ST. MARTIN'S ENGINEERING COLLEGE

An UGC Autonomous Institute
NBA & NAAC A+ Accredited
Dhulapally, Secunderabad-500 100
www.smec.ac.in

PROJECT REPORTS OF IT

A

PROJECT REPORT

On

A DEEP LEARNING FACIAL EXPRESSION RECOGNITION BASED SCORING SYSTEM FOR RESTAURANTS

Submitted by

Ms. PATHI POOJA	(17K81A1241)
Ms. CHANDANA KARTHALA	(17K81A1205)
Mr. POLISHETTY SRIRAM	(17K81A1243)
Mr. PITLA AKHIL SAI	(16K81A1247)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Mr. D. BABU RAO

ASSOCIATE PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN'S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **A DEEP LEARNING FACIAL EXPRESSION RECOGNITION BASED SCORING SYSTEM FOR RESTAURANTS**, is being submitted by **PATHI POOJA (17K81A1241), CHANDANA KARTHALA (17K81A1205), POLISHETTY SRIRAM (17K81A1243), PITLA AKHIL SAI (16K81A1247)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

	Head of the Department
Mr. D.BABU RAO	Dr. R.NAGARAJU
Department of IT	Department of IT

Internal Examiner External Examiner

Place:

Date:

DECLARATION

We, the students of **Bachelor of Technology** in Department of **Information Technology,** session: <2017 – 2021>, St. Martin’s Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled “**A DEEP LEARNING FACIAL EXPRESSION BASED SCORING SYSTEM FOR RESTAURANTS**” is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

PATHI POOJA	**17K81A1241**
CHANDANA KARTHALA	**17K81A1205**
POLISHETTY SRIRAM	**17K81A1243**
PITLA AKHIL SAI	**16K81A1247**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R. NAGARAJU**, Head of the Department, Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

PATHI POOJA	**17K81A1241**
CHANDANA KARTHALA	**17K81A1205**
POLISHETTY SRIRAM	**17K81A1243**
PITLA AKHIL SAI	**16K81A1247**

TABLE OF CONTENTS

ABSTRACT

Recently, the popularity of automated and unmanned restaurants has increased. Due to the absence of staff, there is no direct perception of the customers' impressions in order to find out what their experiences with the restaurant concept are like. For this purpose, this paper presents a rating system based on facial expression recognition with pre-trained convolutional neural network (CNN) models. For interactive human and computer interface (HCI) it is important that the computer understand facial expressions of human. With HCI the gap between computers and humans will reduce. The computers can interact in more appropriate way with humans by judging their expressions. There are various techniques for facial expression recognition which focuses on getting good results of human expressions and then the food is supposed to be rated. Currently, three expressions (satisfied, neutral and disappointed) are provided by the scoring system

LIST OF FIGURES

LIST OF SCREENSHOTS

LIST OF ACRONYMS

<CNN>	Convolutional Neural Network
<HCI>	Human and Computer Interface
<FER>	Facial Expression Recognition
<FACS>	Facial Action Coding System
<LBP>	Local Binary Patterns
<NMF>	Non-negative Matrix Factorization

1. INTRODUCTION

1.1 Project Overview

FACIAL expression is one of the most powerful, natural and universal signals for human beings to convey their emotional states and intentions. Numerous studies have been conducted on automatic facial expression analysis because of its practical importance in sociable robotics, medical treatment, driver fatigue surveillance, and many other human-computer interaction systems. In the field of computer vision and machine learning, various facial expression recognition (FER) systems have been explored to encode expression information from facial representations. As early as the twentieth century, Ekman and Friesen defined six basic emotions based on cross-culture study, which indicated that humans perceive certain basic emotions in the same way regardless of culture.

These prototypical facial expressions are anger, disgust, fear, happiness, sadness, and surprise. Contempt was subsequently added as one of the basic emotions. Recently, advanced research on neuroscience and psychology argued that the model of six basic emotions are culture-specific and not universal. Although the affect model based on basic emotions is limited in the ability to represent the complexity and subtlety of our daily affective displays and other emotion description models, such as the Facial Action Coding System (FACS) and the continuous model using affect dimensions, are considered to represent a wider range of emotions, the categorical model that describes emotions in terms of discrete basic emotions is still the most popular perspective for FER, due to its pioneering investigations along with the direct and intuitive definition of facial expressions. And in this survey, we will limit our discussion on FER based on the categorical model. FER systems can be divided into two main categories according to the feature representations: static image FER and dynamic sequence FER. In static-based methods, the feature representation is encoded with only spatial information from the current single image, whereas dynamic-based methods consider the temporal relation among contiguous frames in the input facial expression sequence. Based on these two vision based methods, other modalities, such as audio and physiological channels, have also been used in multimodal systems to assist the recognition of expression. Most of the traditional methods have used handcrafted features or shallow learning (e.g., local binary patterns (LBP), LBP on three orthogonal planes (LBP-TOP) , non-negative matrix factorization (NMF) and sparse learning) for FER. However, since 2013, emotion recognition competitions such as FER2013 and Emotion Recognition in the Wild (EmotiW) have collected relatively sufficient training data from challenging real-world scenarios, which implicitly promote the transition of FER from lab-controlled to in-the-wild settings. In the meanwhile, due to the dramatically increased chip processing abilities (e.g., GPU units) and well-designed network architecture, studies in various fields have begun to transfer to deep learning methods, which have achieved the state-of-the-art recognition accuracy and exceeded previous results by a large margin. Likewise, given with more effective training data of facial expression, deep learning techniques have increasingly been implemented to handle the challenging factors for emotion recognition in the wild. Figure 1 illustrates this evolution on FER in the aspect of algorithms and datasets.

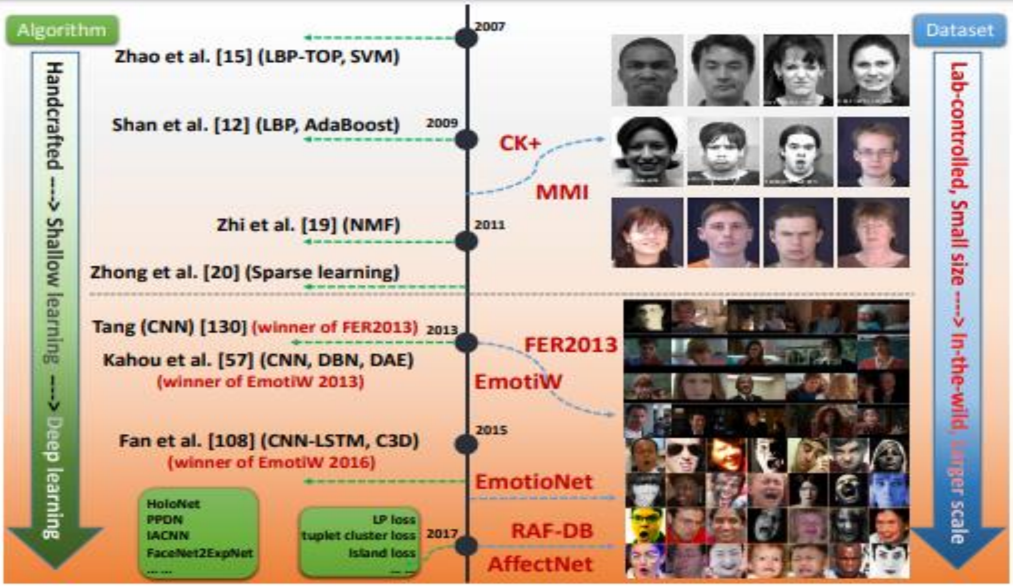

Algorithm
Dataset
Handcrafted ---> Shallow learning ---> Deep learning
Lab-controlled, Small size ---> In-the-wild, Larger scale
2007
Zhao et al. [15] (LBP-TOP, SVM)
Shan et al. [12] (LBP, AdaBoost)
2009
CK+
MMI
Zhi et al. [19] (NMF)
2011
Zhong et al. [20] (Sparse learning)
Tang (CNN) [130] (winner of FER2013)
2013
Kahou et al. [57] (CNN, DBN, DAE)
(winner of EmotiW 2013)
FER2013
EmotiW
2015
Fan et al. [108] (CNN-LSTM, C3D)
(winner of EmotiW 2016)
EmotioNet
HoloNet
PPDN
IACNN
FaceNet2ExpNet
LP loss
tuplet cluster loss
Island loss
2017
RAF-DB
AffectNet

Figure 1.1.1 Evolution on FER in the aspect of algorithms and datasets

Exhaustive surveys on automatic expression analysis have been published in recent years. These surveys have established a set of standard algorithmic pipelines for FER. However, they focus on traditional methods, and deep learning has rarely been reviewed. Very recently, FER based on deep learning has been surveyed in which is a brief review without introductions on FER datasets and technical details on deep FER. Therefore, in this paper, we make a systematic research on deep learning for FER tasks based on both static images and videos (image sequences). We aim to give a newcomer to this filed an overview of the systematic framework and prime skills for deep FER.

Despite the powerful feature learning ability of deep learning, problems remain when applied to FER. First, deep neural networks require a large amount of training data to avoid overfitting. However, the existing facial expression databases are not sufficient to train the well-known neural network with deep architecture that achieved the most promising results in object recognition tasks. Additionally, high inter-subject variations exist due to different personal attributes, such as age, gender, ethnic backgrounds and level of expressiveness. In addition to subject identity bias, variations in pose, illumination and occlusions are common in unconstrained facial expression scenarios. These factors are nonlinearly coupled with facial expressions and therefore strengthen the requirement of deep networks to address the large intra-class variability and to learn effective expression-specific representations.

1.2 Project Objectives

The Internet has opened the new doors for information exchange and the growth of social media has created unprecedented opportunities for citizens to publicly raise their opinions, but it has serious bottlenecks when it comes to doing analysis of these opinions. So the main objectives of our project are:

1. Avoids errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.
2. User-friendly screens for the data entry to handle large volume of data.
3. Unmanned task can be done by just detecting the Facial emotion of the person.

1.3 Scope of the Project

The project scope statement is a key document that provides all stakeholders with a clear understanding of why the project was initiated and defines its key goals. Our goal is to predict the Restaurant reviews, to get User opinions by analysing reviews and to know the performance of our trained CNN Model. We experiment with different linguistically motivated models of sentiment expression, again using the results to improve the performance of our classifier. Last but not the least, human expressive behaviors in realistic applications involve encoding from different perspectives, and the facial expression is only one modality. Although pure expression recognition based on visible face images can achieve promising results, incorporating with other models into a high-level framework can provide complementary information and further enhance the robustness. For example, participants in the EmotiW challenges and Audio Video Emotion Challenges (AVEC) considered the audio model to be the second most important element and employed various fusion techniques for multimodal affect recognition. Additionally, the fusion of other modalities, such as infrared images, depth information from 3D face models and physiological data, is becoming a promising research direction due to the large complementarity for facial expressions.

1.4 Organization of Chapters

1.4.1 Introduction

1.4.2 Literature Survey

This chapter deal with the study of the existing applications. The pros and cons of the existing application are carefully examined. The user expectations from the system, their experiences and the overall perception of the system is stated. The developer’s perceptions are also recorded.

1.4.3 Software and Hardware Requirements

The scope of our application is determined and software requirement specifications are hence concluded in this chapter.

1.4.4 Software Development Analysis

The existing systems are given a deeper study from the developers view. The infrastructure, the logic and the implementation methods are analyzed. The drawbacks and the problems faced by the developers in developing were deeply studied

1.4.5 Project System Design

The proposed system details are stated in this chapter. The infrastructure and the workflow of the application were discussed. The algorithms to be used are also stated here with their pros and cons.

1.4.6 Project Coding

The implementation details are explained in this chapter. The system design is converted into code by developing the required features.

1.4.7 Project Testing

The developed application is vigorously tested. The metrics that are used for testing the application are recorded. The expected and the actual behavior of the application is reported in this chapter.

1.4.8 Output Screens

This chapter contains the information of the deployment of the application in the form of screenshots and navigation is explained.

1.4.8 Conclusion

This chapter contains the final conclusions which are drawn after deployment of the application.

2. LITERATURE SURVEY

2.1 Survey on Background

2.1.1 P. Ekman and W. V. Friesen, "Constants across societies in the face and feeling." Journal of character and social brain research, vol. 17, no. 2, pp. 124–129, 1971

Explored the subject of whether any outward appearances of feeling are widespread. Ongoing investigations indicating that individuals from educated societies connected a similar feeling ideas with a similar facial practices couldn't exhibit that probably some outward appearances of feeling are all inclusive; the way of life contrasted had all been uncovered with a portion of a similar broad communications introductions of outward appearance, and these may have shown the individuals in each culture to perceive the remarkable outward appearances of different societies. To show that individuals from a preliterate culture who had negligible introduction to proficient societies would connect a similar feeling ideas with indistinguishable facial practices from do individuals from Western and Eastern educated societies, information were accumulated in New Guinea by recounting to 342 Ss a story, demonstrating them a lot of 3 faces, and requesting that they select the face which indicated the feeling proper to the story. Ss were individuals from the Fore phonetic social gathering, which up until 12 yr. prior was a confined, Neolithic, material culture. Results give proof on the side of the theory.

2.1.2 R. E. Jack, O. G. Garrod, H. Yu, R. Caldara, and P. G. Schyns, "Outward appearances of feeling are not socially all inclusive," Proceedings of the National Academy of Sciences, vol. 109, no. 19, pp. 7241–7244, 2012.

Since Darwin's original works, the all-inclusiveness of outward appearances of feeling has stayed one of the longest standing discussions in the organic and sociologies. Quickly expressed, the comprehensiveness speculation guarantees that all people impart six fundamental inward enthusiastic states (glad, shock, dread, disturb, outrage, and tragic) utilizing a similar facial developments by uprightness of their organic and transformative starting points [Susskind JM, et al. (2008) Nat Neurosci 11:843–850]. Here, we disprove this expected all-inclusiveness. Utilizing a one of a kind PC designs stage that consolidates generative syntaxes [Chomsky N (1965) MIT Press, Cambridge, MA] with visual discernment, we got to the imagination of 30 Western and Eastern culture people and remade their psychological portrayals of the six fundamental outward appearances of feeling. Multifaceted examinations of the psychological portrayals challenge all-inclusiveness on two separate checks. To begin with, while Westerners speak to every one of the six fundamental feelings with an unmistakable arrangement of facial developments normal to the gathering, Easterners don't. By discrediting the long-standing all-inclusiveness speculation, our information feature the ground-breaking impact of culture on molding essential practices once considered naturally designed. Therefore, our information open an extraordinary nature–sustain banter across expansive fields from transformative brain science and social neuroscience to long range interpersonal communication by means of advanced symbols.

2.1.3 Y.-I. Tian, T. Kanade, and J. F. Cohn, "Recognizing action units for facial expression analysis," IEEE Transactions on pattern analysis and machine intelligence, vol. 23, no. 2, pp. 97–115, 2001.

Ekman and Friesen built up the Facial Action Coding System (FACS) for portraying outward appearances by activity units (AUs). Of 44 FACS AUs that they characterized, 30 AUs are anatomically identified with the withdrawals of explicit facial muscles: 12 are for upper face, and 18 are for lower face. AUs can happen either separately or in blend. At the point when AUs happen in blend they might be added substance, in which the mix doesn't change the presence of the constituent AUs, or no additive, in which the presence of the constituents changes. Even though the quantity of nuclear activity units is moderately little, more than 7,000 diverse AU mixes have been watched [30] . FACS gives the elucidating power important to portray the subtleties of outward appearance. Usually happening AUs and a portion of the added substance and no additive AU blends are appeared in Tables 1 and 2. For instance of a no additive impact, AU 4 shows up contrastingly relying upon whether it happens alone or in mix with AU 1 (as in AU 1 ‡

4). At the point when AU 4 happens alone, the foreheads are drawn together and brought down. In AU 1 ‡ 4, the foreheads are drawn together however are raised because of the activity of AU 1. AU 1 ‡ 2 is another case of no additive mixes. At the point when AU 2 happens alone, it raises the external forehead, yet in addition regularly pulls up the internal temple which brings about a fundamentally the same as appearance to AU 1 ‡ 2. These impacts of the no additive AU mixes increment the challenges of AU acknowledgment.

2.2 Conclusion on Survey

For interactive human and computer interface (HCI) it is important that the computer understand facial expressions of human. With HCI the gap between computers and humans will reduce. The computers can interact in more appropriate way with humans by judging their expressions. There are various techniques for facial expression recognition which focuses on getting good results of human expressions in order to get reviews in the restaurants. We decided to present a rating system based on facial expression recognition with pre-trained convolutional neural network (CNN) models.

3. SOFTWARE AND HARDWARE REQUIREMENTS

3.1 Software Requirements

- Operating System : Windows family
- Technology : Python 3.6 or Higher
- IDE : PyCharm

3.1.1 Python

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. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython 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.

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.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, PyCharm, NetBeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

3.1.2 Purpose

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

3.1.3 Functional Requirements

Graphical User interface with the User.

3.1.4 Non-functional Requirements

- **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.
- **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.
- **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.
- **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.
- **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.
- **Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life. However, the allowable wattage can’t be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

3.1.5 Input and Output Design

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

- 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.
- 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.
- 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

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.

3.2 Hardware Requirements

- Processer : Intel i3 or Higher
- Ram : Min 4 GB
- Hard Disk : Min 100 GB

4. SOFTWARE DEVELOPMENT ANALYSIS

4.1 Overview of Problem:

The purpose of this analysis is to build a prediction model to detect the facial expression in order to predict the review on a restaurant's food/service. To do so, we will work on various Facial Expression datasets, we will load it into Viola Jones Object detection model which makes use of Haar based classifiers. In the end, we hope to find a "best" model for predicting the customer reviews based on facial expressions.

To build the CNN model to predict the review based on customer facial expressions, following steps are performed.

The network uses a cascade structure with three networks:

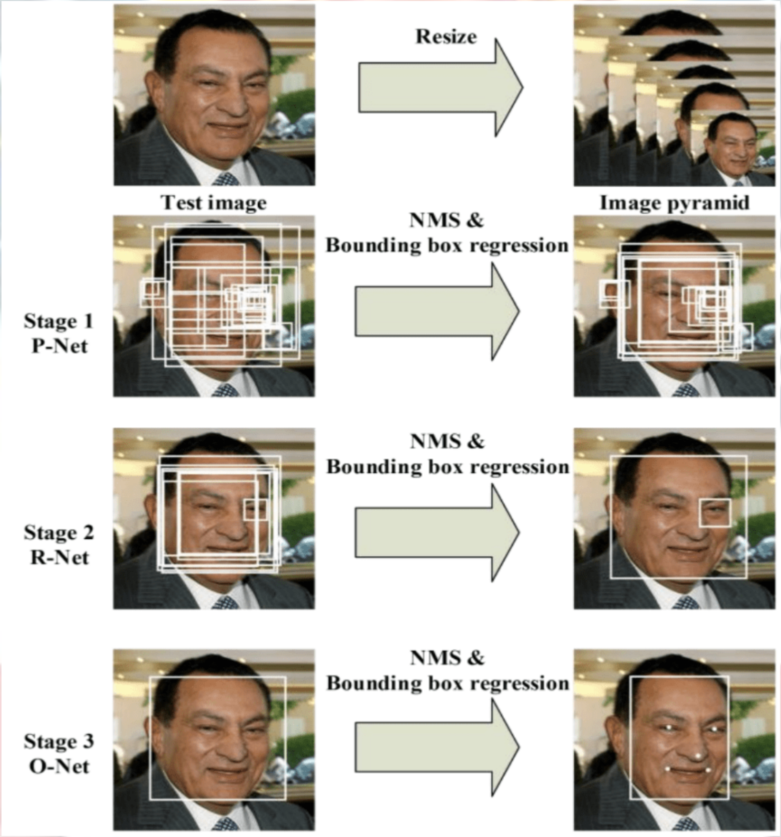


Figure 4.1.1. Cascade Structure of network

- First the image is rescaled to a range of different sizes (called an image pyramid), then the first model (Proposal Network or P-Net) proposes candidate facial regions.

- The second model (Refine Network or R-Net) filters the bounding boxes
- The third model (Output Network or O-Net) proposes facial landmarks.

4.1.1 Existing System

As there is no staff available in unmanned restaurants, it is difficult for the restaurant management to estimate how the concept and the food is experienced by the customers. Existing rating systems, such as Google and TripAdvisor, only partially solve this problem, as they only cover a part of the customer's opinions.

4.1.2 Disadvantages of Existing System

1. These rating systems are only used by a subset of the customers who rate the restaurant on independent rating platforms on their own initiative.
2. This applies mainly to customers who experience their visit as very positive or negative.
3. Anonymous reviews posted on independent rating platforms will often lead to a mixed perception on the restaurants making it hard to decide its genuinely.

4.2 Define the Problem :

Let's define the problem in order to think about the solution so as to get an optimal solution. Currently there are many restaurants which are operating all around the world. But hardly restaurants know about what their customers think of the restaurant's service. Restaurants aren't aware of the reasons for their thrusts and troughs , facial recognition analysis helps them for the reason for fluctuation of their financial condition. We collect facial reviews from various customers and perform face detection followed by facial expression recognition using Viola Jones Object Detection Algorithm which is based on Haar based classifiers.

4.2.1 Proposed System

In order to solve the above problem, all customers must be motivated to give a rating. The proposed system introduces an approach for a restaurant rating system that asks every customer for a rating after their visit to increase the number of ratings as much as possible. This system can be used unmanned restaurants; the scoring system is based on facial expression detection using pre- trained convolutional neural network (CNN) models. It allows the customer to rate the food by taking or capturing a picture of his face that reflects the corresponding feelings.

4.2.2 Advantages of Proposed System

1. Compared to text-based rating system, there is much less information and no individual experience reports collected.
2. It is a simple, fast and playful rating system.
3. It gives a wider range of opinions about the experiences of the customers with the restaurant concept.
4. There is no need for any independent rating platforms.
5. Can know every person's review who visited the restaurants.

4.3 Modules Overview :

In Development and analysis of the data set, there are three modules involved. They are:

1. Face Detection
2. Facial Expression Recognition Classification
3. Convolutional Neural Network (CNN)

4.4 Define the Modules

4.4.1 Face Detection

Face detection or localization is an important step for image classification since only the principal component of face such as nose, eyes, mouth are needed for classification. Face detection algorithms can be broadly classified into feature, knowledge, template and appearance based methods. Our proposed system uses Viola Jones object detection algorithm for face localization which comes under feature based classification. Viola Jones object detection algorithm uses Haar feature based cascade classifiers. The Haar Cascade classifier is extremely important element of the face detection. The presence of the features in any of the input image is determined by the Haar features.

4.4.2 Facial Expression Recognition classification:

After learning the deep features, the final step of FER (Facial Expression Recognition) is to classify the given face into one of the basic emotion categories. Unlike the traditional methods, where the feature extraction step and the feature classification step are independent, deep networks can perform FER in an end-to-end way. Specifically, a loss layer is added to the end of the network to regulate the back-propagation error; then, the prediction probability of each sample can be directly output by the network.

4.4.3 Convolutional neural network (CNN):

In CNN, SoftMax loss is the most common used function that minimizes the cross-entropy between the estimated class probabilities and the ground truth distribution.CNN has been extensively used in diverse computer vision applications, including FER. At the beginning of the 21st century, several studies in the FER literature found that the CNN is robust to face location changes and scale variations and behaves better than the multilayer perceptron (MLP) in the case of previously unseen face pose variations, employed the CNN to address the problems of subject independence as well as translation, rotation, and scale invariance in the recognition of facial expressions.

4.4 Module Functionality:

The CNN model collects the reviews from a .xml file which contains various facial expressions based on which the emotion label is detected, and the results are subjected to Viola Jones Object Detection Algorithm and Haar based classifiers to obtain the results.

5. PROJECT SYSTEM DESIGN

5.1 Architecture Diagram:

An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components. It is an important tool as it provides an overall view of the physical deployment of the software system and its evolution roadmap.

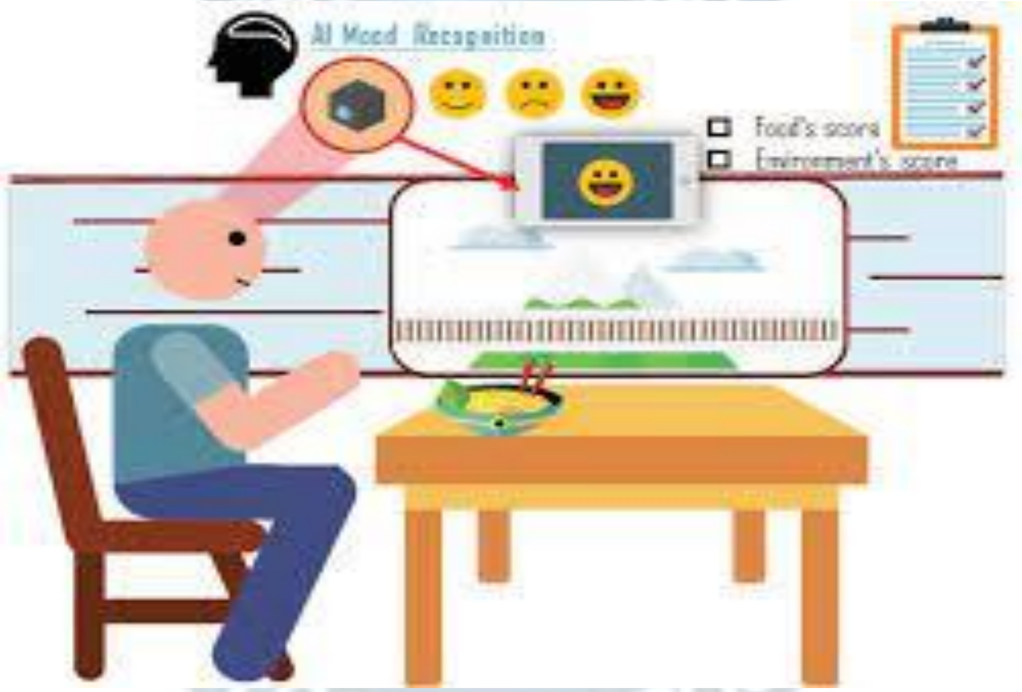


Figure 5.1.1 Architecture Diagram

5.2 UML Diagrams:

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML comprises two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML. The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development processes.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of the OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

5.2.1 Usecase Diagram :

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

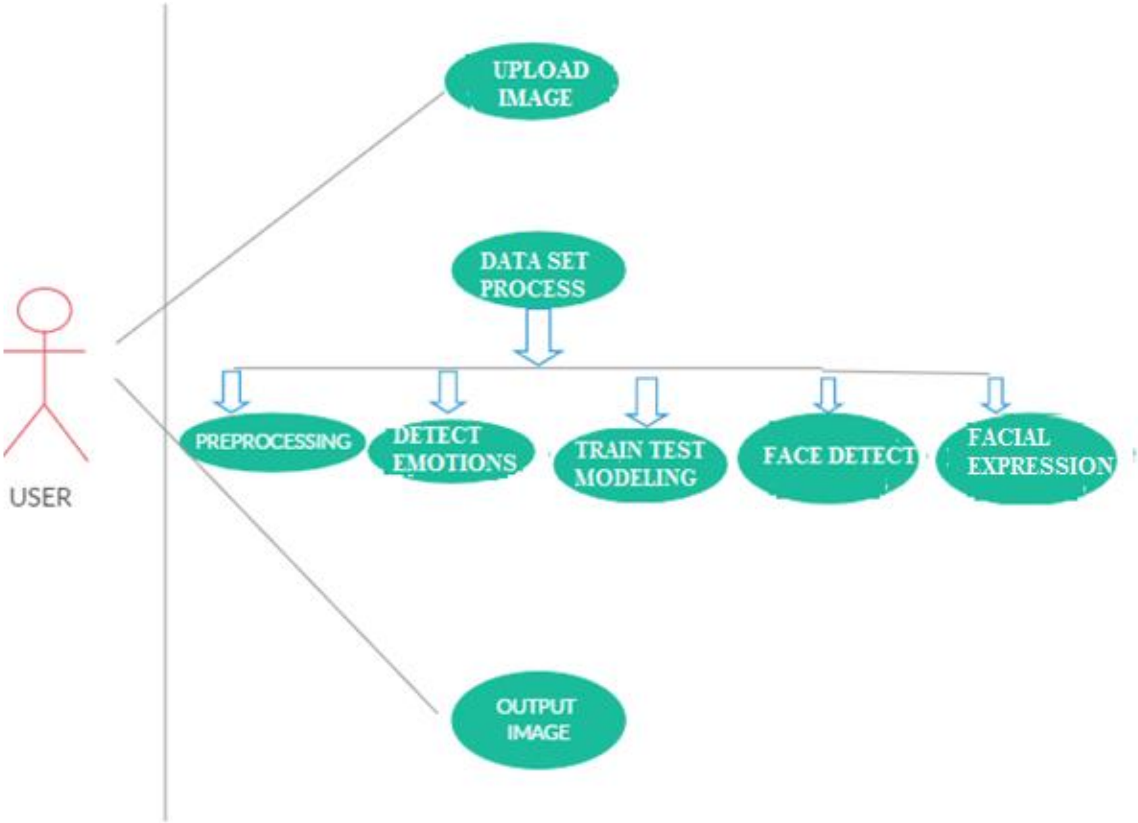


Figure 5.2.1.1. Usecase diagram of User

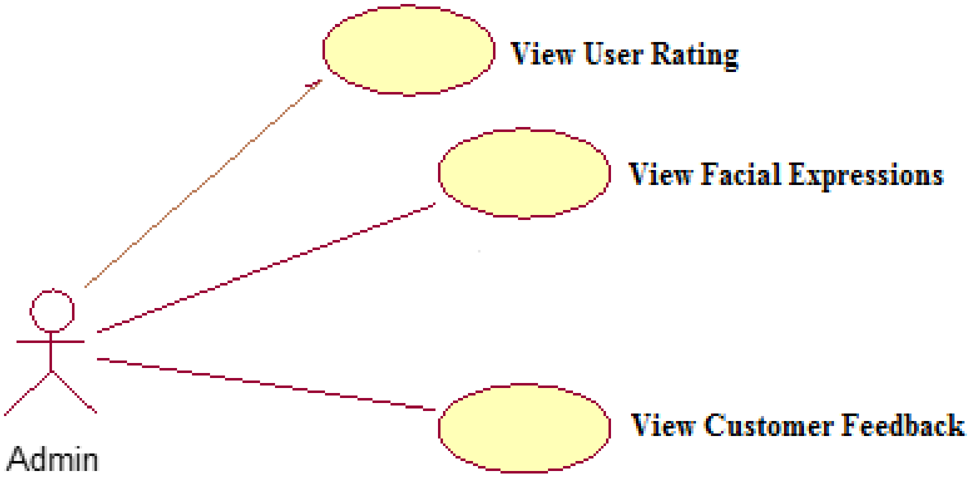


Figure 5.2.1.2 Usecase Diagram of Admin

5.2.2 Sequence Diagram :

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagram.

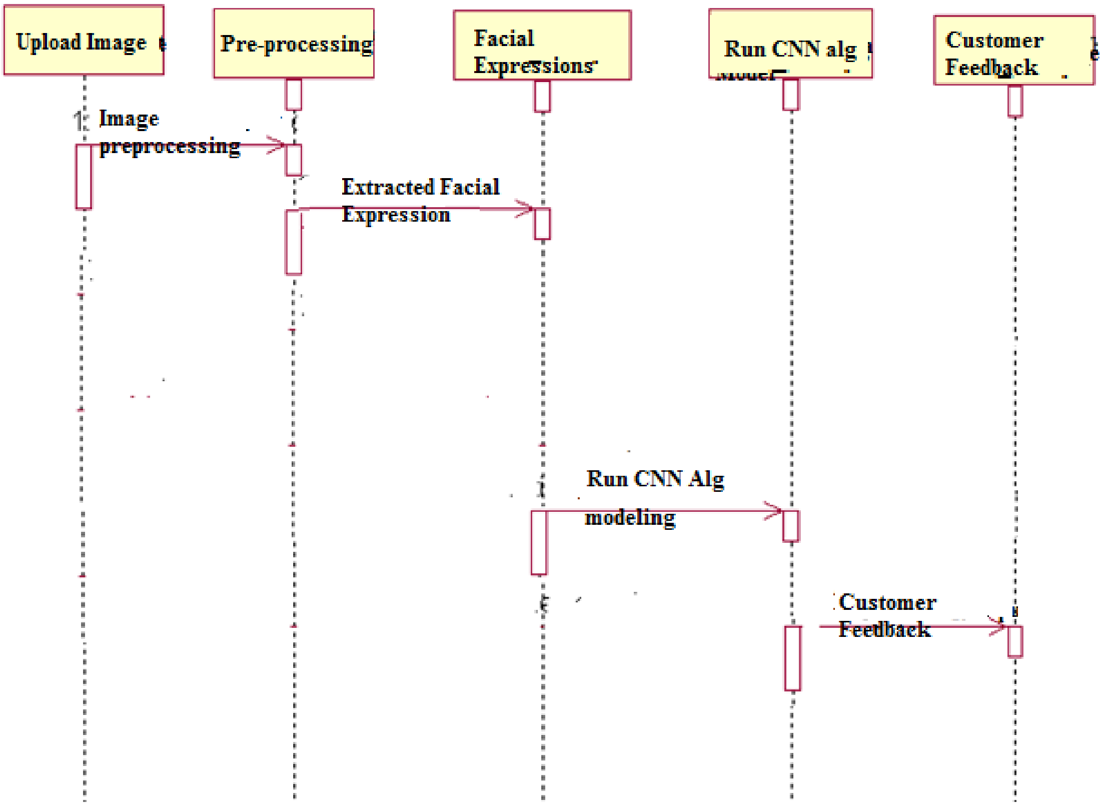


Figure 5.2.2.1 Sequence Diagram

5.2.3 Component Diagram:

The component diagram extends the information given in a component notation element. One way of illustrating the provided and required interfaces by the specified component is in the form of a rectangular compartment attached to the component element.[2] Another accepted way of presenting the interfaces is to use the ball-and-socket graphic convention. A provided dependency from a component to an interface is illustrated with a solid line to the component using the interface from a "lollipop", or ball, labelled with the name of the interface. A required usage dependency from a component to an interface is illustrated by a half-circle, or socket, labelled with the name of the interface, attached by a solid line to the component that requires this interface. Inherited interfaces may be shown with a lollipop, preceding the name label with a caret symbol. To illustrate dependencies between the two, use a solid line with a plain arrowhead joining the socket to the lollipop.

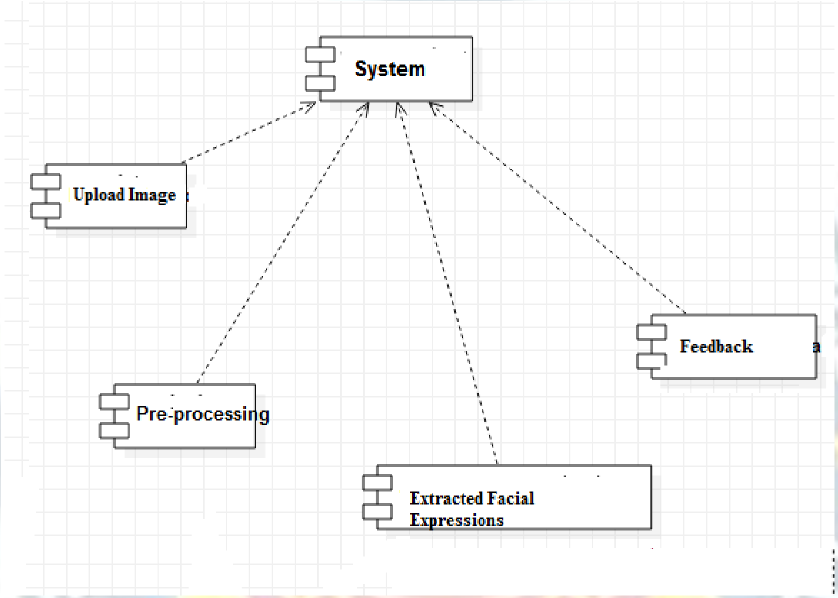


Figure 5.2.3.1 Component Diagram

5.2.4 Activity diagrams:

Activity diagrams are graphical representations of workflows of stepwise activities and actions[1] with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores. Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

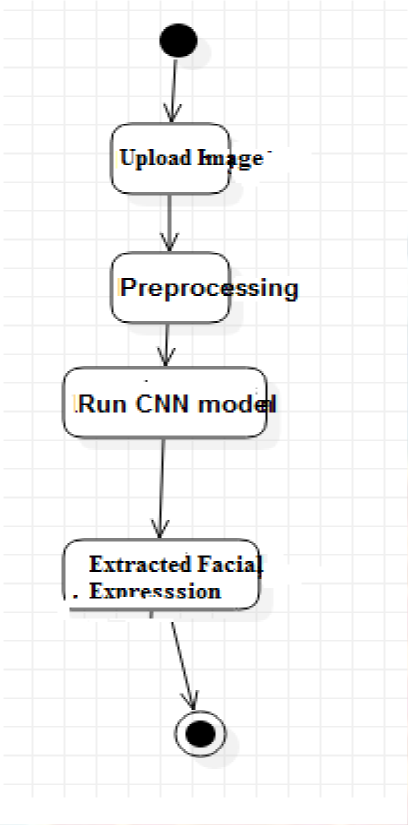


Fig 5.2.4.1 Activity Diagram

5.2.5 Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application, however class diagram is a bit different. It is the most popular UML diagram in the coder community.

The purpose of the class diagram can be summarized as −

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

Class diagram is basically a graphical representation of the static view of the system and represents different aspects of the application. A collection of class diagrams represent the whole system.

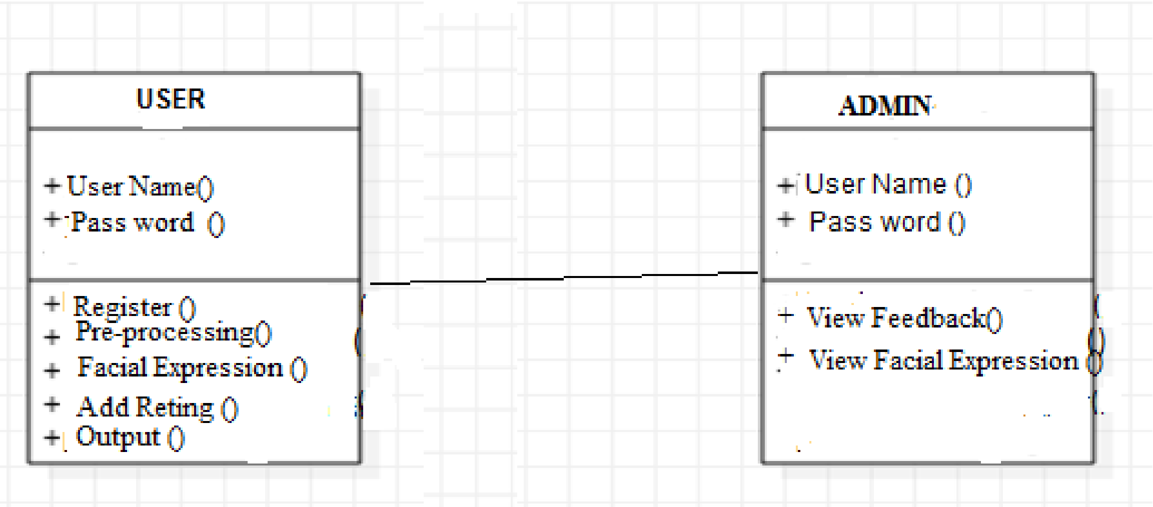


Figure 5.2.5.1 Class Diagram

6. PROJECT CODING

6.1 Code Templates

FacialApp\views.py

```
from django.shortcuts import render
from django.template import RequestContext
import pymysql
from django.http import HttpResponse
from django.conf import settings
from django.core.files.storage import FileSystemStorage
import datetime
import cv2
from keras.models import load_model
from keras.preprocessing.image import img_to_array
import numpy as np



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('t1', False)
      password = request.POST.get('t2', 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 ViewRating(request):
    if request.method == 'GET':
       strdata = '<table border=1 align=center width=100%><tr><th>Customer Name</th><th>Rating</th><th>Facial Expression</th><th>Photo</th> <th>Date & Time</th></tr><tr>'
       con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'facial',charset='utf8')
       with con:
          cur = con.cursor()
          cur.execute("select * FROM rating")
          rows = cur.fetchall()
          for row in rows:

strdata+='<td>'+row[0]+'</td><td>'+str(row[1])+'</td><td>'+row[2]+'</td><td><img src=/static/photo/'+row[0]+'.png width=200 height=200></img></td><td>'+str(row[4])+'</td></tr>'
    context= {'data':strdata}
    return render(request, 'ViewRatings.html', context)
```

```
def Rating(request):
    if request.method == 'POST' and request.FILES['t3']:
        output = ''
        myfile = request.FILES['t3']
        name = request.POST.get('t1', False)
        rating = request.POST.get('t2', False)
        fs = FileSystemStorage()
        filename = fs.save('C:/Python/Facial/Facial/FacialApp/static/photo/'+name+'.png',
myfile)
        now = datetime.datetime.now()
        current_time = now.strftime("%Y-%m-%d %H:%M:%S")
        detection_model_path =
'C:/Python/Facial/Facial/FacialApp/haarcascade_frontalface_default.xml'
        emotion_model_path = 'C:/Python/Facial/Facial/FacialApp/_mini_XCEPTION.106-
0.65.hdf5'
        face_detection = cv2.CascadeClassifier(detection_model_path)
        emotion_classifier = load_model(emotion_model_path, compile=False)
        EMOTIONS = ["angry","disgust","scared", "happy", "sad", "surprised","neutral"]
        orig_frame =
cv2.imread('C:/Python/Facial/Facial/FacialApp/static/photo/'+name+'.png')
        orig_frame = cv2.resize(orig_frame, (48, 48))
        frame = cv2.imread(filename,0)
        faces =
face_detection.detectMultiScale(frame,scaleFactor=1.1,minNeighbors=5,minSize=(30,30
),flags=cv2.CASCADE_SCALE_IMAGE)
        print("==================="+str(len(faces)))
        print(emotion_classifier)
        if len(faces) > 0:
            faces = sorted(faces, reverse=True,key=lambda x: (x[2] - x[0]) * (x[3] - x[1]))[0]
            (fX, fY, fW, fH) = faces
            roi = frame[fY:fY + fH, fX:fX + fW]
            roi = cv2.resize(roi, (48, 48))
            roi = roi.astype("float") / 255.0
            roi = img_to_array(roi)
            roi = np.expand_dims(roi, axis=0)
            preds = emotion_classifier.predict(roi)[0]
            emotion_probability = np.max(preds)
            label = EMOTIONS[preds.argmax()]
            if label == 'happy':
              output = 'Satisfied'
            if label == 'neutral':
              output = 'Neutral'
            if label == 'angry' or label == 'sad' or label == 'disgust' or label == 'scared' or label == 'surprised':
              output = 'Disappointed'
        print("==================="+output)
        db_connection = pymysql.connect(host='127.0.0.1',port = 3308,user = 'root', password = 'root', database = 'facial',charset='utf8')
        db_cursor = db_connection.cursor()
        query = "INSERT INTO
rating(customer_name,rating,facial_expression,photo_path,rating_date)
VALUES('"+name+"','"+rating+"','"+output+"','"+name+'.png'+"','"+current_time+"')"
        db_cursor.execute(query)
        db_connection.commit()
        print(db_cursor.rowcount, "Record Inserted")
        if db_cursor.rowcount == 1:
            context= {'data':'Your Rating is : '+rating+' and Facial Expression : '+output}
            return render(request, 'User.html', context)
```

```
else:
        context= {'data':'Error in request process'}
        return render(request, 'User.html', context)
```

FacialApp\urls.py

```
from django.urls import path
from . import views
urlpatterns = [path("index.html", views.Index, name="Index"),
            path("User.html", views.User, name="User"),
            path("Rating", views.Rating, name="Rating"),
            path("Admin.html", views.Admin, name="Admin"),
            path("AdminLogin", views.AdminLogin, name="AdminLogin"),
            path("ViewRating", views.ViewRating, name="ViewRating"),
]
```

Facial\settings.py

```
"""
Django settings for Facial project.

Generated by 'django-admin startproject' using Django 2.2.7.

For more information on this file, see
https://docs.djangoproject.com/en/2.2/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/2.2/ref/settings/
"""

import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
```

```
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.2/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'x9at8+ndi2w522k3r&54&8gv6zc^#pv4ol_t^1sl#8c&fjv0hr'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'FacialApp'
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'Facial.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [
                    os.path.join('C:/Python/Facial/Facial/FacialApp', 'templates'),
          ],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'Facial.wsgi.application'
# Database
# https://docs.djangoproject.com/en/2.2/ref/settings/#databases
```

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'chatbot',
        'HOST': '127.0.0.1',
        'PORT': '3306',
        'USER': 'root',
        'PASSWORD': 'root',
            'OPTIONS': {
          'autocommit': True,
        },
    }
}



# Password validation
# https://docs.djangoproject.com/en/2.2/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]


# Internationalization
# https://docs.djangoproject.com/en/2.2/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/2.2/howto/static-files/

STATIC_URL = '/static/'
```

Facial\urls.py

```
"""Facial URL Configuration

The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/2.2/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path, include.

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('FacialApp.urls')),
]
```

Facial\wsgi.py

```
"""
WSGI config for Facial project.
It exposes the WSGI callable as a module-level variable named ``application``.
For more information on this file, see
https://docs.djangoproject.com/en/2.2/howto/deployment/wsgi/
"""
```

```
import os
from django.core.wsgi import get_wsgi_application
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'Facial.settings')
application = get_wsgi_application()
```

templates\index.html

```
{% load static %}
<html>
<head>
<title>Facial Expression Recognition</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link href="{% static 'style.css' %}" rel="stylesheet" type="text/css" />
</head>
<body>
<div class="main">
  <div class="main_resize">
    <div class="header">
      <div class="logo">
        <h1><span><center>A Deep Learning Facial Expression Recognition Based
Scoring
<br/><center>System for Restaurants</center></span><small></small></h1>
      </div>
    </div>
    <div class="content">
      <div class="content_bg">
        <div class="menu_nav">
          <ul>
          <li><a href="{% url 'Index' %}">Home</a></li>
                    <li><a href="{% url 'User' %}">User</a></li>
                    <li><a href="{% url 'Admin' %}">Administrator</a></li>
          </ul>
  </div>
    <div class="hbg"><img src="{% static 'images/header_images.jpg' %}"
width="915" height="286" alt="" /></div>
<p align="justify"><font size="3" style="font-family: Comic Sans MS"
color="black">Recently, the popularity of automated and unmanned restaurants has
increased. Due to the absence of staff, there is no direct perception of the customers'
impressions in order to find out what their experiences with the restaurant
concepts are like. For this purpose, this paper presents a rating system based on facial
expression recognition with pre-trained convolutional neural network (CNN) models. It is
composed of a web application, a web server, and a pre-trained AIserver. Both the food
and the environment are supposed to be rated. Currently, three expressions (satisfied,
neutral and disappointed) are provided by the scoring system.</p>
</body>
</html>
```

templates\user.html

```
{% load static %}
<html>
<head>
<title>Facial Expression Recognition</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link href="{% static 'style.css' %}" rel="stylesheet" type="text/css" />
<script language="javascript">
        function validate(formObj)
        {
        if(formObj.t1.value.length==0)
        {
        alert("Please enter customer name");
        formObj.t1.focus();
        return false;
        }
        if(formObj.t2.value.length==0)
        {
        alert("Please enter rating");
        formObj.t2.focus();
        return false;
        }
        if(formObj.t3.value.length==0)
        {
        alert("Please upload photo");
        formObj.t3.focus();
        return false;
        }
        formObj.actionUpdateData.value="update";
        return true;
        }
        </script>
</head>
<body>
<div class="main">
  <div class="main_resize">
    <div class="header">
      <div class="logo">
        <h1><span><center>A Deep Learning Facial Expression Recognition Based Scoring
<br/><center>System for Restaurants</center></span><small></small></h1>
      </div>
    </div>
    <div class="content">
      <div class="content_bg">
        <div class="menu_nav">
          <ul>
```

```
<li><a href="{% url 'Index' %}">Home</a></li>
                    <li><a href="{% url 'User' %}">User</a></li>
                    <li><a href="{% url 'Admin' %}">Administrator</a></li>
        </ul>
      </div>
      <div class="hbg"><img src="{% static 'images/header_images.jpg' %}"
width="915" height="286" alt="" /></div>
                                      <center>
<form name="f1" method="post" action="{% url 'Rating' %}" enctype="multipart/form-
data" onsubmit="return validate(this);">
{% csrf_token %}
<br/>
  <h2><b>User Rating Screen</b></h2>

<font size="" color="red"><center>{{ data }}</center></font>

<table align="center" width="40" >
<tr><td><b>Customer&nbsp;Name</b></td><td><input type="text" name="t1"
style="font-family: Comic Sans MS" size="20"/></td></tr>
<tr><td><b>Rating</b></td><td><input type="text" name="t2" style="font-family:
Comic Sans MS" size="10"/></td></tr>
<tr><td><b>Upload&nbsp;Photo</b></td><td><input type="file" name="t3"
style="font-family: Comic Sans MS" size="30"/></td></tr>
<tr><td></td><td><input type="submit" value="Submit"></td>
</table>
</div>
</div>
</body>
</html>
```

templates\adminscreen.html

```
{% load static %}
<html>
<head>
<title>Facial Expression Recognition</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link href="{% static 'style.css' %}" rel="stylesheet" type="text/css" />
</head>
<body>
<div class="main">
  <div class="main_resize">
    <div class="header">
      <div class="logo">
        <h1><span><center>A Deep Learning Facial Expression Recognition Based
Scoring
<br/><center>System for Restaurants</center></span><small></small></h1>
      </div>
    </div>
    <div class="content">
      <div class="content_bg">
        <div class="menu_nav">
          <ul>
          <li><a href="{% url 'ViewRating' %}">View Users Rating</a></li>
                          <li><a href="{% url 'Index' %}">Logout</a></li>
                          </ul>
        </div>
        <div class="hbg"><img src="{% static 'images/header_images.jpg' %}"
width="915" height="286" alt="" /></div>
```

```
<p align="justify"><font size="3" style="font-family: Comic Sans MS" color="black">
<font size="" color="red"><center>{{ data }}</center></font></p>
</body>
</html>
```

manage.py

```
#!/usr/bin/env python
"""Django's command-line utility for administrative tasks."""
import os
import sys
def main():
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'Facial.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)

if __name__ == '__main__':
    main()
```

6.2 Outline for Various Files

app1\views.py

This is the important file of our project. First the required modules are imported. Then the facial expressions from the .xml haar classifier file are read. We need to pre-process our image by removing any light effects or unwanted effects to enhance the image. Next, we resize the image into blocks of 48*48 to detect the face in image. Once this process is done, we will have all the faces detected in the image. Then the face with the highest accuracy is processed further for classification. In the next step, the trained CNN model detects the emotion label of the face detected and provides us the result.

urls.py

Every page on the Internet needs its own URL. This way your application knows what it should show to a user who opens that URL. In Django, we use something called URLconf (URL configuration). URLconf is a set of patterns that Django will try to match the requested URL to find the correct view. This happens in urls.py file.

settings.py

settings.py is a core file in Django projects. It holds all the configuration values that your web app needs to work; database settings, logging configuration, where to find static files, API keys if you work with external APIs, and a bunch of other stuff.

wsgi.py

Django's primary deployment platform is WSGI, the Python standard for web servers and applications. Django's startproject management command sets up a minimal default WSGI configuration for you, which you can tweak as needed for your project, and direct any WSGI-compliant application server to use.

index.html , user.html and adminscreen.html

An HTML file contains Hypertext Markup Language (HTML), which is used to format the structure of a webpage. It is stored in a standard text format and contains tags that define the page layout and content of the webpage, including the text, tables, images, and hyperlinks displayed on the webpage.

manage.py

A command-line utility that lets you interact with this **Django** project in various ways. The **manage.py** script is used to create applications, work with databases, and start the development web server.

7. PROJECT 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.

7.1 Various Testcases

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output, often with the help of a test oracle or a previous result that is known to be good, without any knowledge of the test object's internal structure.

7.2 Block Box Testing

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing. The main focus of black box testing is on the validation of your functional requirements. Black box testing gives abstraction from code and focuses on testing effort on the software system behavior. Black box testing facilitates testing communication amongst modules

Input
Black Box
Output

Fig 7.2.1 Black Box Structure

The above Black-Box can be any software system you want to test. For Example, an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation.

Here are the generic steps followed to carry out any type of Black Box Testing.

- Initially, the requirements and specifications of the system are examined.
- Tester chooses valid inputs (positive test scenario) to check whether SUT processes them correctly. Also, some invalid inputs (negative test scenario) are chosen to verify that the SUT is able to detect them.
- Tester determines expected outputs for all those inputs.
- Software tester constructs test cases with the selected inputs.
- The test cases are executed.
- Software tester compares the actual outputs with the expected outputs.
- Defects if any are fixed and re-tested.

7.2.1 Types of Black Box Testing

There are many types of Black Box Testing but the following are the prominent ones -

- **Functional testing** - This black box testing type is related to the functional requirements of a system; it is done by software testers.
- **Non-functional testing** - This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.

- **Regression testing** - Regression Testing is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

7.2.2 Tools used for Black Box Testing:

Tools used for Black box testing largely depends on the type of black box testing you are doing.

- For Functional/ Regression Tests you can use - QTP, Selenium.
- For Non-Functional Tests, you can use - LoadRunner, Jmeter.

7.2.3 Black Box Testing Techniques

Following are the prominent Test Strategy amongst the many used in Black box Testing

- **Equivalence Class Testing:** It is used to minimize the number of possible test cases to an optimum level while maintains reasonable test coverage.
- **Boundary Value Testing:** Boundary value testing is focused on the values at boundaries. This technique determines whether a certain range of values are acceptable by the system or not. It is very useful in reducing the number of test cases. It is most suitable for the systems where an input is within certain ranges.
- **Decision Table Testing**: A decision table puts causes and their effects in a matrix. There is a unique combination in each column.

7.3 White Box Testing

White Box Testing is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers, so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing and Glass box testing.

It is one of two parts of the Box Testing approach to software testing. Its counterpart, Blackbox testing, involves testing from an external or end-user type perspective. On the other hand, White box testing in software engineering is based on the inner workings of an application and revolves around internal testing.

The term "Whitebox" was used because of the see-through box concept. The clear box or Whitebox name symbolizes the ability to see through the software's outer shell (or "box") into its inner workings. Likewise, the "black box" in "Black Box Testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested. White box testing involves the testing of the software code for the following:

- Internal security holes
- Broken or poorly structured paths in the coding processes
- The flow of specific inputs through the code
- Expected output
- The functionality of conditional loops
- Testing of each statement, object, and function on an individual basis

The testing can be done at system, integration and unit levels of software development. One of the basic goals of Whitebox testing is to verify a working flow for an application. It involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output, you have encountered a bug.

7.3.1 Steps in White Box Testing:

we have divided it into **two basic steps**. This is what testers do when testing an application using the white box testing technique:

Step 1) Understand the Source Code

The first thing a tester will often do is learn and understand the source code of the application. Since white box testing involves the testing of the inner workings of an application, the tester must be very knowledgeable in the programming languages used in the applications they are testing. Also, the testing person must be highly aware of secure coding practices. Security is often one of the primary objectives of testing software. The tester should be able to find security issues and prevent attacks from hackers and naive users who might inject malicious code into the application either knowingly or unknowingly.

Step 2) Create Test Cases and Execute

The second basic step to white box testing involves testing the application's source code for proper flow and structure. One way is by writing more code to test the application's source code. The tester will develop little tests for each process or series of processes in the application. This method requires that the tester must have intimate knowledge of the code and is often done by the developer. Other methods include Manual Testing, trial, and error testing and the use of testing tools as we will explain further on in this article.

7.3.2 White Box Testing Techniques

A major White box testing technique is Code Coverage analysis. Code Coverage analysis eliminates gaps in a Test Case suite. It identifies areas of a program that are not exercised by a set of test cases. Once gaps are identified, you create test cases to verify untested parts of the code, thereby increasing the quality of the software product. There are automated tools available to perform Code coverage analysis. Below are a few coverage analysis techniques a box tester can use:

Statement Coverage:- This technique requires every possible statement in the code to be tested at least once during the testing process of software engineering.

Branch Coverage - This technique checks every possible path (if-else and other conditional loops) of a software application.

Apart from above, there are numerous coverage types such as Condition Coverage, Multiple Condition Coverage, Path Coverage, Function Coverage etc. Each technique has its own merits and attempts to test (cover) all parts of software code. Using Statement and Branch coverage you generally attain 80-90% code coverage which is sufficient.

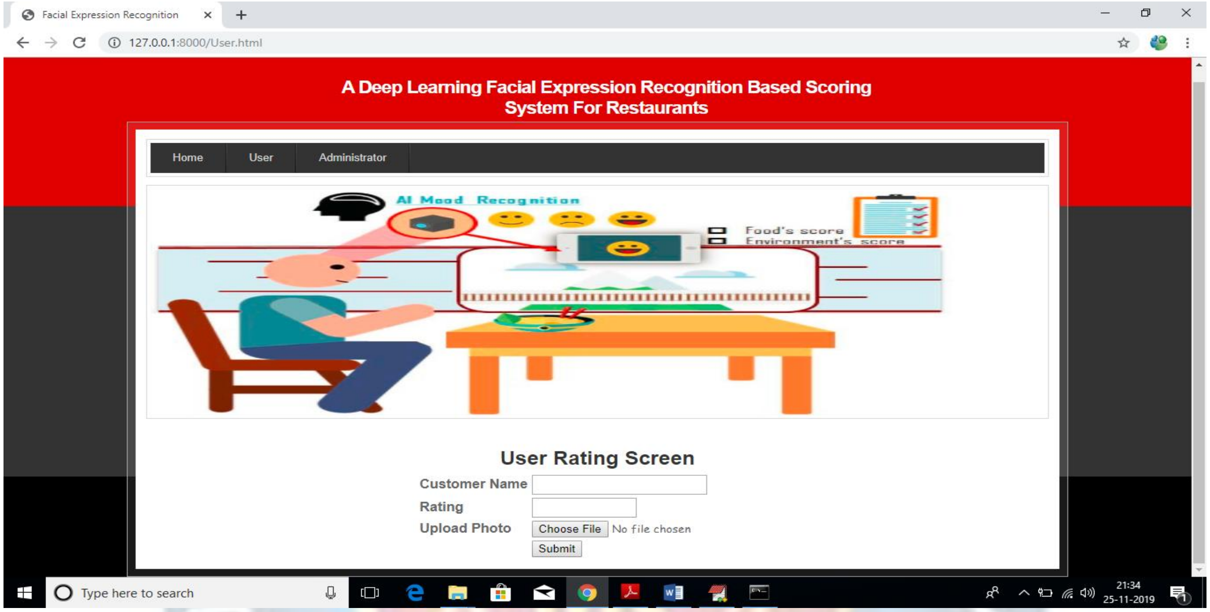
Following are important Whitebox Testing Techniques:

- Statement Coverage
- Decision Coverage
- Branch Coverage

- Condition Coverage
- Multiple Condition Coverage
- Finite State Machine Coverage
- Path Coverage
- Control flow testing
- Data flow testing

8.OUTPUT SCREENS

8.1 Input Screens:-



A Deep Learning Facial Expression Recognition Based Scoring System For Restaurants

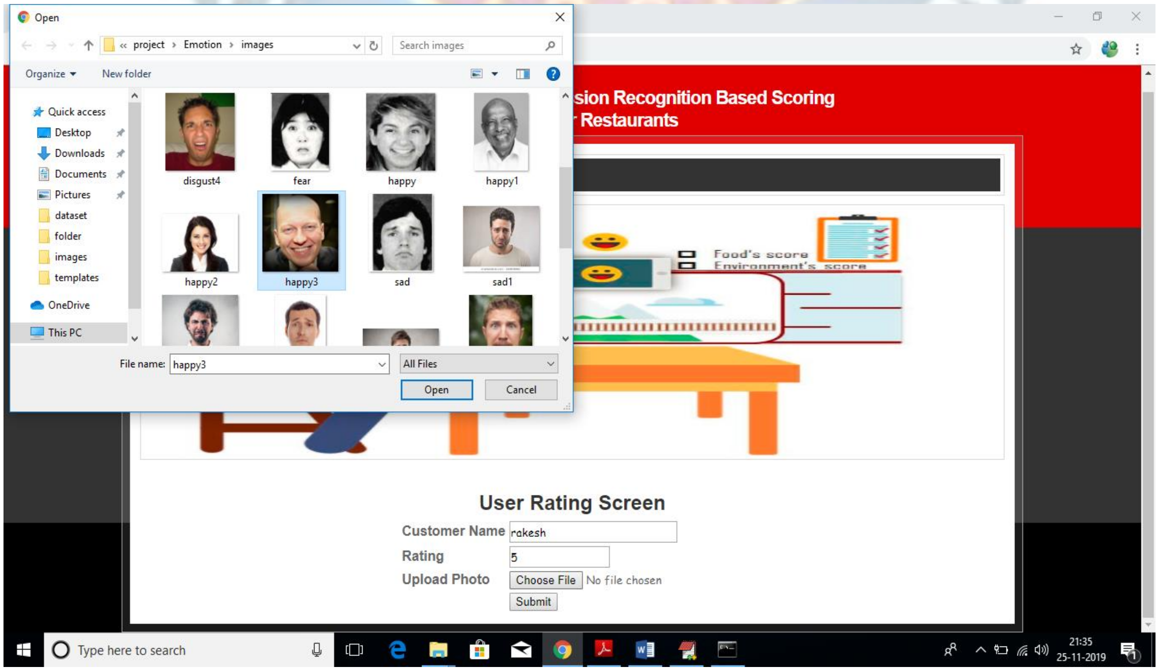
User Rating Screen

Customer Name

Rating

Upload Photo No file chosen

Screenshot 8.1.1 User uploading rating



File name: happy3

…sion Recognition Based Scoring …r Restaurants

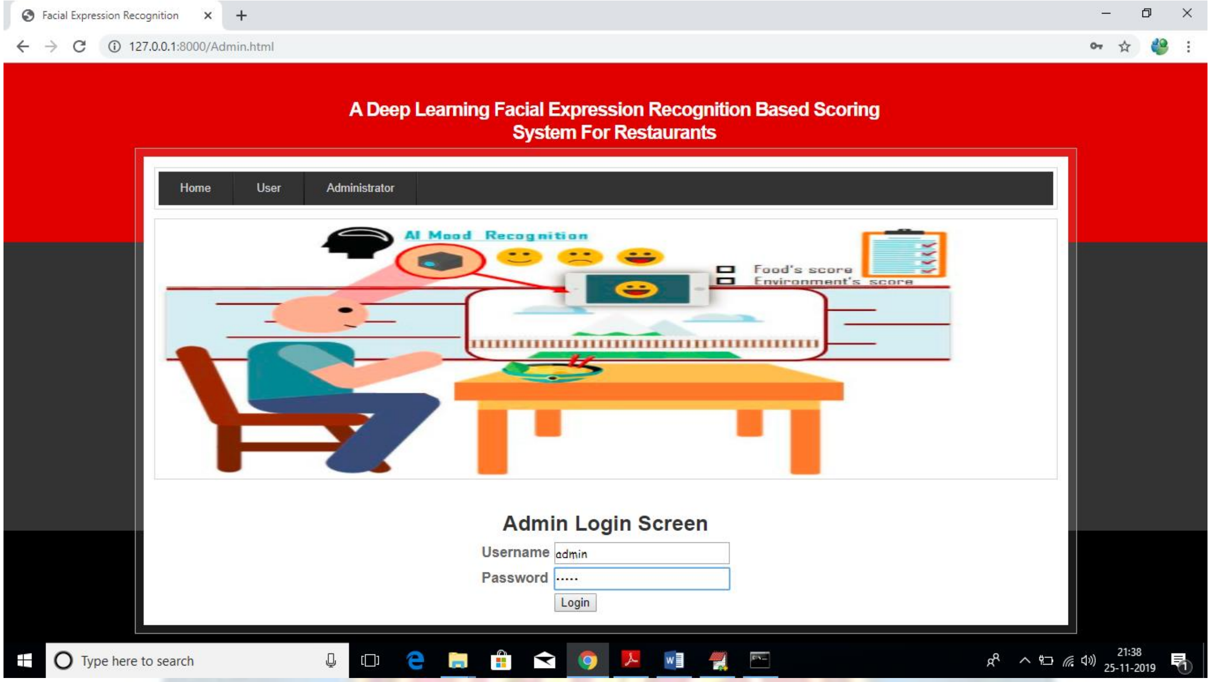
User Rating Screen

Customer Name rakesh

Rating 5

Upload Photo No file chosen

Screenshot 8.1.2 User uploading image



Facial Expression Recognition
127.0.0.1:8000/Admin.html
A Deep Learning Facial Expression Recognition Based Scoring System For Restaurants
Home
User
Administrator
AI Mood Recognition
Food's score
Environment's score
Type here to search
21:38
25-11-2019

Admin Login Screen

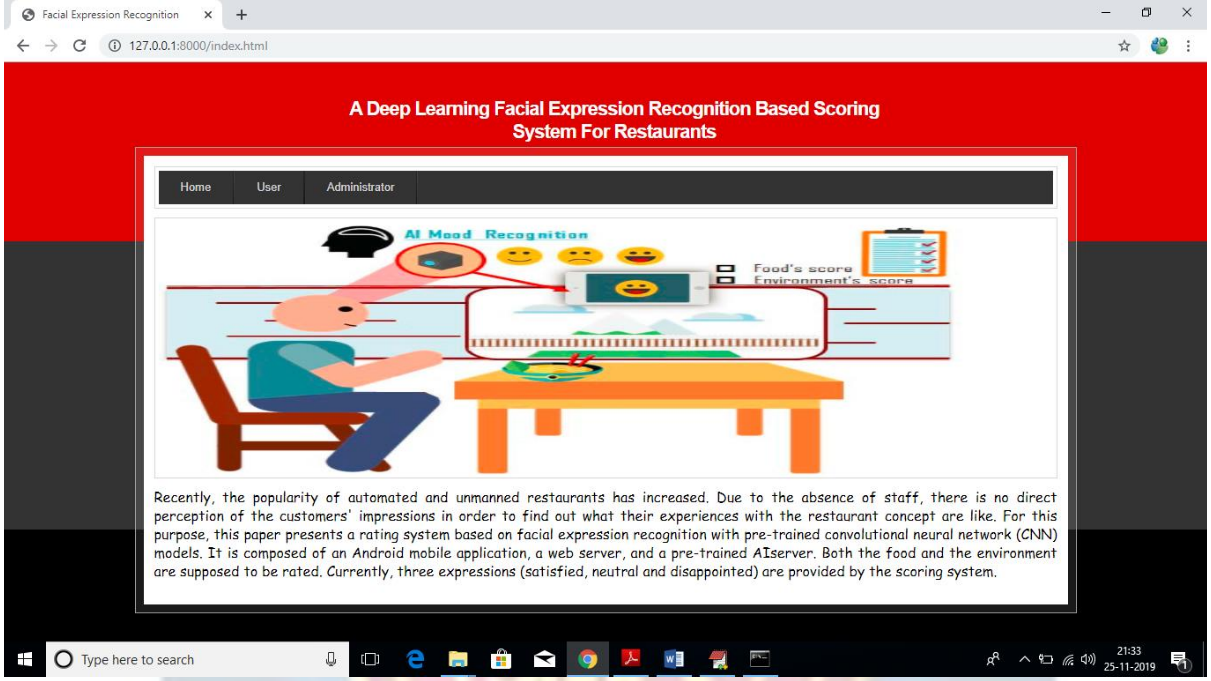
Username admin

Password .....

Login

Screenshot 8.1.3 Admin login

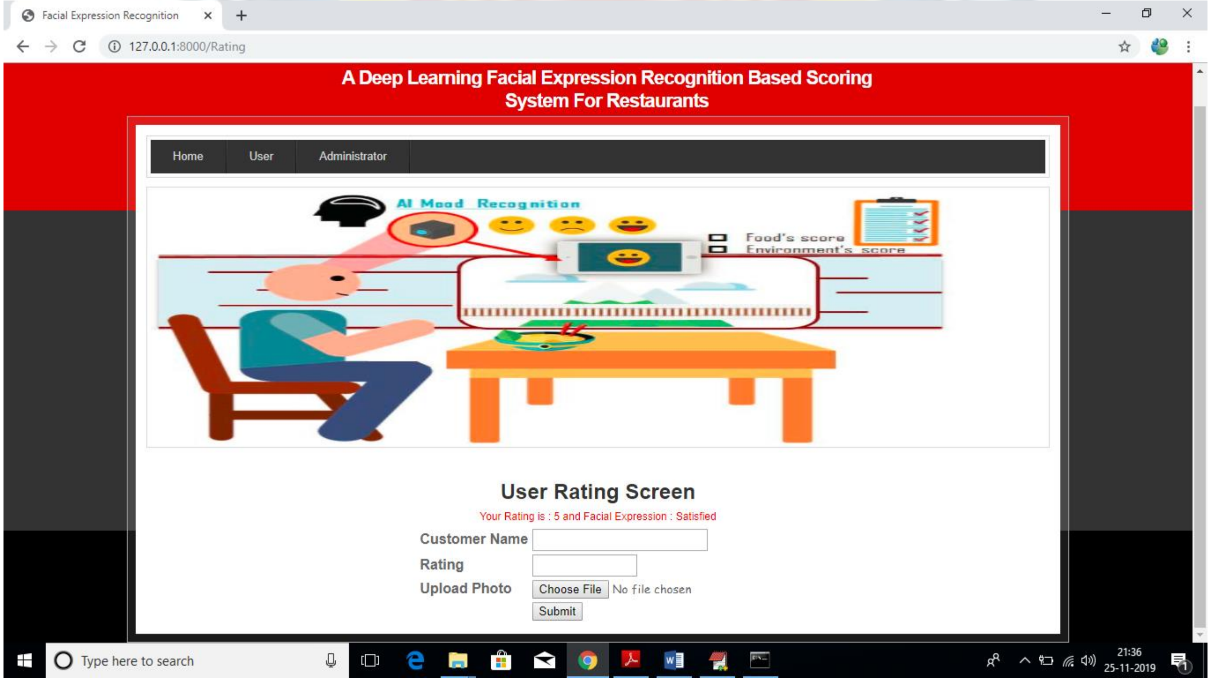
8.2 Output Screens:-



A Deep Learning Facial Expression Recognition Based Scoring System For Restaurants

Recently, the popularity of automated and unmanned restaurants has increased. Due to the absence of staff, there is no direct perception of the customers' impressions in order to find out what their experiences with the restaurant concept are like. For this purpose, this paper presents a rating system based on facial expression recognition with pre-trained convolutional neural network (CNN) models. It is composed of an Android mobile application, a web server, and a pre-trained AIserver. Both the food and the environment are supposed to be rated. Currently, three expressions (satisfied, neutral and disappointed) are provided by the scoring system.

Screenshot 8.2.1 Home Page



A Deep Learning Facial Expression Recognition Based Scoring System For Restaurants

User Rating Screen

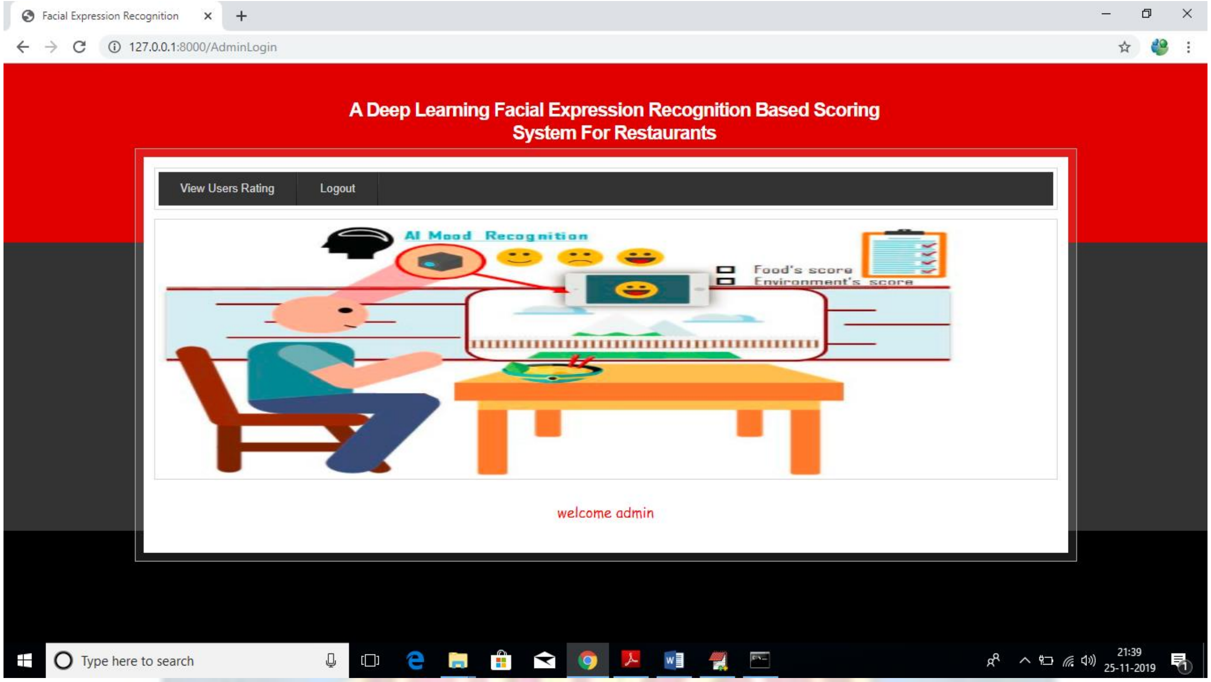
Your Rating is : 5 and Facial Expression : Satisfied

Customer Name

Rating

Upload Photo No file chosen

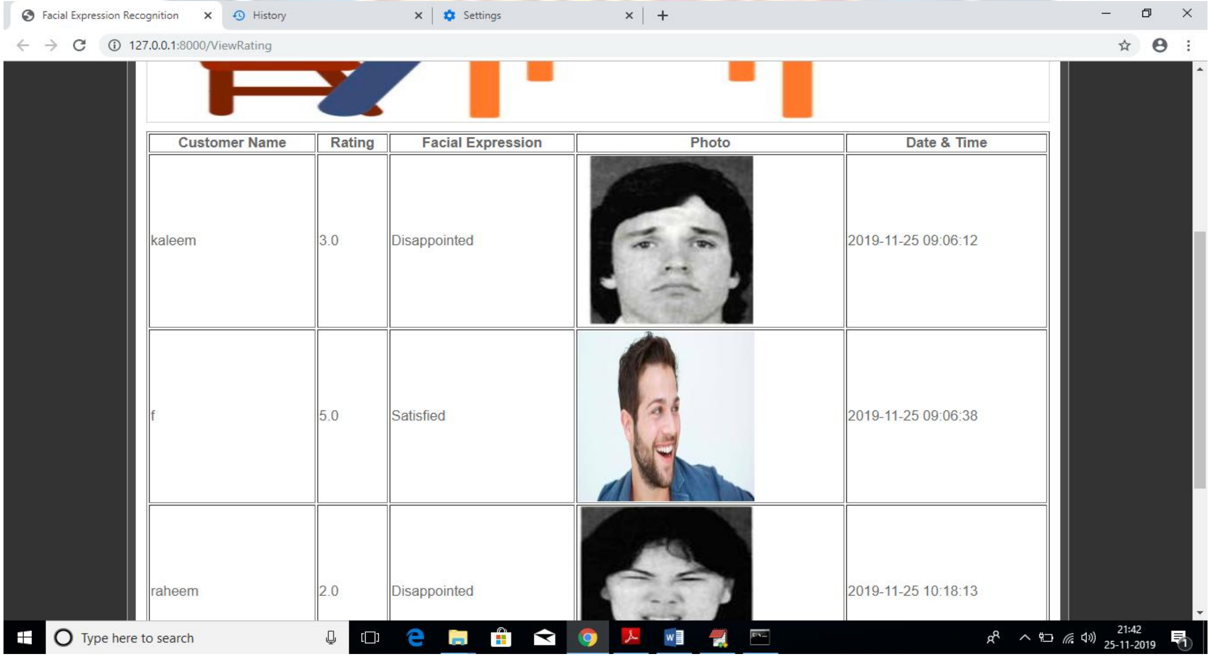
Screenshot 8.2.2 User rating output



A Deep Learning Facial Expression Recognition Based Scoring System For Restaurants

welcome admin

Screenshot 8.2.3 Admin welcome page



Customer Name	Rating	Facial Expression	Photo	Date & Time
kaleem	3.0	Disappointed		2019-11-25 09:06:12
f	5.0	Satisfied		2019-11-25 09:06:38
raheem	2.0	Disappointed		2019-11-25 10:18:13

Screenshot 8.2.4 Admin accessing reviews with emotion label results

9. EXPERIMENTAL RESULTS

9.1 Analyzing the result with the help of input image:

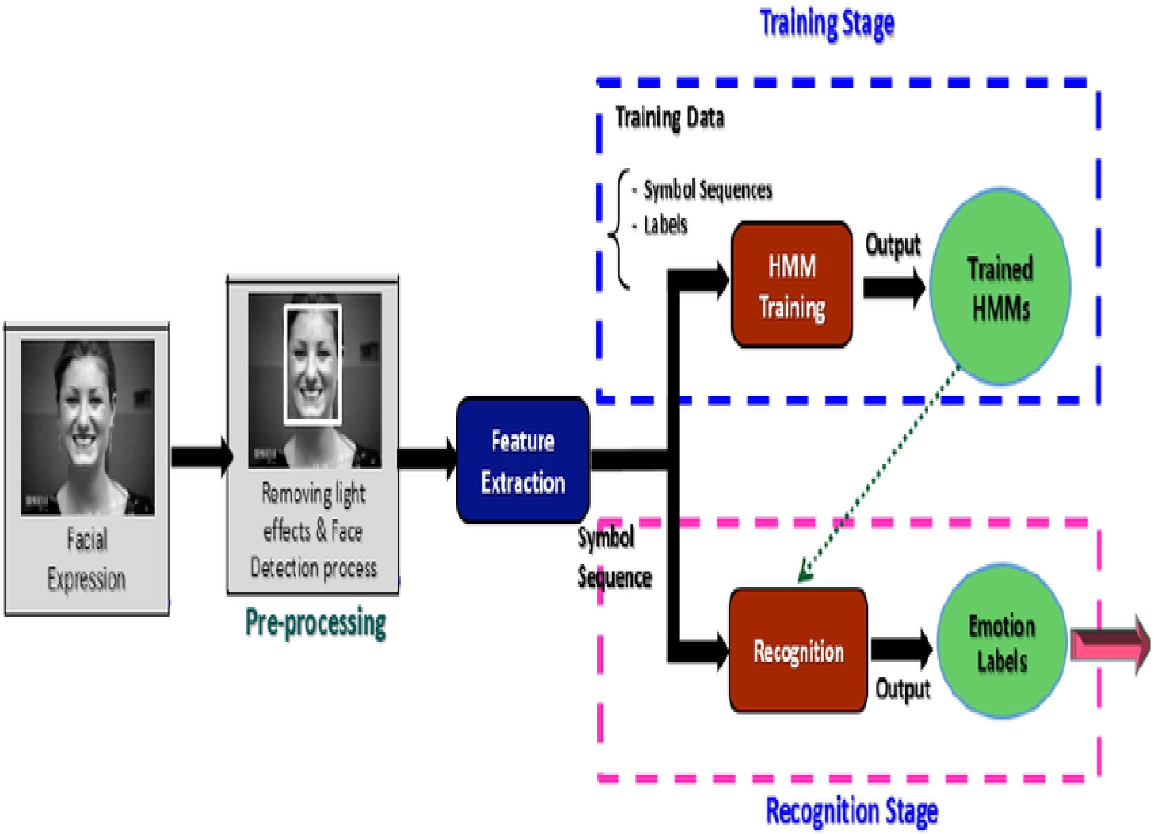


Fig 9.1.1 Processing of input image

From the above flow, we can analyze the input image and predict the emotion label just by viewing the image. The results from the experimental results can thus be said to match with the actual generated results.

10. CONCLUSION AND FUTURE ENHANCEMENT

10.1 Conclusion

We proposed a deep learning facial expression based scoring system for restaurants for the purpose of classification of customer reviews. We filtered input images, performed face detection and finally generated the emotion label using the CNN Model. We achieved an accuracy of 85% with Haar based classifiers.

10.2 Future Enhancement

1. A further development could lead to a system where the customer can rate touch-less in the restaurant. For this, it must be ensured that the accuracy of the facial expression recognition is high enough.

2. It is also an idea to extend the image-based rating system with a speech recognition feature. The customer could express his opinion and impressions verbally or make suggestions for improvement like it is already done with Google ratings.

3. It is planned to extend the system with a web application that will enable the restaurant management to get a quick graphical overview and easy insights into the statistics.

11. REFERENCES

1. Hussain Saeed, Ali Shouman,Malis Elfar,Mostafa Shabka,Shikharesh Majumdar, and Chung Horng-Lung, “Near-field communication sensors and cloud based smart restaurant management system,” in Proceedings of the 2016 IEEattern Recognition(CVPRE 3rd World Forum on Internet of Things(WTIOT),pp.686-691,2016

2. Florian Schroff,Dmitry Kalenichenko,and James Philbin,”FaceNet:a unified embedding for face recognition and clustering” n Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition(CVPR),pp.815-823,2015

3. Andrew G. Howard,Menglong Zhu, Bo Chen,Dmitry Kalenichenko, Weijun Wang, Tobias sWeyand,Macro Andreeto,and Hartwig Adam,”MobileNets:efficient convolutional neural networks for mobile vision applications,”

4. Janne Tommala,Pedram Ghazi,Bishwo Adhikara,and Hekki Huttunen,”Real time system for facial analysis,” in Proceedings of the 7 th European Workshop on Visual Information Processing.

5. Muscles of The Face and Their Functions Facial Expression Recognition (Face recognition Techniques) Part 1 M Anatomy "Muscles of the Face and Their Functions facial Expression Recognition (Face Recognition Techniques) Part 1."

6. FERA 2015-Second Facial Expression Recognition and Analysis challenge - IEEE Conference publication

7. Dr. Chetana Tukkoji et.al “ITM-CLD: Intelligent traffic management to handling cloudlets of the large data”, published in CSOC 2018-Cybernetics and Algorithms in Intelligent Systems, book series AISC- volume 765, 17 May 2018.

A

Project report

On

USE OF ARTIFICIAL NEURAL NETWORKS TO IDENTIFY FAKE PROFILES

Submitted by

Ms. Y. SINDHU	(17K81A1256)
Mr. PEDDI YUVARAJ	(17K81A1242)
Ms. M. HARSHINI NAGA SRI	(17K81A1232)
Ms. T. KEERTHANA	(17K81A1252)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Dr. R. NAGARAJU

PROFESSOR & HOD

DEPARTMENT OF INFORMATION TECHNOLOGY

ST. MARTIN'S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **USE OF ARTIFICIAL NEURAL NETWORKS TO IDENTIFY FAKE PROFILES**, is being submitted by **Y. SINDHU (17K81A1256), PEDDI YUVARAJ (17K81A1242), M. HARSHINI NAGA SRI (17K81A1232), TIKKISETTY KEERTHANA (17K81A1252),** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN Information Technology** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

	Head of the Department
Dr. R. NAGARAJU	Dr. R. NAGARAJU
Department of Information Technology	Department of Information Technology

Internal Examiner

External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of **Information Technology**, session: 2017 – 2021, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, here by declare that work presented in this Project Work entitled USE OF ARTIFICIAL NEURAL NETWORKS TO IDENTIFY FAKE PROFILES is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

Y. SINDHU	**17K81A1256**
PEDDI YUVARAJ	**17K81A1242**
M.HARSHINI NAGA SRI	**17K81A1232**
T. KEERTHANA	**17K81A1252**

LASYA
INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **M.HARSHINI NAGA SRI** WITH ROLL NO.**17K81A1232, P.YUVRAJ** WITH ROLL NO.**17K81A1242**, **T.KEERTHANA** WITH ROLL NO.**17K81A1252, Y.SINDHU** WITH ROLL NO.**17K81A1256**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**USE OF ARTIFICIAL NEURAL NETWORKS TO IDENTIFY FAKE PROFILES**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE

Lasya IT Solutions Pvt. Ltd. ★ Kompally-500014.

ENDEVOUR.

ORUGANTI VENKAT

DIRECTOR

TRAININGS & PLACEMENTS LASYA IT
SOLUTIONS PVT LTD.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin’s Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R. NAGARAJU**, Head of the Department and as well as our project coordinator, Information technology St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Dr. R. NAGARAJU**, Department of Information Technology, St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

Y. SINDHU	**17K81A1256**
PEDDI YUVARAJ	**17K81A1242**
M. HARSHINI NAGA SRI	**17K81A1232**
T. KEERTHANA	**17K81A1252**

TABLE OF CONTENTS

ABSTRACT

we use machine learning, namely an artificial neural network to determine what are the chances that Facebook friend request is authentic 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 is 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.

LIST OF TABLES

LIST OF FIGURES

1. 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. Currently, 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.

1.1 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.

1.2 PROJECT OBJECTIVES

In this paper, 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.

1.3 SCOPE OF THE PROJECT

The project currently takes input fields attributes such as user age, gender, status count, friend count, location, location most of the fields take binary value except the status count, friend_count.so the evaluation criteria depends on these attributes and ANN algorithm.

1.4 ORGANIZATION OF CHAPTERS

1.4.1 INTRODUCTION

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.

1.4.2 LITERATURE SURVEY

There is a tremendous increase in technologies these days. Mobiles are becoming smart. Technology is associated with online social networks which has become a part in every one’s life in making new friends and keeping friends, their interests are known easier. But this increase in networking online makes many problems like faking their profiles, online impersonation having become more and more in present days. Users are fed with more unnecessary knowledge during surfing which are posted by fake users. Research have observed that 20% to 40% profiles in online social networks like Facebook are fake profiles. Thus, this detection of fake profiles in online social networks results into solution using frameworks.

1.4.3 SOFTWARE & HARDWARE REQUIREMENTS

HARDWARE REQUIREMENTS

Processor: Pentium IV or higher
RAM :256 MB
Hard Disk: 40GB (Minimum 512 MB)
Monitor: 14’ Color Monitor

SOFTWARE REQUIREMENTS

Operating System: Windows 7/XP/8
Framework: Django (python framework)
Database: MySQL, MySQL client
Web server: WampServer 2.4
Coding Languages: PYTHON
Designing: HTML, CSS, JavaScript

1.4.4 SOFTWARE DEVELOPMENT ANALYSIS

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.

Proposed System

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

We utilize Microsoft Excel to store old and new fake data profiles. The algor ithm 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.

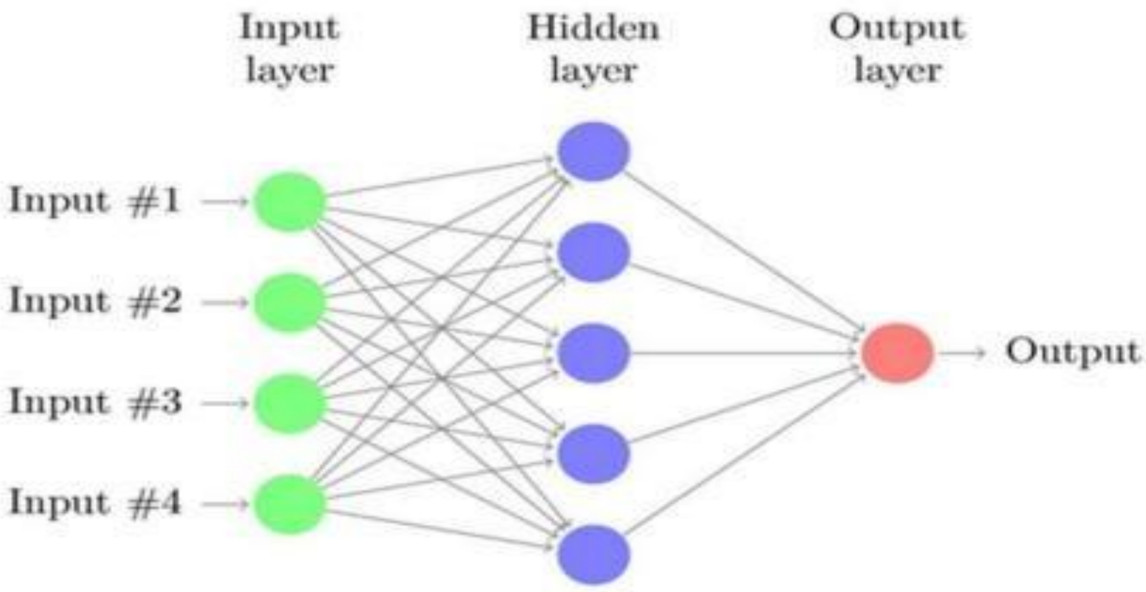
Analysis can do by using feasibility study techniques like

1)Economical Feasibility

2)Technical Feasibility

3)Social Feasibility

1.4.5 PROJECTSYSTEM DESIGN



1.4.6 PROJECT CODING

```
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)
```

1.4.7 PROJECT 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.

1) Black box testing

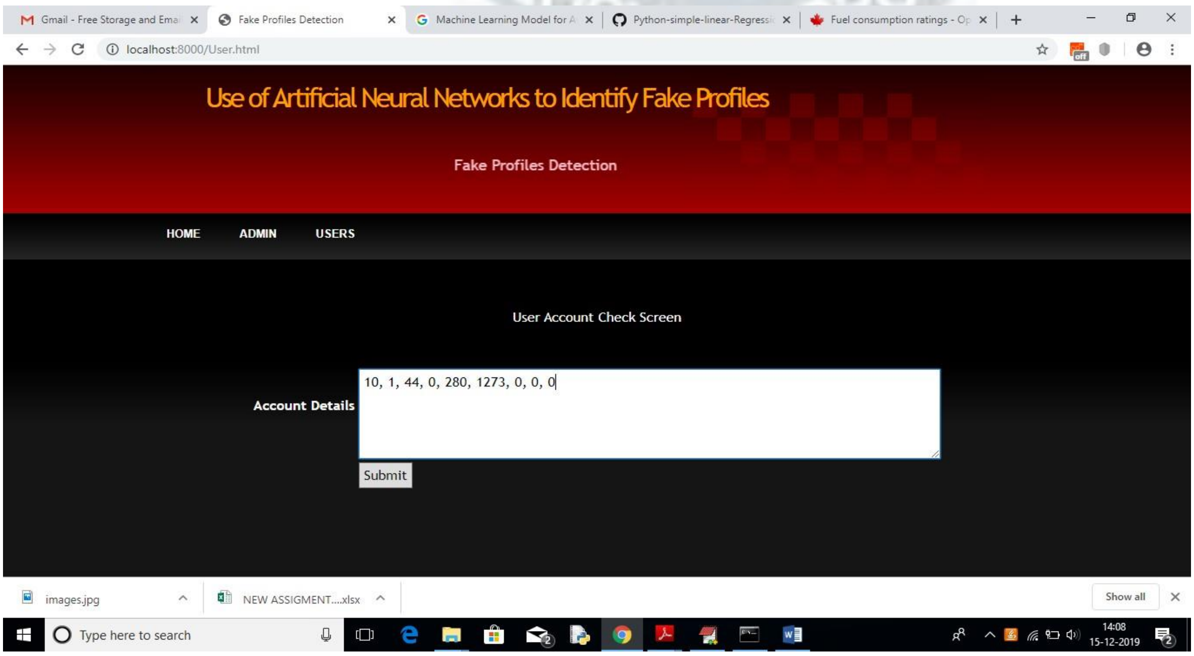
2) White box testing

3) Performance testing

4) Load testing

5) Manual testing

1.4.8 INPUT SCREENS

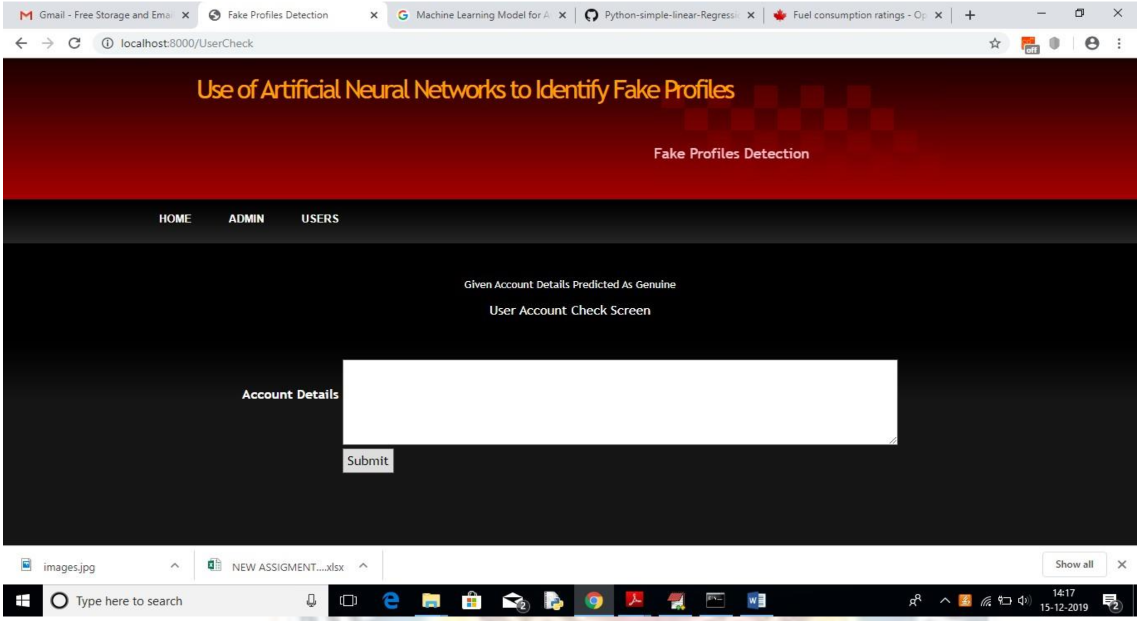

Use of Artificial Neural Networks to Identify Fake Profiles

Fake Profiles Detection

User Account Check Screen

Account Details: 10, 1, 44, 0, 280, 1273, 0, 0, 0

1.4.9 OUTPUT SCREENS



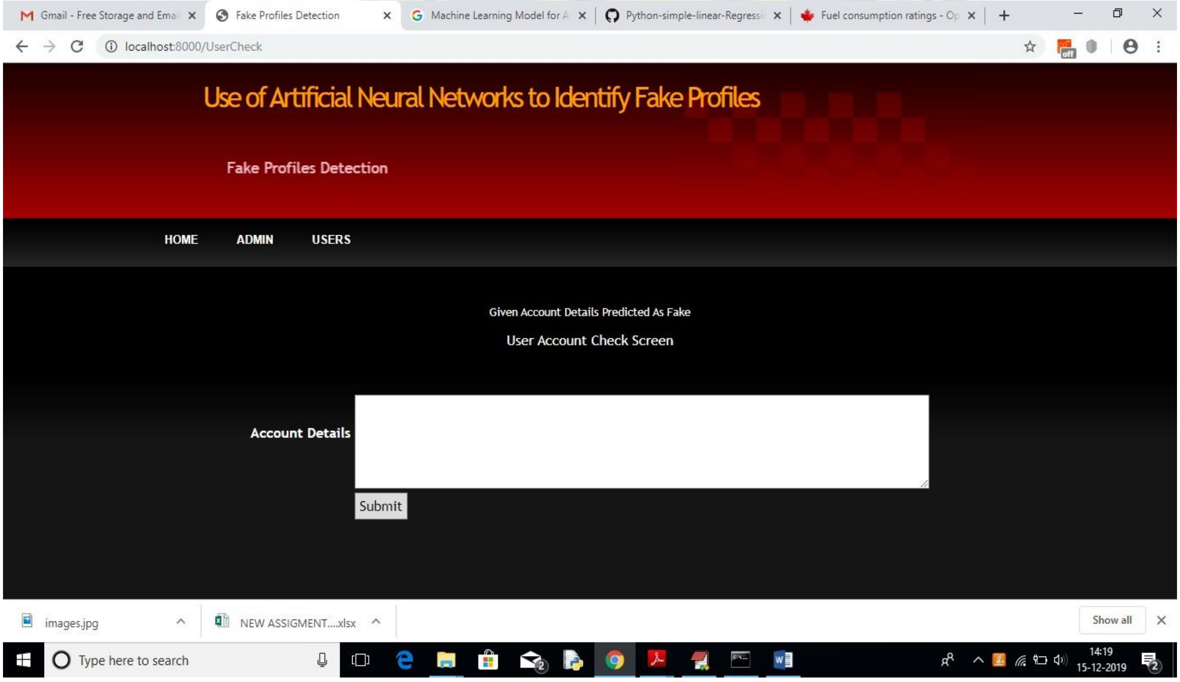
Use of Artificial Neural Networks to Identify Fake Profiles

Fake Profiles Detection

Given Account Details Predicted As Genuine

User Account Check Screen

Account Details



Use of Artificial Neural Networks to Identify Fake Profiles

Fake Profiles Detection

Given Account Details Predicted As Fake

User Account Check Screen

Account Details

2. LITERATURE SURVEY

2.1 SURVEY ON BACKGROUNDS

There is a tremendous increase in technologies these days. Mobiles are becoming smart. Technology is associated with online social networks which has become a part in every one’s life in making new friends and keeping friends, their interests are known easier. But this increase in networking online makes many problems like faking their profiles, online impersonation having become more and more in present days. Users are fed with more unnecessary knowledge during surfing which are posted by fake users. Research have observed that 20% to 40% profiles in online social networks like Facebook are fake profiles. Thus, this detection of fake profiles in online social networks results into solution using frameworks.

In this paper, we use machine learning, namely an artificial neural network to determine what are the chances that Facebook friend request is authentic 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 is 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 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.

2.2 CONCLUSION ONSURVEY

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 func tion. 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 a nd bias.
In this paper, 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.

3. SOFTWARE AND HARDWARE REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS

Operating System: Windows 7/XP/8

Framework: Django (python framework)

Database: MySQL, MySQL client

Web server: WampServer 2.4

Coding Languages: PYTHON

Designing: HTML, CSS, JavaScript

3.1.1 Python

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 of 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.

What is Python?

Python is a popular programming language. It was created by Guido van Rossum, 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.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

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 Quickstart

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.

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!
```

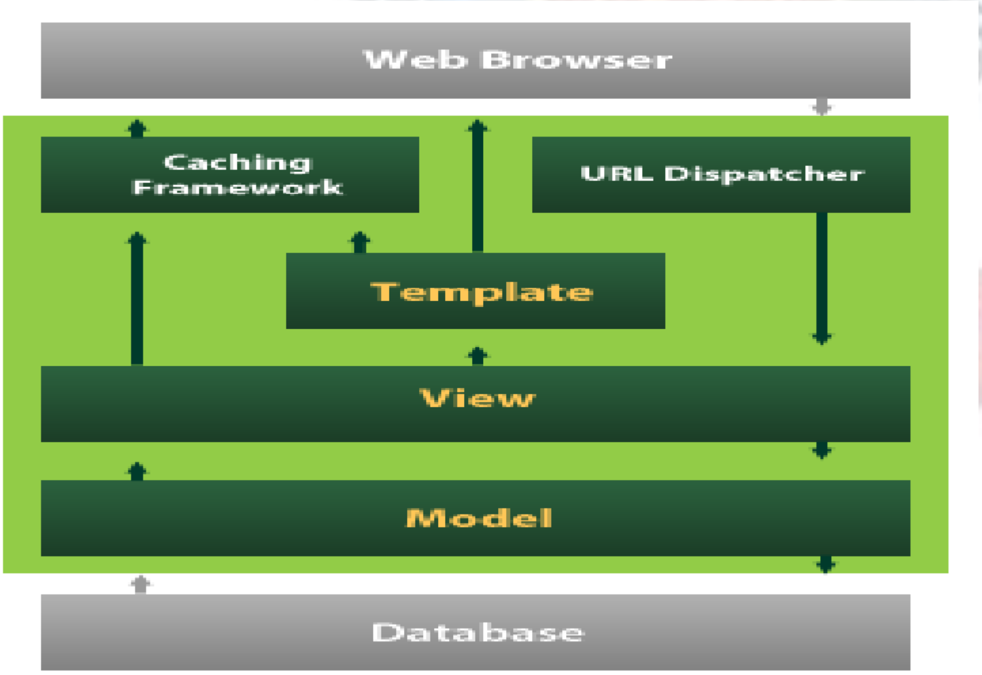
Whenever you are done in the python command line, you can simply type the following to quit the python command line interface:

```
exit()
```

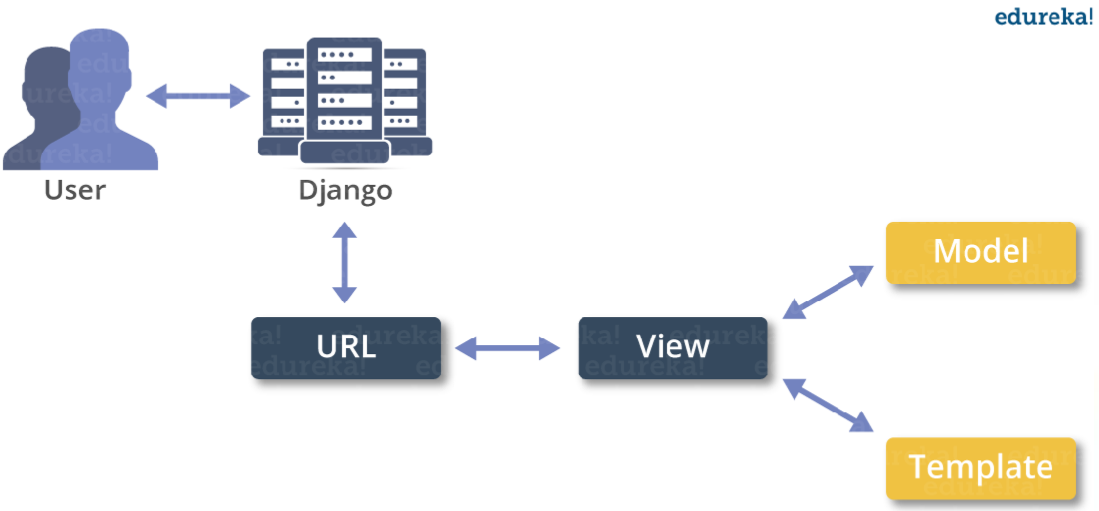
3.1.2 Django

Django is a high- level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source.

Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and "pluggability" of components, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings files and data models.



Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models



3.1.3 Purpose

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was important to keep the navigations from one screen to the other well-ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

3.1.4 Functional Requirements

- Graphical User interface with the User.

3.1.5 Non-Functional Requirements

- **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.
- **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.

- **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.
- **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.
- **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.
- **Powe r Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery
  life. However, the allowable wattage can't be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

3.1.6 Input & Output Design

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 into 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.

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 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.

3.2 HARDWARE REQUIREMENTS

Processor: Pentium IV or higher

RAM :256 MB

Hard Disk: 40 GB (Minimum 512 MB)

Monitor: 14’ Color Monitor

4. SOFTWARE DEVELOPMENT ANALYSIS

4.1 OVERVIEW OF THE PROJECT

4.1.1 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.

4.2 DEFINE THE PROBLEM

4.2.1 Proposed System

In our solution, we use machine learning, namely an artificial neural network to determine what are the chances that a friend request is authentic 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.

4.2.2 Advantages of Proposed system

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 a ccounts that were already compromised being sold.

4.2.3 Algorithms

4.2.3.1 CNN:

To demonstrate how to build a convolutional neural network-based image classifier, we shall build a 6-layer neural network that will identify and separate one image from other. This network that we shall build is a very small network that we can run on a CPU as well. Traditional neural networks that are very good at doing image classification have many more parameters and take a lot of time if trained on normal CPU. However, our objective is to show how to build a real-world convolutional neural network using TENSORFLOW.

Neural Networks are essentially mathematical models to solve an optimization problem. They are made of neurons, the basic computation unit of neural networks. A neuron takes an input (say x), do some comp utation on it (say: multiply it with a variable w and adds another variable b) to produce a value (say; z= wx + b). This value is passed to a non- linear function called activation function (f) to produce the

final output (activation) of a neuron. There are many kinds of activation functions. One of the popular activation functions is Sigmoid. The neuron which uses sigmoid function as an activation function will be called sigmoid neuron. Depending on the activation functions, neurons are named and there are many kinds of them like RELU, TanH.

Random Forest Classification Technique:

This classifier classifies collection of decision trees to subset of randomly generated training set. Then it augments the likes from decision sub trees to know subclass of handling object for tests. Random forest will generate NA missing

values for attributes increase accuracy for larger sets of data. If more number of tress, it does not allow to tree to fit model.

4.2.4 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

4.2.4.1 Economic 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.

4.2.4.2 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 a modest requirement, as only minimal or null.

changes are required for implementing this system.

4.2.4.3 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.

4.3 MODULES OVERVIEW

we use machine learning, namely an artificial neural network to determine what are the chances that Facebook friend request is authentic 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.

4.4 DEFINE THE MODULES

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

a) **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.

b) **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.

4.5 MODULES FUNCTIONALITY

4.5.1 Face Detection:

Face detection or localization is an important step for image classification since only the principal component of face such as nose, eyes, mouth are needed for classification. Face detection algorithms can be broadly classified into feature, knowledge, template, and appearance-based methods. Our proposed system uses Viola Jones object detection algorithm for face localization which comes under feature-based classification. Viola Jones object detection algorithm uses Haar feature based cascade classifiers. The Haar Cascade classifier is extremely important element of the face detection. The presence of the features in any of the input image is determined by the Haar features.

4.5.2 Facial Expression Recognition classification:

After learning the deep features, the final step of FER (Facial Expression Recognition) is to classify the given face into one of the basic emotion categories. Unlike the traditional methods, where the feature extract ion step and the feature classification step are independent, deep networks can perform FER in an end-to-end way. Specifically, a loss layer is added to the end of the network to regulate the back-propagation error; then, the prediction probability of each sample can be directly output by the network. In CNN, SoftMax loss is the most common used function that minimizes the cross-entropy between the estimated class probabilities and the ground truth distribution.

4.5.3 Convolutional neural network (CNN):

CNN has been extensively used in diverse computer vision applications, including FER. At the beginning of the 21st century, several studies in the FER literature found that the CNN is robust to face location changes and scale variations and behaves better than the multi- layer perceptron (MLP) in the case of previously unseen face pose variations, employed the CNN to address the problems of subject independence as well as translation, rotation, and scale invariance in the recognition of facial expressions.

5. PROJECT SYSTEM DESIGN

5.1 DATADESIGN

5.1.1 Databases SQLite

Name
Use of Artificial

Table 5.1.1 SQLite Database

5.1.2 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 5.1.2 List of Database Table

5.2 SYSTEM ARCHITECTURE

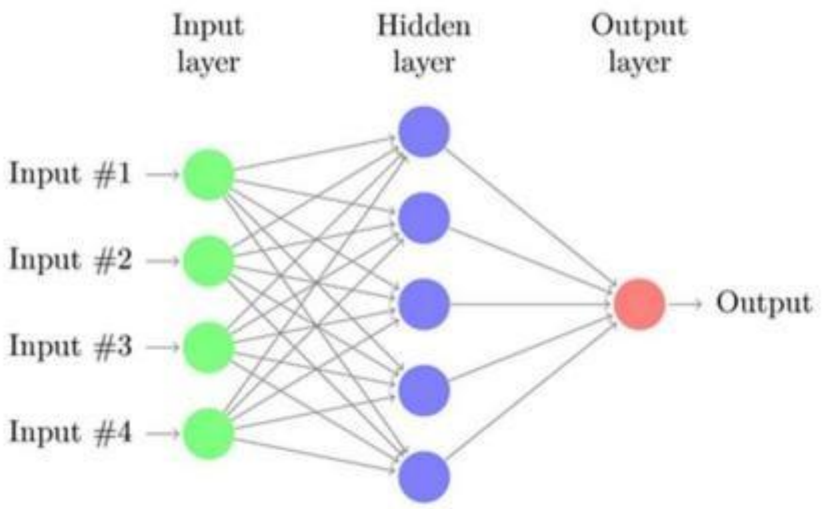


Figure: Neural Network

5.3 UML Diagrams

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta- model and a notation. In the future, some form of method or process may also be added to; or associated with, UML. The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modelling language.
- Encourage the growth of OO tools market.
- Support higher level development concepts such as collaborations, frameworks, patterns, and components.
- Integrate best practices.

5.3.1 CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

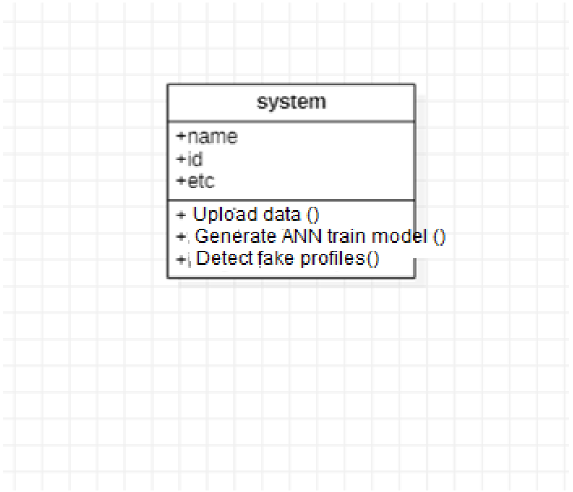


Figure: Class Diagram

5.3.2 USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

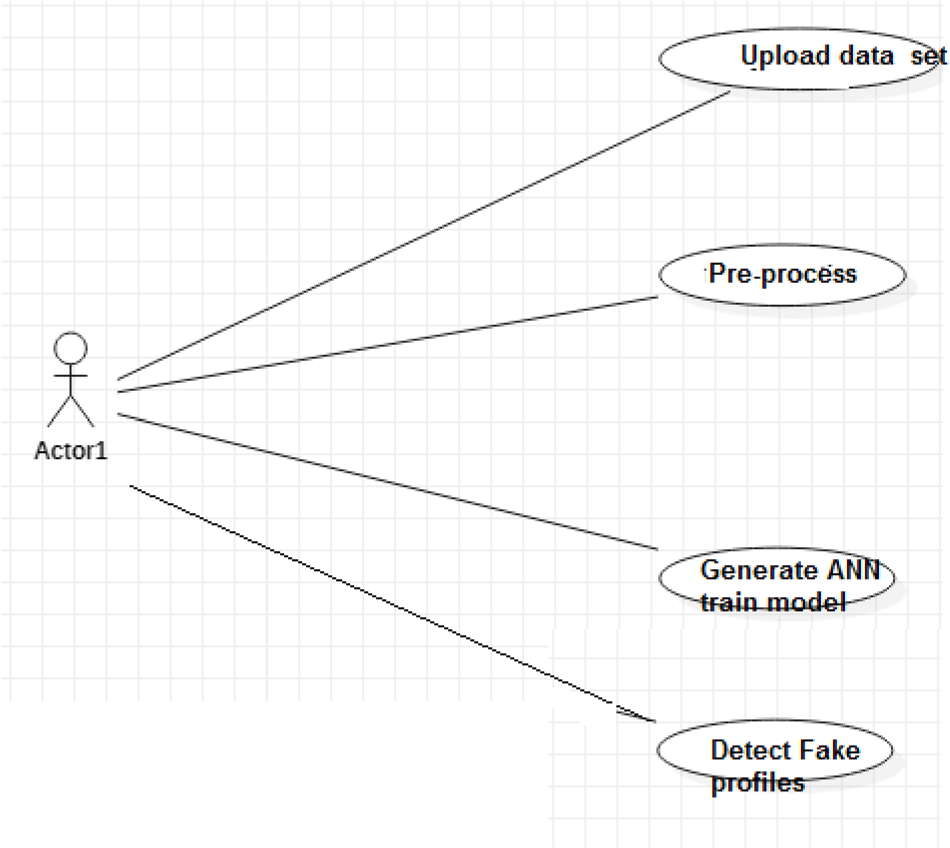


Figure: Use Case Diagram

5.3.3 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

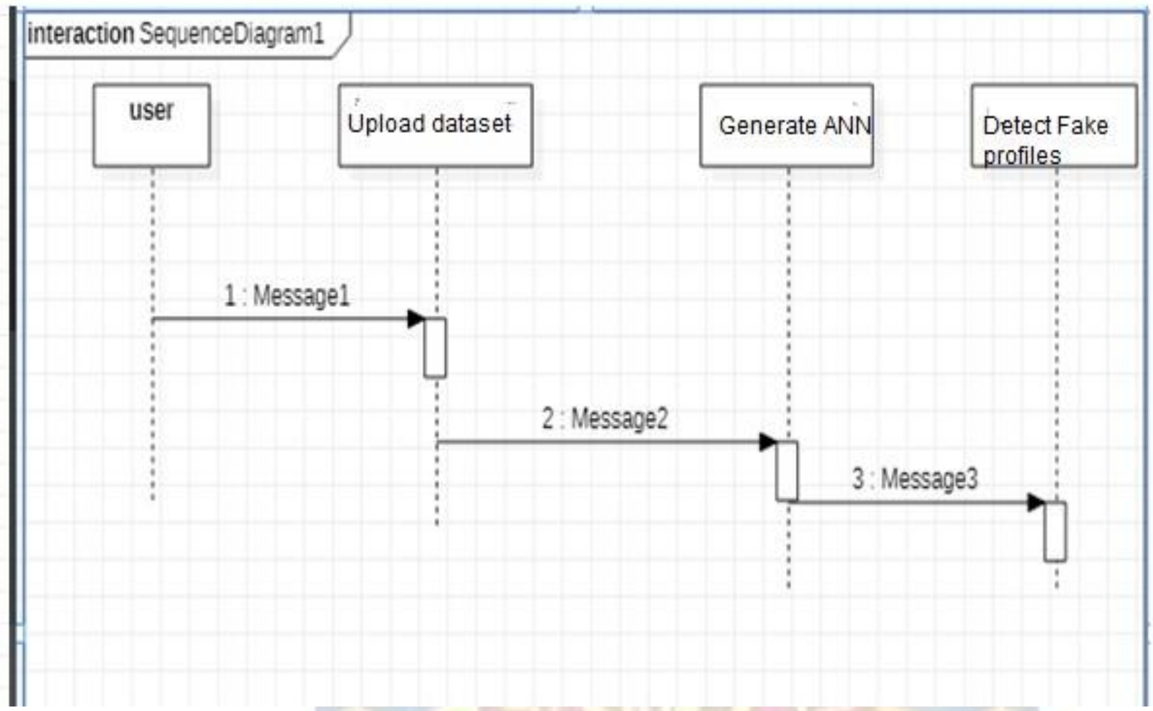


Figure: Sequence Diagram

5.3.4 Activity diagrams

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration, and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores. [citation needed]Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores. [citation needed]

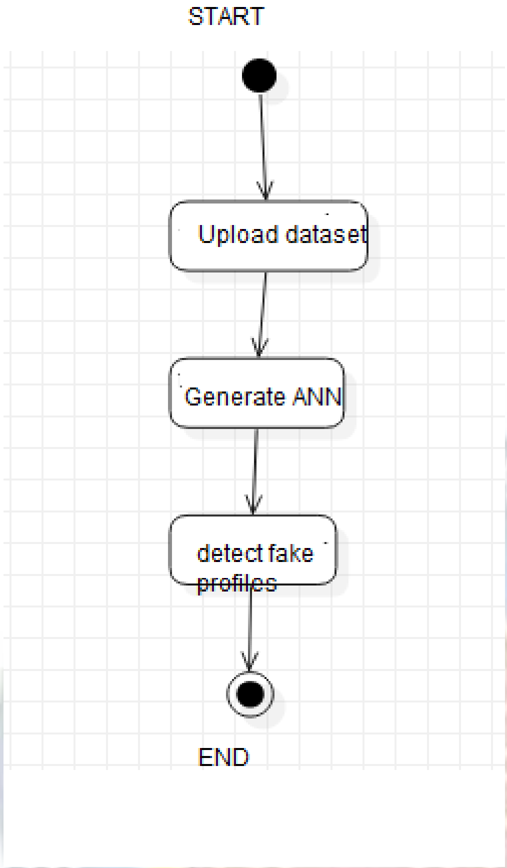


Figure: Activity Diagram

5.3.5 Component Diagram

The component diagram extends the information given in a component notation element. One way of illustrating the provided and required interfaces by the specified component is in the form of a rectangular compartment attached to the component element. Another accepted way of presenting the interfaces is to use the ball-and-socket graphic convention. A provided dependency from a component to an interface is illustrated with a solid line to the component using the interface from a "lollipop", or ball, labelled with the name of the interface. A required usage dependency from a component to an interface is illustrated by a half-circle, or socket, labelled with the name of the interface, attached by a solid line to the component that requires this interface. Inherited interfaces may be shown with a lollipop, preceding the name label with a caret symbol. To illustrate dependencies between the two, use a solid line with a plain arrowhead joining the socket to the lollipop.

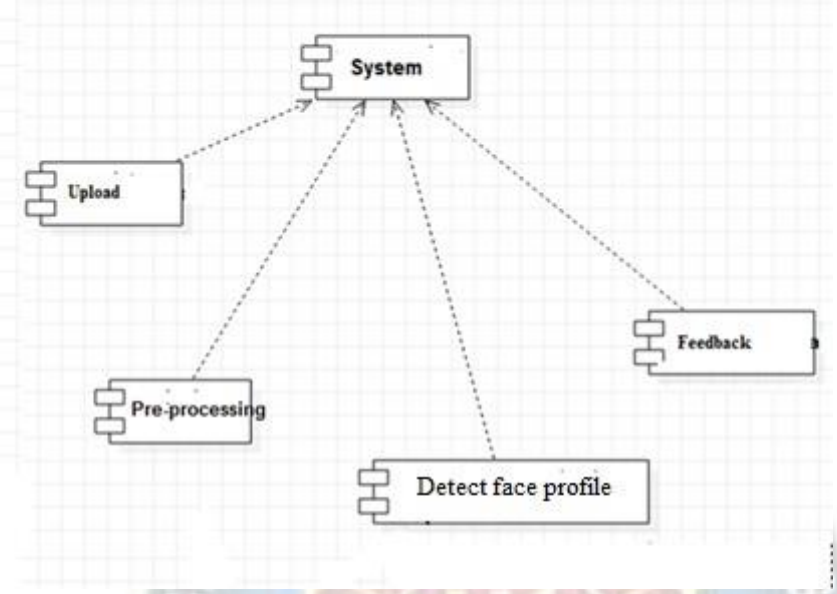


Figure: Component Diagram

5.3.6 Deployment diagram

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.[1] To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

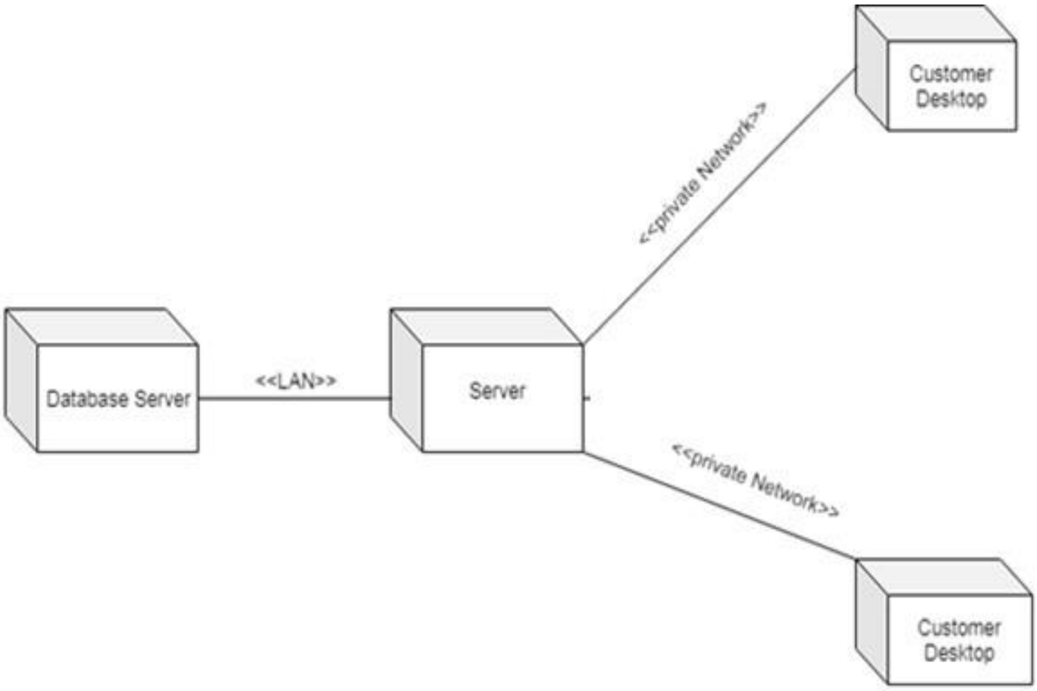


Figure: Deployment Diagram

5.3.7 Object Diagram

An object diagram is a UML structural diagram that shows the instances of the classifiers in models. Object diagrams use notation that is similar to that used in class diagrams. Class diagrams show the actual classifiers and their relationships in a system.

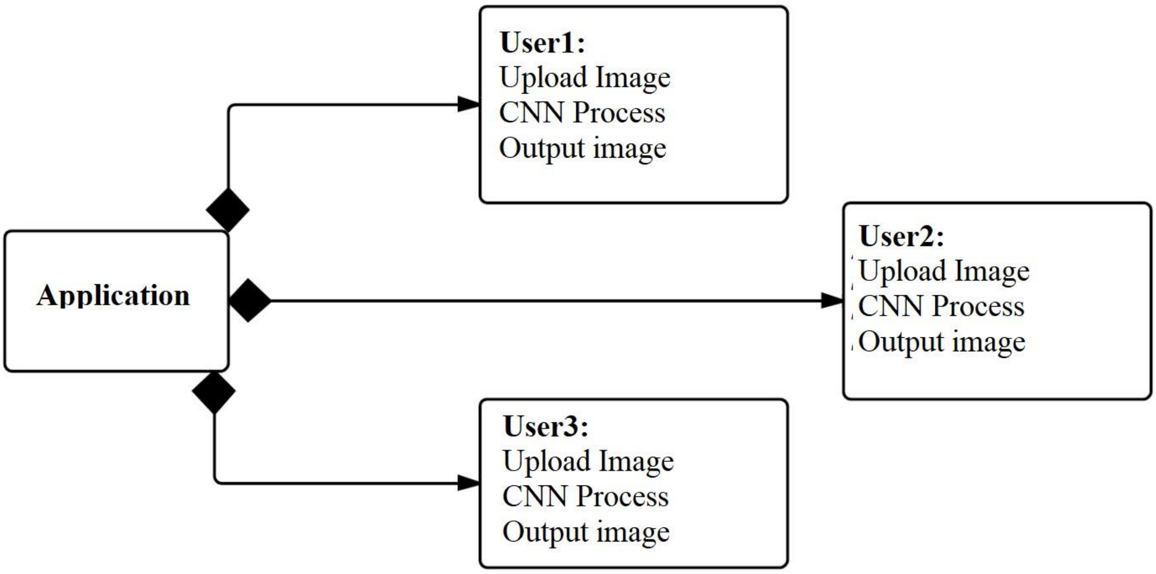


Figure: Object Diagram

5.3.8 Package Diagram

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.

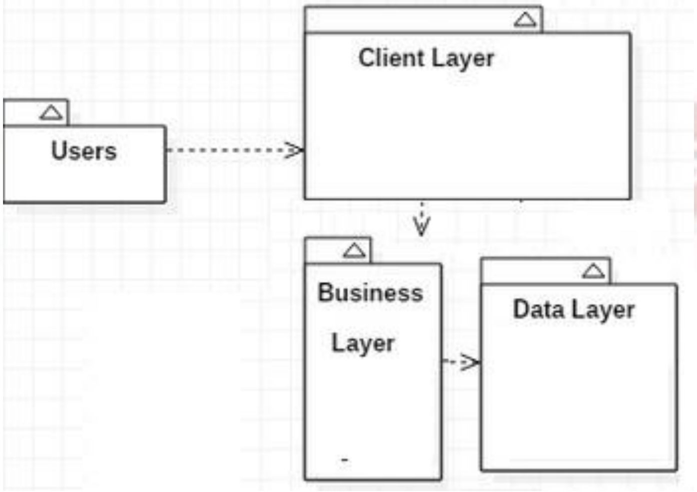


Figure: Package Diagram

5.3.9 Profile Diagram

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connecto rs, and their components.

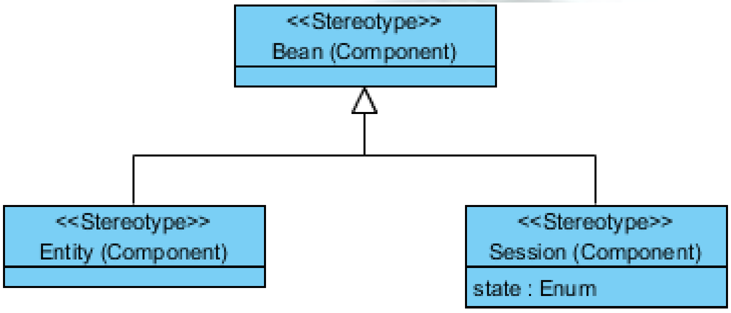


Figure: Profile Diagram

6. PROJECT CODING

6.1 CODE TEMPLATES:

Views.py

```
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

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()
    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,Location_IP\n';
      input+=data+"\n"
      f = open("C:/FakeProfile/Profile/dataset/test.txt", "w")
      f.write(input)
      f.c lose()
      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)
    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)
```

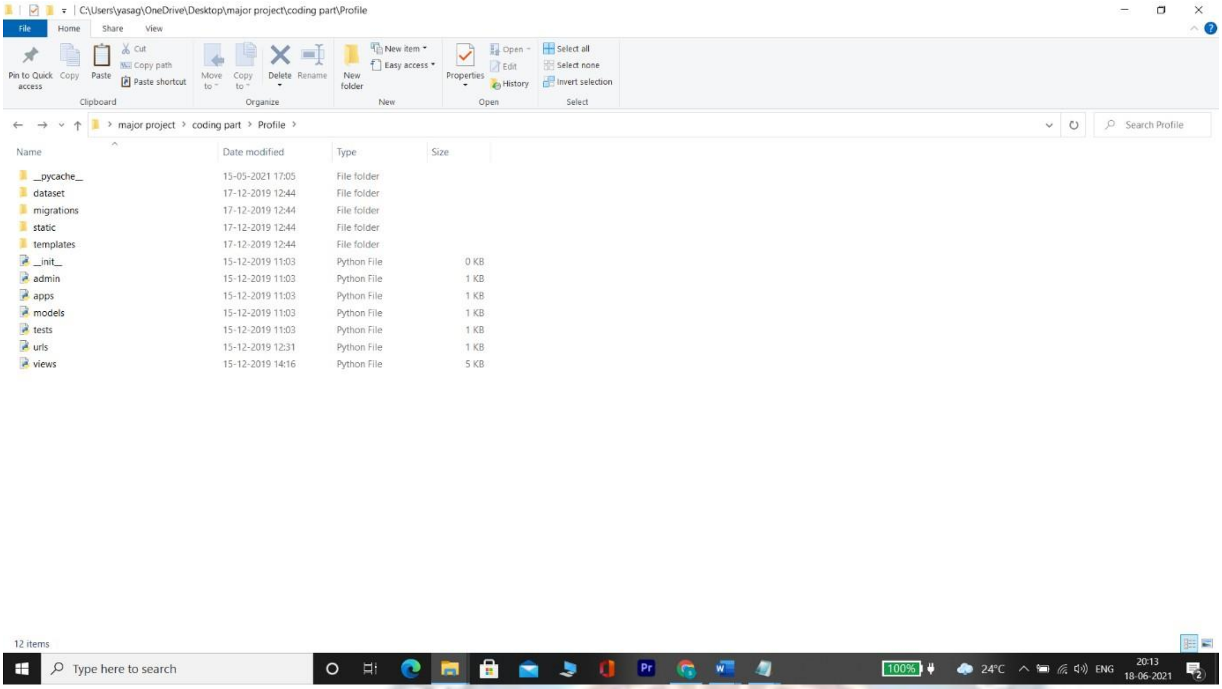
6.2 Outline of various Files

1. urls.py:

This is the basic file which takes all kinds of urls for our project that contains the links for all the flow of modules .Their synchronization and links to particular html pages work.

2. Views.py

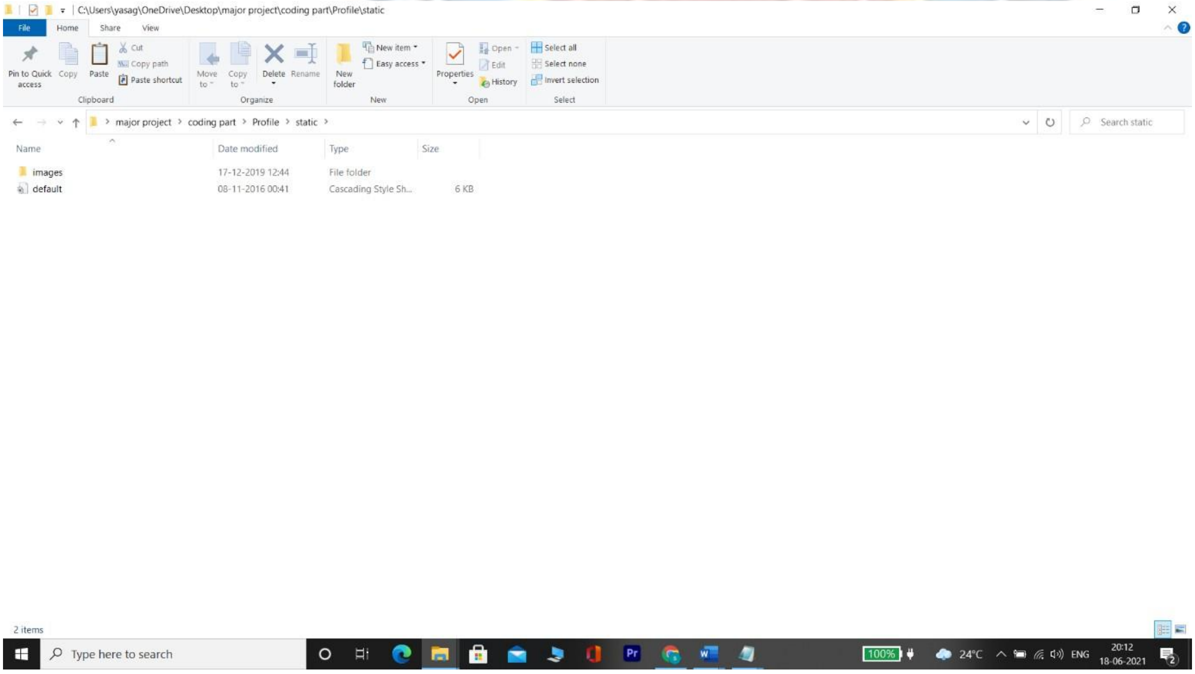
All the manipulations of the api calls is done here with the respective methods. The index method is rendering the information to index.html page and it is not passing any arguments. The request method type is 'GET'. The User Method takes the similar functionality of index and it renders to user.html page ,Similarly the admin function takes care of redirecting to the admin.html. The major validation lies in the AdminLogin method which takes the form from admin.html page with attributes such as username and password and validates against set username=`admin' and password=`admin' Once this criteria is able to meet then we move to adminscreen.html else we will still be in th admin.html page. UserCheck method will work on taking the input using a textfield in the front-end these information will be written down in write mode to test.txt file and this test.txt file is used to feed input to the ANN algorithm and according to criteria set the input is validated and judged as fake or genuine profile. GenerateModel this is basically doing the loading process of our trained dataset to the dataset.txt and we its loading accuracy and other things backend like which dataset is loading currently etc kind of information. ViewTrain does the functionality of showing our trained dataset to the admin in the format of table with columns as our attributes and rows as our tuples of datasets.



Name	Date modified	Type	Size
__pycache__	15-05-2021 17:05	File folder	
dataset	17-12-2019 12:44	File folder	
migrations	17-12-2019 12:44	File folder	
static	17-12-2019 12:44	File folder	
templates	17-12-2019 12:44	File folder	
__init__	15-12-2019 11:03	Python File	0 KB
admin	15-12-2019 11:03	Python File	1 KB
apps	15-12-2019 11:03	Python File	1 KB
models	15-12-2019 11:03	Python File	1 KB
tests	15-12-2019 11:03	Python File	1 KB
urls	15-12-2019 12:31	Python File	1 KB
views	15-12-2019 14:16	Python File	5 KB

3. Static Files

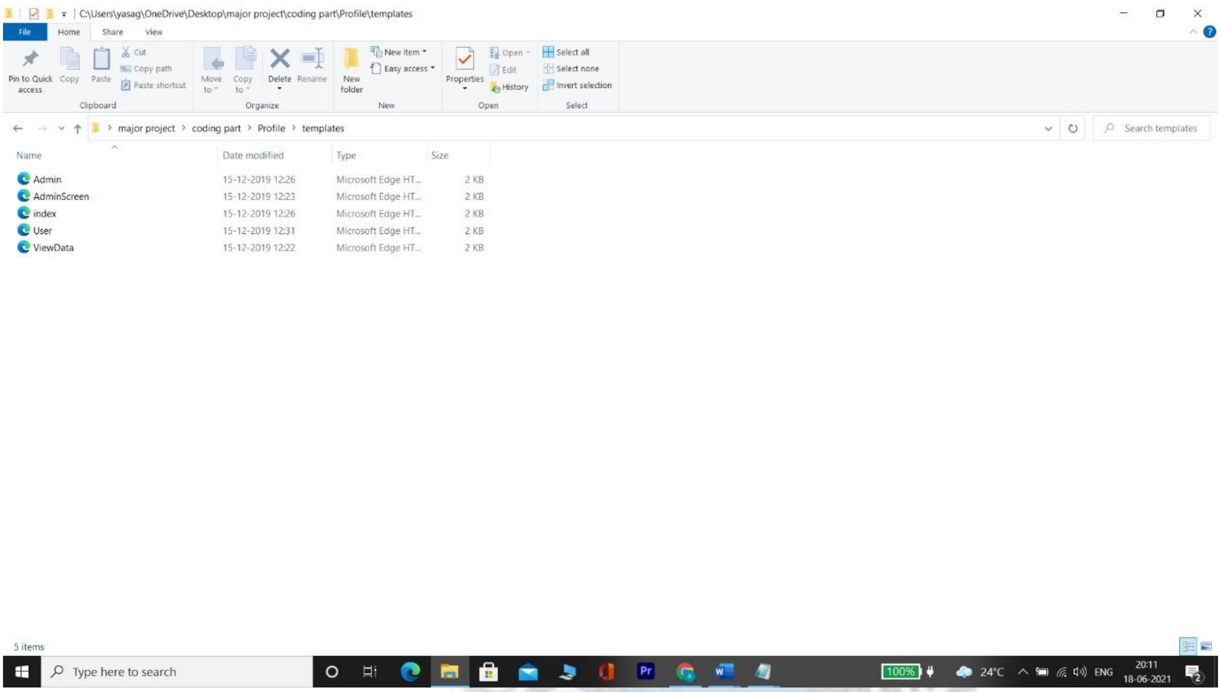
All the images are by default kept in static folder by developers it is the best practice because Django searches for resources in static folders.



Name	Date modified	Type	Size
images	17-12-2019 12:44	File folder	
default	08-11-2016 00:41	Cascading Style Sh...	6 KB

4. Templates:

All the html pages respective css files are arranged here and it makes it easy for rendering purpose and Django follows MVT architecture which means T stands for template in this context.


C:\Users\yasag\OneDrive\Desktop\major project\coding part\Profile\templates

Name	Date modified	Type	Size
Admin	15-12-2019 12:26	Microsoft Edge HT...	2 KB
AdminScreen	15-12-2019 12:23	Microsoft Edge HT...	2 KB
index	15-12-2019 12:26	Microsoft Edge HT...	2 KB
User	15-12-2019 12:31	Microsoft Edge HT...	2 KB
ViewData	15-12-2019 12:22	Microsoft Edge HT...	2 KB

7. PROJECT 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 tests. Each test type addresses a specific testing requirement.

7.1. VARIOUS TEST CASES

7.1.1 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.

7.1.2 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.

7.1.3 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.

7.1.4 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.

7.1.5 Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

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.

7.1.6 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

7.1.6.1 Test Results

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

7.1.7 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.

7.1.7.1 Test Results

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

Test cases

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 1 Test Case for Empty Login Fields

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 signups need to provide all data.
Output	Signup Fails and an alert message appears asking to enter valid email and name.

Table 6:3 Test Case for Signup fail

7.2 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.

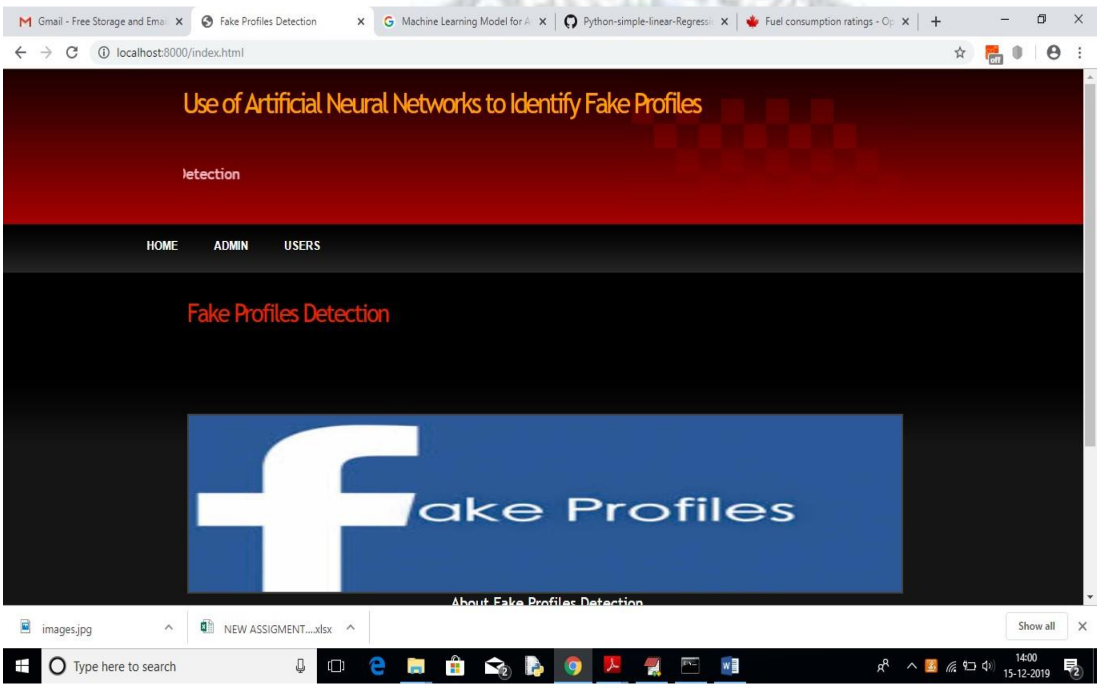
7.3 WHITE BOX TESTING

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.

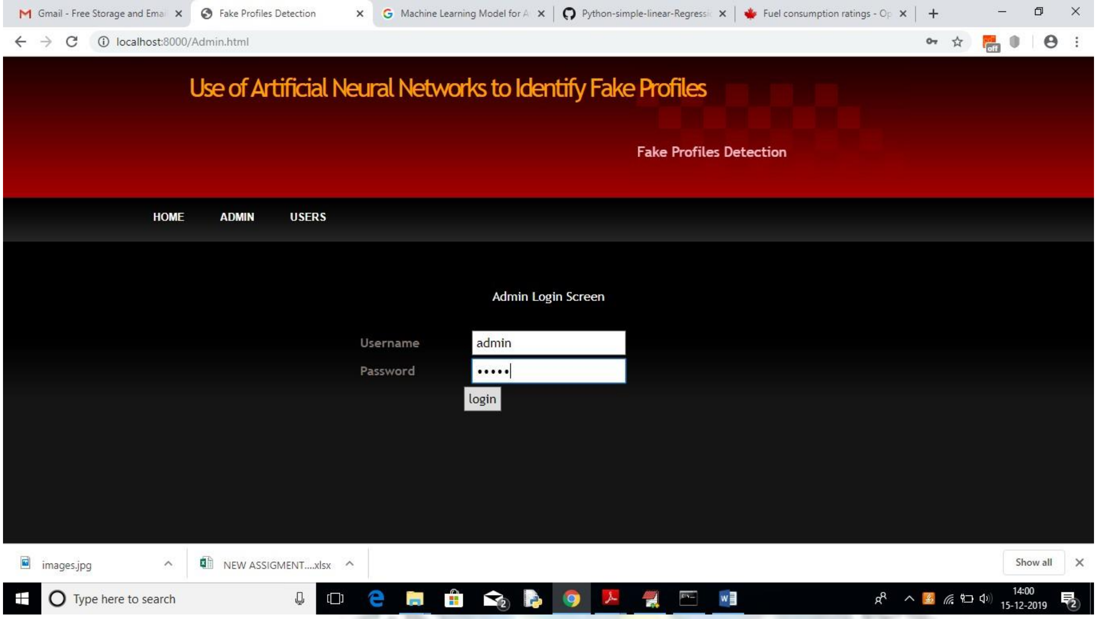
8. OUTPUT SCREENS

8.1 USER INTERFACES

Deploy this application on DJANGO server and then run- in browser enter URL as ‘http://localhost:8000/index.html’ to get below screen


Gmail - Free Storage and Emai
Fake Profiles Detection
Machine Learning Model for A
Python-simple-linear-Regressi
Fuel consumption ratings - Op
localhost:8000/index.html
Use of Artificial Neural Networks to Identify Fake Profiles
Detection
HOME
ADMIN
USERS
Fake Profiles Detection
ake Profiles
About Fake Profiles Detection
images.jpg
NEW ASSIGMENT....xlsx
Show all
Type here to search
14:00
15-12-2019

In above screen click on ‘ADMIN’ link to get below login screen

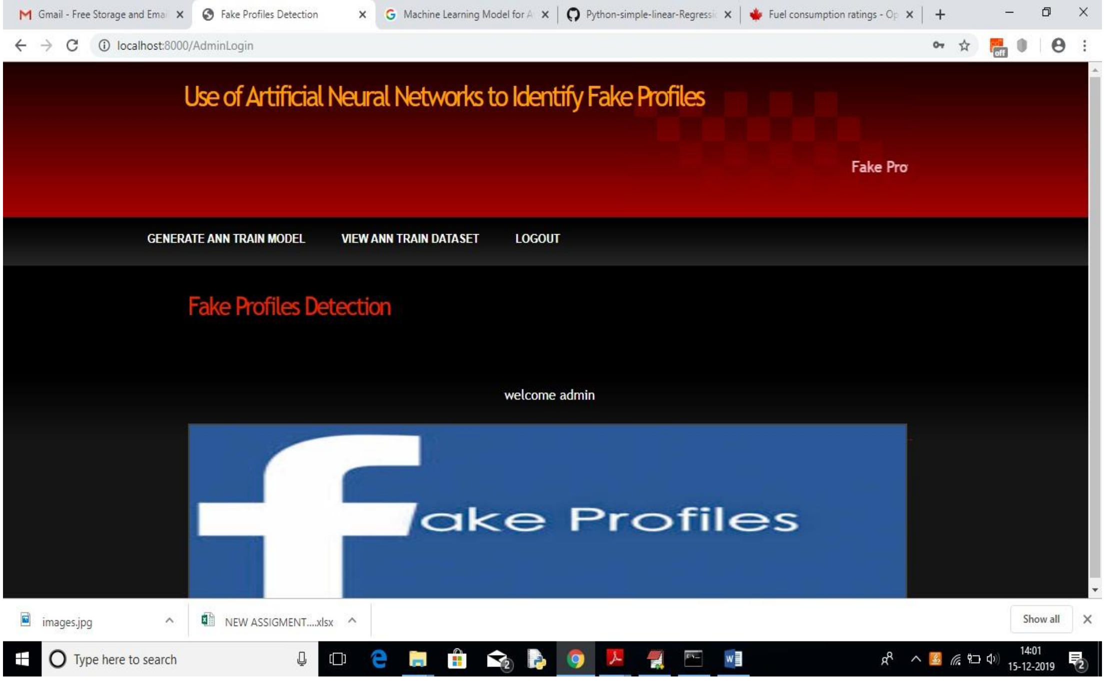

localhost:8000/Admin.html
Use of Artificial Neural Networks to Identify Fake Profiles
Fake Profiles Detection
HOME
ADMIN
USERS
login

Admin Login Screen

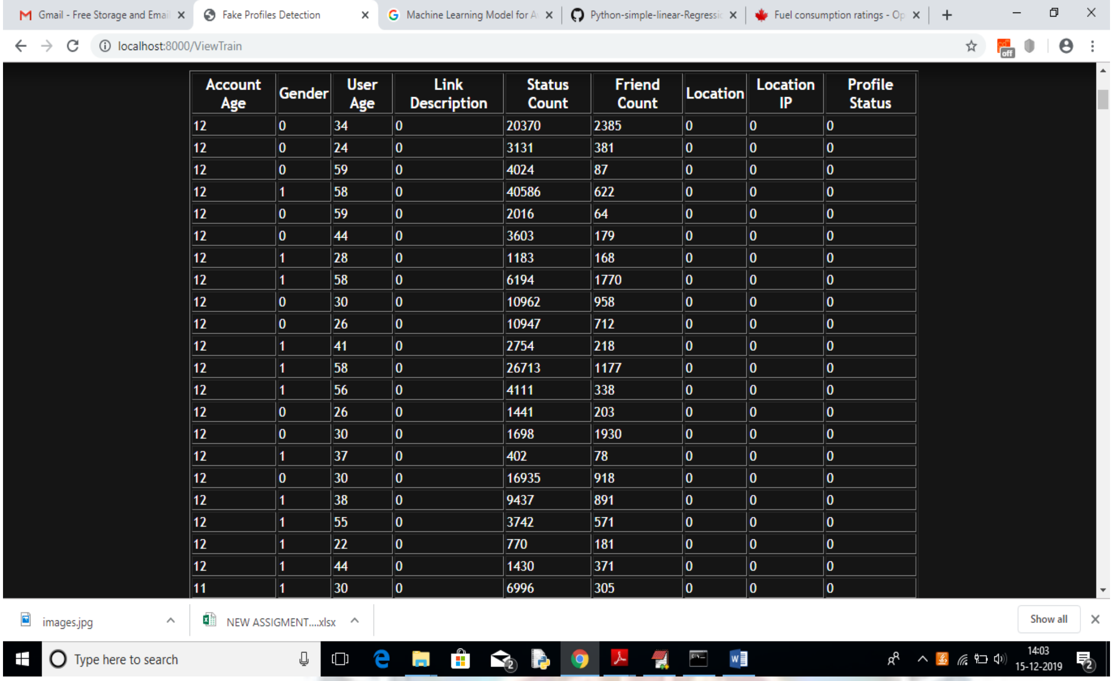
Username admin

Password •••••

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



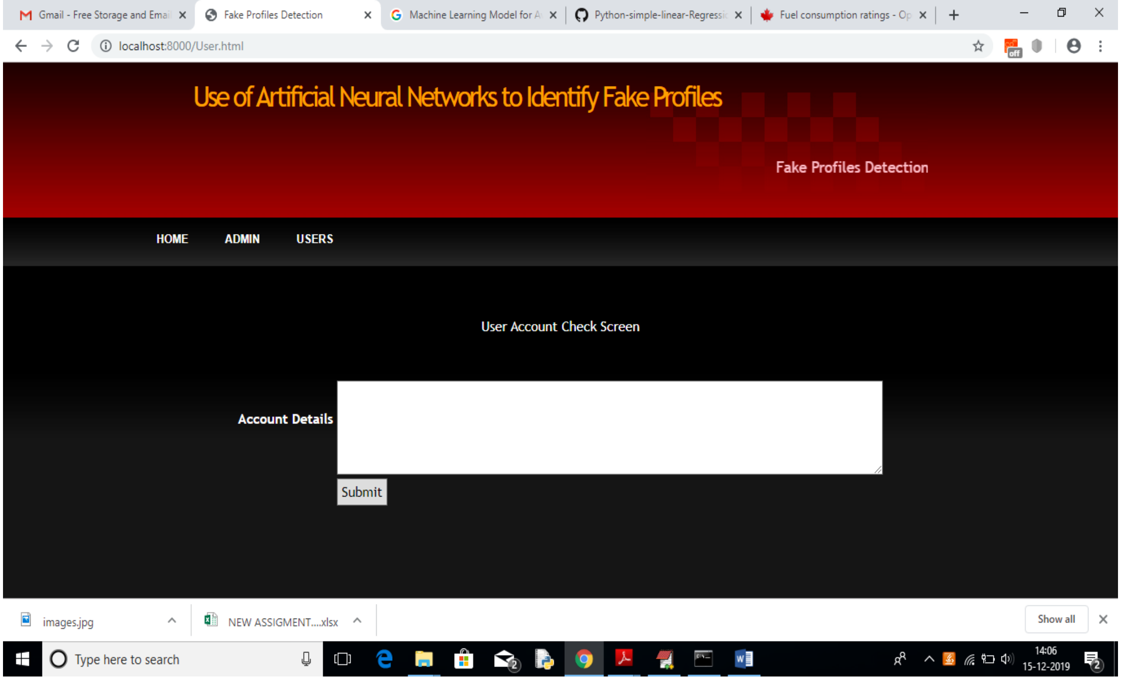
In above screen click on 'Generate ANN Train Model' to generate training model on dataset. After clicking on that link, you can see server console to check ANN processing details with accuracy.



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.

8.2 OUTPUT SCREENS



Use of Artificial Neural Networks to Identify Fake Profiles

Fake Profiles Detection

User Account Check Screen

Account Details

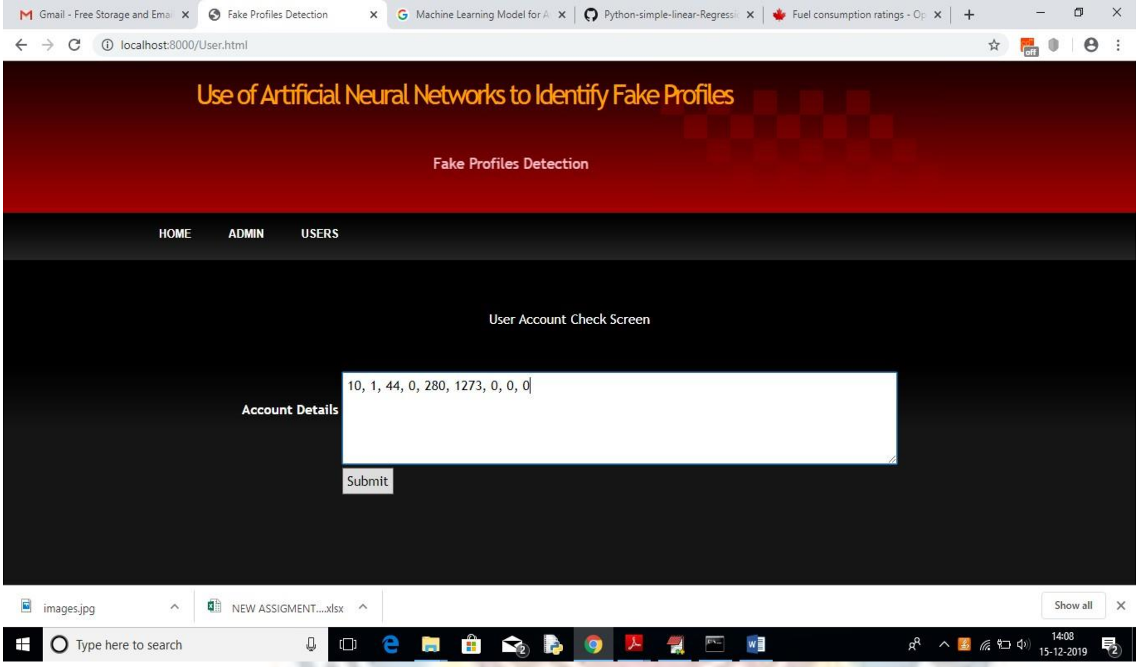
In above screen enter some test account details to get prediction/identification from ANN. You can use below records to check.

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



Use of Artificial Neural Networks to Identify Fake Profiles

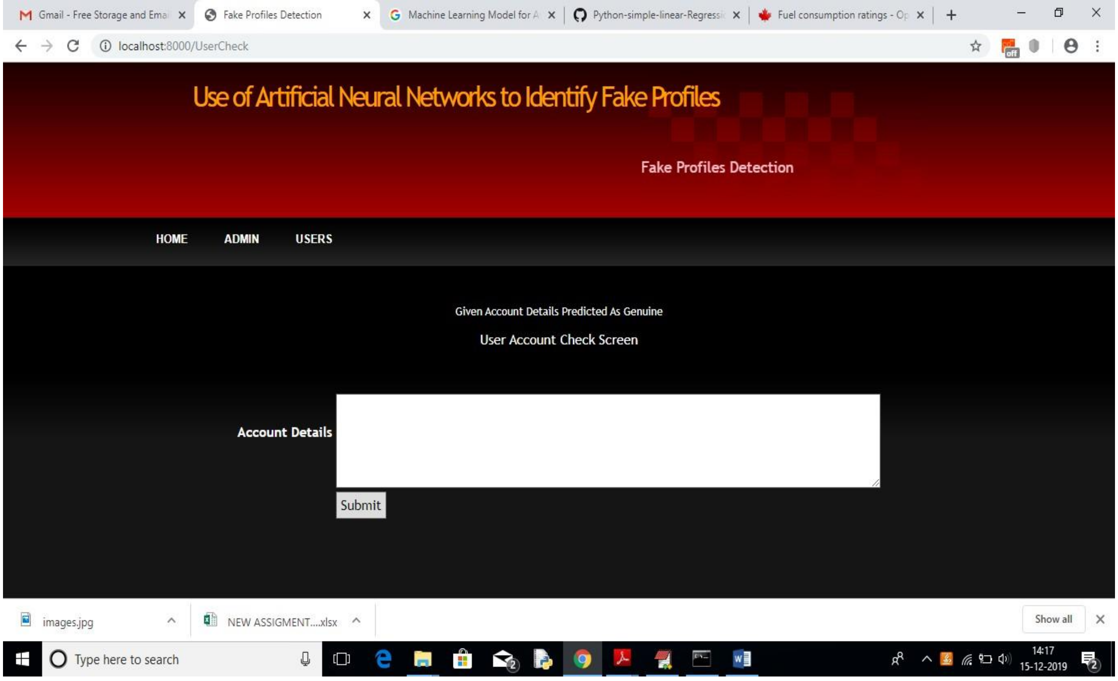
Fake Profiles Detection

User Account Check Screen

Account Details: 10, 1, 44, 0, 280, 1273, 0, 0

Submit

For above input will get below result



Use of Artificial Neural Networks to Identify Fake Profiles

Fake Profiles Detection

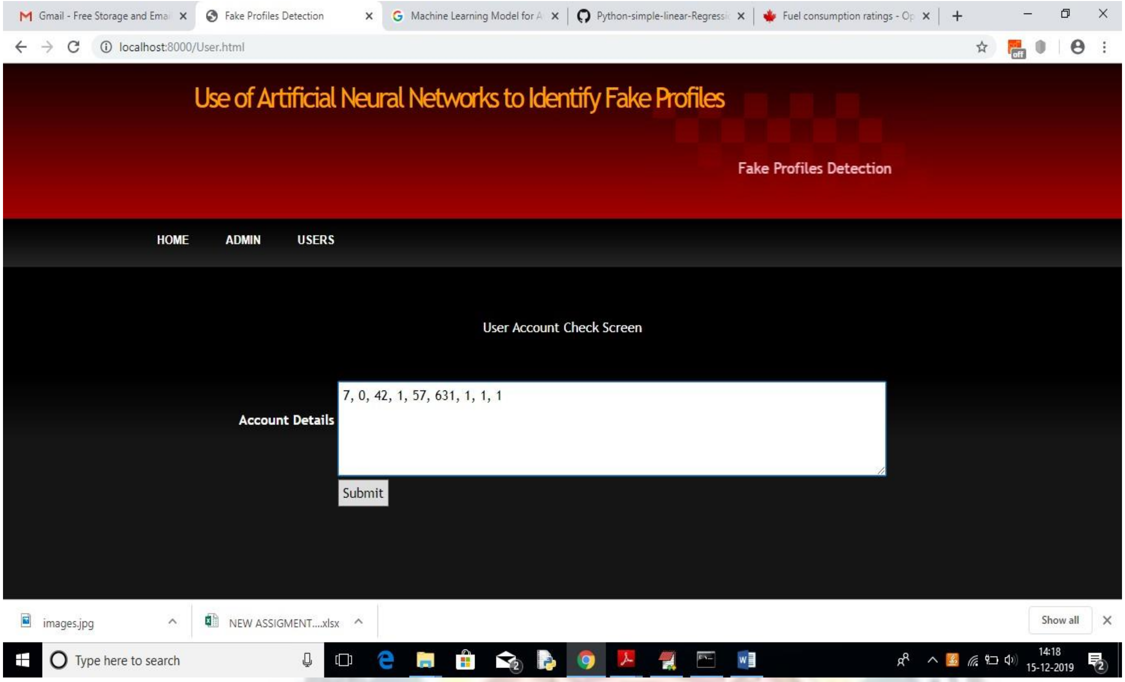
Given Account Details Predicted As Genuine

User Account Check Screen

Account Details

Submit

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

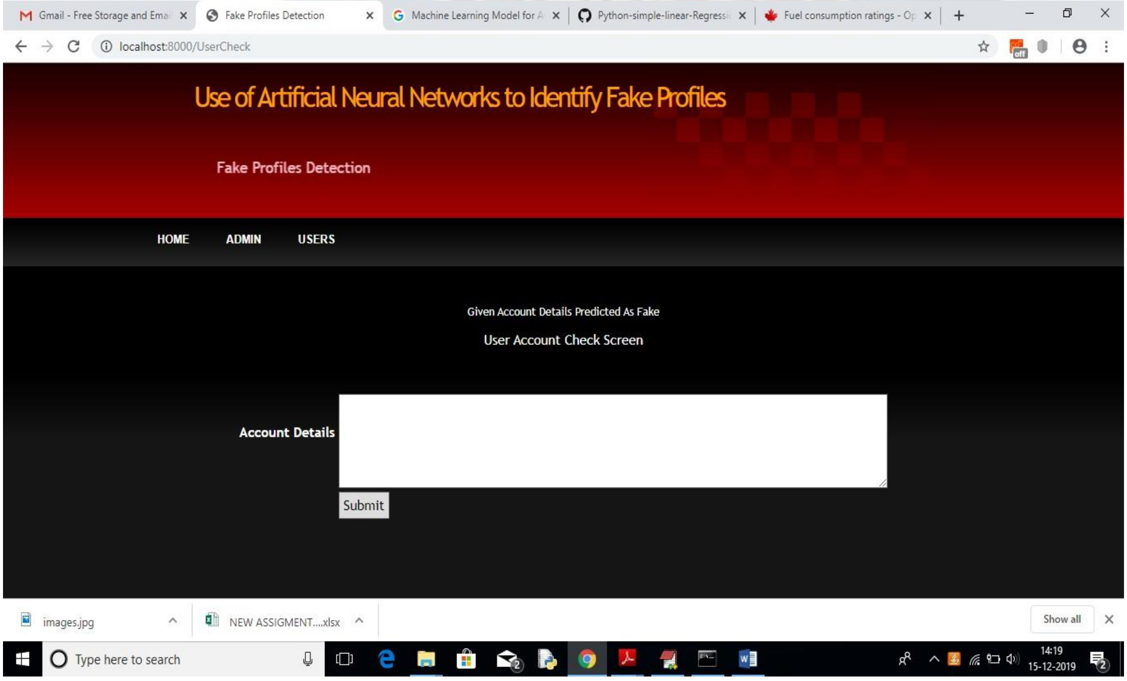


localhost:8000/User.html

Account Details: 7, 0, 42, 1, 57, 631, 1, 1, 1

Gmail - Free Storage and Ema
Fake Profiles Detection
Machine Learning Model for A
Python-simple-linear-Regressi
Fuel consumption ratings - Op
Use of Artificial Neural Networks to Identify Fake Profiles
Fake Profiles Detection
HOME
ADMIN
USERS
User Account Check Screen
Account Details
Submit
images.jpg
NEW ASSIGMENT....xlsx
Show all
Type here to search
14:18
15-12-2019

For above account details we got below result



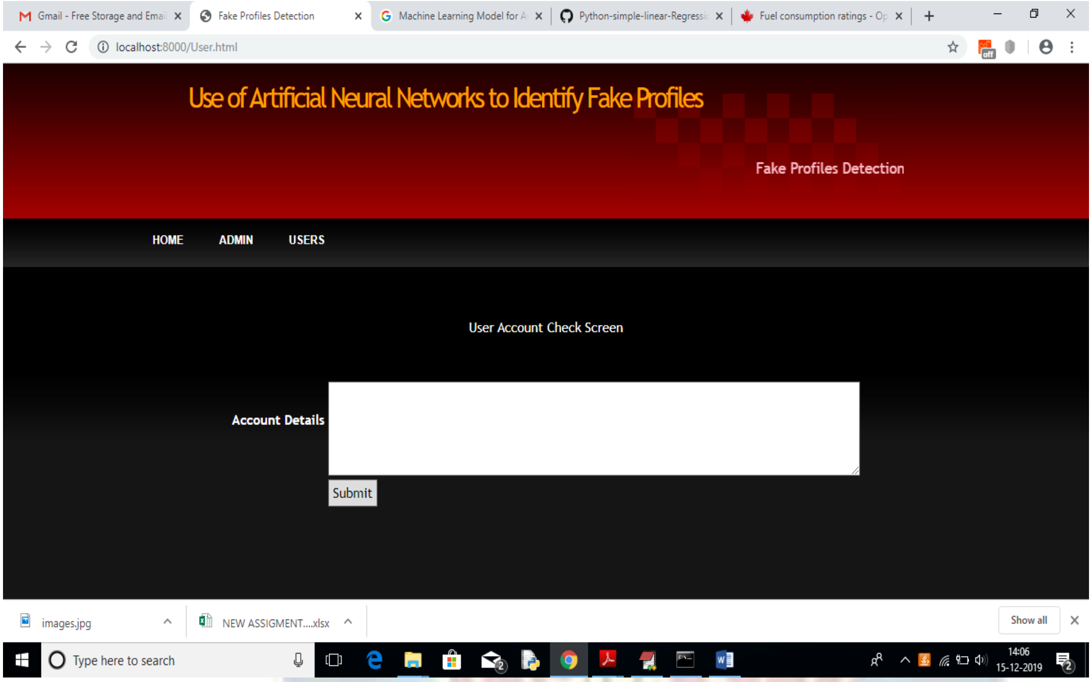
localhost:8000/UserCheck

Given Account Details Predicted As Fake

Gmail - Free Storage and Ema
Fake Profiles Detection
Machine Learning Model for A
Python-simple-linear-Regressi
Fuel consumption ratings - Op
Use of Artificial Neural Networks to Identify Fake Profiles
Fake Profiles Detection
HOME
ADMIN
USERS
User Account Check Screen
Account Details
Submit
images.jpg
NEW ASSIGMENT....xlsx
Show all
Type here to search
14:19
15-12-2019

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

9. EXPERIMENTAL RESULTS



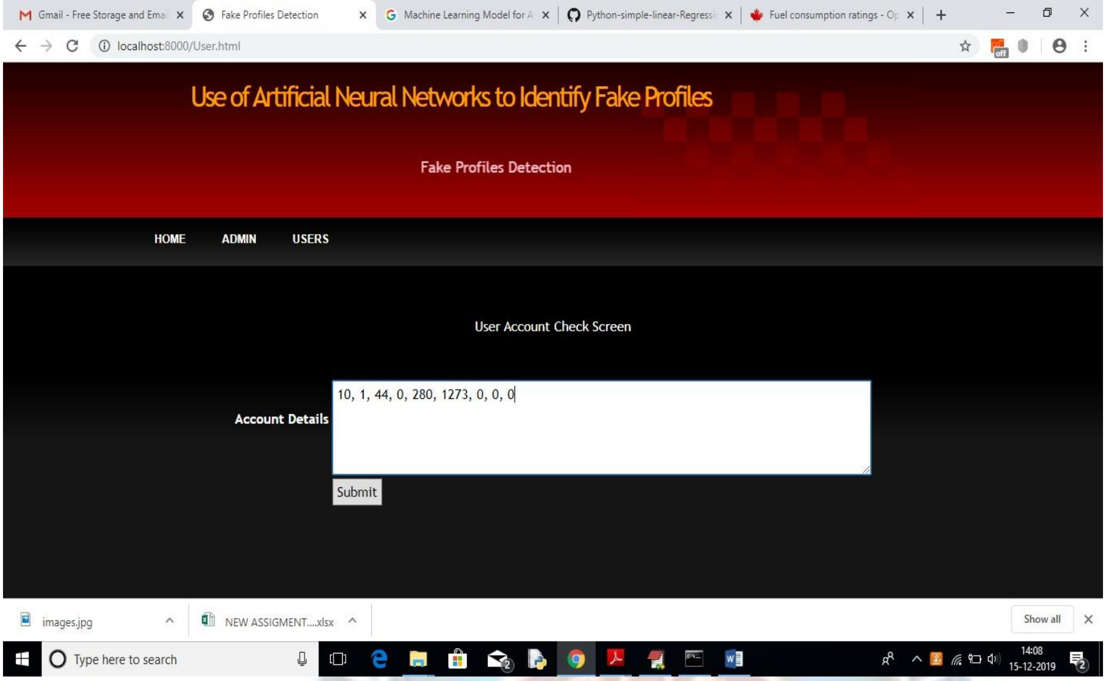
In above screen enter some test account details to get prediction/identification from ANN. You can use below records to check.

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



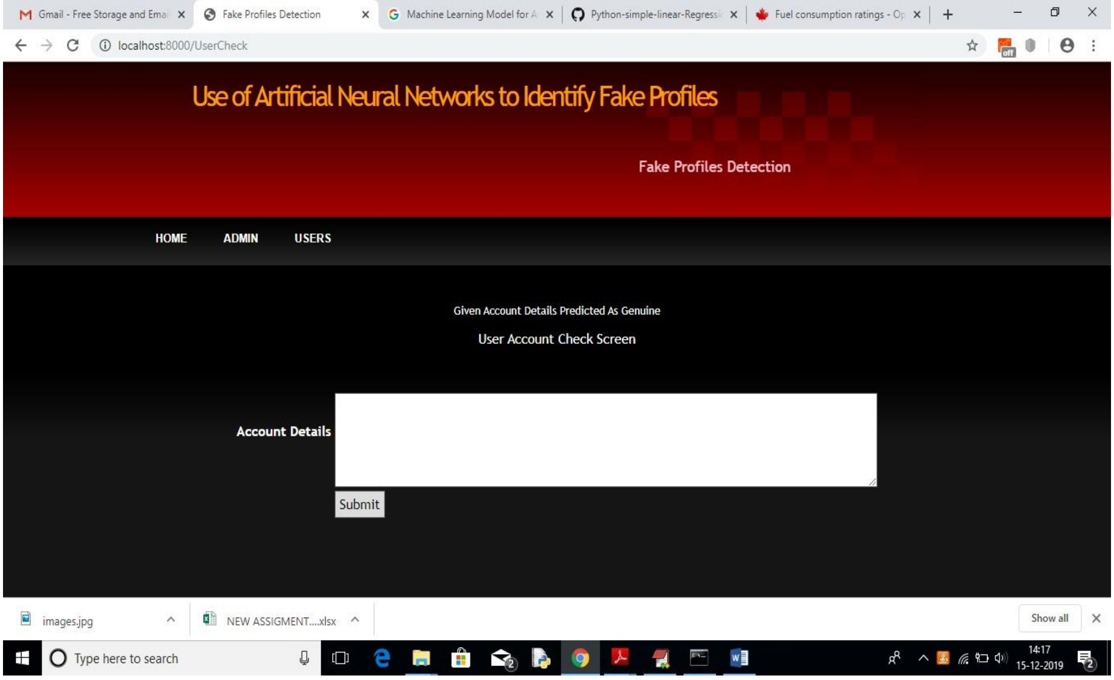
Use of Artificial Neural Networks to Identify Fake Profiles

Fake Profiles Detection

User Account Check Screen

Account Details: 10, 1, 44, 0, 280, 1273, 0, 0, 0

For above input will get below result



Use of Artificial Neural Networks to Identify Fake Profiles

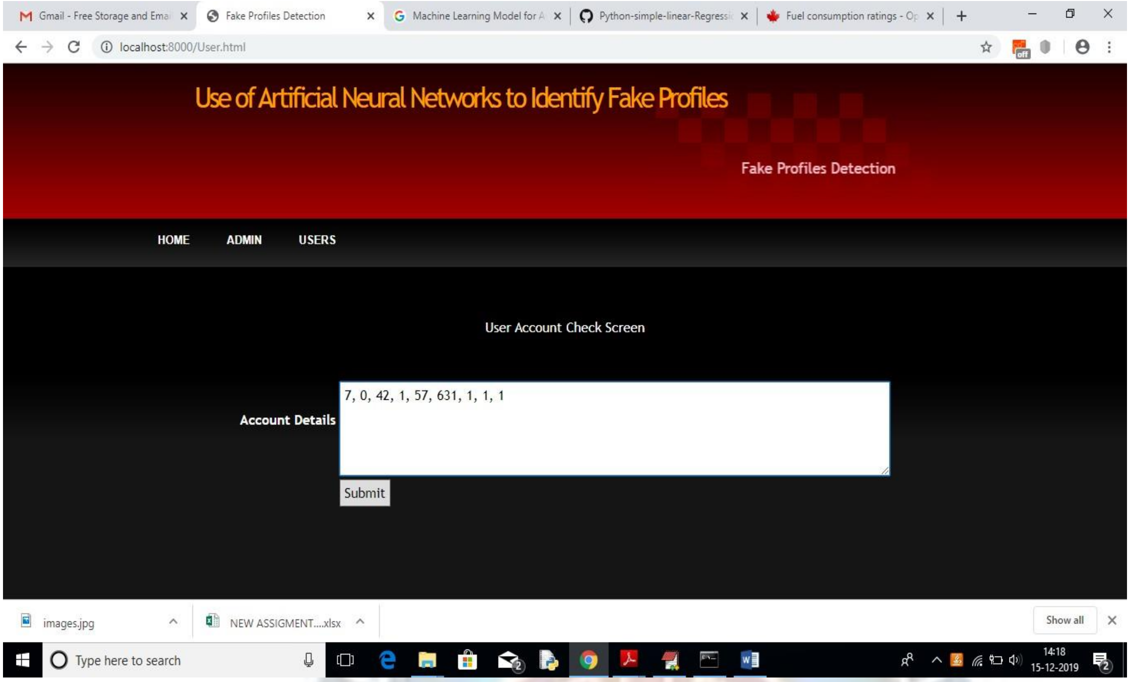
Fake Profiles Detection

Given Account Details Predicted As Genuine

User Account Check Screen

Account Details

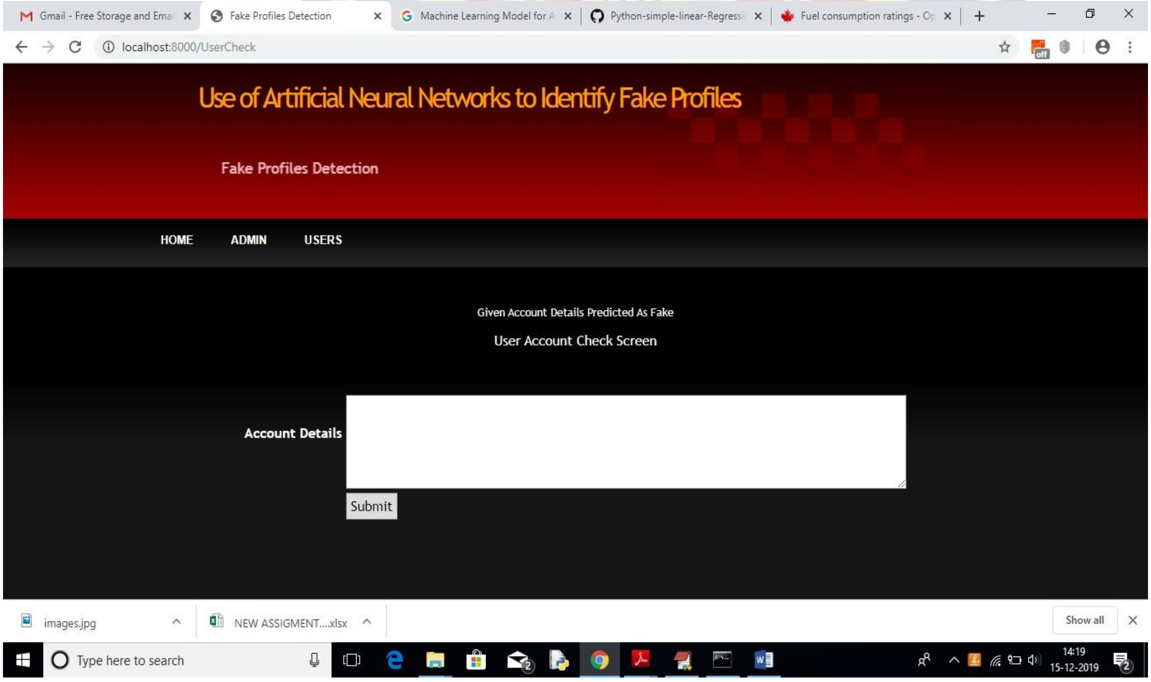
In above screen we can see the result predicted as genuine account



User Account Check Screen

Account Details: 7, 0, 42, 1, 57, 631, 1, 1, 1

For above account details we got below result



Given Account Details Predicted As Fake

User Account Check Screen

Account Details

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

10. CONCLUSION AND FUTURE ENHANCEMENT

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.

FUTURE ENHANCEMENT

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.

REFERENCES

[1] Nazir, Atif, Saqib Raza, Chen-Nee Chuah, Burkhard Schipper, and C. A. Davis. “Ghostbusting Facebook: Detecting and Characterizing Phantom Profiles in Online Social Gaming Applications.” In *WOSN*. 2010.

[2] Adikari, Shalinda, and Kaushik Dutta. “Identifying Fake Profiles in LinkedIn.” In *PACIS*, p. 278. 2014.

[3] Chu, Zi, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. “Who is tweeting on Twitter: human, bot, or cyborg?.” In Proceedings of the 26th annual computer security applications conference, pp. 21-30. ACM, 2010.

[4] Stringhini, Gianluca, Gang Wang, Manuel Egele, Christopher Kruegel, Giovanni Vigna, Haitao Zheng, and Ben Y. Zhao. “Follow the green: growth and dynamics in twitter follower markets.” In Proceedings of the 2013 conference on Internet measurement conference, pp. 163-176. ACM, 2013.

[5] Thomas, Kurt, Damon McCoy, Chris Grier, Alek Kolcz, and Vern Paxson. “Trafficking Fraudulent Accounts: The Role of the Underground Market in Twitter Spam and Abuse.” *In* Presented as part of the 22nd {USENIX} Security Symposium ({USENIX} Security13), pp. 195-210. 2013.

[6] Farooqi, Gohar Irfan, Emiliano De Cristofaro, Arik Friedman, Guillaume Jourjon, Mohamed Ali Kaafar, M. Zubair Shafiq, and Fareed Zaffar. “Characterizing Seller-Driven Black-Hat Marketplaces.” arXiv preprint arXiv: 1505.01637 (2015).

[7] Viswanath, Bimal, M. Ahmad Bashir, Mark Crovella, Saikat Guha, Krishna P. Gummadi, Balachander Krishnamurthy, and Alan Mislove. “Towards detecting anomalous user behavior in online social networks.” In 23rd {USENIX} Security Symposium ({USENIX} Security 14), pp. 223-238. 2014.

BIBLIOGRAPHY

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

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

Xml and Layout Guide https://www.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

A

Project report

On

FILTERING INSTAGRAM HASHTAGS THROUGH CROWDTAGGING AND THE HITS ALGORITHM

Submitted by

Mr.V. ROSHAN	(17K81A1255)
Ms.R. SHARANYA	(17K81A1257)
Ms.CHAITANYAPRIYA	(17K81A1249)
Ms.P. AMULYA	(17K81A1260)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Mr.J.LAKSHMINARAYANA

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN'S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **FILTERING INSTAGRAM HASHTAGS THROUGH CROWDTAGGING AND HITS ALGORITHM**, is being submitted by **V. ROSHAN (17K81A1255), R. SHARANYA (17K81A1257), S. CHAITANYA PRIYA (17K81A1249), P.AMULYA (17K81A1260)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

	Head of the Department
J.LAKSHMI NARAYANA	Dr.R.NAGARAJU
Department of IT	Department of IT

Internal Examiner

External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of **Information Technology**, session: 2017 – 2021, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **Filtering Instagram Hashtags Through Crowdtagging and the Hits Algorithm** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

V.ROSHAN	**17K81A1255**
R.SHARANYA	**17K81A1257**
S.CHAITANYA PRIYA	**17K81A1249**
P.AMULYA	**17K81A1260**

LASYA
INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **P.AMULYA** WITH ROLL NO.**17K81A1260, R.SHARANYA** WITH ROLL NO.**17K81A1257**, **S.CHAITANYA PRIYA** WITH ROLL NO.**17K81A1249**, **V.ROSHAN REDDY** WITH ROLL NO.**17K81A1255**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**FILTERING INSTAGRAM HASHTAGS THROUGH CROWD TAGGING AND THE HITS ALGORITHM**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd.
★ Kompally-500014. ★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin’s Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr.R.NAGARAJU**, Head of the Department, **INFORMATION TECHNOLOGY**, St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr.D.BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Mr. J. LAKSHMI NARAYANA**, Assistant Professor, Department of Information Technology, St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

V.ROSHAN	**17K81A1255**
R. SHARANYA	**17K81A1257**
S.CHAITANYA PRIYA	**17K81A1249**
P.AMULYA	**17K81A1260**

INDEX

Title **Page No**

ABSTRACT

Instagram is a rich source for mining descriptive tags for images and multimedia in general. The tags–image pairs can be used to train automatic image annotation (AIA) systems in accordance with the learning by example paradigm. In previous studies, we had concluded that, on average, 20% of the Instagram hashtags are related to the actual visual content of the image they accompany, i.e., they are descriptive hashtags, while there are many irrelevant hashtags, i.e., stop-hashtags, that are used across totally different images just for gathering clicks and for searchability enhancement. In this project, we present a novel methodology, based on the principles of collective intelligence that helps in locating those hashtags.

We show that the application of a modified version of the well-known hyper link induced topic search (HITS) algorithm, in a crowd tagging context, provides an effective and consistent way for finding pairs of Instagram images and hashtags, which lead to representative and noise-free training sets for content-based image retrieval. As a proof of concept, we used the crowdsourcing platform to allow collective intelligence to be gathered in the form of tag selection (crowd tagging) for Instagram hashtags. The crowd tagging data are used to form bipartite graphs in which the first type of nodes corresponds to the annotators and the second type to the hashtags they selected. The HITS algorithm is first used to rank the annotators in terms of their effectiveness in the crowd tagging task and then to identify the right hashtags per image.

LIST OF FIGURES

LIST OF SCREENSHOTS

1. INTRODUCTION

1.1 PROJECT OVERVIEW

SOCIAL media are online communication channels dedicated to community-based input, interaction, content sharing, and collaboration. These media give the users the opportunity to share their content such as text, video, and images. Users usually accompany the content they post with text such as comments or hashtags. This alternative text (comment, hashtags, etc.) provides valuable information about the user posts and other information. Preece et al. construct a Sentinel platform that can enhance social media data in order to understand different situations they based also in YouTube video comments. Sagduyu et al. present a novel system that can present large-scale synthetic data from social media. In their system, they use textual content (hashtags and hyperlinks in tweets) to produce topics and train the n-gram model. The users in several of those media, e.g., Twitter, Instagram, and Facebook, use hashtags to annotate the digital content they upload.

Hashtags are, usually, words or nonspaced phrases preceded by the symbol # that allow creators/content contributors to apply tagging that makes it easier for other users to locate their posts. A great portion of the digital content shared on social media platforms consists of images and short videos. Thus, effective retrieval of images from social media and the web, in general, becomes harder and more challenging day by day. Contemporary search engines are basically based on text descriptions to retrieve images; however, inaccurate text descriptions and the plethora of contextually annotated images led to extended research for content-based image retrieval techniques. The main problem of the content-based image retrieval is the so-called semantic gap: content-based retrieval is associated with low-level features while humans use high-level concepts for their search. To overcome this problem, automatic image annotation (AIA) methods were developed, that is, processes by which computing systems automatically assign metadata in the form of captions or keywords to images.

Among the AIA methods, those based on the learning by example paradigm are probably the most common one. A small set of manually annotated training images are used to train models, which learn the correlation between image features and textual words (high-level concepts) and then allow automatic annotation of other (unseen) images. Obviously, good training examples, i.e., representative and accurate pairs of images and related tags are vital in this case. Social media, and especially the Instagram, provide a rich source of image–tag pairs. Mining the right ones, automatically or semiautomatically, to be used as training examples is extremely important.

We must consider, however, that, in many cases, hashtags that accompany images in social media are not related with the image's content but serve several other purposes such as the expression of user's emotional state, the increase in user's clicks and findability, and the beginning of a new communication or discussion. We have also noticed that many Instagram hashtags are used across images that have nothing in common, just for searchability enhancement. We named those hashtags as stop hashtags. Thus, filtering the Instagram hashtags in terms of the visual content of the image they accompany is required. Hyperlink-induced topic search (HITS) is a ranking algorithm than we could use to filter Instagram hashtags and locate the most relevant.

The purpose of the HITS algorithm, developed by Jon Kleinberg, is to rate webpages. The basic idea is that a webpage can provide information about a topic and also relevant links for a topic. Thus, webpages belong to two groups: pages that provide good information about a topic ("authoritative") and those that give to the user good links about a topic ("hubs"). The HITS algorithm gives to each webpage both a hub and an authoritative value.

1.2 PROJECT OBJECTIVES

The main purpose of hashtags is to categorize content. They allow you to find relevant content from other people and connect with other users based on a common interest. However, there is a recent disengagement with hashtags.

People are embarrassed to use them because they look long and messy in captions, and hashtags are deemed “socially weird.” With this stigma associated with hashtags, there is no way to organize content and discover certain themes via a search. Instagram has gone through a lot of changes since 2010, but through it all, one thing has stayed consistent: the importance of Instagram hashtags. Even in 2021, using relevant, targeted hashtags on your posts and stories is one of the best ways to get discovered by new audiences on Instagram. And this can translate into more engagement, more followers, and more customers for your business. Instagram hashtags work by organizing and categorizing photos and videos.

If you have a public Instagram account and add a hashtag to a post, that post will be visible on the corresponding hashtag page (it’s basically a directory of all the photos and videos that were tagged with that hashtag).Since hashtags are used with an intent to discover content, the right hashtags can put you in front of your target audience, even if they haven’t connected with you before.

For example, a food blogger might post a picture of a gorgeous smoothie bowl, and then use the hashtags #superfoods, #cleaneating, and #vegansofig when it’s uploaded to Instagram. By using these three hashtags, the image is catalogued so other Instagram users who enjoy healthy foods can easily find it.

A few things to keep in mind:

- When people with private profiles tag posts, they won’t appear publicly on hashtag pages.
- Numbers are allowed in hashtags. However, spaces and special characters, like $ or %, won’t work.
- You can only add hashtags to your own posts. You can’t tag other people’s photos/videos.
- You can use up to 30 hashtags on a post and 10 on Instagram Stories.

1.3 SCOPE OF THE PROJECT

3.4.1 EXISTING SYSTEM:

Instagram is a rich source for mining descriptive tags for images and multimedia in general. The tags–image pairs can be used to train automatic image annotation (AIA) systems in accordance with the learning by example paradigm. In previous studies, we had concluded that, on average, 20% of the Instagram hashtags are related to the actual visual content of the image they accompany, i.e., they are descriptive hashtags, while there are many irrelevant hashtags, i.e., stop-hashtags, that are used across totally different images just for gathering clicks and for searchability enhancement.

3.4.2 PROPOSED SYSTEM:

we present a novel methodology, based on the principles of collective intelligence that helps in locating those hashtags. We show that the application of a modified version of the well-known hyperlink induced topic search (HITS) algorithm, in a crowd tagging context, provides an effective and consistent way for finding pairs of Instagram images and hashtags, which lead to representative and noise-free training sets for content-based image retrieval.

As a proof of concept, we used the crowdsourcing platform to allow collective intelligence to be gathered in the form of tag selection (crowd tagging) for Instagram hashtags.

The crowd tagging data are used to form bipartite graphs in which the first type of nodes corresponds to the annotators and the second type to the hashtags they selected. The HITS algorithm is first used to rank the annotators in terms of their effectiveness in the crowd tagging task and then to identify the right hashtags per image.

1.4 ORGANIZATION OF CHAPTERS

1.4.1 INTRODUCTION

Hashtags are, usually, words or non spaced phrases preceded by the symbol # that allow creators/content contributors to apply tagging that makes it easier for other users to locate their posts. A great portion of the digital content shared on social media platforms consists of images and short videos. Thus, effective retrieval of images from social media and the web, in general, becomes harder and more challenging day by day. Contemporary search engines are basically based on text descriptions to retrieve images; however, inaccurate text descriptions and the plethora of contextually annotated images led to extended research for content-based image retrieval techniques.

1.4.2 LITERATURE SURVEY

REVIEW OF RELATED LITERATURE

1.TOPIC MODELLING ON INSTAGRAM HASHTAGS:AN ALTERNATIVE WAY TO AUTOMATIC IMAGE ANNOTATION

2.CROWDSOURCING FOR MULTIPLE-CHOICE QUESTION ANSWERING

3.A SURVEY AND ANALYSIS ON AUTOMATIC IMAGE ANNOTATION

4.VALIDITY AND RELIABILITY OF NATURALISTIC DRIVING SCENE CATEGORIZATION JUDGEMENTS FROM CROWDSOURCING

1.4.3 REQUIREMENTS SPECIFICATION

SOFTWARE REQUIREMENTS

• Operating System	: Windows family(8/7/XP)
• Technology	: Python 3.6 and Django
• IDE	: PyCharm
• Database Connectivity	: MySQL

HARDWARE REQUIREMENTS

Processor	: Pentium IV or higher
RAM	: 4 GB
Space on Hard Disk	: minimum 80GB

1.4.4 SOFTWARE DEVELOPMENT ANALYASIS

HYPERLINK INDUCED TOPIC SEARCH (HITS) ALGORITHM

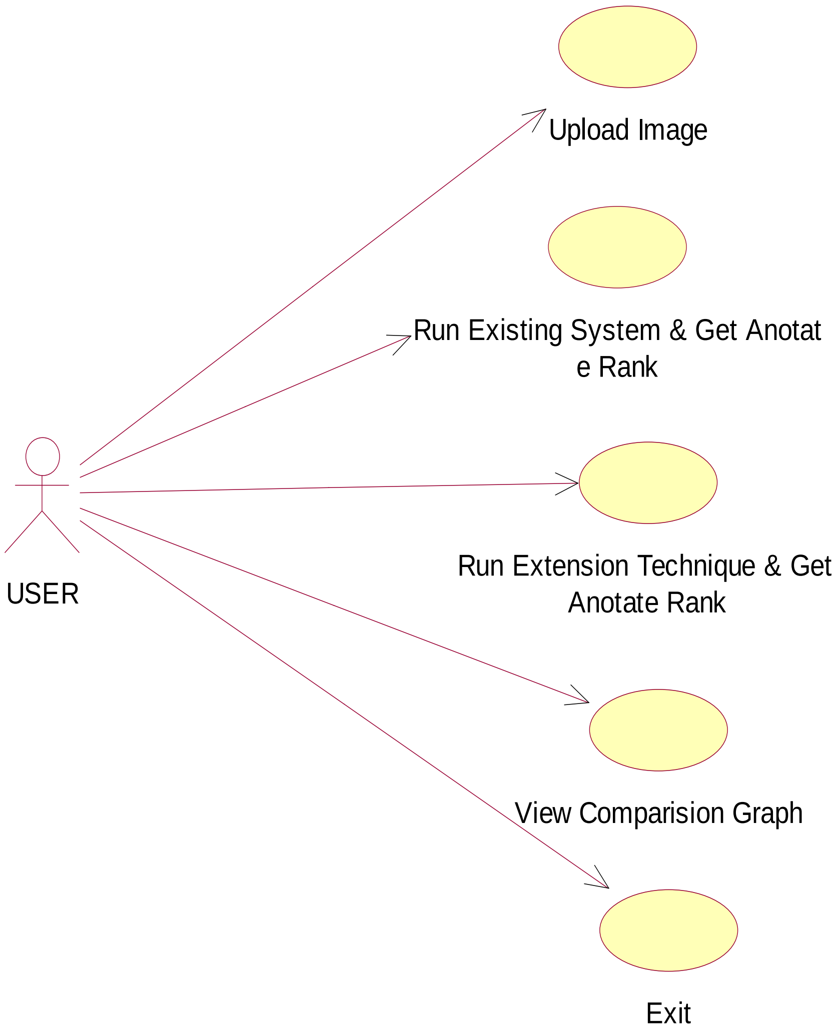
Hyperlink Induced Topic Search (HITS) Algorithm is a Link Analysis Algorithm that rates webpages, developed by Jon Kleinberg. This algorithm is used to the web link-structures to discover and rank the webpages relevant for a particular search. HITS uses hubs and authorities to define a recursive relationship between webpages. Before understanding the HITS Algorithm, we first need to know about Hubs and Authorities.

Given a query to a Search Engine, the set of highly relevant web pages are called Roots. They are potential Authorities. Pages that are not very relevant but point to pages in the Root are called Hubs. Thus, an Authority is a page that many hubs link to whereas a Hub is a page that links to many authorities.

1.4.5 PROJECT SYSTEM DESIGN

USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.


Upload Image
Run Existing System & Get Anotat e Rank
Run Extension Technique & Get Anotate Rank
USER
View Comparision Graph
Exit

1.4.6 PROJECT CODING

```
def RCNN(filename):
    transform = transforms.Compose([transforms.ToTensor(),
transforms.Normalize((0.485, 0.456, 0.406), (0.229, 0.224, 0.225))])
    with open('model/vocab.pkl', 'rb') as f:
        vocab = pickle.load(f)
    # Build models
    encoder = EncoderCNN(256).eval()  # eval mode (batchnorm uses moving
mean/variance)
    decoder = DecoderRNN(256, 512, len(vocab), 1)
    encoder = encoder.to(device)
    decoder = decoder.to(device)

    # Load the trained model parameters
    encoder.load_state_dict(torch.load('model/encoder-5-3000.pkl'))
    decoder.load_state_dict(torch.load('model/decoder-5-3000.pkl'))

    # Prepare an image
    image = loadImage(filename, transform)
    image_tensor = image.to(device)

    # Generate an caption from the image
    feature = encoder(image_tensor)
    sampled_ids = decoder.sample(feature)
    sampled_ids = sampled_ids[0].cpu().numpy()          # (1, max_seq_length) -
> (max_seq_length)

    # Convert word_ids to words
    sampled_caption = []
    for word_id in sampled_ids:
        word = vocab.idx2word[word_id]
        sampled_caption.append(word)
        if word == '<end>':
            break
    sentence = ' '.join(sampled_caption)
    sentence = sentence.replace('kite','umbrella')
    sentence = sentence.replace('flying','with')

    image = Image.open(filename)
    plt.imshow(np.asarray(image))

    text.insert(END,"Automatic Extracted Sentence From Image :
"+sentence+"\n\n")
    if len(sentence) > 0:
        length = len(sentence)-5
        sentence = sentence[8:length]
        print(sentence)
    sentence = regex.sub('', sentence)
    for word,pos in nltk.pos_tag(nltk.word_tokenize(str(sentence))):
        if (pos == 'NN' or pos == 'NNP' or pos == 'NNS' or pos == 'NNPS'):
            word = getUpper(word)
            if word not in attributes:
                attributes.append(word.lower())

    text.insert(END,"Extracted Main Attributes From Image:
"+str(attributes)+"\n")
```

1.4.7 PROJECT TESTING

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.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

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

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.

1.4.8 OUTPUT SCREEN

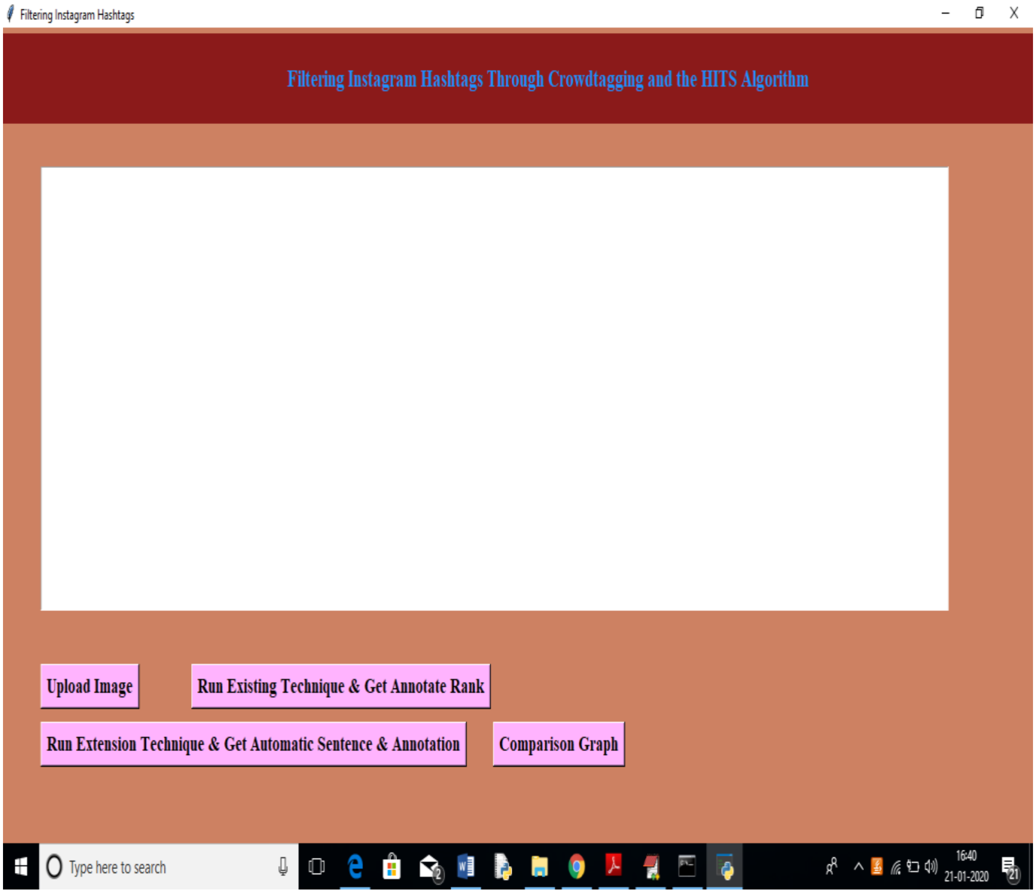

Filtering Instagram Hashtags
Filtering Instagram Hashtags Through Crowdtagging and the HITS Algorithm
Upload Image
Run Existing Technique & Get Annotate Rank
Run Extension Technique & Get Automatic Sentence & Annotation
Comparison Graph
Type here to search
16:40
21-01-2020

Figure 1.4.8 Output Screen

1.4.9 CONCLUSIONS

A final remark of this project refers to the importance of using weighted user–tag bipartite graphs for the crowdtagged images. It appears that weighting the bipartite graphs with the hub scores of the annotators provides the best results.

However, even in the case that the reliability metric of the crowdsourcing platform itself (the _trust variable of Figure-eight in our case) is used to weight the bipartite graphs, the results are not significantly worse. We are a little bit reluctant to generalize this conclusion because, in this paper, we have used too many annotations (499) per image. Thus, one of our future tests will involve a more typical image crowdtagging scenario in which much more images will be used and much fewer (typically less than five) annotations per image will be considered. In that case, only partial coannotation of the same images by the same annotators will take place in contrast to this paper in which all annotators annotated all images.

2. LITERATURE SURVEY

2.1 REVIEW OF RELATED LITERATURE

1. TOPIC MODELLING ON INSTAGRAM HASHTAGS:AN ALTERNATIVE WAY TO AUTOMATIC IMAGE ANNOTATION

Automatic Image Annotation (AIA) is the process of assigning tags to digital images without the intervention of humans. Most of the modern automatic image annotation methods are based on the learning by example paradigm. In those methods building the training examples, that is, pairs of images and related tags, is the first critical step. We have shown in our previous studies that hashtags accompanying images in social media and especially the Instagram provide a reach source for creating training sets for AIA. However, we concluded that only 20% of the Instagram hashtags describe the actual content of the image they accompany, thus, a series of filtering steps need to apply in order to identify the appropriate hashtags. In this paper we apply topic modelling with Latent Dirichlet Allocation (LDA) on Instagram hashtags in order to predict the subject of the related images. Since a topic is composed by a set of related terms, the identification of the visual topic of an Instagram image, through the proposed method, provides a plausible set of tags to be used in the context of training AIA methods.

2. CROWDSOURCING FOR MULTIPLE-CHOICE QUESTION ANSWERING

We leverage crowd wisdom for multiple-choice question answering and employ lightweight machine learning techniques to improve the aggregation accuracy of crowdsourced answers to these questions. In order to develop more effective aggregation methods and evaluate them empirically, we developed and deployed a crowdsourced system.

Analyzing our data (which consist of more than 200,000 answers), we find that by just going with the most selected answer in the aggregation, we can answer over 90% of the questions correctly, but the success rate of this technique plunges to 60% for the later/harder questions in the quiz show.

To improve the success rates of these later/harder questions, we investigate novel weighted aggregation schemes for aggregating the answers obtained from the crowd. By using weights optimized for reliability of participants (derived from the participants' confidence), we show that we can pull up the accuracy rate for the harder questions by 15%, and to overall 95% average accuracy. Our results provide a good case for the benefits of applying machine learning techniques for building more accurate crowdsourced question answering systems.

3. A SURVEY AND ANALYSIS ON AUTOMATIC IMAGE ANNOTATION

In recent years, image annotation has attracted extensive attention due to the explosive growth of image data. With the capability of describing images at the semantic level, image annotation has many applications not only in image analysis and understanding but also in some relative disciplines, such as urban management and biomedical engineering. Because of the inherent weaknesses of manual image annotation, Automatic Image Annotation (AIA) has been raised since the late 1990s.We classify AIA methods into five categories: 1) Generative model-based image annotation, 2) Nearest neighbor-based image annotation, 3) Discriminative model-based image annotation, and 4) Tag completion-based image annotation, 5) Deep Learning-based image annotation.

Comparisons of the five types of AIA methods are made based on the underlying idea, main contribution, model framework, computational complexity, computation time, and annotation accuracy. We also give an overview of five publicly available image datasets and four standard evaluation metrics commonly used as benchmarks for evaluating AIA methods. Then the performance of some typical or well-behaved models is assessed based on benchmark dataset and standard evaluation metrics. Finally, we share our viewpoints on the open issues and challenges in AIA as well as research trends in the future.

4. VALIDITY AND RELIABILITY OF NATURALISTIC DRIVING SCENE CATEGORIZATION JUDGEMENTS FROM CROWDSOURCING

A common challenge with processing naturalistic driving data is that humans may need to categorize great volumes of recorded visual information. By means of the online platform CrowdFlower, we investigated the potential of crowdsourcing to categorize driving scene features. (i.e., presence of other road users, straight road segments, etc.) at greater scale than a single person or a small team of researchers would be capable of. In total, 200 workers from 46 different countries participated in 1.5 days. Validity and reliability were examined, both with and without embedding researcher generated control questions via the CrowdFlower mechanism known as Gold Test Questions (GTQs).

By employing GTQs, we found significantly more valid (accurate) and reliable (consistent) identification of driving scene items from external workers. Specifically, at a small scale CrowdFlower Job of 48 three-second video segments, an accuracy (i.e., relative to the ratings of a confederate researcher) of 91% on items was found with GTQs compared to 78% without. A difference in bias was found, where without GTQs, external workers returned more false positives than with GTQs. At a larger scale CrowdFlower Job making exclusive use of GTQs, 12,862 three-second video segments were released for annotation. Infeasible (and self-defeating) to check the accuracy of each at this scale, a random subset of 1012 categorizations were validated and returned similar levels of accuracy (95%).

2.2 SOFTWARE REQUIREMENT

For developing the project, the following are the Software Requirements:

- Python
- Django

2.2.1 PYTHON

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. C Python, the reference implementation of Python, is opensource software and has a community-based development model, as do nearly all its variant implementations. C Python is managed by the non-profit Python Software Foundation. Python features a dynamic type of 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.

2.2.2 DJANGO

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source.

Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and "pluggability" of components, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings files and data models.

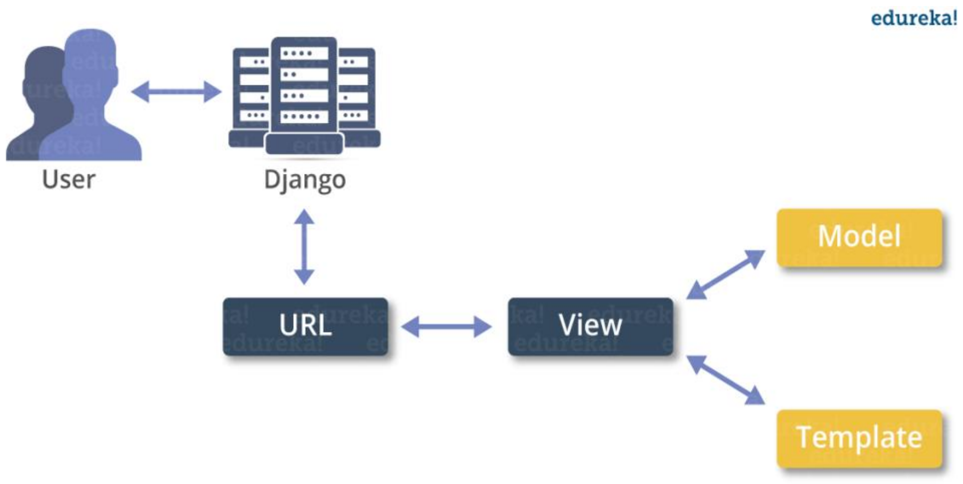


Figure 2.2.2 Django via Admin Model

2.3 ALGORITHMS FOR RESEARCH ACTIVITY

HYPERLINK INDUCED TOPIC SEARCH (HITS) ALGORITHM

Hyperlink Induced Topic Search (HITS) Algorithm is a Link Analysis Algorithm that rates webpages, developed by Jon Kleinberg. This algorithm is used to the web link-structures to discover and rank the webpages relevant for a particular search. HITS uses hubs and authorities to define a recursive relationship between webpages. Before understanding the HITS Algorithm, we first need to know about Hubs and Authorities.

Given a query to a Search Engine, the set of highly relevant web pages are called Roots. They are potential Authorities. Pages that are not very relevant but point to pages in the Root are called Hubs. Thus, an Authority is a page that many hubs link to whereas a Hub is a page that links to many authorities.

Algorithm –

- Let number of iterations be k.
- Each node is assigned a Hub score = 1 and an Authority score = 1.
- Repeat k times:

Hub update: Each node’s Hub score = \Sigma (Authority score of each node it points to).

Authority update: Each node’s Authority score = \Sigma (Hub score of each node pointing to it).

Normalize the scores by dividing each Hub score by square root of the sum of the squares of all Hub scores and dividing each Authority score by square root of the sum of the squares of all Authority scores. (optional)

Now, let us see how to implement this algorithm using Networks Module.

Let us consider the following Graph:

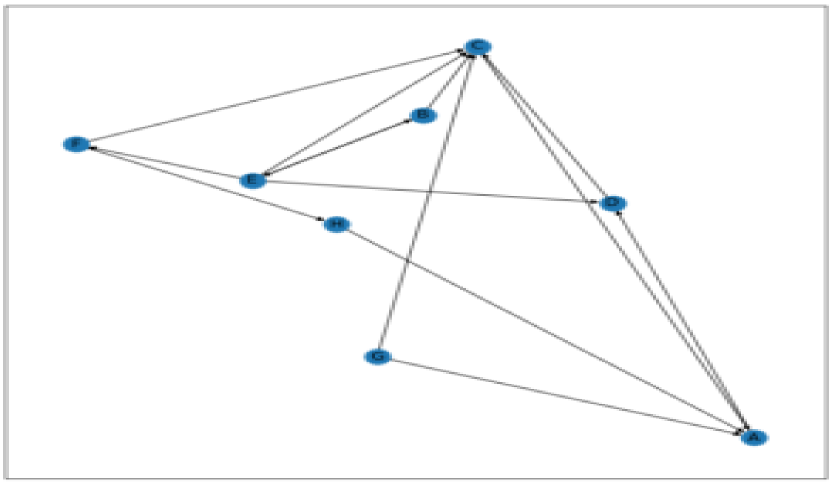


Figure 2.3 Network Module Graph

2.4 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

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 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 a modest requirement, as only minimal or null changes are required for implementing this system.

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.

3. REQUIREMENTS SPECIFICATION

3.1 SOFTWARE REQUIREMENTS

Operating System	: Windows family(8/7/XP)
Technology	: Python 3.6 and Django
IDE	: PyCharm
Database Connectivity	: MySQL

3.2 HARDWARE REQUIREMENTS

Processor	: Pentium IV or higher
RAM	: 4 GB
Space on Hard Disk	: minimum 80GB

4. SOFTWARE REQUIREMENTS ANALASIS

4.1 DEFINE THE PROBLEM

Instagram is a rich source for mining descriptive tags for images and multimedia in general. The tags–image pairs can be used to train automatic image annotation (AIA) systems in accordance with the learning by example paradigm. In previous studies, we had concluded that, on average, 20% of the Instagram hashtags are related to the actual visual content of the image they accompany, i.e., they are descriptive hashtags, while there are many irrelevant hashtags, i.e., stop-hashtags.

In This project, we show that the application of a modified version of the well-known hyper link induced topic search (HITS) algorithm, in a crowd tagging context, provides an effective and consistent way for finding pairs of Instagram images and hashtags, which lead to representative and noise-free training sets for content-based image retrieval. As a proof of concept, we used the crowdsourcing platform to allow collective intelligence to be gathered in the form of tag selection (crowd tagging) for Instagram hashtags. The crowd tagging data are used to form bipartite graphs in which the first type of nodes corresponds to the annotators and the second type to the hashtags they selected. The HITS algorithm is first used to rank the annotators in terms of their effectiveness in the crowd tagging task and then to identify the right hashtags per image.

4.2 DEFINE THE MODULES

1 SYSTEM FRAMEWORKS:

In this framework, we develop a model prototype of Instagram. Instagram is free photo and video sharing app available on web, we develop here in the localhost. People can upload photos to our service and share them with their followers or with a select group of friends.

They can also view, comment and like posts shared by their friends on Instagram. Anyone can create an account by registering an email address and selecting a username.

2 USER:

In User module, Initially User must have to register their detail and after login user can view the profile details. User can post photos. While posting images, the user should enter the tag and description of the image and then post the image is developed in this module. After posting the image, the users can be able to type comment of the image. We have developed the system to comment the important words with the Hashtag.

In this user module, the user has the option of Viewing their profile, View Friends status, post image, See the posted images of own and view their friend's images are developed in this user module.

3 INSTAGRAM SERVER:

In the Instagram server module, we develop the system with functionalities of developing the options of viewing all the users, all friend's status, view all images, view all image reviews, view dislikes, view image hit results etc. Overall, this module of Instagram server is like the admin of the overall system.

4.3 MODULE FUNTIONALITIES

1. Functional Requirements -Graphical User interface with the User.

2. Non-Functional Requirements

A. **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.

B. **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.

C. **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.

D. **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.

E. **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.

F. **Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life. However, the allowable wattage cannot be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

5 SOFTWARE DESIGN

5.1 SYSTEM ARCHITECTURAL DESIGN

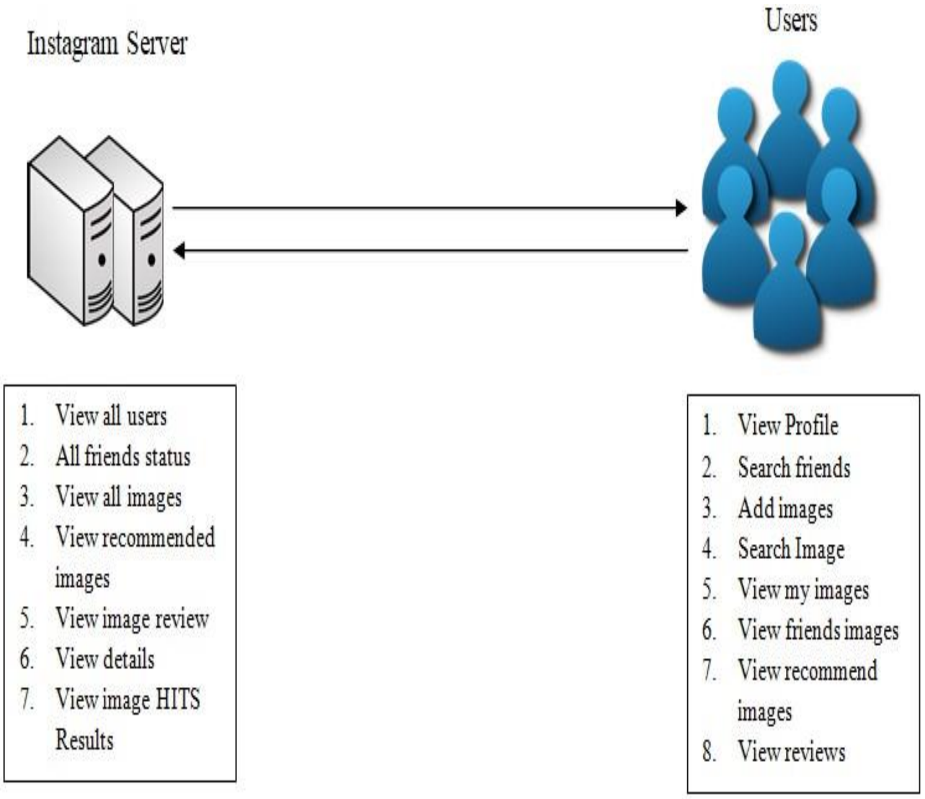


Figure 5.1 System Architectural Design

5.2 CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

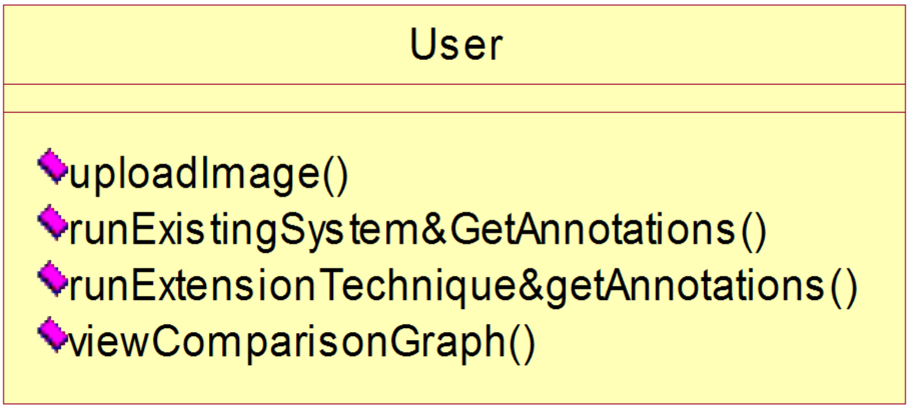

User
uploadImage()
runExistingSystem&GetAnnotations()
runExtensionTechnique&getAnnotations()
viewComparisonGraph()

Figure 5.2 Class Diagram

5.3 USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

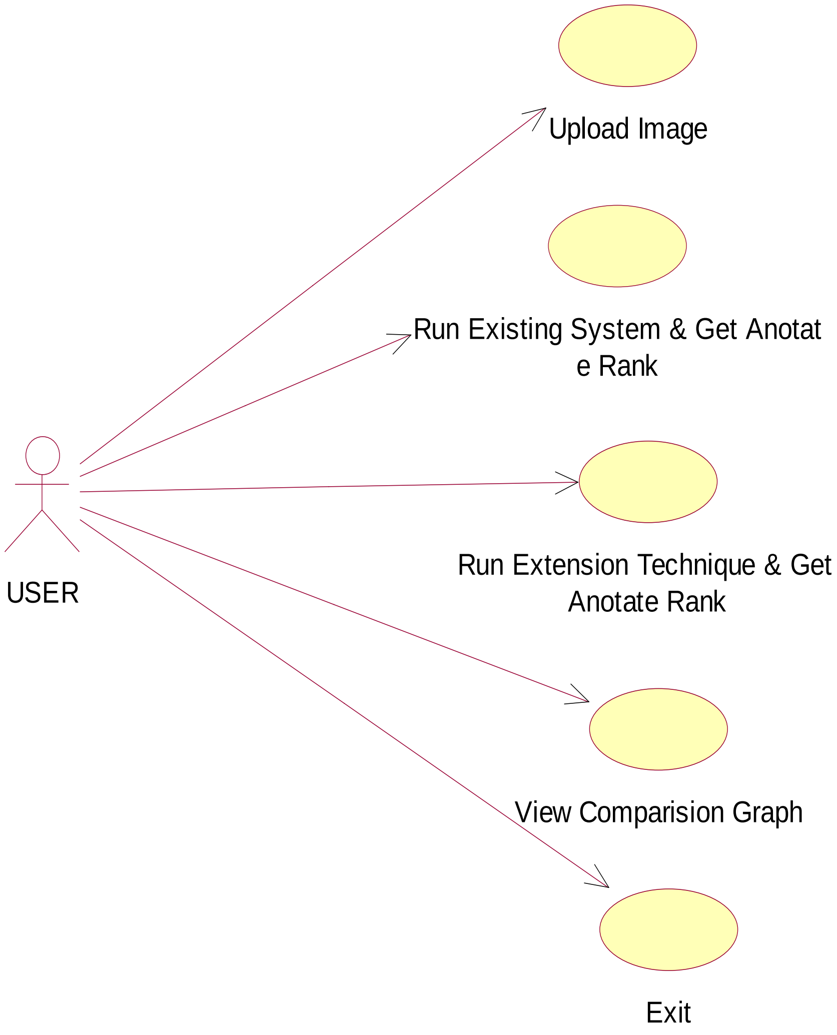

Upload Image
Run Existing System & Get Anotat
e Rank
USER
Run Extension Technique & Get
Anotate Rank
View Comparision Graph
Exit

Figure 5.3 Use Case Diagram

5.4 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

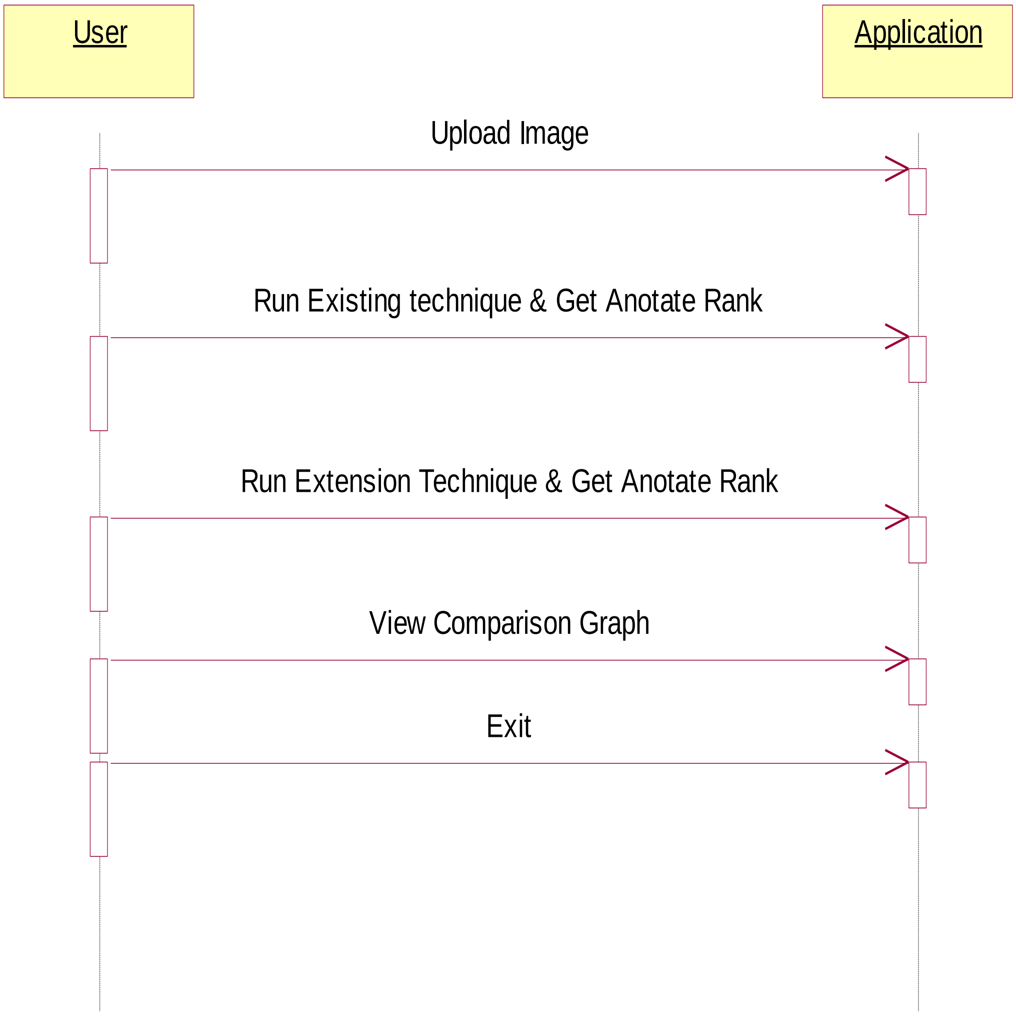

User
Application
Upload Image
Run Existing technique & Get Anotate Rank
Run Extension Technique & Get Anotate Rank
View Comparison Graph
Exit

Figure 5.4 Sequence Diagram

5.5 COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

A Communication diagram models the interactions between objects or parts in terms of sequenced messages. Communication diagrams represent a combination of information taken from Class, Sequence, and Use Case Diagrams describing both the static structure and dynamic behaviour of a system.

However, communication diagrams use the free-form arrangement of objects and links as used in Object diagrams. In order to maintain the ordering of messages in such a free-form diagram, messages are labelled with a chronological number and placed near the link the message is sent over. Reading a communication diagram involves starting at message 1.0 and following the messages from object to object.

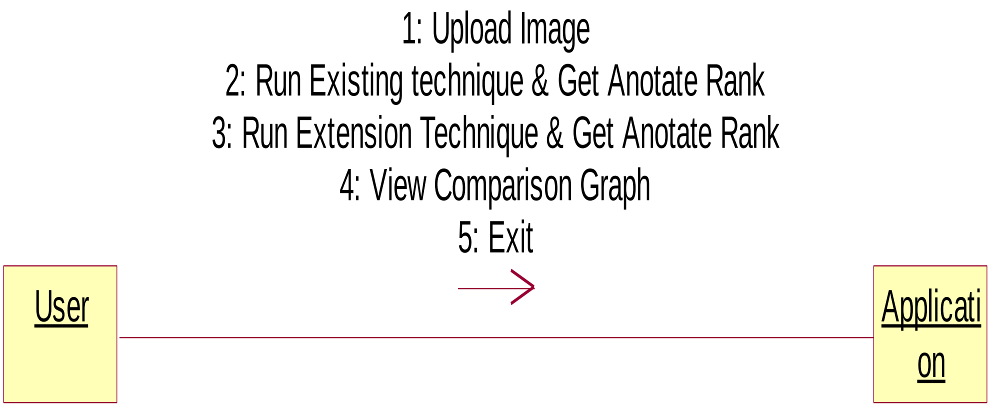

1: Upload Image
2: Run Existing technique & Get Anotate Rank
3: Run Extension Technique & Get Anotate Rank
4: View Comparison Graph
5: Exit
User
Application

Figure 5.5 Collaboration Diagram

5.6 DEPLOYMENT DIAGRAM

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.[1] To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

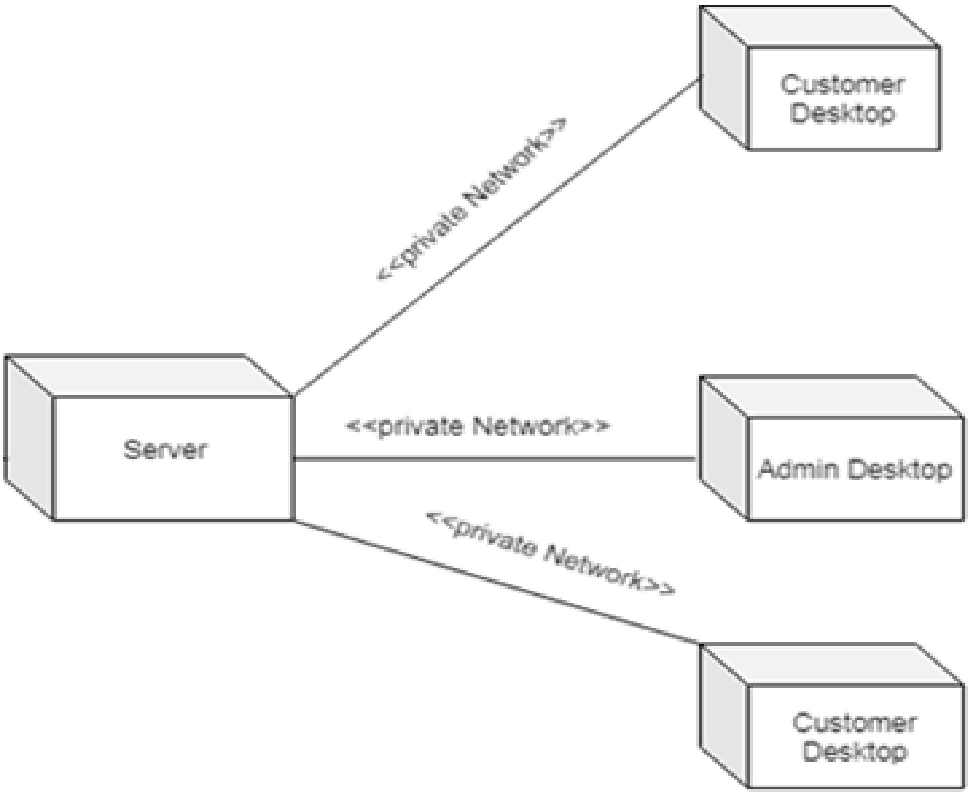


Figure 5.6 Deployment Diagram

5.7 PACKAGE DIAGRAM

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.

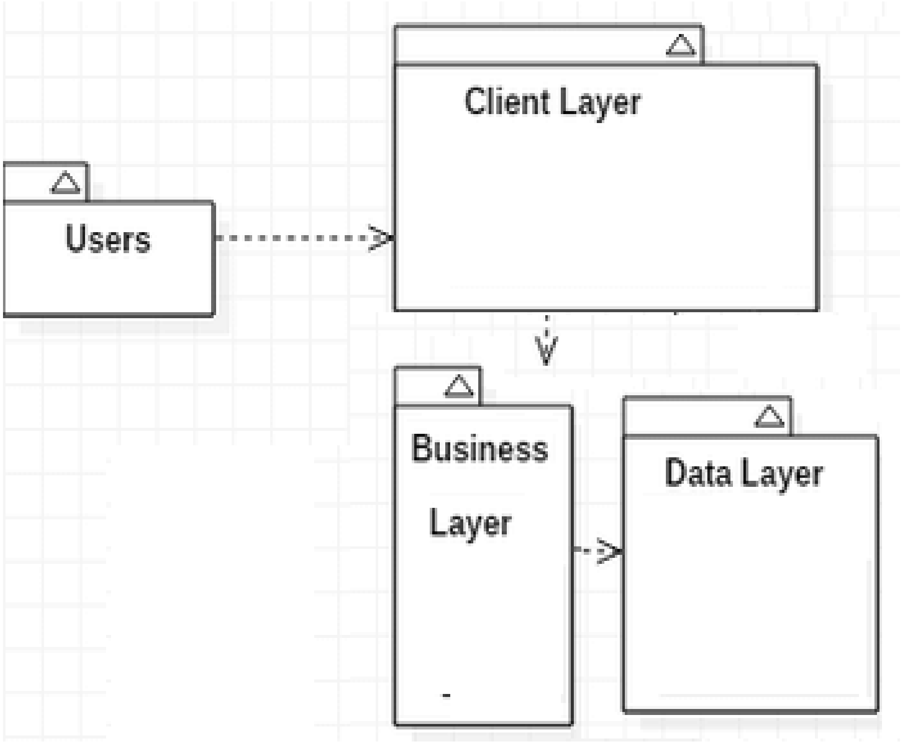


Figure 5.7 Package Diagram

5.8 PROFILE DIAGRAM

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connectors and their components.

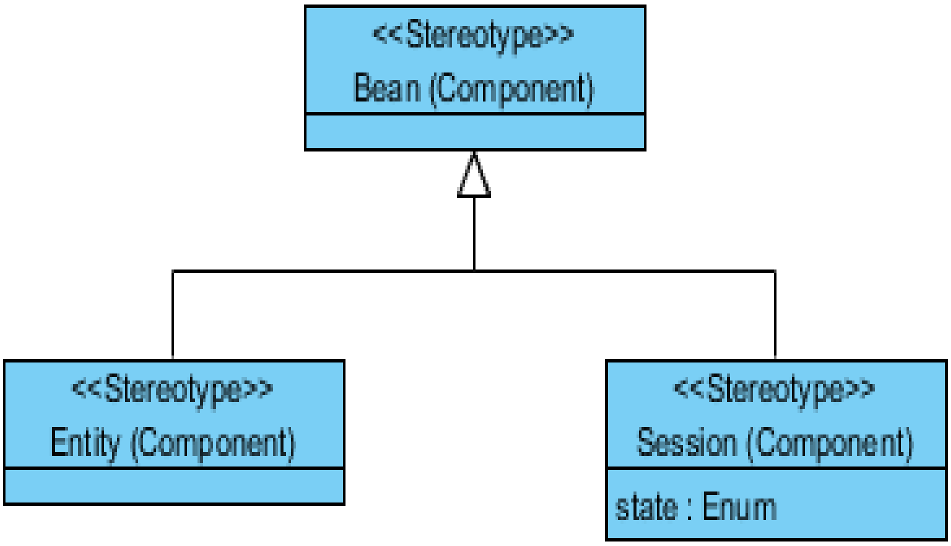


Figure 5.8 Profile Diagram

6. CODING/CODE TEMPLATES

IMPLEMTATION OF CODE

HASHTAG.py

```
from tkinter import messagebox
from tkinter import *
from tkinter import simpledialog
import tkinter
from tkinter import filedialog
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
from tkinter.filedialog import askopenfilename
import json
import networkx as nx
import operator
from RCNN import EncoderCNN, DecoderRNN
import nltk
from nltk.corpus import wordnet
from torchvision import transforms
import re
from PIL import ImageTk, Image
import torch
import pickle
from build_vocab import Vocabulary
matplotlib.use( 'tkagg' )

main = tkinter.Tk()
main.title("Filtering Instagram Hashtags") #designing main screen
main.geometry("1300x1200")
```

```
global filename
attributes = []
mytags = []
global existing_correct
global extension_correct
sorted_a7 = []

device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
pattern = r'[^A-Za-z ]'
regex = re.compile(pattern)

def loadImage(image_path, transform=None):
    image = Image.open(image_path)
    image = image.resize([224, 224], Image.LANCZOS)
    if transform is not None:
        image = transform(image).unsqueeze(0)
    return image

def getUpper(word):
  data = word[0:1]
  data = data.upper();
  data = data+word[1:len(word)]
  return data

def RCNN(filename):
    transform = transforms.Compose([transforms.ToTensor(),
transforms.Normalize((0.485, 0.456, 0.406), (0.229, 0.224, 0.225))])
    with open('model/vocab.pkl', 'rb') as f:
        vocab = pickle.load(f)
    # Build models
    encoder = EncoderCNN(256).eval()  # eval mode (batchnorm uses moving
mean/variance)
```

```
    decoder = DecoderRNN(256, 512, len(vocab), 1)
    encoder = encoder.to(device)
    decoder = decoder.to(device)

    # Load the trained model parameters
    encoder.load_state_dict(torch.load('model/encoder-5-3000.pkl'))
    decoder.load_state_dict(torch.load('model/decoder-5-3000.pkl'))

    # Prepare an image
    image = loadImage(filename, transform)
    image_tensor = image.to(device)

    # Generate an caption from the image
    feature = encoder(image_tensor)
    sampled_ids = decoder.sample(feature)
    sampled_ids = sampled_ids[0].cpu().numpy()          # (1, max_seq_length) -
> (max_seq_length)

    # Convert word_ids to words
    sampled_caption = []
    for word_id in sampled_ids:
        word = vocab.idx2word[word_id]
        sampled_caption.append(word)
        if word == '<end>':
            break
    sentence = ' '.join(sampled_caption)
    sentence = sentence.replace('kite','umbrella')
    sentence = sentence.replace('flying','with')

    image = Image.open(filename)
    plt.imshow(np.asarray(image))
    text.insert(END,"Automatic Extracted Sentence From Image :
"+sentence+"\n\n")
    if len(sentence) > 0:
        length = len(sentence)-5
        sentence = sentence[8:length]
        print(sentence)
    sentence = regex.sub('', sentence)
    for word,pos in nltk.pos_tag(nltk.word_tokenize(str(sentence))):
        if (pos == 'NN' or pos == 'NNP' or pos == 'NNS' or pos == 'NNPS'):
            word = getUpper(word)
            if word not in attributes:
                attributes.append(word.lower())

    text.insert(END,"Extracted Main Attributes From Image:
"+str(attributes)+"\n")

def upload(): #function to upload tweeter profile
    global filename
    text.delete('1.0', END)
    filename = filedialog.askopenfilename(initialdir="imgs")
    text.delete('1.0', END)
    text.insert(END,filename+" loaded\n");

def existing():
    global existing_correct
    attributes.clear()
    mytags.clear()
    G7 = nx.read_pajek('data/img7.net')
    [annotators,tags] = nx.bipartite.sets(G7)
    for val in tags:
        if len(attributes) < 15:
            attributes.append(val)
    for val in tags:
        mytags.append(val)
    text.delete('1.0', END)
    text.insert(END,"Tags for image 7\n\n");
    text.insert(END,str(list(sorted(tags))))
    existing_correct = 0
    G7 = nx.DiGraph(G7)
    [h7,a7] = nx.hits(G7)
    sorted_a7 = sorted(a7.items(),key=operator.itemgetter(1), reverse=True)
    text.delete('1.0', END)
    text.insert(END,filename+" loaded\n");
    for i in range(0,8):
        data = sorted_a7[i];
        if data[0] in tags:
            existing_correct = existing_correct + 1
            text.insert(END,"Existing Correct Annotation : "+data[0]+"\n")
    text.insert(END,"Existing technique Correctly Found Annotation :
"+str(existing_correct)+"\n\n\n")

def extension():
    global extension_correct
    text.delete('1.0', END)
    RCNN(filename)
    temp = []
    extension_correct = 0
    for i in range(len(attributes)):
        for syn in wordnet.synsets(attributes[i].lower()):
            for l in syn.lemmas():
                if l.name() in mytags and l.name not in temp:
                    temp.append(l.name)
                    extension_correct = extension_correct + 1
                    text.insert(END,"Extension Correct Annotation : "+l.name()+"\n")
    if attributes[i] in mytags:
        extension_correct = extension_correct + 1
        text.insert(END,"Extension Correct Annotation : "+attributes[i]+"\n")
    text.insert(END,"Extension technique Correctly Found Annotation : 
"+str(extension_correct)+"\n\n\n")




def graph():
    height = [existing_correct,extension_correct]
    bars = ('Existing Correct Annotation', 'Extension Correct Annotation')
    y_pos = np.arange(len(bars))
    plt.bar(y_pos, height)
    plt.xticks(y_pos, bars)
    plt.show()

font = ('times', 16, 'bold')
title = Label(main, text='Filtering Instagram Hashtags Through Crowdtagging 
and the HITS Algorithm')
title.config(bg='firebrick4', fg='dodger blue')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=0,y=5)

font1 = ('times', 12, 'bold')
text=Text(main,height=20,width=150)
scroll=Scrollbar(text)
text.configure(yscrollcommand=scroll.set)
text.place(x=50,y=120)
text.config(font=font1)



font1 = ('times', 14, 'bold')
```

```
uploadButton = Button(main, text="Upload Image", command=upload,
bg='#ffb3fe')
uploadButton.place(x=50,y=550)
uploadButton.config(font=font1)

modelButton = Button(main, text="Run Existing Technique & Get Annotate
Rank", command=existing, bg='#ffb3fe')
modelButton.place(x=250,y=550)
modelButton.config(font=font1)

runforest = Button(main, text="Run Extension Technique & Get Automatic
Sentence & Annotation", command=extension, bg='#ffb3fe')
runforest.place(x=50,y=600)
runforest.config(font=font1)

rundcnn = Button(main, text="Comparison Graph", command=graph,
bg='#ffb3fe')
rundcnn.place(x=650,y=600)
rundcnn.config(font=font1)
main.config(bg='LightSalmon3')
main. mainloop ()
```

RCNN.py

```
import torch

import torch. nn as nn

import torchvision.models as models

from torch.nn.utils.rnn import pack_padded_sequence

class EncoderCNN(nn.Module):

 #method to load pretrained resnet model
```

```
    def __init__(self, embed_size):
        """Load the pretrained ResNet-152 and replace top fc layer."""
        super(EncoderCNN, self).__init__()
        resnet = models.resnet152(pretrained=True)
        modules = list(resnet.children())[:-1]      # delete the last fc layer.
        self.resnet = nn.Sequential(*modules)
        self.linear = nn.Linear(resnet.fc.in_features, embed_size)
        self.bn = nn.BatchNorm1d(embed_size, momentum=0.01)
    def forward(self, images):
        """Extract feature vectors from input images."""
        with torch.no_grad():
            features = self.resnet(images)
        features = features.reshape(features.size(0), -1)
        features = self.bn(self.linear(features))
        return features
class DecoderRNN(nn.Module):
    def __init__(self, embed_size, hidden_size, vocab_size, num_layers,
  max_seq_length=20):
        """Set the hyper-parameters and build the layers."""
        super(DecoderRNN, self).__init__()
        self.embed = nn.Embedding(vocab_size, embed_size)
        self.lstm = nn.LSTM(embed_size, hidden_size, num_layers,
  batch_first=True)
        self.linear = nn.Linear(hidden_size, vocab_size)
        self.max_seg_length = max_seq_length
```

```
def forward(self, features, captions, lengths):
        """Decode image feature vectors and generates captions."""
        embeddings = self.embed(captions)
        embeddings = torch.cat((features.unsqueeze(1), embeddings), 1)
        packed = pack_padded_sequence(embeddings, lengths, batch_first=True)
        hiddens, _ = self.lstm(packed)
        outputs = self.linear(hiddens[0])
        return outputs


    def sample(self, features, states=None):
        """Generate captions for given image features using greedy search."""
        sampled_ids = []
        inputs = features.unsqueeze(1)
        for i in range(self.max_seg_length):
            hiddens, states = self.lstm(inputs, states)          # hiddens: (batch_size, 1, hidden_size)
            outputs = self.linear(hiddens.squeeze(1))            # outputs:  (batch_size, vocab_size)
            _, predicted = outputs.max(1)                        # predicted: (batch_size)
            sampled_ids.append(predicted)
            inputs = self.embed(predicted)                       # inputs: (batch_size, embed_size)
            inputs = inputs.unsqueeze(1)                         # inputs: (batch_size, 1, embed_size)
        sampled_ids = torch.stack(sampled_ids, 1)                # sampled_ids: (batch_size, max_seq_length)
        return sampled_ids
```

7 PROJECT 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 tests. Each test type addresses a specific testing requirement.

7.1 VARIOUS TESTING CASES

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.

2. INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they 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.

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.

Systems/Procedure: 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.

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.

5. 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.

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.

7. UNIT TESTING

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

7.2 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.

.

8 RESULTS/OUTPUT

Double click on ‘run.bat’ file to get below screen.

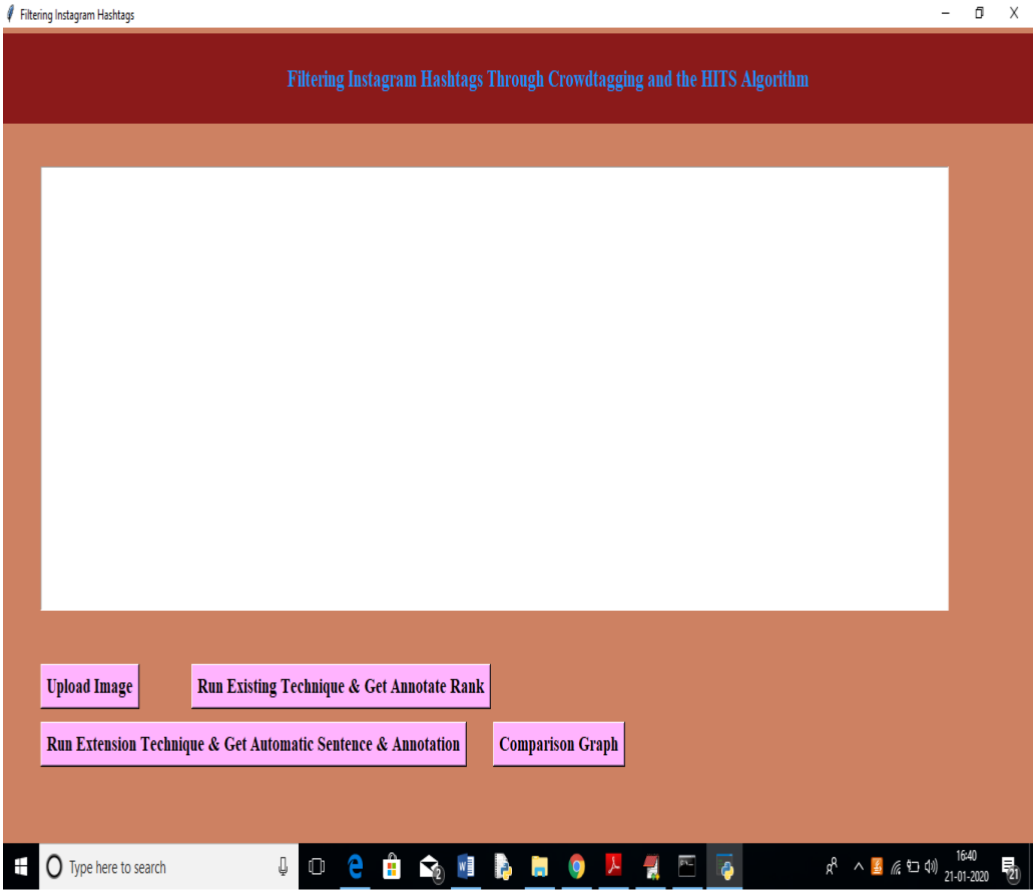

Filtering Instagram Hashtags
Filtering Instagram Hashtags Through Crowdtagging and the HITS Algorithm
Upload Image
Run Existing Technique & Get Annotate Rank
Run Extension Technique & Get Automatic Sentence & Annotation
Comparison Graph
Type here to search
16:40
21-01-2020

Figure 8.1 Output webpage

In above screen click on ‘Upload Image’ button to upload image

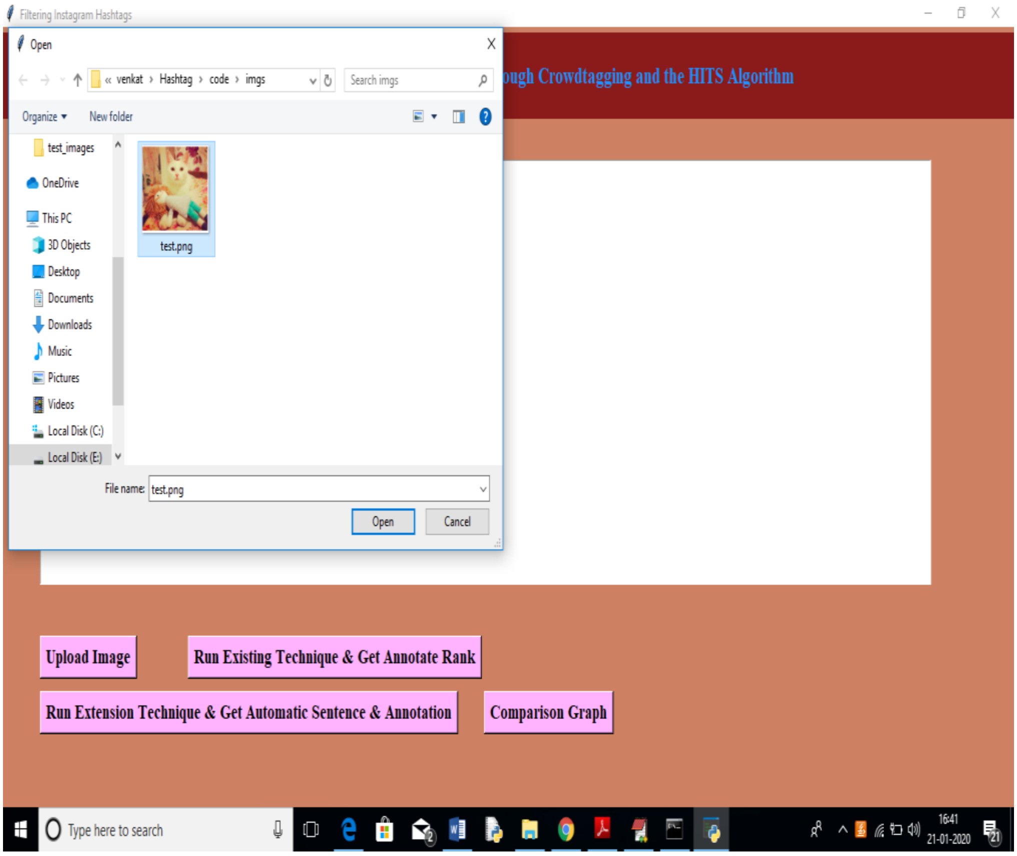
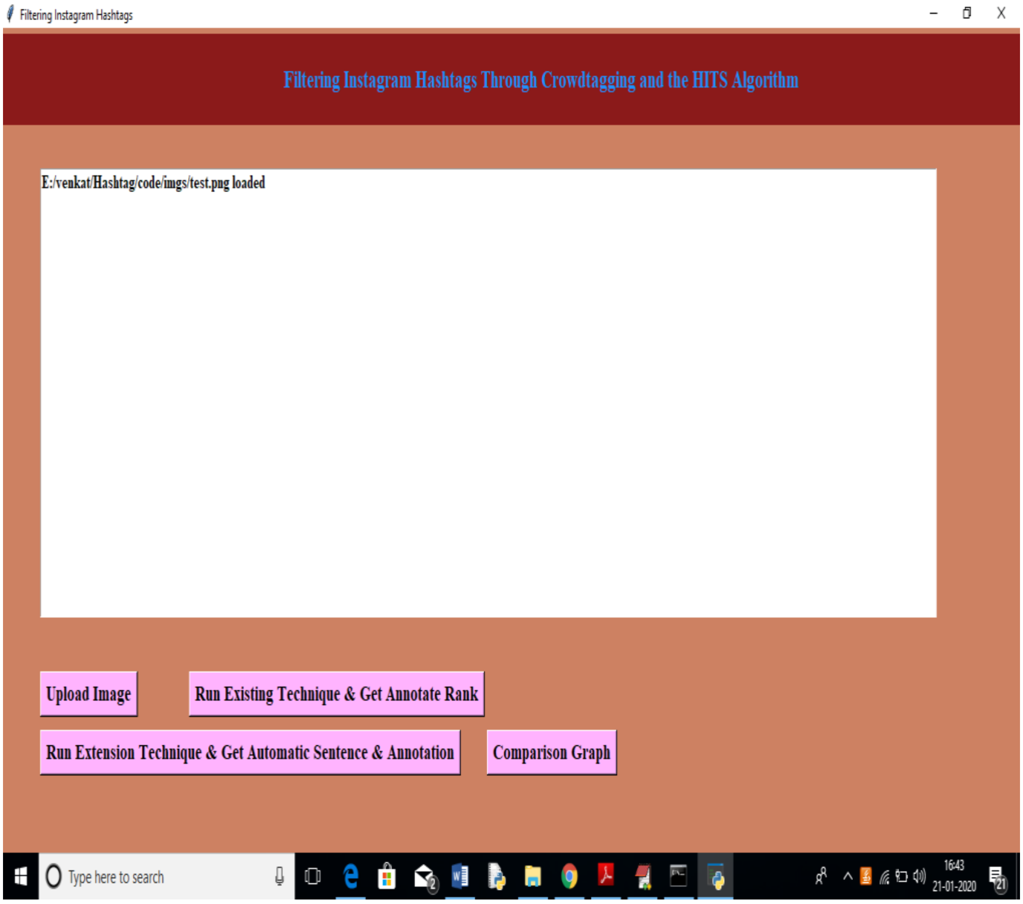


Figure 8.2 Uploading An Image

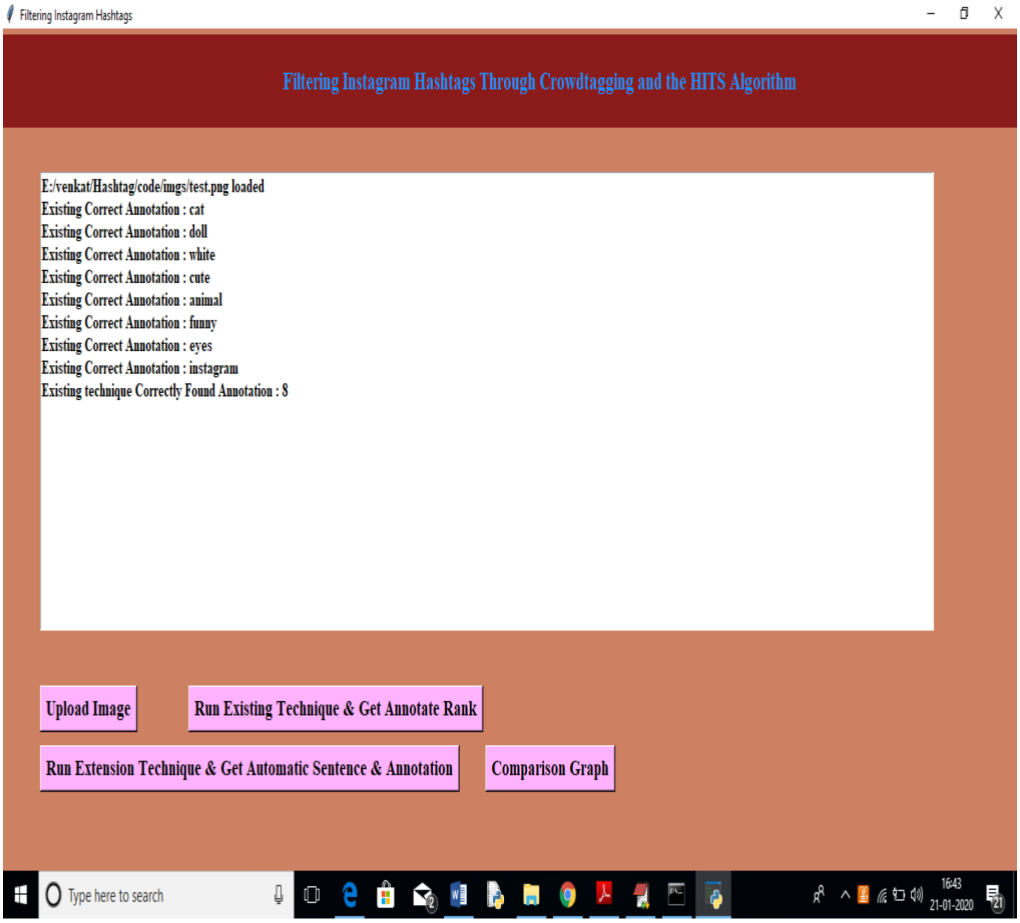
In above screen I am uploading one image and by seeing that image anybody can say that cat or kitten sitting on a bed with some stuff and our extension will describe same sentence or extract same data from image, but existing technique just will check whether given hash tag and annotator tags are similar or relevant or not relevant. After uploading image will get below screen


Filtering Instagram Hashtags
Filtering Instagram Hashtags Through Crowdtagging and the HITS Algorithm
Upload Image
Run Existing Technique & Get Annotate Rank
Run Extension Technique & Get Automatic Sentence & Annotation
Comparison Graph
Type here to search
16:43
21-01-2020

E:/venkat/Hashtag/code/imgs/test.png loaded

Figure 8.3 Click Run Existing Technique and Get Annotation Rank

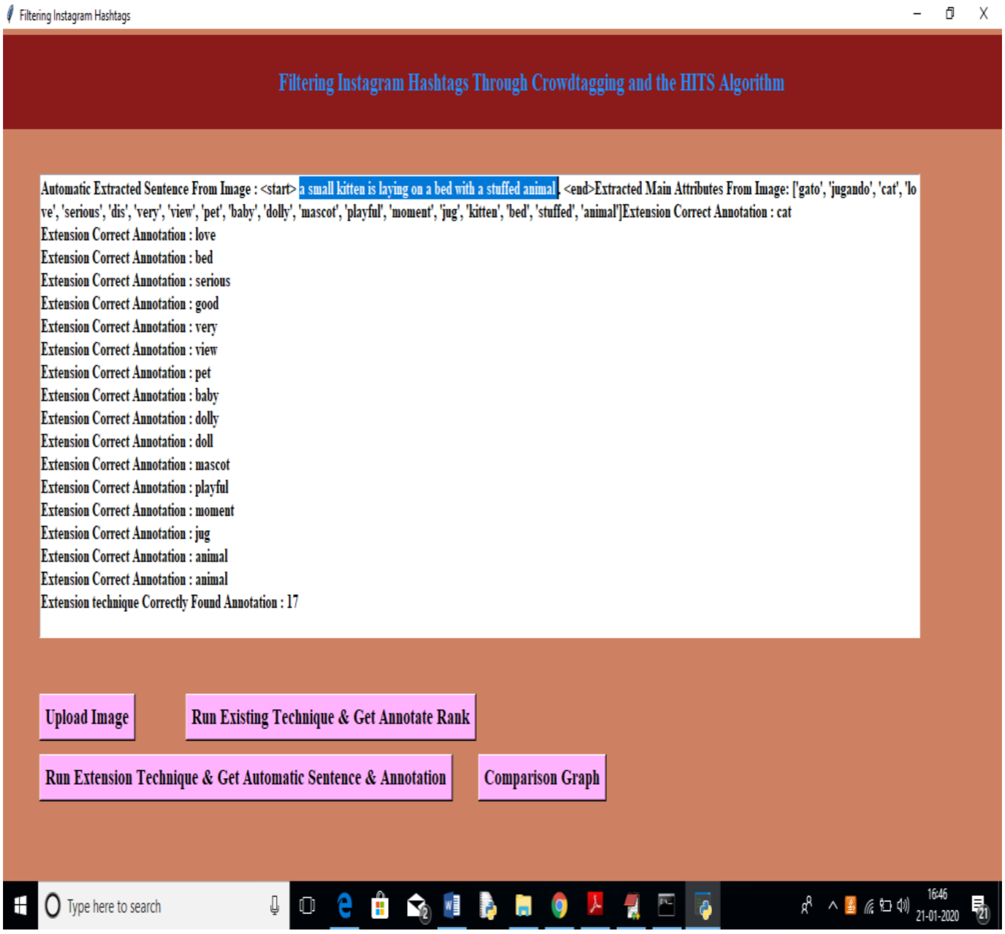
Now click on ‘Run Existing Technique & Get Annotation Rank’ button to get below screen.


Filtering Instagram Hashtags
Filtering Instagram Hashtags Through Crowdtagging and the HITS Algorithm
Upload Image
Run Existing Technique & Get Annotate Rank
Run Extension Technique & Get Automatic Sentence & Annotation
Comparison Graph
Type here to search
16:43
21-01-2020

```
E:/venkat/Hashtag/code/imgs/test.png loaded
Existing Correct Annotation : cat
Existing Correct Annotation : doll
Existing Correct Annotation : white
Existing Correct Annotation : cute
Existing Correct Annotation : animal
Existing Correct Annotation : funny
Existing Correct Annotation : eyes
Existing Correct Annotation : instagram
Existing technique Correctly Found Annotation : 8
```

Figure 8.4 Existing Correct Annotations

In above screen we can see from loaded images above annotations are correct as image contains cat, doll, cute etc. Existing technique able to extract 8 correct annotations from all annotated text. Now click on ‘Run Extension Technique & get Automatic Sentence & Annotation’ button to describe image in sentence and to check extracted words are matching with annotators words or not.

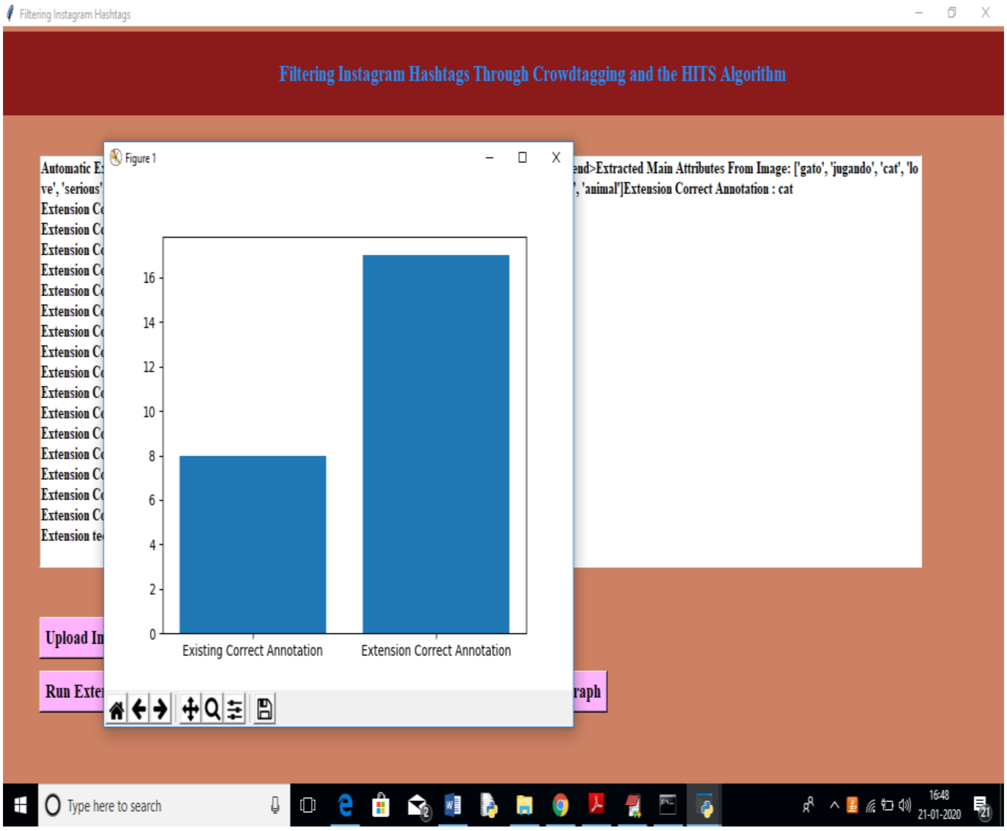


Filtering Instagram Hashtags
Filtering Instagram Hashtags Through Crowdtagging and the HITS Algorithm
Upload Image
Run Existing Technique & Get Annotate Rank
Run Extension Technique & Get Automatic Sentence & Annotation
Comparison Graph
Type here to search
16:46
21-01-2020

```
Automatic Extracted Sentence From Image : <start> a small kitten is laying on a bed with a stuffed animal . <end>Extracted Main Attributes From Image: ['gato', 'jugando', 'cat', 'lo
ve', 'serious', 'dis', 'very', 'view', 'pet', 'baby', 'dolly', 'mascot', 'playful', 'moment', 'jug', 'kitten', 'bed', 'stuffed', 'animal']Extension Correct Annotation : cat
Extension Correct Annotation : love
Extension Correct Annotation : bed
Extension Correct Annotation : serious
Extension Correct Annotation : good
Extension Correct Annotation : very
Extension Correct Annotation : view
Extension Correct Annotation : pet
Extension Correct Annotation : baby
Extension Correct Annotation : dolly
Extension Correct Annotation : doll
Extension Correct Annotation : mascot
Extension Correct Annotation : playful
Extension Correct Annotation : moment
Extension Correct Annotation : jug
Extension Correct Annotation : animal
Extension Correct Annotation : animal
Extension technique Correctly Found Annotation : 17
```

Figure 8.5 Extension Correct Annotation

In above screen in selected text, you can see our extension technique describing image in a sentence and then extracting words from image and compare with annotator's tags to get relevant details. Extension technique able to extract 17 related annotations. Now click on 'Comparison Graph' button to get below graph.


Filtering Instagram Hashtags
Filtering Instagram Hashtags Through Crowdtagging and the HITS Algorithm
Figure 1
16
14
12
10
8
6
4
2
0
Existing Correct Annotation
Extension Correct Annotation
Upload In
Run Exte
raph
Type here to search
16:48
21-01-2020

```
Automatic E                end>Extracted Main Attributes From Image: ['gato', 'jugando', 'cat', 'lo
ve', 'serious'             ', 'animal']Extension Correct Annotation : cat
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension C
Extension te
```

Figure 8.6 Comparison Graph

In above graph x-axis represents technique name and y-axis represents count of extracted matching annotations and we can see extension technique able to extract more related words compare to existing technique.

Note: existing technique can be able to check with only one image as author given only one image details in paper and what other images, he has used that information is not available. But extension technique can work with any image. See another image example.

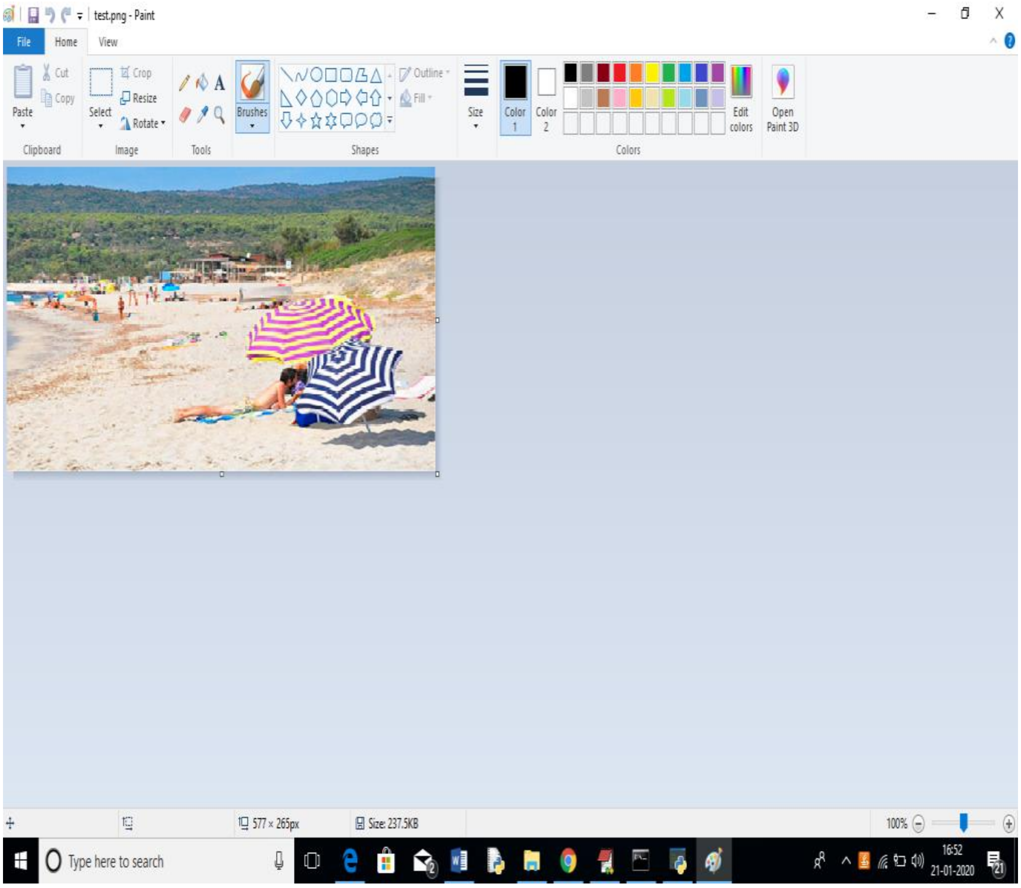


Figure 8.7 Image Comparison

In above image we can see peoples are on beach with umbrellas and extension technique can extract this information but cannot compare with existing technique as author not include this image in his annotation dataset

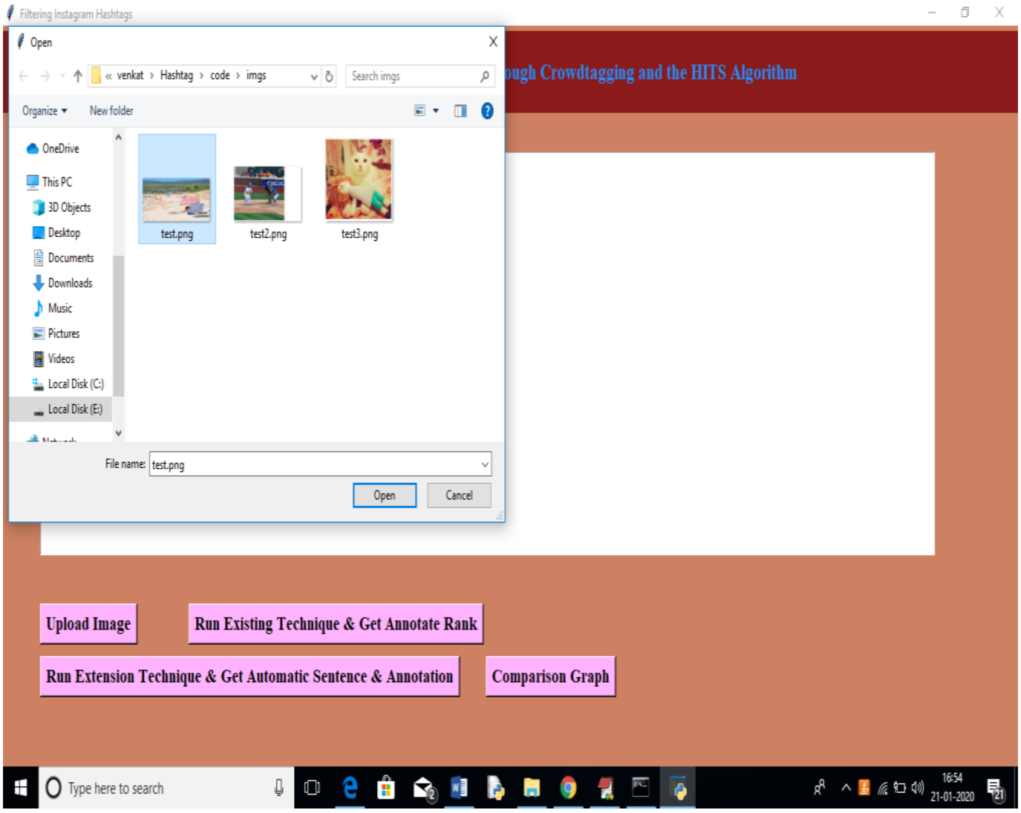
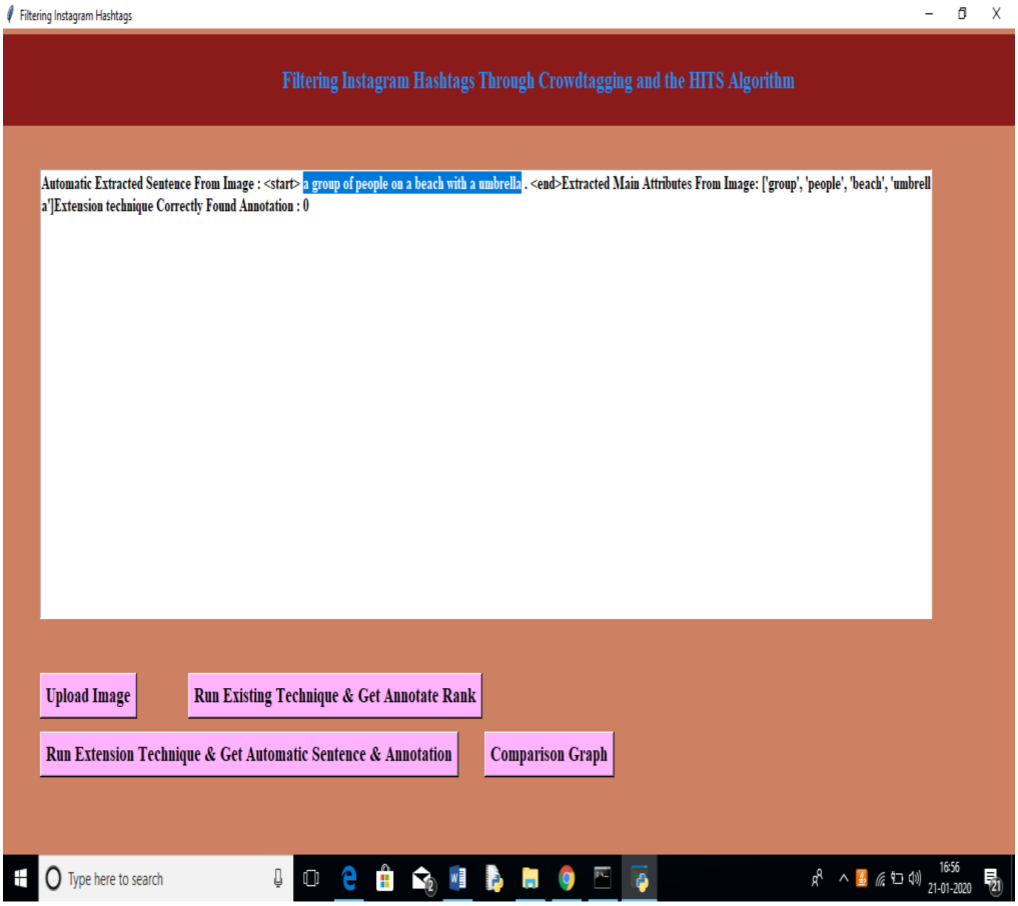


Figure 8.8 Uploading Same Image

In above screen uploading same image and then click on ‘Run Extension Technique & get Automatic Sentence & Annotation’ button to get below results


Filtering Instagram Hashtags
Filtering Instagram Hashtags Through Crowdtagging and the HITS Algorithm
Upload Image
Run Existing Technique & Get Annotate Rank
Run Extension Technique & Get Automatic Sentence & Annotation
Comparison Graph
Type here to search
16:56
21-01-2020

```
Automatic Extracted Sentence From Image : <start> a group of people on a beach with a umbrella . <end>Extracted Main Attributes From Image: ['group', 'people', 'beach', 'umbrell
a']Extension technique Correctly Found Annotation : 0
```

Figure 8.9 Image Description

In above screen in selected text, you can see sentence describing image and its related attributes or hashtag also displaying.

9 CONCLUSIONS

In this project, we have presented an innovative methodology, based on the HITS algorithm and the principles of collective intelligence, for the identification of Instagram hashtags that describe the visual content of the images they are associated with. We have empirically shown that the application of a two-step HITS algorithm in a crowdtagging context provides an easy and effective way to locate pairs of Instagram images and hashtags that can be used as training sets for content-based image retrieval systems in the learning by example paradigm. As a proof of concept, we have used 25 000 evaluations (500 annotations for each one of 50 images) collected from the Figure-eight crowdsourcing platform to create a bipartite graph composed of users (annotators) and the tags they selected to describe the 50 images. The hub scores of the HITS algorithm applied to this graph, called hereby full bipartite graph, give us a measure of the reliability of the annotators. The approach is based on the findings of Theodosiou et al. [39], in which the reliability of annotators is better approximated if we consider all the annotations, they have performed rather than the subset of gold test questions. In the second step, a weighted bipartite graph for each image is composed in the same way as the full bipartite graph. The weights of these graphs are the hub scores computed in the previous step. By thresholding the authority scores of the per image graphs, obtained by the application of the HITS algorithm on the weighted graphs, we can rank and then effectively locate the hashtags that are relevant to their visual content as per the annotator’s evaluation.

Some important findings of this paper are briefly summarized here. The first refers to the value of crowdtagging itself. In several studies before, we found that the crowd can substitute the experts in the evaluation of images with respect to relevant tags. However, even with many annotators (499 in our case), it seems that a perfect agreement between annotators and experts cannot be achieved. It was found that from the 145 different tags suggested for the 50 images used in this paper by the two experts, only 135 were also identified by the 499 annotators.

This leads to a maximum achievable recall value equal to 0.931. Thus, in subjective evaluation tasks, such as those referring to the identification of tags that are related to the visual content of images, no perfect agreement between the experts and the crowd should be expected.

A second finding is that crowdtagging of images can be effectively modeled through user–tag bipartite graphs, one per image. Thresholding the authority score of the HITS algorithm applied on these graphs is a robust way to identify the tags that characterize the visual content of the corresponding images. Getting the top ranked tags based on the authority score is an alternative solution, but, with a little bit lower effectiveness.

A final remark of this project refers to the importance of using weighted user–tag bipartite graphs for the crowdtagged images. It appears that weighting the bipartite graphs with the hub scores of the annotators provides the best results. However, even in the case that the reliability metric of the crowdsourcing platform itself (the _trust variable of Figure-eight in our case) is used to weight the bipartite graphs, the results are not significantly worse. We are a little bit reluctant to generalize this conclusion because, in this paper, we have used too many annotations (499) per image. Thus, one of our future tests will involve a more typical image crowdtagging scenario in which much more images will be used and much fewer (typically less than five) annotations per image will be considered. In that case, only partial coannotation of the same images by the same annotators will take place in contrast to this paper in which all annotators annotated all images.

10. FUTURE ENHANCEMENT

One of our future tests will involve a more typical image crowd tagging scenario in which much more images will be used and much fewer (typically less than five) annotations per image will be considered. In that case, only partial co-annotation of the same images by the same annotators will take place in contrast to this paper in which all annotators annotated all images.

We are currently working to check, in practice, that the image–hashtags pairs mined from the Instagram through the approach described in this paper can be used, indeed, for a large-scale AIA in a content-based image retrieval scenario as proposed by Theodosiou and Tsapatsoulis.

11. REFERENCES

[1] A. Argyrou, S. Giannoulakis, and N. Tsapatsoulis, “Topic modelling on Instagram hashtags: An alternative way to automatic image annotation?” in Proc. 13th Int. Workshop Semantic Social Media Adaptation Personalization, 2018, pp. 61–67.

[2] B. I. Aydin, Y. S. Yilmaz, Y. Li, Q. Li, J. Gao, and M. Demirbas, “Crowdsourcing for multiple-choice question answering,” in Proc. 28th. AAAI Conf. Artif. Intell., 2014, pp. 2946–2953.

[3] C. D. D. Cabrall et al., “Validity and reliability of naturalistic driving scene categorization judgments from crowdsourcing,” Accident Anal. Prevention, vol. 114, pp. 25–33, May 2018.

[4] Q. Cheng, Q. Zhang, P. Fu, C. Tu, and S. Li, “A survey and analysis on automatic image annotation,” Pattern Recognit., vol. 79, pp. 242–259, Jul. 2018.

[5] N. Craswell, “Mean reciprocal rank,” in Encyclopedia of Database Systems. London, U.K.: Springer, 2009, p. 1703.

[6] H. Cui, Q. Li, H. Li, and Z. Yan, “Healthcare fraud detection based on trustworthiness of doctors,” in Proc. Trustcom/BigDataSE/I SPA, 2016, pp. 74–81.

[7] A. R. Daer, R. Hoffman, and S. Goodman, “Rhetorical functions of hashtag forms across social media applications,” in Proc. 32nd ACM Int. Conf. Design Commun. CD-ROM, 2014, Art. no. 16.

[8] E. Ferrara, R. Interdonato, and A. Tagarelli, “Online popularity and topical interests through the lens of instagram,” in Proc. 25th ACM Conf. Hypertext social media, 2014, pp. 24–34.

[9] J. M. Fletcher and T. Wennekers, “From structure to activity: Using centrality measures to predict neuronal activity,” Int. J. Neural Syst., vol. 28, no. 2, 2018, Art. no. 1750013.

A

PROJECT REPORT

On

DRUG DISEASE PREDICTION USING MACHINE LEARNING

Submitted by

Mr. K ROHIT KUMAR REDDY	(17K81A1219)
Mr. RAJESH JALIGMA	(17K81A1217)
Mr. K ROHIT RAY	(17K81A1228)
Mr. M ROHIT REDDY	(17K81A1233)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Dr.MAALYADRI MEDIDA

PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN'S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **DRUG DISEASE PREDICTION USING MACHINE LEARNING**, is being submitted by **K.ROHITH KUMAR REDDY (17K81A1219), RAJESH JALIGAMA (17K81A1217),K.ROHIT RAY (17K81A1228), M.ROHIT REDDY (17K81A1233)**in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN** INFORMATION TECHNOLOGY is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Project Guide	Head of the Department
Dr.MAALYADRI MEDIDA	**DR.R.NAGARAJU**
Department of Information Technology	Department of Information Technology

Internal Examiner

External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of Information Technology, session: 2017 – 2021, St. Martin’s Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **Drug Disease Prediction Using Machine Learning** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

K.ROHITH KUMAR REDDY	**(17K81A1219)**
RAJESH JALIGAMA	**(17K81A1217)**
K.ROHIT RAY	**(17K81A1228)**
M.ROHIT REDDY	**(17K81A1233)**

LASYA INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
TECHNOLOGY FOR PROSPERITY
SMEC
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **J.RAJESH** WITH ROLL NO.**17K81A1217**, **K.ROHIT RAY** WITH ROLL NO.**17K81A1228**, **K.ROHITH KUMAR REDDY** WITH ROLL NO.**17K81A1219**, **M.ROHIT REDDY** WITH ROLL NO.**17K81A1233**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**DRUG DISEASE PREDICTION USING MACHINE LEARNING**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd. ★ Kompally-500014 ★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin’s Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr.Mr.R.Nagaraju**, Head of the Department, Information Technology , St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **DR.MAALYADRI MEDIDA** Professor, Department of Information Technology, St. Martin’s Engineering College, Dhulapally, for her support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

K.ROHITH KUMAR REDDY	**(17K81A1219)**
RAJESH JALIGAMA	**(17K81A1217)**
K.ROHIT RAY	**(17K81A1228)**
M.ROHIT REDDY	**(17K81A1233)**

TABLE OF CONTENTS

ABSTRACT

Using Machine learning, our project proposes disease prediction system. For small problems, the users have to go personally to the hospital for check-up which is more time consuming. Also handling the telephonic calls for appointments is quite hectic. Such a problem can be solved by using disease prediction application by giving proper guidance regarding healthy living. Over the past decade, the use of the specific disease prediction tools along with the concerning health has been increased due to a variety of diseases and less doctor-patient ratio. Thus, in this system, we are concentrating on providing immediate and accurate disease prediction to the users about the symptoms they enter along with the severity of disease predicted. Best suitable algorithm and doctor consultation will be given in this project. For prediction of diseases, different machine learning algorithms are used to ensure quick and accurate predictions. In one channel, the symptoms entered will be crosschecked with the database. Further, it will be preserved in the database if the symptom is new which its primary work is and the other channel will provide severity of disease predicted. A web/android application is deployed for user for easy portability, configuring and being able to access remotely where doctors cannot reach easily. Normally users are not aware about all the treatment regarding the particular disease, this project also looks forward to providing medicine and drug consultation of disease predicted. Therefore, this arrangement helps in easier health management.

LIST OF FIGURES

1.INTRODUCTION

Drug research and development is a complex, lengthy and expensive process. It often takes 10-15 years of research and 0.8-15 billion dollars to make a drug from abstract concept to market-ready product . Annually, 90% of drugs fail to get access to FDA evaluations, thereby preventing their use in actual therapy . Accordingly, Drug Repositioning (DR) based on computing method appears. The repositioning method bypasses many pre-approval tests that are critical to newly developed therapeutic compounds, and it can shorten the drug development cycle to 3-12 years for a repositioned drug. In recent years, DR has received increased interest from governments, nongovernmental agencies and academic researchers

1.1 PROJECT OVERVIEW

Drug and Disease features were obtained by querying open linked data to train our classifier for predicting new drug indications, and the predictive performance of the classifier for different validation schemes was evaluated. We collected the drug and disease data from Bio2RDF, an open source project that uses semantic web technologies to link data from multiple sources. A binary feature matrix was generated using drug target, substructure and side effects and disease ontology terms. We collected a broader collection of data containing 816 drugs and 1393 diseases with their features and gold standard data we generated by combining multiple drug indication data sources. We tried our method on a different dataset, compiled by other researchers, that confirmed the predictive value of our method independent of the primary data. A crucial flaw in the typical evaluation scheme for drug indication predictions that would yield unrealistic predictions is to fail to consider the paired nature of inputs. We partitioned the data in distinct training and test sets where not only pairs but also drugs/diseases are were not overlapped. We tested several classifiers under different cross validation schemes and compared our approach with existing methods. We observed that our model had better predictive performance than the existing models in disjoint cross-validation settings. Keywords: linked open data, SPARQL, drug repositioning, machine learning, drug indication prediction**.**

1.2 PROJECT OBJECTIVE

Drug research and development is a complex, lengthy and expensive process. It often takes 10-15 years of research and 0.8-15 billion dollars to make a drug from abstract concept to market-ready product . Annually, 90% of drugs fail to get access to FDA evaluations, thereby preventing their use in actual therapy . Accordingly, Drug Repositioning (DR) based on computing method appears. The repositioning method bypasses many pre-approval tests that are critical to newly developed therapeutic compounds, and it can shorten the drug development cycle to 3-12 years for a repositioned drug. In recent years, DR has received increased interest from governments, nongovernmental agencies and academic researchers. In general, DR seeks to find new uses for existing drugs, with established and demonstrated human safety. In technical terminology, DR is the process by which new indications are found for approved drugs.Recently, the usage The associate editor coordinating the review of this manuscript and approving it for publication was Ying Song. of computational DR in drug discovery has become a popular practice, and an increasing number of machine learning , network analysis , text mining and semantic inference methods have been proposed

1.3 SCOPE OF THE PROJECT

The Earth is passing through a purplish patch of technology, where there is increasing demand of intelligence and accuracy behind it. Today’s people are more likely addicted to Internet but they are not concerned about their personal health. In this 21st Century humans are surrounded with technology as they are the constituent of our day to day life cycle.So,We are proposing such a system which will flaunt a simple and elegant User Interface and also be time efficient. In order to make it less time consuming we are aiming at a more specific questionnaire which will be followed by the system. Our aim with this system is to be the connecting bridge between doctors and patients. The main feature will be the machine learning, in which we will be using algorithms such as Naïve Bayes Algorithm, K-Nearest Algorithm, Decision Tree Algorithm, Random Forest Algorithm and Support Vector Machine, which will help us in getting accurate predictions and Also, will find which algorithm gives a faster and efficient result by comparatively-comparing. Another feature that our system will comprise of is Doctor’s Consultation. After delivering the results, our system will also suggest the user to get a doctors consultation on this report.By using this feature, we will not only address the other class of users i.e. the Doctors but we will also gain their trust in this system as in that this system is not affecting their business.

1.4 ORGANIZATION OF CHAPTERS

1.4.1 INTRODUCTION

The Earth is passing through a purplish patch of technology, where there is increasing demand of intelligence and accuracy behind it. Today’s people are more likely addicted to Internet but they are not concerned about their personal health. In this 21st Century humans are surrounded with technology as they are the constituent of our day to day life cycle. With this we are always focusing on the health for ourselves and our earned valuables respectively. People avoid to go in hospital for small problem which may become a major disease in future. Establishing question answer forums is becoming a simple way to answer those queries rather than browsing through the list of potentially relevant document from the web. Our basic idea is to develop a system which will predict and give the details of the disease predicted along with its severity which as symptoms are given as input by the user. The system will compare the symptoms with the datasets provided in the database. If the symptom matches the datasets then it should ask other relevant symptoms specifying the name of the symptom. If not, the symptom entered should be notified as wrong symptom. After this a prompt will come up asking whether you want to still save the symptom in the database. If you click on yes, it will be saved in the database, if not it will go to the recycle bin. The main feature will be the machine learning, in which we will be using algorithms such as Naïve Bayes Algorithm, K-Nearest Algorithm, Decision Tree Algorithm, Random Forest Algorithm and Support Vector Machine, which will predict accurate disease and Also, will find which algorithm gives a faster and efficient result by comparatively-comparing.

1.4.2 LITERATURE SURVEY

“Prediction of Cardiovascular Disease Using Machine Learning Algorithms” (2018).

This paper contributes the correlative application and analysis of distinct machine learning algorithms in the R software which gives an immediate mechanism for the user to use the machine learning algorithms in R software for forecasting the cardiovascular diseases. “A Proposed Model for Lifestyle Disease Predict Vectorion Using Support Machine” (2018).

This study aims to understand support vector machine and use it to predict lifestyle diseases that an individual might be susceptible to.

“Multi Disease Prediction Using Data Mining Techniques” (2017).

In this study two different data mining classification techniques was used for the prediction of various diseases and their performance was compared in order to evaluate the best classifier. An important challenge in data mining and machine learning areas is to build precise and computationally efficient classifiers for Medical applications.

“Prediction of Heart Disease Using Machine Learning Algorithms” (2018).

In this paper, two supervised data mining algorithm was applied on the dataset to predict the possibilities of having heart disease of a patient, were analyzed with classification model namely Naïve Bayes Classifier and Decision tree classification. The Decision tree model has predicted the heart disease patient with an accuracy level of 91% and Naïve Bayes classifier has predicted heart disease patient with an accuracy level of 87%.

1.4.4 SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements

For developing the application the following are the Software Requirements:

Operating Systems supported

1. Windows 7 and above.

Technologies and Languages used to Develop

1. Python

Debugger and Emulator

- Any Browser (Particularly Chrome)

Hardware Requirements

For developing the application the following are the Hardware Requirements:

- Processor: Pentium IV or higher
- RAM:8GB
- Hard Disk:1TB

1.4.4 SOFTWARE DEVELOPMENT ANALAYASIS

Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It is used for web development (server-side),software development,mathematics,system scripting.

Machine Learning

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.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.

1.4.5 PROJECT SYSTEM DESIGN

1.USER :

Upload Your Symptoms : Using this module user will upload his Symptoms .

2. APPLICATION :

Based on the symptoms disease of the user will be predicted by machine learning algorithms and then the system will suggests drugs related to the predicted disease.

1.4.6 PROJECT CODING

Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It is used for web development (server-side),software development,mathematics,system scripting.

Tkinter

Out of all the GUI methods, **tkinter** is the most commonly used method. It is a standard **Python** interface to the Tk GUI toolkit shipped with **Python**. **Python** with **tkinter** is the fastest and easiest way to create the GUI applications.

NumPy

NumPy is the fundamental package for scientific computing in **Python**. ... **NumPy** arrays facilitate advanced mathematical and other types of operations on large numbers of data. Typically, such operations are executed more efficiently and with less code than is possible using **Python's** built-in sequences.

Decision Tree

Decision Trees are a type of Supervised **Machine Learning** where the data is continuously split according to a certain parameter. The **tree** can be explained by two entities, namely **decision** nodes and leaves

Support Vector Machine

"**Support Vector Machine**" (**SVM**) is a supervised **machine learning** algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems.

1.4.7 PROJECT 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.

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.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program

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.

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.

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.

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.

1.4.8 INPUT SCREENS

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

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.

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.

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.

1.4.9 OUTPUT SCREENS

The **design** of **output** is the most important task of any system. During **output design**, developers identify the type of **outputs** needed, and consider the necessary **output** controls and prototype report layouts.

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.

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.Select methods for presenting information.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.

1.4.10 CONCLUSIONS

This project gives research of multiple researches done in this field. Our Proposed System aims at bridging gap between Doctors and Patients which will help both classes of users in achieving their goals. This system provides support for multiple disease prediction using different Machine Learning algorithms. The present approach of many systems focuses only on automating this process which lacks in building the user’s trust in the system. By providing Doctor’s recommendation in our system, we ensure user’s trust side by side ensuring that the Doctor’s will not feel that their Business is getting affected due to this System.

2 . LITERATURE SURVEY

2.1 SURVEY ON BACKGROUND

“Prediction of Cardiovascular Disease Using Machine Learning Algorithms” (2018).

This paper contributes the correlative application and analysis of distinct machine learning algorithms in the R software which gives an immediate mechanism for the user to use the machine learning algorithms in R software for forecasting the cardiovascular diseases.

“A Proposed Model for Lifestyle Disease Predict Vectorion Using Support Machine” (2018).

This study aims to understand support vector machine and use it to predict lifestyle diseases that an individual might be susceptible to.

“Multi Disease Prediction Using Data Mining Techniques” (2017).

In this study two different data mining classification techniques was used for the prediction of various diseases and their performance was compared in order to evaluate the best classifier. An important challenge in data mining and machine learning areas is to build precise and computationally efficient classifiers for Medical applications.

“Prediction of Heart Disease Using Machine Learning Algorithms” (2018).

In this paper, two supervised data mining algorithm was applied on the dataset to predict the possibilities of having heart disease of a patient, were analyzed with classification model namely Naïve Bayes Classifier and Decision tree classification. The Decision tree model has predicted the heart disease patient with an accuracy level of 91% and Naïve Bayes classifier has predicted heart disease patient with an accuracy level of 87%.

“Analysis of Heart Disease Prediction Using Datamining Techniques”(2017)

Heart disease is one of the leading causes of deaths worldwide and the early prediction of heart disease is very important.

In this study prove that the proposed new algorithm achieves a highest accuracy compare with another algorithm.

“Review of Medical Disease Symptoms Prediction Using Data Mining Technique”(2017)

In this paper evaluate the performance of medical disease prediction based on data mining technique. The classifier classified the medical diagnosis of disease data such as cancer, liver problem, and heart disease and so on. SVM method better classified data in compression of conventional cluster ensemble technique.

2.2 CONCLUSIONS ON SURVEY

This study aims to understand support vector machine and use it to predict lifestyle diseases that an individual might be susceptible to.In this study two different data mining classification techniques was used for the prediction of various diseases and their performance was compared in order to evaluate the best classifier. An important challenge in data mining and machine learning areas is to build precise and computationally efficient classifiers for Medical applications.In this paper, two supervised data mining algorithm was applied on the dataset to predict the possibilities of having heart disease of a patient, were analyzed with classification model namely Naïve Bayes Classifier and Decision tree classification. The Decision tree model has predicted the heart disease patient with an accuracy level of 91% and Naïve Bayes classifier has predicted heart disease patient with an accuracy level of 87%.In this paper evaluate the performance of medical disease prediction based on data mining technique. The classifier classified the medical diagnosis of disease data such as cancer, liver problem, and heart disease and so on. SVM method better classified data in compression of conventional cluster ensemble technique.

3.SOFTWARE AND HARDWARE REQUIREMENTS

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

REQUIREMENT SPECIFICATION

Functional Requirements

- Graphical User interface with the User.

3.1 SOFTWARE REQUIREMENTS

For developing the application the following are the Software Requirements:

Operating Systems supported

- Windows 7 and above

Technologies and Languages used to Develop

- Python

Debugger and Emulator

- Any Browser (Particularly Chrome)

3.2 HARDWARE REQUIREMENTS

For developing the application the following are the Hardware Requirements:

- Processor: Pentium IV or higher
- RAM: 8GB
- Hard Disk: 1 TB

4 . SOFTWARE DEVELOPMENT ANALAYASIS

4.1.EXISTING SYSTEM

In the existing system the data set is typically small, for patients and diseases with specific conditions. These systems are mostly designed for the more colossal diseases such as Heart Disease, Cancer etc. The pre-selected characteristics may sometimes not satisfy the changes in the disease and its influencing factors which could lead to inaccuracy in results. As we live in continuously evolving world, the symptoms of diseases also evolve over a course of time. Also most of the current systems make the users wait for long periods by making them answer lengthy questionnaires.

4.2. PROPOSED SYSTEM

We are proposing such a system which will flaunt a simple and elegant User Interface and also be time efficient. In order to make it less time consuming we are aiming at a more specific questionnaire which will be followed by the system. Our aim with this system is to be the connecting bridge between doctors and patients. The main feature will be the machine learning, in which we will be using algorithms such as Naïve Bayes Algorithm, K-Nearest Algorithm, Decision Tree Algorithm, Random Forest Algorithm and Support Vector Machine, which will help us in getting accurate predictions and Also, will find which algorithm gives a faster and efficient result by comparatively-comparing. Another feature that our system will comprise of is Doctor’s Consultation. After delivering the results, our system will also suggest the user to get a doctors consultation on this report.By using this feature, we will not only address the other class of users i.e. the Doctors but we will also gain their trust in this system as in that this system is not affecting their business.

SYSTEM ARCHITECTURE

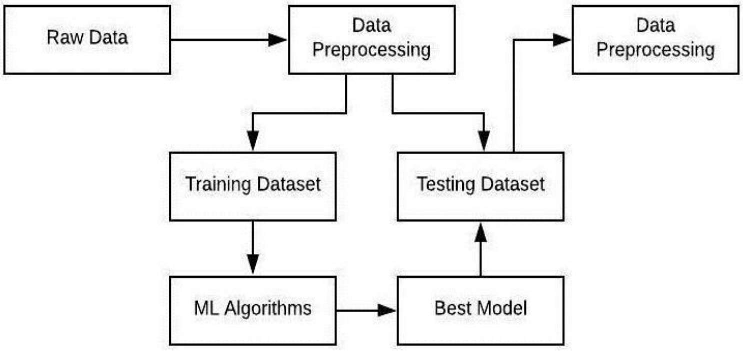


FIG.4.3 SYSTEM ARCHITECTURE

As shown in the above figure, the raw data from the original dataset is passed onto the first phase i.e. Data pre-processing. In Data pre-processing this raw data is then cleaned of all redundancies, missing values etc. The new clean data is fit for training different algorithmic models on it.

The process of training models is fundamental process in Machine learning Projects. There are two approaches to machine learning mainly Supervised Learning and Unsupervised Learning. Our model mostly applies the first approach initially. i.e. Supervised Learning.

Now in Supervised Learning, the system is trained on some examples i.e. Training set and then the model is asked to predict new values based on the test set.

The partitioning of dataset becomes crucial for getting good accuracy in models. The percentage mostly used while partitioning is 80/20 .i.e. 80% for training and 20% for testing purposes.

In our system we aim at first applying different algorithms on the training dataset and based on the model's Confidence and testing dataset accuracy, we select the best model algorithm and apply it on testing dataset to generate accurate results.

4.3 MODULE FUNCTIONALITY

1.USER :

Upload Your Symptoms : Using this module user will upload his Symptoms .

2. APPLICATION :

And based on the symptoms disease of the user will be predicted by machine learning algorithms and then the system will suggests drugs related to the predicted disease.

5.PROJECT SYSTEM DESIGN

5.1 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.

2. Provide extendibility and specialization mechanisms to extend the core concepts.

3. Be independent of particular programming languages and development process.

4. Provide a formal basis for understanding the modeling language.

5. Encourage the growth of OO tools market.

6. Support higher level development concepts such as collaborations, frameworks, patterns and components.

7. Integrate best practices.

USE-CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

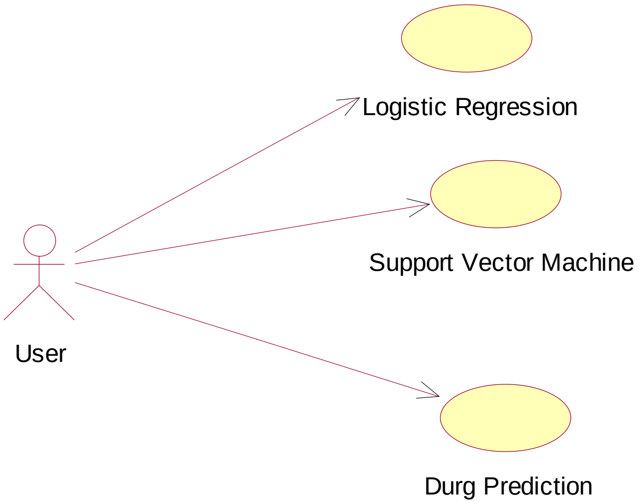


FIG.5.1.1 USE CASE DIAGRAM

CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

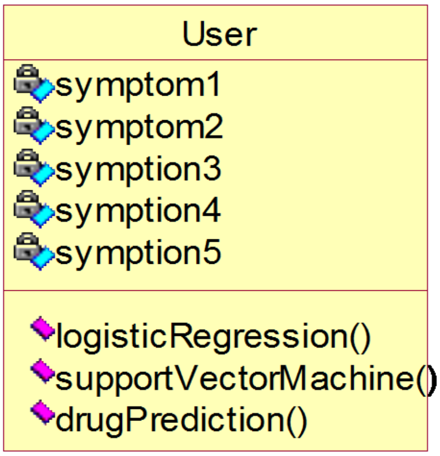


FIG.5.1.2 CLASS DIAGRAM

SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

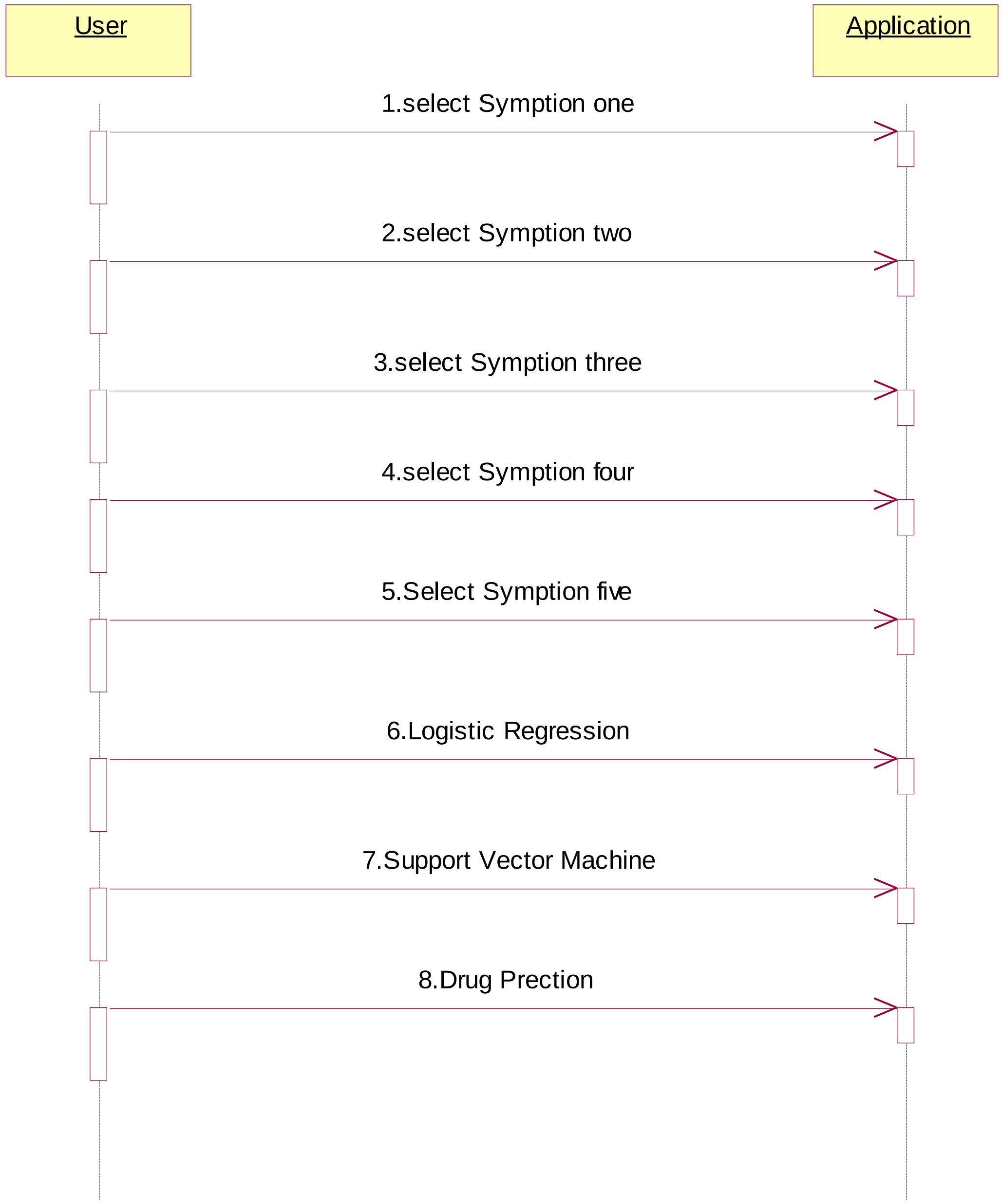


FIG.5.1.3 SEQUENCE DIAGRAM

COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular **use** case and define the role of each object.

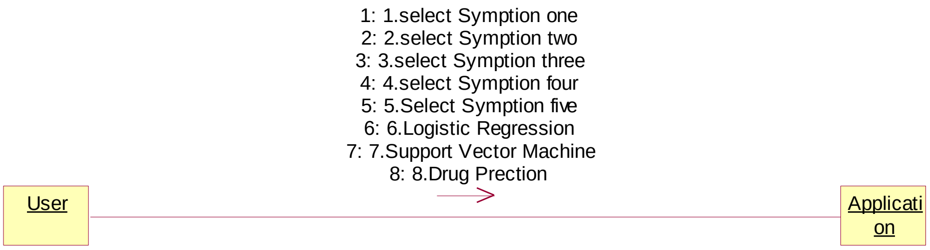


FIG.5.1.4 COLLABORATION DIAGRAM

6. PROJECT CODING

6.1 TECHNOLOGY

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. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython 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

What is Python

Python is a popular programming language. It was created by Guido van Rossum, 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.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.

- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

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 Quickstart

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":

Virtual Environments and Packages

Introduction

Python applications will often use packages and modules that don’t come as part of the standard library. Applications will sometimes need a specific version of a library, because the application may require that a particular bug has been fixed or the application may be written using an obsolete version of the library’s interface.

This means it may not be possible for one Python installation to meet the requirements of every application. If application A needs version 1.0 of a particular module but application B needs version 2.0, then the requirements are in conflict and installing either version 1.0 or 2.0 will leave one application unable to run.

The solution for this problem is to create a virtual environment, a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.

Different applications can then use different virtual environments. To resolve the earlier example of conflicting requirements, application A can have its own virtual environment with version 1.0 installed while application B has another virtual environment with version 2.0. If application B requires a library be upgraded to version 3.0, this will not affect application A's environment.

Creating Virtual Environments

The module used to create and manage virtual environments is called venv. venv will usually install the most recent version of Python that you have available. If you have multiple versions of Python on your system, you can select a specific Python version by running python3 or whichever version you want.

To create a virtual environment, decide upon a directory where you want to place it, and run the venv module as a script with the directory path:

```
python3 -m venv tutorial-env
```

This will create the tutorial-env directory if it doesn't exist, and also create directories inside it containing a copy of the Python interpreter, the standard library, and various supporting files.

A common directory location for a virtual environment is .venv. This name keeps the directory typically hidden in your shell and thus out of the way while giving it a name that explains why the directory exists. It also prevents clashing with .env environment variable definition files that some tooling supports.

Once you've created a virtual environment, you may activate it.

On Windows, run:

```
tutorial-env\Scripts\activate.bat
```

On Unix or MacOS, run:

```
source tutorial-env/bin/activate
```

(This script is written for the bash shell. If you use the csh or fish shells, there are alternate activate.csh and activate.fish scripts you should use instead.)

Activating the virtual environment will change your shell's prompt to show what virtual environment you're using, and modify the environment so that running python will get you that particular version and installation of Python. For example:

```
$ source ~/envs/tutorial-env/bin/activate

(tutorial-env) $ python

Python 3.5.1 (default, May  6 2016, 10:59:36)

  ...

>>> import sys

>>>sys.path

['', '/usr/local/lib/python35.zip', ...,

'~/envs/tutorial-env/lib/python3.5/site-packages']

>>>
```

12.3. Managing Packages with pip

You can install, upgrade, and remove packages using a program called pip. By default pip will install packages from the Python Package Index, <https://pypi.org>. You can browse the Python Package Index by going to it in your web browser, or you can use pip's limited search feature:

```
 (tutorial-env) $ pip search astronomy

skyfield              - Elegant astronomy for Python

gary                  - Galactic astronomy and gravitational dynamics.

novas                 - The United States Naval Observatory NOVAS astronomy library
```

astroobs - Provides astronomy ephemeris to plan telescope observations

PyAstronomy - A collection of astronomy related tools for Python.

Machine Learning

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.

6.2 MACHINE LEARNING ALGORITHMS

DECISION TREE LEARNING

Decision Tree is a **Supervised learning technique** that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where **internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome.**In a Decision tree, there are two nodes, which are the **Decision Node** and **Leaf Node.** Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.The decisions or the test are performed on the basis of features of the given dataset.It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.In order to build a tree, we use the **CART algorithm,** which stands for **Classification and Regression Tree algorithm.**A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

RANDOM FOREST

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of **ensemble learning,** which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.As the name suggests, **"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."** Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.**The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.**

NAIVE BAYES CLASSIFIER

Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems.It is mainly used in text classification that includes a high-dimensional training dataset.Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.**It is a probabilistic classifier, which means it predicts on the basis of the probability of an object**.Some popular examples of Naïve Bayes Algorithm are **spam filtration, Sentimental analysis, and classifying articles.**

SUPPORT VECTOR MACHINE

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression

problems. However, primarily, it is used for Classification problems in Machine Learning.The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

6.3 CODE IMPLEMENTATION

```
import tkinter as tk

from tkinter import LabelFrame, Label,END,
              Tk,StringVar,LEFT,Entry,OptionMenu,W,Button,Text

import numpy as np

import pandas as pd

l1=['back_pain','constipation','abdominal_pain','diarrhoea','mild_fever','yellow_urine',

'yellowing_of_eyes','acute_liver_failure','fluid_overload','swelling_of_stomach',

'swelled_lymph_nodes','malaise','blurred_and_distorted_vision','phlegm','throat_irritation',

'redness_of_eyes','sinus_pressure','runny_nose','congestion','chest_pain','weakness_in_limbs',

'fast_heart_rate','pain_during_bowel_movements','pain_in_anal_region','bloody_stool',

'irritation_in_anus','neck_pain','dizziness','cramps','bruising','obesity','swollen_legs',

'swollen_blood_vessels','puffy_face_and_eyes','enlarged_thyroid','brittle_nails',

'swollen_extremeties','excessive_hunger','extra_marital_contacts','drying_and_tingling_lips',

'slurred_speech','knee_pain','hip_joint_pain','muscle_weakness','stiff_neck','swelling_joints',

'movement_stiffness','spinning_movements','loss_of_balance','unsteadiness',
'weakness_of_one_body_side','loss_of_smell','bladder_discomfort','foul_smell_of urine',

'continuous_feel_of_urine','passage_of_gases','internal_itching','toxic_look_(typhos)',

'depression','irritability','muscle_pain','altered_sensorium','red_spots_over_body','belly_pain',

'abnormal_menstruation','dischromic _patches','watering_from_eyes','increased_appetite','polyuria','family_history','mucoid_sputum',

'rusty_sputum','lack_of_concentration','visual_disturbances','receiving_blood_transfusion',

'receiving_unsterile_injections','coma','stomach_bleeding','distention_of_abdomen',

'history_of_alcohol_consumption','fluid_overload','blood_in_sputum','prominent_veins_on_calf',

'palpitations','painful_walking','pus_filled_pimples','blackheads','scurring','skin_peeling',

'silver_like_dusting','small_dents_in_nails','inflammatory_nails','blister','red_sore_around_nose',

'yellow_crust_ooze']

disease=['Fungal infection','Allergy','GERD','Chronic cholestasis','Drug Reaction',

'Peptic ulcer diseae','AIDS','Diabetes','Gastroenteritis','Bronchial Asthma','Hypertension',

' Migraine','Cervical spondylosis',

'Paralysis (brain hemorrhage)','Jaundice','Malaria','Chicken pox','Dengue','Typhoid','hepatitis A',

'Hepatitis B','Hepatitis C','Hepatitis D','Hepatitis E','Alcoholic hepatitis','Tuberculosis',

'Common Cold','Pneumonia','Dimorphic hemmorhoids(piles)',

'Heartattack','Varicoseveins','Hypothyroidism','Hyperthyroidism','Hypoglycemia','Osteoarthristis',
'Arthritis','(vertigo) Paroymsal  Positional Vertigo','Acne','Urinary tract infection','Psoriasis',

'Impetigo']

l2=[]

for x in range(0,len(l1)):

    l2.append(0)

# TESTING DATA df -------------------------------------------------------------------------------------

df=pd.read_csv("Training.csv")

df.replace({'prognosis':{'Fungal infection':0,'Allergy':1,'GERD':2,'Chronic
cholestasis':3,'Drug Reaction':4,

'Peptic ulcer diseae':5,'AIDS':6,'Diabetes ':7,'Gastroenteritis':8,'Bronchial
                  Asthma':9,'Hypertension ':10,

'Migraine':11,'Cervical spondylosis':12,

'Paralysis (brain hemorrhage)':13,'Jaundice':14,'Malaria':15,'Chicken
                  pox':16,'Dengue':17,'Typhoid':18,'hepatitis A':19,

'Hepatitis B':20,'Hepatitis C':21,'Hepatitis D':22,'Hepatitis E':23,'Alcoholic
                  hepatitis':24,'Tuberculosis':25,

'Common Cold':26,'Pneumonia':27,'Dimorphic hemmorhoids(piles)':28,'Heart
                  attack':29,'Varicose veins':30,'Hypothyroidism':31,

'Hyperthyroidism':32,'Hypoglycemia':33,'Osteoarthristis':34,'Arthritis':35,

'(vertigo) Paroymsal  Positional Vertigo':36,'Acne':37,'Urinary tract
                  infection':38,'Psoriasis':39,

'Impetigo':40}},inplace=True)
```

```
# print(df.head())



X= df[l1]



y = df[["prognosis"]]

np.ravel(y)

print(y)



# TRAINING DATA tr --------------------------------------------------------------------------------

tr=pd.read_csv("Testing.csv")

tr.replace({'prognosis':{'Fungal infection':0,'Allergy':1,'GERD':2,'Chronic cholestasis':3,'Drug Reaction':4,

'Peptic ulcer diseae':5,'AIDS':6,'Diabetes ':7,'Gastroenteritis':8,'Bronchial Asthma':9,'Hypertension ':10,

'Migraine':11,'Cervical spondylosis':12,

'Paralysis (brain hemorrhage)':13,'Jaundice':14,'Malaria':15,'Chicken pox':16,'Dengue':17,'Typhoid':18,'hepatitis A':19,

'Hepatitis B':20,'Hepatitis C':21,'Hepatitis D':22,'Hepatitis E':23,'Alcoholic hepatitis':24,'Tuberculosis':25,

'Common Cold':26,'Pneumonia':27,'Dimorphic hemmorhoids(piles)':28,'Heart attack':29,'Varicose veins':30,'Hypothyroidism':31,

'Hyperthyroidism':32,'Hypoglycemia':33,'Osteoarthristis':34,'Arthritis':35,

'(vertigo) Paroymsal  Positional Vertigo':36,'Acne':37,'Urinary tract
```

```
            infection':38,'Psoriasis':39,

'Impetigo':40}},inplace=True)

X_test= tr[l1]

y_test = tr[["prognosis"]]

np.ravel(y_test)

def DecisionTree():

    from sklearn import tree

        clf3 = tree.DecisionTreeClassifier()   # empty model of the decision tree

    clf3 = clf3.fit(X,y)

    # calculating accuracy-------------------------------------------------------------------

    from sklearn.metrics import accuracy_score

    y_pred=clf3.predict(X_test)

    print(accuracy_score(y_test, y_pred))

   print(accuracy_score(y_test, y_pred,normalize=False))

 # -----------------------------------------------------

    psymptoms =
[Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]

    for k in range(0,len(l1)):

        # print (k,)

        for z in psymptoms:

            if(z==l1[k]):
```

```
            l2[k]=1

    inputtest = [l2]

    predict = clf3.predict(inputtest)

    predicted=predict[0]

    h='no'

    for a in range(0,len(disease)):

        if(predicted == a):

            h='yes'

            break

    if (h=='yes'):

        t1.delete("1.0", END)

        t1.insert(END, disease[a])

    else:

        t1.delete("1.0", END)

        t1.insert(END, "Not Found")

def randomforest():

    from sklearn.ensemble import RandomForestClassifier

    clf4 = RandomForestClassifier()

    clf4 = clf4.fit(X,np.ravel(y))

    # calculating accuracy-------------------------------------------------------------------
    from sklearn.metrics import accuracy_score

    y_pred=clf4.predict(X_test)

    print(accuracy_score(y_test, y_pred))

    print(accuracy_score(y_test, y_pred,normalize=False))

    # -----------------------------------------------------

        psymptoms =
[Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]

for k in range(0,len(l1)):

        for z in psymptoms:

            if(z==l1[k]):

                l2[k]=1

    inputtest = [l2]

    predict = clf4.predict(inputtest)

    predicted=predict[0]

    h='no'

    for a in range(0,len(disease)):

        if(predicted == a):

            h='yes'

            break

    if (h=='yes'):

        t2.delete("1.0", END)
        t2.insert(END, disease[a])

    else:

        t2.delete("1.0", END)

        t2.insert(END, "Not Found")

def NaiveBayes():

    from sklearn.naive_bayes import GaussianNB

    gnb = GaussianNB()

    gnb=gnb.fit(X,np.ravel(y))

    Gnb

    # calculating accuracy-------------------------------------------------------------------

    from sklearn.metrics import accuracy_score

    y_pred=gnb.predict(X_test)

    print(accuracy_score(y_test, y_pred))

    print(accuracy_score(y_test, y_pred,normalize=False))

    # -----------------------------------------------------



    psymptoms =
            [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.ge
            t()]

    for k in range(0,len(l1)):

        for z in psymptoms:
            if(z==l1[k]):
                l2[k]=1
    inputtest = [l2]
    predict = gnb.predict(inputtest)
    predicted=predict[0]
    h='no'
    for a in range(0,len(disease)):
        if(predicted == a):
            h='yes'
            break

    if (h=='yes'):
        t3.delete("1.0", END)
        t3.insert(END, disease[a])
    else:
        t3.delete("1.0", END)
        t3.insert(END, "Not Found")

def drugPred():
    global svcdisease
    global logitdisease

    diabetes = "Glucagon or Alpha-glucosidase"

    tension = "Beta Blockers or ACE inhinitors"

    t6.delete('1.0',END)

    if(logitdisease == "Diabetes" or svcdisease == "Diabetes"):

        t6.insert(END,diabetes)

    elif(logitdisease == "Hypertension" or svcdisease == "Hypertensio"):

        t6.insert(END,tension)

    else:

        t6.insert(END,"Need to implement")

        def LogisticRegression():

    global logitdisease

    from sklearn.linear_model import LogisticRegression

    clf5 = LogisticRegression()  # empty model of the Logistic Regression

    clf5 = clf5.fit(X, np.ravel(y))

# calculating accuracy---------------------------------------------------------------

    from sklearn.metrics import accuracy_score

    y_pred=clf5.predict(X_test)

    print(accuracy_score(y_test, y_pred))

    print(accuracy_score(y_test, y_pred,normalize=False))
```

```
# -----------------------------------------------------
psymptoms = [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]
for k in range(0,len(l1)):
    # print (k,)
    for z in psymptoms:
        if(z==l1[k]):
            l2[k]=1
inputtest = [l2]
predict = clf5.predict(inputtest)
predicted=predict[0]
h='no'
for a in range(0,len(disease)):
    if(predicted == a):
        h='yes'
        break
if (h=='yes'):
    t4.delete("1.0", END)
    logitdisease = disease[a]
    t4.insert(END, disease[a])
    else:

        t4.delete("1.0", END)

        t4.insert(END, "Not Found")

def SVC():

    global svcdisease

    from sklearn.svm import SVC

    clf6 = SVC()  # empty model of the SVM

    clf6 = clf6.fit(X, y)

    # calculating accuracy------------------------------------------------------------------------

    from sklearn.metrics import accuracy_score

    y_pred=clf6.predict(X_test)

    print(accuracy_score(y_test, y_pred))

    print(accuracy_score(y_test, y_pred,normalize=False))

    # --------------------------------------------------------

    psymptoms = [Symptom1.get(),Symptom2.get(),Symptom3.get(),Symptom4.get(),Symptom5.get()]

    for k in range(0,len(l1)):

        # print (k,)

        for z in psymptoms:

            if(z==l1[k]):

                l2[k]=1
    inputtest = [l2]

    predict = clf6.predict(inputtest)

    predicted=predict[0]

    h='no'

    for a in range(0,len(disease)):

        if(predicted == a):

            h='yes'

            break

    if (h=='yes'):

        t5.delete("1.0", END)

        svcdisease = disease[a]

        t5.insert(END, disease[a])

    else:

        t5.delete("1.0", END)

        t5.insert(END, "Not Found")

# gui_stuff------------------------------------------------------------------------------------

root = Tk()

root.configure(background='blue')

# entry variables

Symptom1 = StringVar()
```

```
Symptom1.set(None)
Symptom2 = StringVar()
Symptom2.set(None)
Symptom3 = StringVar()
Symptom3.set(None)
Symptom4 = StringVar()
Symptom4.set(None)
Symptom5 = StringVar()
Symptom5.set(None)
Name = StringVar()
# Heading
w2 = Label(root, justify=LEFT, text="Drug Disease Prediction using Machine Learning",
           fg="white", bg="blue")
w2.config(font=("Elephant", 25))
w2.grid(row=1, column=0, columnspan=2, padx=100)
w2 = Label(root, justify=LEFT, text="", fg="white", bg="blue")
w2.config(font=("Aharoni", 40))
w2.grid(row=2, column=0, columnspan=2, padx=100)
# labels
NameLb = Label(root, text="Name of the Patient", fg="yellow", bg="black")
NameLb.grid(row=6, column=0, pady=15, sticky=W)
```

```
S1Lb = Label(root, text="Symptom 1", fg="yellow", bg="black")

S1Lb.grid(row=7, column=0, pady=10, sticky=W)

S2Lb = Label(root, text="Symptom 2", fg="yellow", bg="black")

S2Lb.grid(row=8, column=0, pady=10, sticky=W)

S3Lb = Label(root, text="Symptom 3", fg="yellow", bg="black")

S3Lb.grid(row=9, column=0, pady=10, sticky=W)

S4Lb = Label(root, text="Symptom 4", fg="yellow", bg="black")

S4Lb.grid(row=10, column=0, pady=10, sticky=W)

S5Lb = Label(root, text="Symptom 5", fg="yellow", bg="black")

S5Lb.grid(row=11, column=0, pady=10, sticky=W)

#lrLb = Label(root, text="DecisionTree", fg="white", bg="red")

#lrLb.grid(row=15, column=0, pady=10,sticky=W)

#destreeLb = Label(root, text="RandomForest", fg="white", bg="red")

#destreeLb.grid(row=16, column=0, pady=10, sticky=W)

#ranfLb = Label(root, text="NaiveBayes", fg="white", bg="red")

#ranfLb.grid(row=17, column=0, pady=10, sticky=W)

log = Label(root, text="Disease Prediction 1", fg="white", bg="red")

log.grid(row=15, column=0, pady=10, sticky=W)

svc = Label(root, text="Disease Prediction 2", fg="white", bg="red")

svc.grid(row=16, column=0, pady=10, sticky=W)
```

```
dp = Label(root, text="Drug Prediction", fg="white", bg="red")

dp.grid(row=17, column=0, pady=10, sticky=W)

# entries

OPTIONS = sorted(l1)



NameEn = Entry(root, textvariable=Name)

NameEn.grid(row=6, column=1)

S1En = OptionMenu(root, Symptom1,*OPTIONS)

S1En.grid(row=7, column=1)

S2En = OptionMenu(root, Symptom2,*OPTIONS)

S2En.grid(row=8, column=1)

S3En = OptionMenu(root, Symptom3,*OPTIONS)

S3En.grid(row=9, column=1)

S4En = OptionMenu(root, Symptom4,*OPTIONS)

S4En.grid(row=10, column=1)

S5En = OptionMenu(root, Symptom5,*OPTIONS)

S5En.grid(row=11, column=1)

#dst = Button(root, text="DecisionTree", command=DecisionTree,bg="green",fg="yellow")

#dst.grid(row=8, column=3,padx=10)

#rnf = Button(root, text="Randomforest", command=randomforest,bg="green",fg="yellow")
```

```
#rnf.grid(row=9, column=3,padx=10)

#lr = Button(root, text="NaiveBayes", command=NaiveBayes,bg="green",fg="yellow")

#lr.grid(row=10, column=3,padx=10)

logit = Button(root, text="Disease Prediction", command=LogisticRegression,bg="green",fg="yellow")

logit.grid(row=8, column=3,padx=10)



svm = Button(root, text="Disease Prediction", command=SVC,bg="green",fg="yellow")

svm.grid(row=9, column=3,padx=10)

dpt = Button(root, text="Drug Prediction", command=drugPred,bg="green",fg="yellow")

dpt.grid(row=10, column=3,padx=10)

#textfileds

#t1 = Text(root, height=1, width=40,bg="orange",fg="black")

#t1.grid(row=15, column=1, padx=10)

#t2 = Text(root, height=1, width=40,bg="orange",fg="black")

#t2.grid(row=16, column=1 , padx=10)

#t3 = Text(root, height=1, width=40,bg="orange",fg="black")

#t3.grid(row=17, column=1 , padx=10)

t4 = Text(root, height=1, width=40,bg="orange",fg="black")

t4.grid(row=15, column=1 , padx=10)

t5 = Text(root, height=1, width=40,bg="orange",fg="black")
```

```
t5.grid(row=16, column=1 , padx=10)

t6 = Text(root, height=2, width=40,bg="orange",fg="black")

t6.grid(row=17, column=1 , padx=10)

root.mainloop()
```

7. PROJECT 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.

TYPES OF TESTS

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.

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.

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.

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.

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.

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.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

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.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

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

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.

8. OUTPUT SCREENS

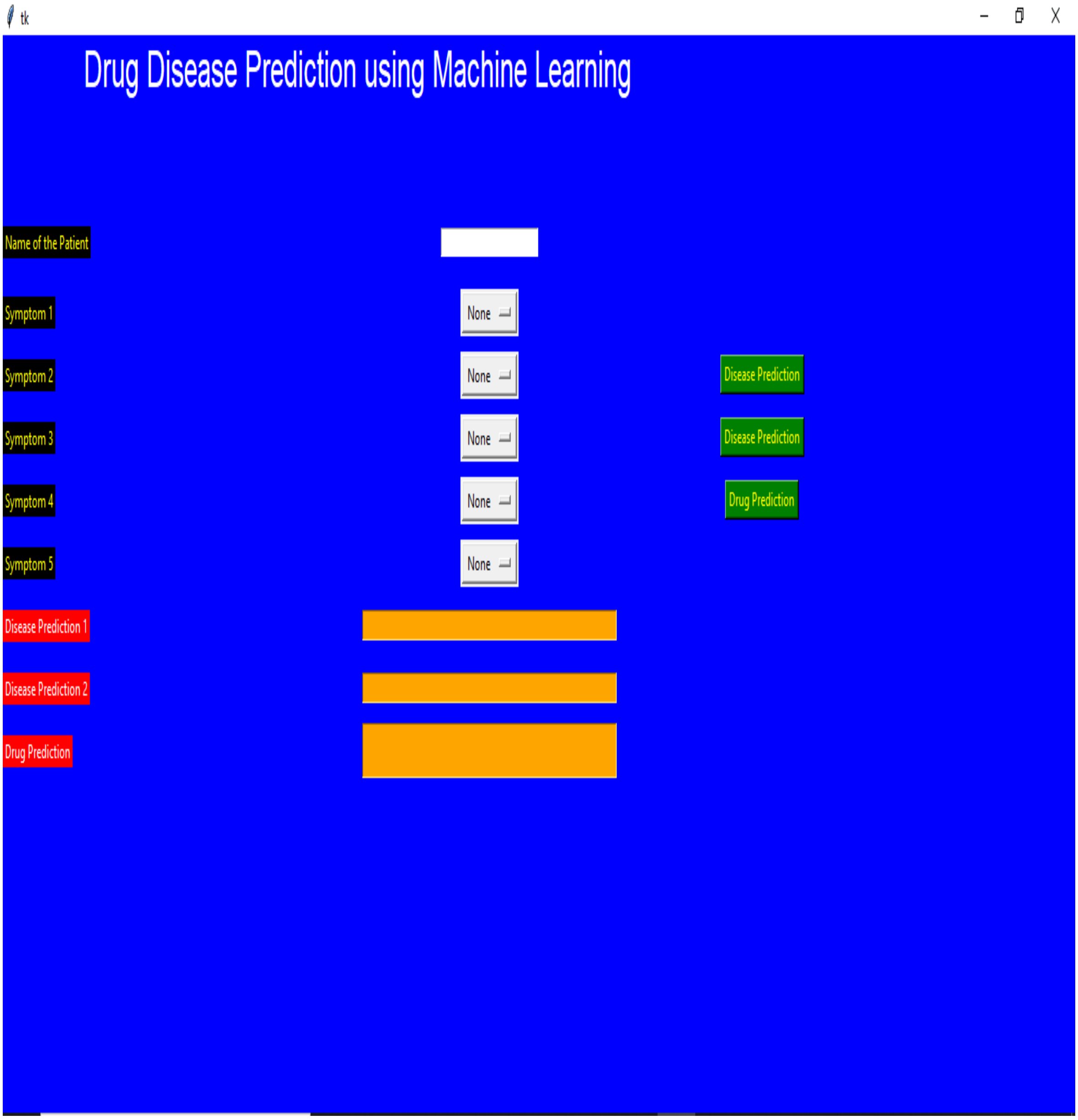
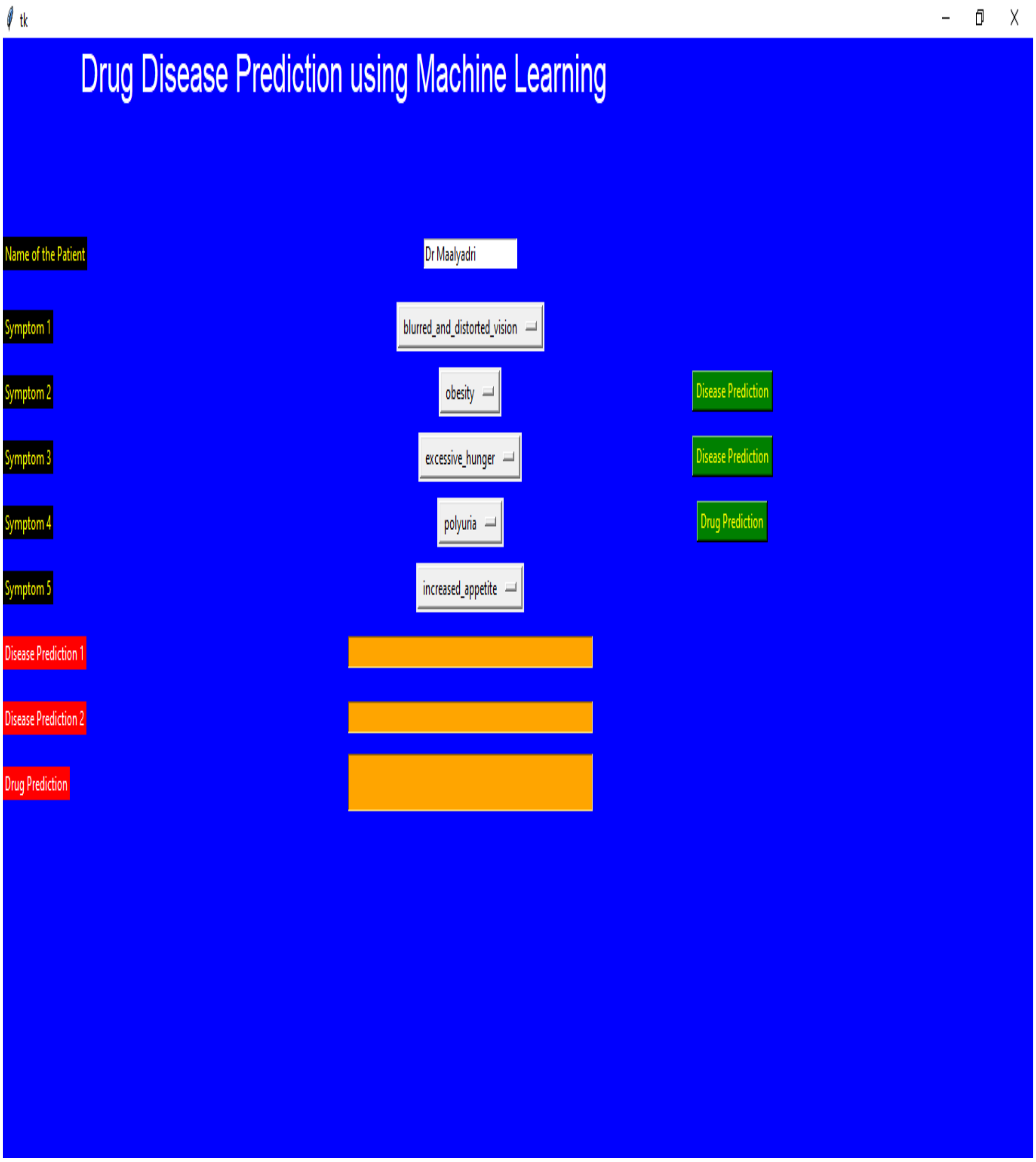


FIG.8.1 OUPUT SCREEN



Name of the Patient: Dr Maalyadri

Symptom 1: blurred_and_distorted_vision

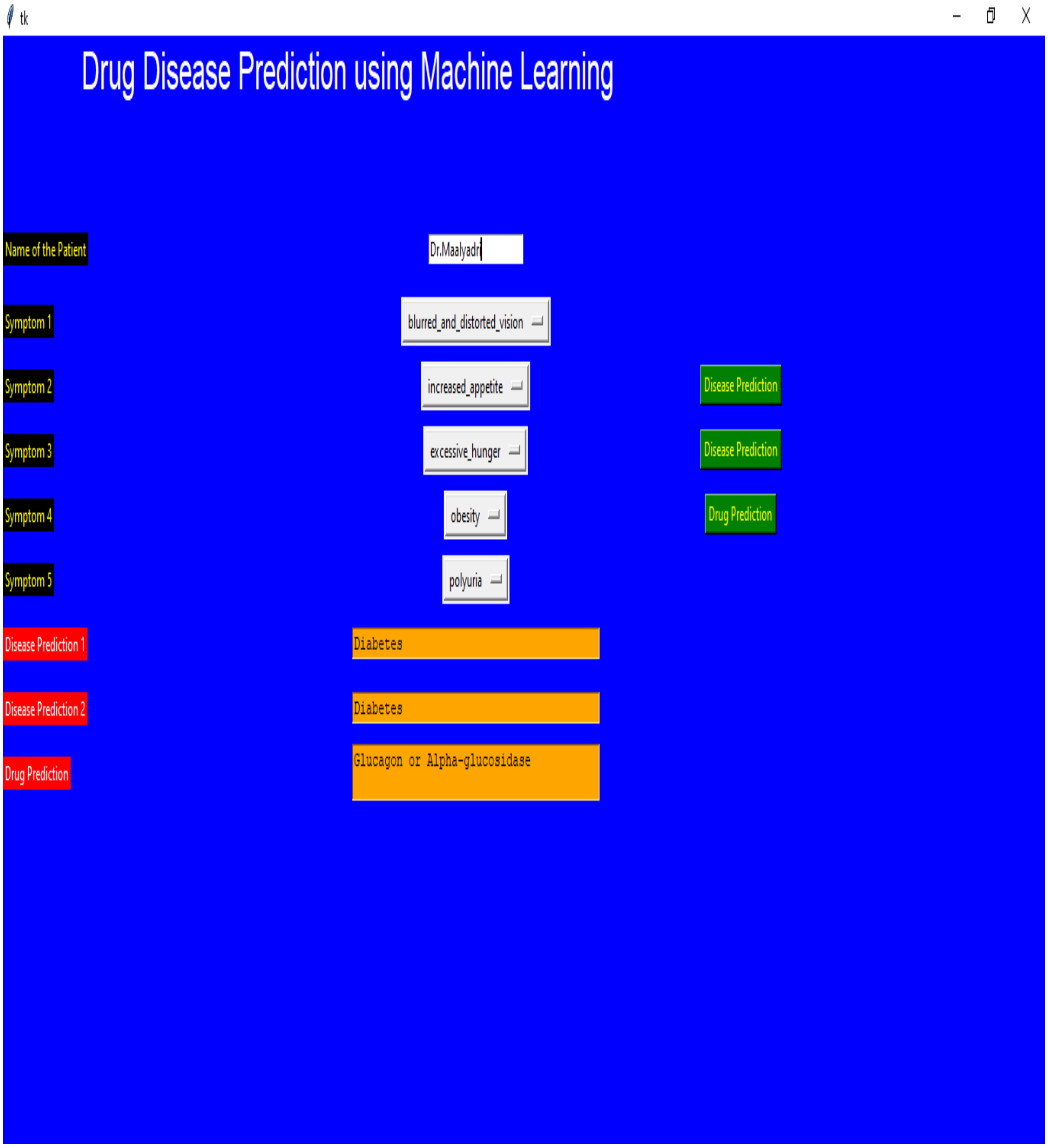
Symptom 2: obesity

Symptom 3: excessive_hunger

Symptom 4: polyuria

Symptom 5: increased_appetite

FIG.8.2 OUPUT SCREEN WITH USER INPUT



Drug Disease Prediction using Machine Learning

Name of the Patient Dr.Maalyadri

Symptom 1 blurred_and_distorted_vision

Symptom 2 increased_appetite

Symptom 3 excessive_hunger

Symptom 4 obesity

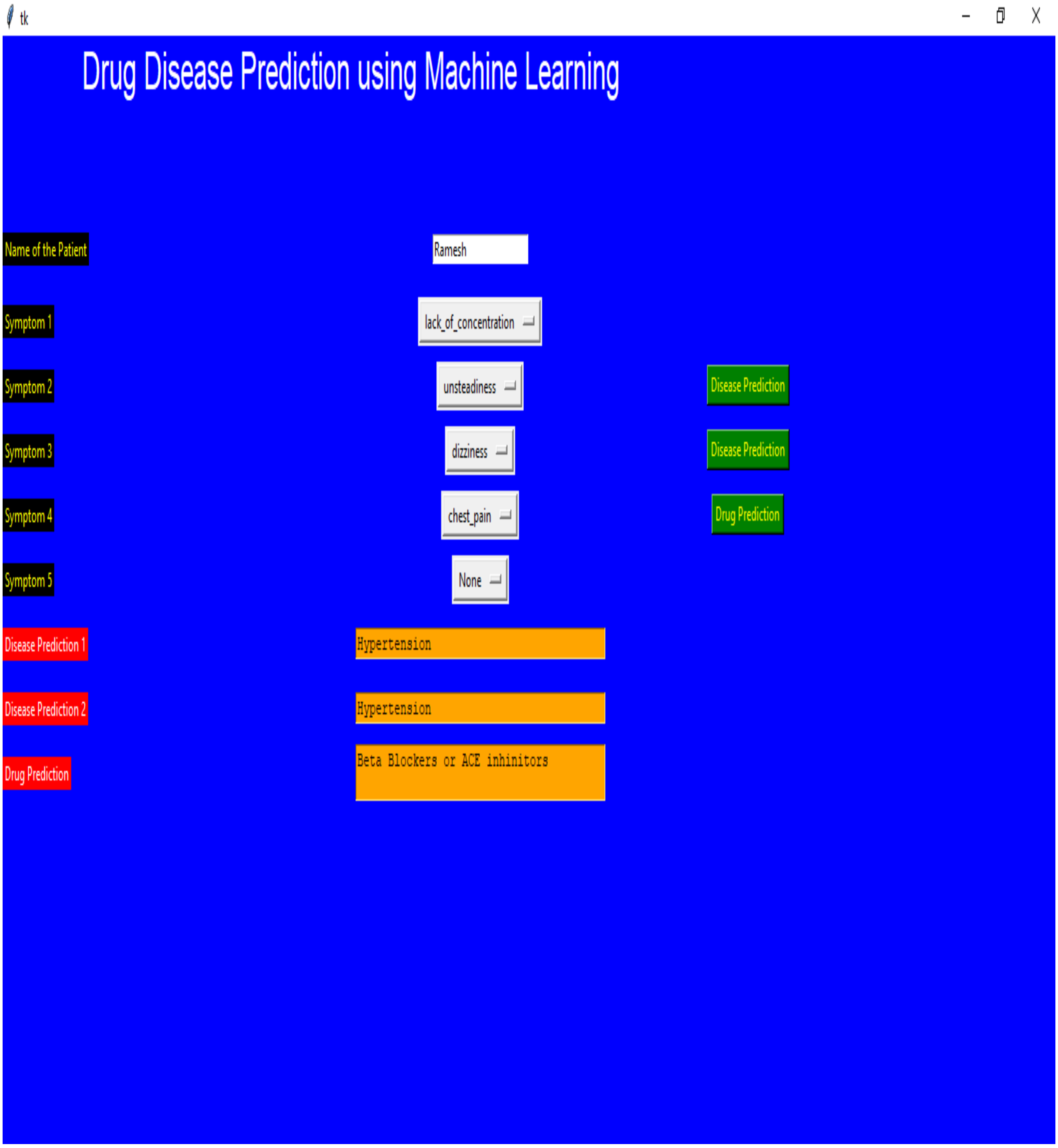
Symptom 5 polyuria

Disease Prediction 1 Diabetes

Disease Prediction 2 Diabetes

Drug Prediction Glucagon or Alpha-glucosidase

FIG.8.3 OUTPUT SCREEN WITH DRUG AND DISEASE PREDICTION



Name of the Patient: Ramesh

Symptom 1: lack_of_concentration

Symptom 2: unsteadiness

Symptom 3: dizziness

Symptom 4: chest_pain

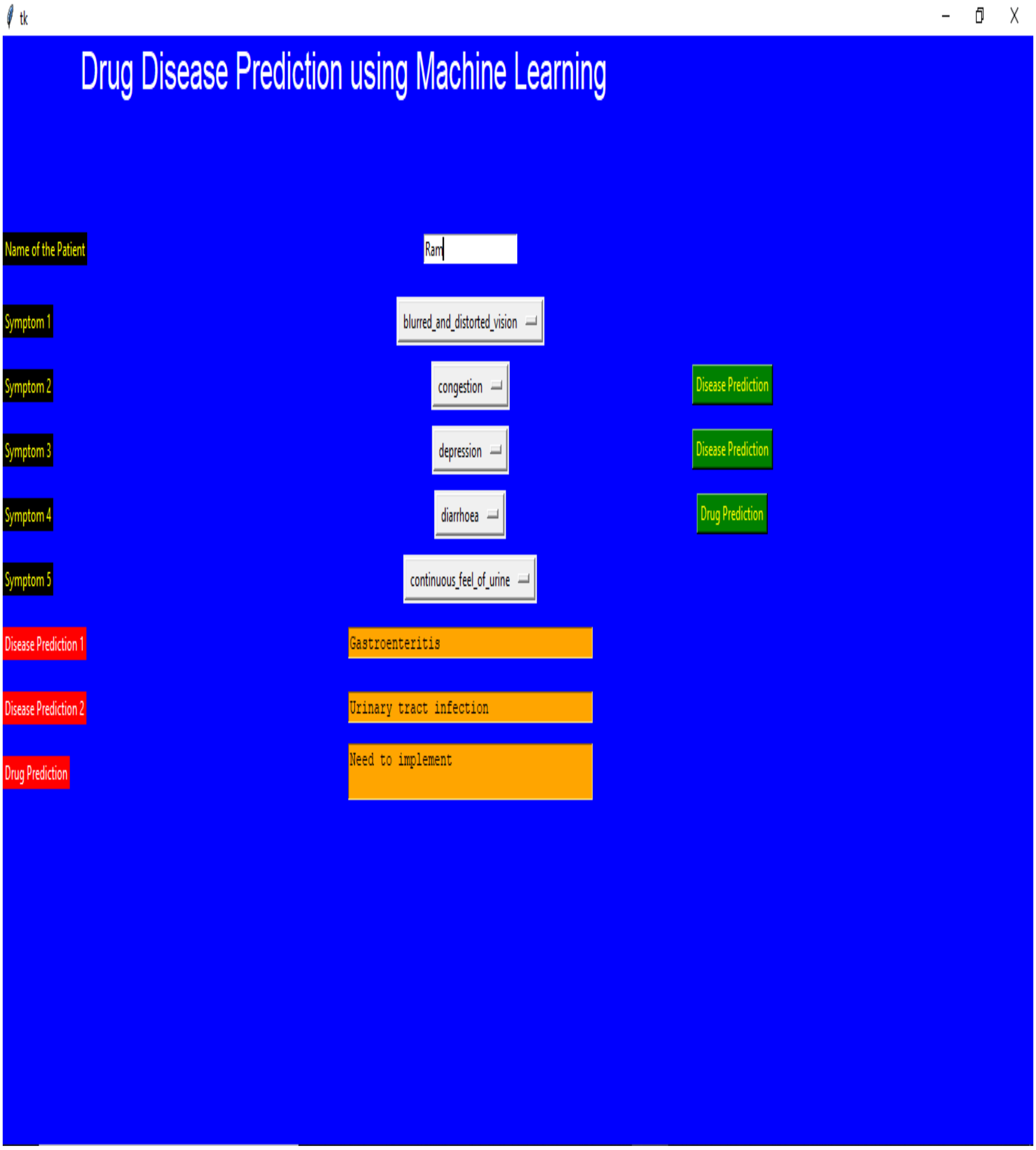
Symptom 5: None

Disease Prediction 1: Hypertension

Disease Prediction 2: Hypertension

Drug Prediction: Beta Blockers or ACE inhinitors

FIG.8.4 OUTPUT SCREEN WITH ANOTHER USER INPUT



Drug Disease Prediction using Machine Learning

Name of the Patient: Ram

Symptom 1: blurred_and_distorted_vision

Symptom 2: congestion

Symptom 3: depression

Symptom 4: diarrhoea

Symptom 5: continuous_feel_of_urine

Disease Prediction 1: Gastroenteritis

Disease Prediction 2: Urinary tract infection

Drug Prediction: Need to implement

FIG.8.5 OUTPUT SCREEN WITH DIFFERENT USER

9. CONCLUSION

This paper gives research of multiple researches done in this field. Our Proposed System aims at bridging gap between Doctors and Patients which will help both classes of users in achieving their goals. This system provides support for multiple disease prediction using different Machine Learning algorithms. The present approach of many systems focuses only on automating this process which lacks in building the user’s trust in the system. By providing Doctor’s recommendation in our system, we ensure user’s trust side by side ensuring that the Doctor’s will not feel that their Business is getting affected due to this System.

10. FUTURE ENHANCEMENT

AS of now we are only doing Drug Disease Recommendation for only two Diseases in future we are going identify Disease Prediction and Drug Recommendation Android Application using Data Mining.The system takes the symptoms from the users which they are feeling at that moment and runs a Machine Learning algorithm in the cloud to detect the disease from which the user may be suffering. The System collects raw data from the user or consumer. As the massive amount of information is already available from healthcare websites, patients can easily compare the diagnosis done by their doctors and the related information which is already present on the internet.And in future we will storing data of every user in our database.And in future we will b training our data to predict drug and disease prediction of every possible disease.we may create a website in future for our project where user can login and give symptoms and our project may detect disease and suggest drug for the disease and information will be stored in the database.

11.BIBILOGRAPHY

[1] K. Goh, M. E. Cusick, D. Valle, B. Childs, M. Vidal, and A. L. Barabási, ‘‘The human disease network,’’ Proc. Nat. Acad. Sci. USA, vol. 104, no. 21, pp. 8685–8690, 2007.

[2] L. Weng, L. Zhang, Y. Peng, and R. S. Huang, ‘‘Pharmacogenetics and pharmacogenomics: A bridge to individualized cancer therapy,’’ Pharmacogenomics, vol. 14, no. 3, pp. 315–324, 2013

[3] H. Luo et al., ‘‘Drug repositioning based on comprehensive similarity measures and bi-random walk algorithm,’’ Bioinformatics, vol. 32, no. 17, pp. 2664–2671, 2016.

[4] S. Naylor and J. Schonfeld, ‘‘Therapeutic drug repurposing, repositioning and rescue—Part I: Overview,’’ Drug Discovery World, vol. 16, pp. 49–62, Dec. 2014

[5] S. J. Cockell et al., ‘‘An integrated dataset for in silico drug discovery,’’ J. Integr. Bioinf., vol. 7, no. 3, pp. 15–27, 2010.

[6] A. Gottlieb, G. Y. Stein, E. Ruppin, and R. Sharan, ‘‘PREDICT: A method for inferring novel drug indications with application to personalized medicine,’’ Mol. Syst. Biol., vol. 7, no. 1, p. 496, 2014.

[7].Jaymin Patel, Prof.Tejal Upadhyay, Dr. Samir Patel “Heart disease prediction using Machine learning and Data Mining Technique" Volume 7.Number1 Sept 2015March 2016.

[8].G. Parthiban, S.K.Srivasta "Applying Machine learning methods in Diagnosing Heart disease for Diabetic Patients" International Journal of Applied Information Systems (IJAIS) – ISSN: 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 3– No.7, August 2012.

[9].“Disease Prediction Using Machine Learning Over Big Data”Vinitha S, Sweetlin S, Vinusha H and Sajini S (2018).

[10].“Heart Disease Prediction using Logistic Regression Algorithm using Machine Learning”Reddy Prasad,Pidaparthi Anjali, S.Adil, N.Deepa (2019)

[11].“Heart Disease Prediction Using Data Mining Techniques”H. Benjamin Fredrick David and S. Antony Belcy(2018).

[12].“Multi Disease Prediction Using Data Mining Techniques”K.Gomathi , Dr. D. Shanmuga Priyaa (2017).

A

PROJECT REPORT

On

ENHANCEMENT OF VEHICLE SPEED DETECTION

Submitted by

Mr. Srikar. T	**(17K81A1248)**
Mr. Ruthvik Reddy	**(17K81A1223)**
Ms. Md. Sana Naaz	**(17K81A1234)**
Mr. Prajwal Yadav	**(17K81A1208)**

in the partial fulfillment for the award of degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

MR. GANDLA SHIVAKANTH

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST. MARTIN'S ENGINEERING COLLEGE

(An Autonomous Institute)

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **ENHANCEMENT OF VEHICLE SPEED DETECTION**", is being submitted by **SRIKAR. T(17K81A1248), K. RUTHVIK REDDY (17K81A1223), MD. SANA NAAZ (17K81A1234), D. PRAJWAL YADAV (17K81A1208)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

	Head of the Department
Mr. GANDLA SHIVAKANTH	Dr. R. NAGARAJU
Department of IT	Department of IT

Internal Examiner	External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of **Information Technology**, session: 2017-2021, St. Martin’s Engineering College, Dhulapally, Kompally, Secundrabad, hereby declare that work presented in this Project Work entitled **ENHANCEMENT OF VEHICLE SPEED DETECTION** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

Mr. Srikar. T	**17K81A1248**
Mr. K. Ruthvik Reddy	**17K81A1223**
Ms. Md. Sana Naaz	**17K81A1234**
Mr. Prajwal Yadav	**17K81A1208**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R. NAGARAJU**, Head of the Department, **DEPARTMENT OF INFORMATION TECHNOLOGY**, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **MR. D. BABU RAO**, Associate Professor, in Department of Information Technology, for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **MR. GANDLA SHIVAKANTH**, Department of Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

SRIKAR.T	**17K81A1248**
K. RUTHVIK REDDY	**17K81A1223**
MD. SANA NAAZ	**17K81A1234**
D. PRAJWAL YADAV	**17K81A1208**

TABLE OF CONTENTS

ABSTRACT

Using the surveillance video camera monitoring system, the speed of vehicle is detected. Apart from vehicle speed detection, this algorithm can be used to monitor the traffic condition along the road or highway. The existing surveillance video cameras are rarely used to measure the vehicle speed and estimate the vehicle. A MATLAB algorithm is proposed and developed to associate the developed algorithm with real-time video sequence and images. Development of vehicle speed detection algorithm is based on the vector-valued function and motion vector technique that estimates the velocity of moving vehicle. Surveillance video camera monitoring system has gained a lot of interest among the research community especially in monitoring vehicle speed. Apart from vehicle speed detection, this algorithm can be used to monitor the traffic condition along the road or highway. The existing surveillance video cameras are rarely used to measure the vehicle speed and estimate the vehicle. Numerous researches have been conducted in order to detect or estimate the speed of a moving vehicle using the image processing technique. Research such as presents various papers on the real-time vehicle detection and speed estimation. Existing methods applied into vehicle speed detection, including speed detection based on digital aerial images, combination value and frame differencing to produce the most successful outcome. Digital aerial images or camera Ultra Cam is used in image processing using the image extraction and detection visually and automated extraction and detection.

LIST OF FIGURES

LIST OF OUTPUT SCREENS

LIST OF ABBREVIATIONS

RADAR	**RADIO DETECTION AND RANGING**
IDE	**INTEGRATED DEVELOPMENT ENVIRONMENT**
SDLC	**SOFTWARE DEVELOPMENT LIFE CYCLE**
GUI	**GRAPHICAL USER INTERFACE**
ER	**ENTITY RELATIONSHIP**
RAM	**RANDOM ACCESS MEMORY**
GPS	**GLOBAL POSITIONING SYSTEM**
OPENCV	**OPEN COMPUTER VISION**

CHAPTER 1: INTRODUCTION

1.1 Project Overview

In recent times, there has been a drastic change in people's lifestyles and with an increase in incomes and lower cost of automobiles there is a huge increment in the number of cars on the roads which has led to traffic and commotion. The manual efforts to keep people from breaking traffic rules such as the speed limit are not enough. There is not enough police and man force available to track the traffic and vehicles on roads and check them for speed control. Hence, we require technologically advanced speed calculators installed that effectively detect cars on the road and calculate their speeds.

To implement the above idea two basic requirements, need to be met which are the effective detection of the cars on roads and their velocity measurement. For this purpose, we can use OpenCV software which uses the Haar cascade to train our machine to detect the object, in this case the car.

We have developed a Haar cascade to detect cars on the roads, whose velocities are then measured using a python script. The real-time application of this project proves to be much useful as it is easy to implement, fast to process and efficient with low-cost development. Also, the tool might be useful to apply in simulation tools to measure velocities of cars. This can be further developed to identify all kinds of vehicles as well as to check anyone who breaks a traffic light.

The improvements in the project can be done by creating a bigger haar cascade since bigger the haar cascade developed, more the number of vehicles that can be detected on the roads. Better search algorithms can allow a faster search and better detection of these vehicles for better efficiency.

This paper is to develop an algorithm to calculate the speed of the object(vehicle) detected. We have implemented the algorithm using Python Script.

The complete implementation uses two basic processes: -

1. Car detection using Haar cascades in OpenCV
2. Measurement of velocity of detected cars using python script.

Car Detection:

Object Location utilizing Haar highlight based course classifiers is a compelling item discovery strategy that uses a machine learning based approach where a course capacity is prepared from a considerable measure of positive and negative pictures. It is then used to recognize protests in different pictures.

- Initially, the calculation needs a considerable measure of positive (pictures of autos) and negative (pictures without autos) to prepare the classifier. At that point, we have to concentrate highlights from it. For this, haar highlights appeared in beneath picture are utilized. They are much the same as our convolutional part. Each component is a solitary esteem acquired by subtracting total of pixels under white rectangle from aggregate of pixels under dark rectangle.

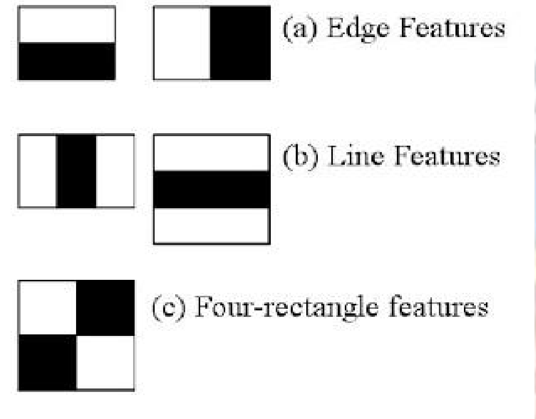


Figure. Features on an image

Now every single conceivable size and areas of every part is utilized to ascertain a lot of components. (Simply envision what amount of calculation it needs? Indeed, even a 24x24 window comes about more than 160000 components). For each component computation, we have to discover whole of pixels under white and dark rectangles. To tackle this, they presented the necessary pictures.

- Now, we apply each component on all the preparation pictures. For each component, it finds the best limit which will characterize the countenances to positive and negative. Be that as it may, clearly, there will be blunders or misclassifications. We select the elements with least mistake rate, which implies they are the elements that best orders the auto and non-auto pictures.
- So now you take a picture. Take each 24x24 window. Apply 6000 elements to it. Check on the off chance that it is auto or not.

But that as it may, clearly, there will be blunders or misclassifications. We select the elements with least mistake rate, which implies they are the elements that best orders the auto and non-auto pictures.

1.2 Project Objectives

This project is subjected to one main objective which is to develop a system that will be able to detect the speed of moving vehicle using a charge-coupled device camera. The speed detection using normal camera will be especially useful in much kind of industries since its main feature is cheap or low cost. This project intends to develop the vehicle speed detection system for speed trap system. Using our proposed system, we were also able to track the traffic on highways by using the xml file and reference point from our video. Beside the main objective, there are a number of sub-objectives of this project. Below are the subobjectives for the project:

1) To develop a new approach of detecting speed using surveillance camera.

2) To measure the speed of moving vehicle.

3) To display height, distance, and speed of vehicle.

4) Track the traffic at highways.

1.3 Scope of the project

This project intends to develop a speed trap system using normal video camera and image processing technique. This project is to help the police enforce the law of vehicle's speeding in Malaysia. Image processing technique is possible using MATLAB software. Here are the scopes of the project:

1) Design a speed detection system for the use in speed trap system for the purpose of traffic speed law enforcement.

2) Data were taken at the straight roads only.

3) Detect only one object at a time.

Road safety: The timely checking of the over speeding vehicle will reduce high. percentage of road accidents.

Automation in law enforcement: The system being completely automatic, reduces the number of traffic police officers needed to deploy in the real field for checking speeding vehicles.

With very few enhancements in the proposed system new features can be easily incorporated such as:

• **Vehicle security:** The lost-out cases of the vehicle are increasing day by day; the stolen vehicle can be easily detected by comparing with the registered entry of stolen vehicles.

• **Parking:** The vehicles can be easily registered using automatic system with this system in the parking lounge or similar purpose complexes.

• **Visitor management:** This system can be effectively used to assist visitor management systems in recognizing guest vehicles.

CHAPTER 2: LITERATURE SURVEY

2.1 Survey on Background

One of the technologies our law enforcement department uses to measure the speed of a moving vehicle is Doppler radar. It beams a radio wave at a vehicle, and then estimate the vehicles speed by measuring change in reflected wave frequency. It is a fixed or hand-held device and is reliable when a moving object is in the field of view and no other moving objects are nearby. Cosine error has to be taken care if the gun is not in the line of sight. Also, Radio interference which causes errors in speed detection has to be taken care. Some of the previous works using image and video processing applied for vehicle detection and speed measurements are vehicle detection based on frame difference, calibrated camera, motion trajectories, Optics and digital aerial images. Also, blurred images were used to find out the vehicle speed along with high-end camera motion detection for automated speed measurements and feature point tracking for vehicle speed measurements were used. Currently highly reliable GPS systems are used to track vehicle speeds in US. Cost-effectiveness is a concern in such a case. In our method moving vehicle video from any video camera or mobile source is utilized. The algorithms are implemented in ‘Python’ language using OpenCV and Visual Studio. Later this code can be ported to a simple processor where vehicle speed can be measured. Example: a simple smart phone with average processing capacity. Our aim was to implement real- time vehicle speed detector.

A video signal is the term used to describe any sequence of time varying images. A still image is a spatial distribution of intensities that remain constant with time while a time varying image has a spatial intensity distribution that varies with time. Videos can be in various formats based on the different cameras or mobile phones used. The video format used is an AVI file with the extension .avi.

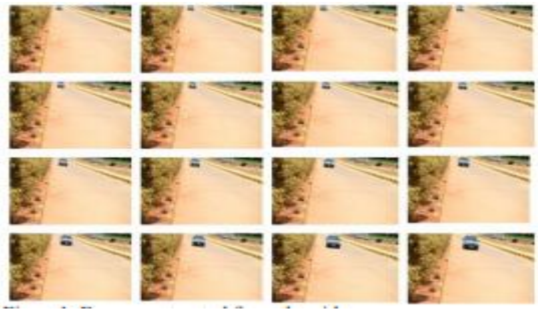


Figure. Video Sequence Images

2.2 Conclusions on Survey

This video is converted into frames. Since the video had 30 frames per second, extracting all the frames would lead to unwanted redundancy and these increases the delay time to execute the program. Hence, we sampled the frames so that we get 3 frames per second, which serves the need. Further above reference frames which are consecutive in nature are selected and converted into grayscale. Conversion to grayscale further reduces the amount of computation. Next steps include preprocessing, moving edge detection, morphological operations, edge detection, vehicle segmentation and corner detection.

Pre-processing

The traffic images have a noise component from several interference sources. The Types of noise include the following: 1) Salt-and-pepper noise, which occurs when an image is coded and transmitted over a noisy channel or degraded by electrical sensor noise, as in video cameras. 2) Convolution noise (blurring), which produces images that are degraded by lens misfocus, motion, or atmospheric turbulence, such as adverse weather conditions. Both noise sources contribute to high-frequency noise components. In our process, median filtering is used to reduce this high-frequency noise. It preserves the edge information required by our algorithm. Edges are a key image feature, as they remain prominent despite the variations in the traffic scenes and ambient lighting.

Algorithm: Median Filtering Algorithm for Image Noise Reduction

Step1: Taking whole video frames as an input.

Step2: Median filter method is applied to remove unwanted noise from the video image frame.

Step3: Applied filtering process at each image frame pixel by pixel that is as shown in below form:

$$Y[r, s] = \text{median}\,\{X[i, j],(i, j)\} \quad (1)$$

Where, X [i, j], (i, j) is a input image, centered around location [r, s] in the image.

Step4: Produced filtered video frame which is not contains any noise.

Background Subtraction

A reference frame is required for subtraction is in some ways similar to calibration. Here calibration is with the input image and not with the camera. Reference frame is used to remove the background of the image which is out of our interest.

Algorithm: Moving object detection using background subtraction algorithm.
Input: Initial frame extracted from input video.
Output: Detected vehicle frame.
Step1: Read all input video frames.
Step2: Identify the foreground in current video frame.
Step3: Apply the morphological opening to eliminate noise in the foreground.
Step4: Detect the connected elements with defined minimum area and measure their bounding boxes.
Step5: Draw bounding boxes throughout the detected cars.
Step6: Display the number of cars and speed detected in the video frame.
Step7: Visualize the results.

Edge Detection

Since there is a background which contains the moving leaves. Further pre-processing and post-processing steps are involved in edge detection. Those steps are 1. Smoothing: Blurring of the image to remove noise. 2. Gradients: The edges should be marked where the gradient of the image has large magnitudes. 3. Non-Maximum Suppression: Only local maxima should be marked as edges. 4. Double Thresholding: Potential edges are determined by thresholding. 5. Edge tracking by Hysteresis: Final edges are determined by suppressing all edges that are not connected to a very certain (strong) edge.

Algorithm: Gradient edge detection algorithm for vehicle edge detection.
Input: Input video frames.
Output: Gradient edge detected video frames.
Step1: Input video frames is taken as input.
Step2: Compute an image gradient vector at each and every pixel by convolving image frame with vertical derivative filters and horizontal derivative filters.

Step3: Display gradient image.

Morphological Operation

In order to remove gaps obtained along the edges, we need to enhance the moving edges. This enhancement uses the morphological operator’s dilation and erosion with an appropriate structural element. The result of sequentially applying dilation (first) and erosion is to remove specific image features smaller than the structural element without affecting the large features of interest. The structural element used is a line of size 8*1

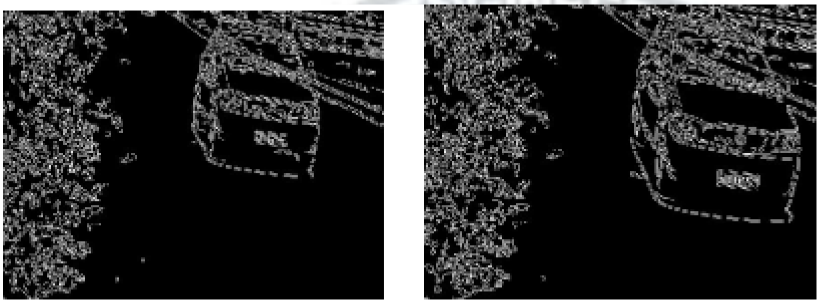


Figure. Canny Edge Detected Images.

CHAPTER 3: SOFTWARE AND HARDWARE REQUIREMENTS

3.1 Software Requirements

- Operating System: Windows 10
- Technology : Python 3.6
- IDE : PyCharm (or) Python Interpreter

3.1.1 Python

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. CPython, the reference implementation of Python, is open-source software and has a community-based development model, as do nearly all its variant implementations. CPython is managed by the non-profit Python Software Foundation. Python features a dynamic type of 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.

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 like 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.

Good to know.

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, PyCharm, NetBeans or Eclipse which are particularly useful when managing larger collections of Python files.

- Python Syntax compared to other programming languages.

- Python was designed for readability and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope, such as the scope of loops, functions and classes. Other programming languages often use curly brackets for this purpose.
- The Python interpreter is usually installed as /user/local/bin/python3.8 on those machines where it is available; putting /user/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., /user/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 does not 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 Read line 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 try 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 shell's -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
- >>>if the_world_is_flat:
- ...   print("Be careful not to fall off!")
- ...
- Be careful not to fall off!
- For more on interactive mode, see Interactive Mode.

3.1.2 Purpose

The project involved analyzing the design of few applications to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well-ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

3.1.3 Functional Requirements

User like police personnel to manage the images extracted images from the video.

3.1.4 Non-functional Requirements

• **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.

• **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.

• **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.

• **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.

• **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.

• **Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life.

However, the allowable wattage can’t be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

3.1.5 Input and Output Design

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 into 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

- 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.
- 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.
- 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.

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.
- Speed Calculation

3.2 Hardware Requirements

- Processor: intel i5 processor
- RAM : 4GB or more
- Hard Disk : 500GB or more

CHAPTER 4: SYSTEM DEVELOPMENT ANALYSIS

4.1 Overview of problem

4.1.1 Existing System

Usually, to measure the vehicle speed police personnel use RADAR guns. The major drawback associated with RADAR (Radio Detection and Ranging) is its cost and accuracy, the device is quite expensive and is less accurate and the biggest drawback is that a line-of-sight connection needed between RADAR and the vehicles. So to overcome the limitations in existing methods, various image processing techniques are used.

In a recent study over-speeding caused most of the accidents, followed by drunken driving. Over-speeding of two-wheelers and three-wheeler's is one of the major reasons of accidents. In order to support traffic management system in our country we need to build economical traffic monitoring systems. In recent times image and video processing has been applied to the field of traffic management system.

Figure. RADAR gun

4.1.2 Limitations of Existing System

- Cost of RADAR guns are high.
- Accuracy is not maintained.
- RADAR guns cannot detect small objects.
- RADARS cannot provide precise image of an object because of high wavelength.
- The wavelength of detection is less using RADAR guns.

4.2 Define the Problem

4.2.1 Proposed System

In the proposed method, the speed is estimated for the vehicle which is coming towards camera by tracking its motion through sequence of images.

In preprocessing, the captured video is converted into frames and the noise is removed using Median Filter technique. Using the motion of vehicle, we calculate the distance travelled from a reference point to captured point where speed of vehicle is calculated.

This project presents a new vehicle speed detection MATLAB algorithm. This algorithm is to detect the vehicle speed based on real-time video sequence through an offline based algorithm to reduce the elapsed processing time.

4.2.2 Advantages of Proposed System

- Using the median filter technique, we can detect the objects having high wavelength.
- Accuracy of the object is maintained.
- Distance travelled by a vehicle is calculated easily using the reference point taken for each vehicle.
- Comparatively less cost to RADAR gun's

4.2.3 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 feasibility study are:

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

4.2.3.1 Economic 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.

4.2.3.2 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 a modest requirement, as only minimal or null changes are required for implementing this system.

4.2.3.3 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.

4.3 Modules Overview

This stage consists of introduction to the approach to create vehicle speed detection from a video scene system. In general, the idea of this project is to calculate the vehicle speed from known distance and time when the first vehicle passes the starting point and the time the vehicle finally reaches end point. Below is the flow chart of the vehicle speed detection. It is to provide a deeper understanding of the details of operation of the vehicle speed detection. We use three modules, and they are uploading the video, capturing images, speed calculation.

4.4 Define the Modules

All the modules are shown in the image below with algorithm developed, video sequencing and block subtraction.

1. Upload Video:

We apply the video to our source code. For each component of video, we

perform threshold operations such that the vehicle is detected using the **Haar Cascade Classifier**. Then the speed and time is detected of each detected vehicle.

2. Capture Image:

Now, each detected car image is divided into 16*16window image, and this image is divided in small pixels. Each image can be divided into 16000 pixels.

3. Speed Calculation:

The velocity of each detected car is shown on the output screen. For this object detection algorithms are used but whereas in our project we have used Haar Cascade Classifier as it is least time consuming, most efficient and exceptionally reliable.

4.5 Module Functionality

The complete implementation uses two basic processes: -

1. Car detection using Haar cascades in OpenCV
2. Measurement of velocity of detected cars using python script.

Car Detection:

Object Location utilizing Haar highlight based course classifiers is a compelling item discovery strategy that uses a machine learning based approach where a course capacity is prepared from a considerable measure of positive and negative pictures. It is then used to recognize protests in different pictures.

• Initially, the calculation needs a considerable measure of positive (pictures of autos) and negative (pictures without autos) to prepare the classifier. At that point, we must concentrate highlights from it. For this, haar highlights appeared in beneath picture are utilized. They are much the same as our convolutional part. Each component is a solitary esteem acquired by subtracting total of pixels under white rectangle from aggregate of pixels under dark rectangle.

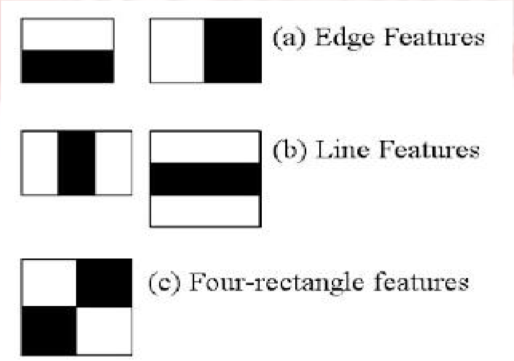


Figure. Features on an image

Now every single conceivable size and areas of every part is utilized to ascertain a lot of components. (Simply envision what amount of calculation it needs? Indeed, even a

24x24 window comes about more than 160000 components). For each component computation, we have to discover whole of pixels under white and dark rectangles. To tackle this, they presented the necessary pictures.

• Now, we apply each component on all the preparation pictures. For each component, it finds the best limit which will characterize the countenances to positive and negative. Be that as it may, clearly, there will be blunders or misclassifications. We select the elements with least mistake rate, which implies they are the elements that best orders the auto and non-auto pictures.

• So now you take a picture. Take each 24x24 window. Apply 6000 elements to it. Check on the off chance that it is auto or not.

Speed Calculation:

Speed calculation is done using the mentioned below method.

- Once a car is detected, using the cascade Classifier () function on the haar cascade developed. Now the time is started which was initialized to 0.
- Using the ratio in the image for each cm travelled by the detected image and real-time distance in meters, the actual distance covered by the car is calculated.
- As soon as the car reaches the center of the detection window whose distance is already known to us the time is stopped.
- Now the actual distance calculated is divided by the time calculated and velocity is obtained.
- This velocity and the distance of the camera in feet from the car (i.e., the height of camera above the car) is printed on the output screen.

From the previous processes, it was already providing the position of each single vehicle in the image frame and the position of mark points found in the reference frame. The speed of the vehicle in each image will be calculated using the position of the vehicle together with position of reference points and the given time stamp. After the centroid of object in each image is acquired, the distance between the two objects can be calculated by subtracting the two centroids taking only the x component.

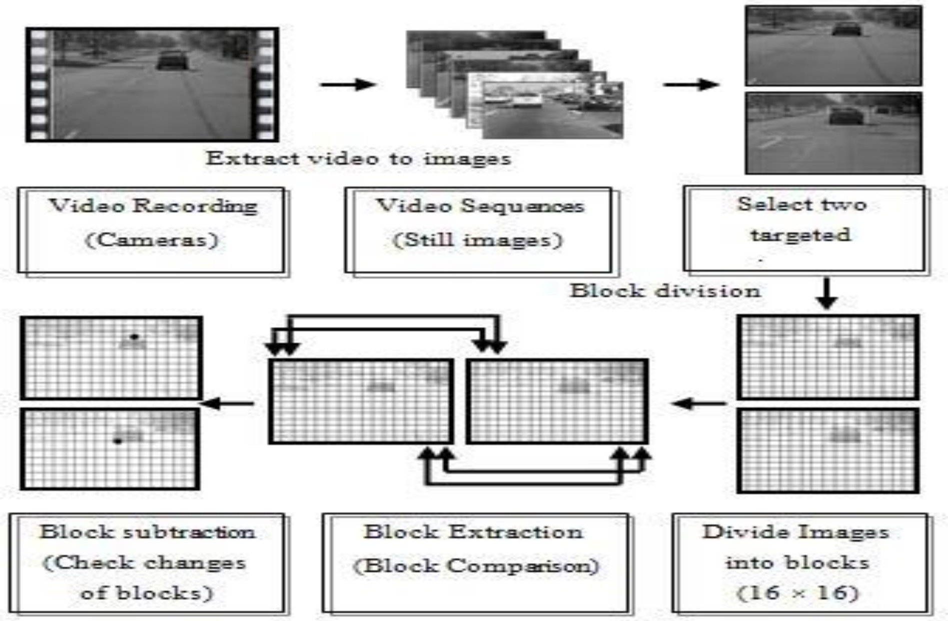

Extract video to images
Video Recording (Cameras)
Video Sequences (Still images)
Select two targeted
Block division
Block subtraction (Check changes of blocks)
Block Extraction (Block Comparison)
Divide Images into blocks (16 × 16)

Figure. Algorithm Development Approach

The video sequences extracted are preprocessed using the video image preprocessing where the block segmentation and block subtraction are done. The video sequence preprocessing is shown in the image below.

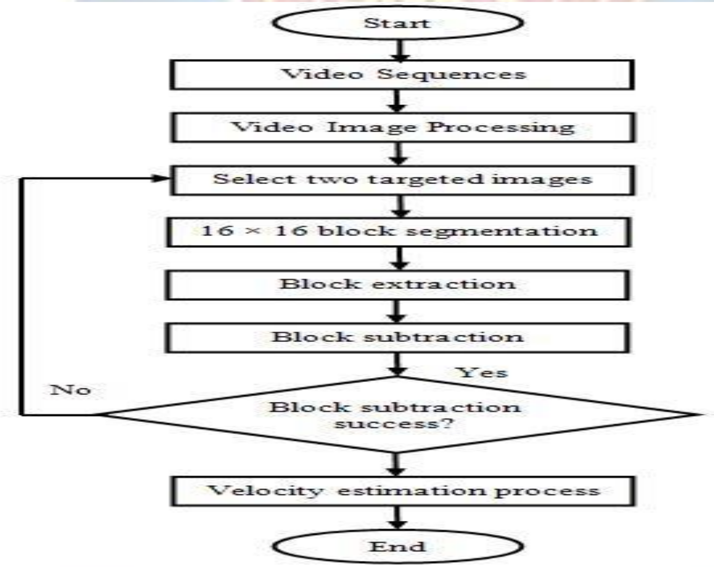

Start
Video Sequences
Video Image Processing
Select two targeted images
16 × 16 block segmentation
Block extraction
Block subtraction
Yes
No
Block subtraction success?
Velocity estimation process
End

Figure. Video Image Processing

Then the velocity of extracted image is calculated using the block extraction method where two blocks are subtracted from the calculated reference point.

Block Subtraction= | Block2 -Block1 |

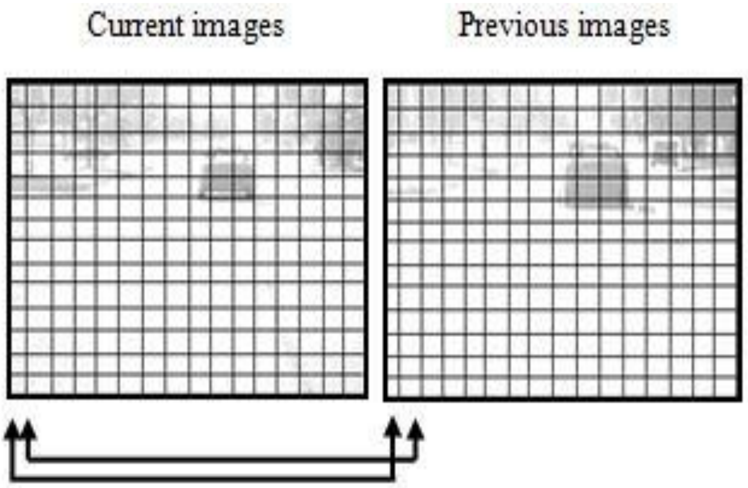


Figure. Block Subtraction

MATLAB Algorithm:

MATLAB is a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numeric computation. It can be used in a wide range of applications, including signal and image processing, communications, control design, test and measurement, financial modeling and analysis, and computational biology.

MATLAB was used to test the algorithms and then to model the system. MATLAB has a variety of toolboxes for different purposes. In this case Image Processing toolbox is used.

CHAPTER 5: PROJECT SYSTEM DESIGN

5.1 System Architecture

An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components. It is an important tool as it provides an overall view of the physical deployment of the software system and its evolution roadmap.

The fundamental properties, and the patterns of relationships, connections, constraints, and linkages among the components and between the system and its environment are known collectively as the architecture of the system.

The purpose of system architecture activities is to define a comprehensive solution based on principles, concepts, and properties logically related to and consistent with each other.

Software architecture is a sort of plan of the system and is primordial for the understanding, the negotiation, and the communication between all the stakeholders (user-side, customer, management, etc.). It makes it easier to understand the whole system and makes the decision-making process more efficient. Architectures must have both form and function and it is a good test of an architecture to measure its elegance. An architecture that is well designed will tend to be elegant and have a simplicity of form that will be obvious to those that take the time study it.

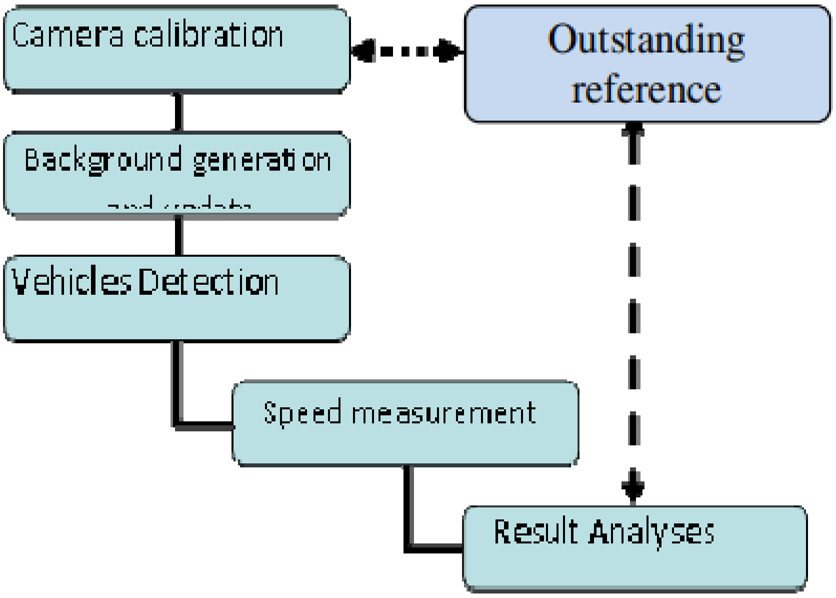


Figure. System Architecture

5.2 E-R Diagram

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

ER diagrams are a visual tool which is helpful to represent the ER model. It was proposed by Peter Chen in 1971 to create a uniform convention which can be used for relational database and network. He aimed to use an ER model as a conceptual modeling approach.

ER diagrams are created based on three basic concepts: entities, attributes and relationships. ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships. At first look, an ER diagram looks similar to the flowchart.

Below points show how to go about creating an ER diagram.

1. Identify all the entities in the system. An entity should appear only once in a particular diagram. Create rectangles for all entities and name them properly.
2. Identify relationships between entities. Connect them using a line and add a diamond in the middle describing the relationship.
3. Add attributes for entities. Give meaningful attribute names so they can be understood easily.

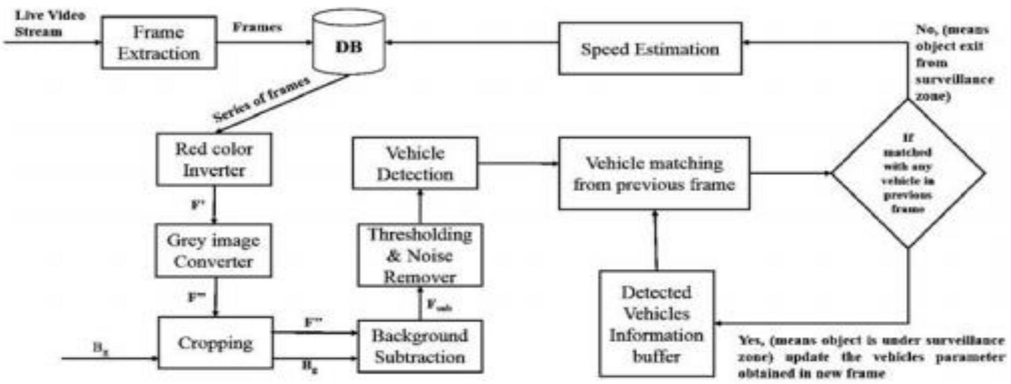

Live Video Stream
Frame Extraction
Frames
DB
Speed Estimation
No, (means object exit from surveillance zone)
Series of frames
Red color Inverter
F'
Grey image Converter
F''
Cropping
B_g
F''
B_g
Background Subtraction
F_{sub}
Thresholding & Noise Remover
Vehicle Detection
Vehicle matching from previous frame
If matched with any vehicle in previous frame
Detected Vehicles Information buffer
Yes, (means object is under surveillance zone) update the vehicles parameter obtained in new frame

Figure. E-R Diagram (Entity Relationship)

5.3 UML Diagrams

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

Goals:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

5.3.1 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

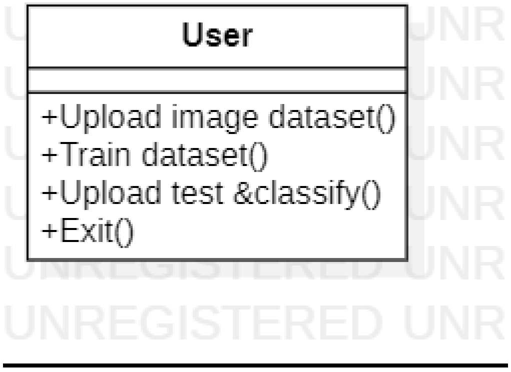


Figure. Class Diagram

5.3.2 Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

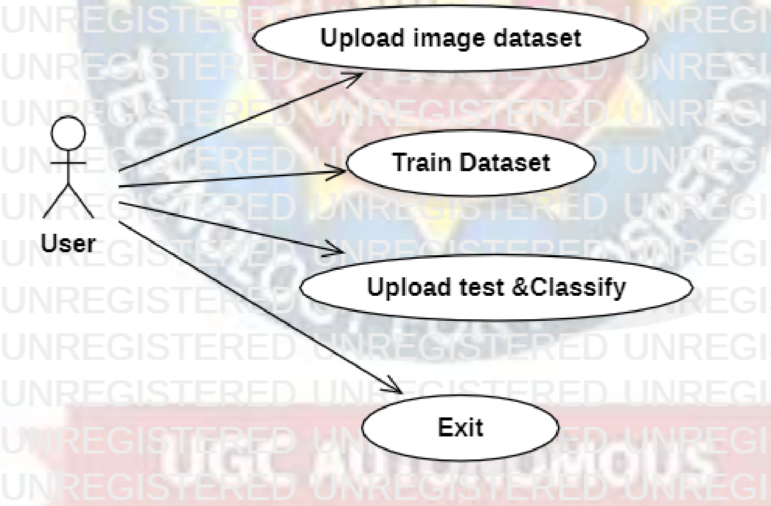


Figure. Use Case Diagram

5.3.3 Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

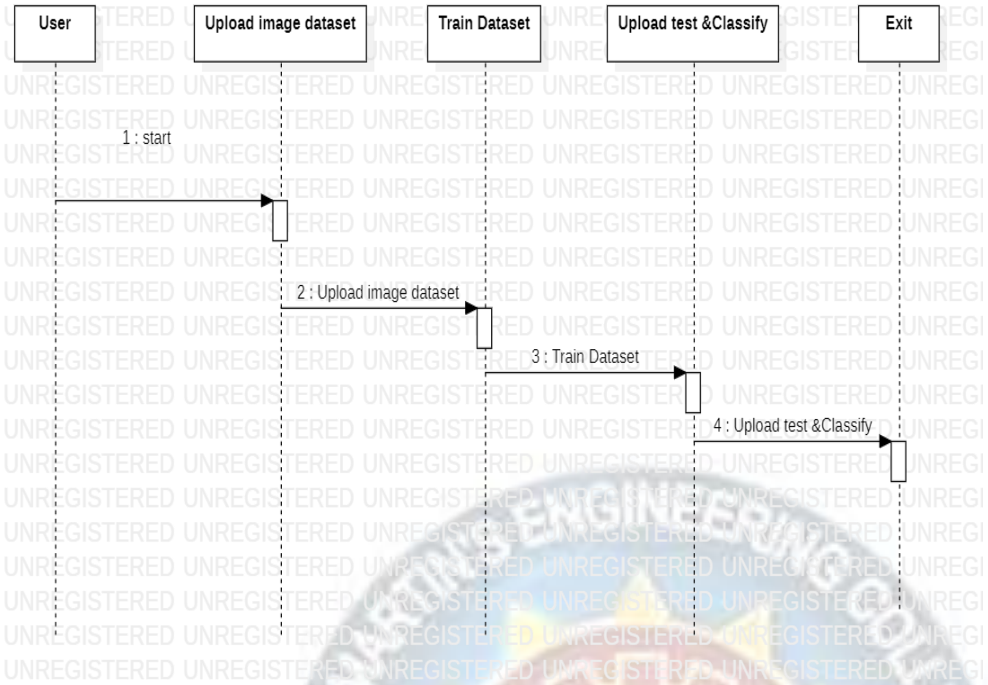


Figure. Sequence Diagram

5.3.4 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

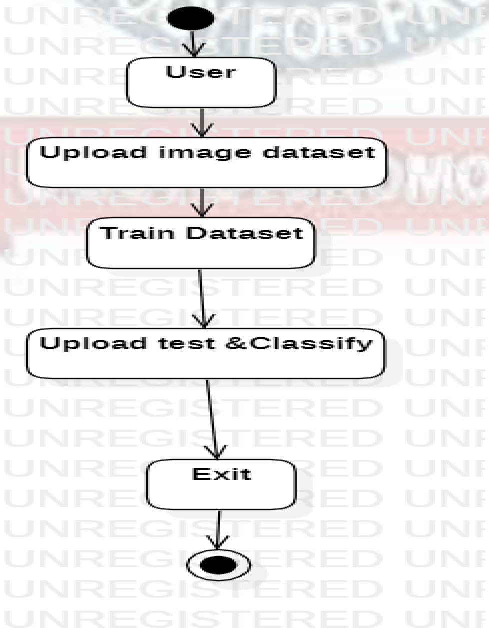


Figure. Activity Diagram

5.3.5 Package Diagram

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.

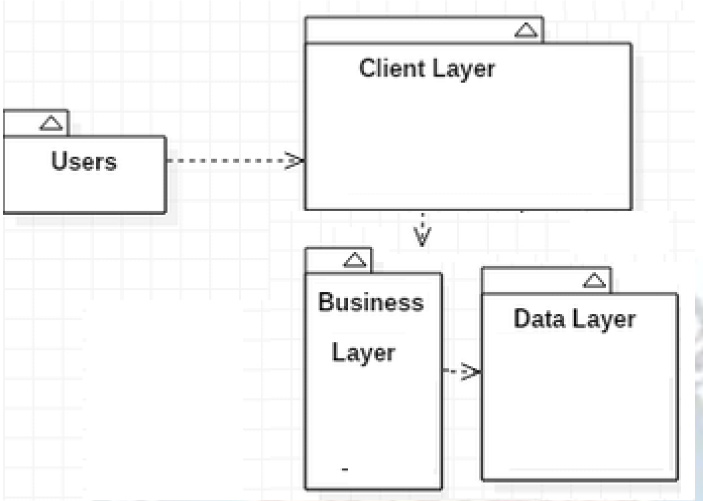


Figure. Package Diagram

5.3.6 Profile Diagram

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connectors and their components.

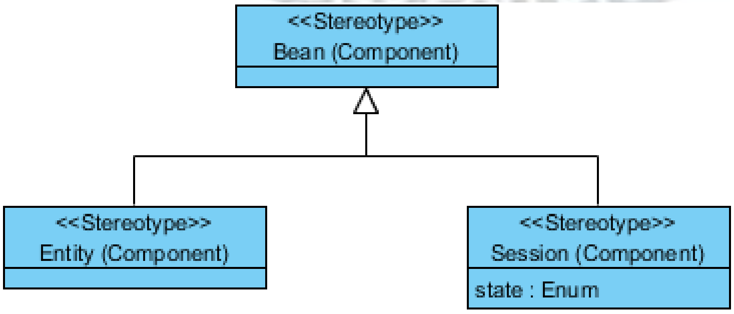


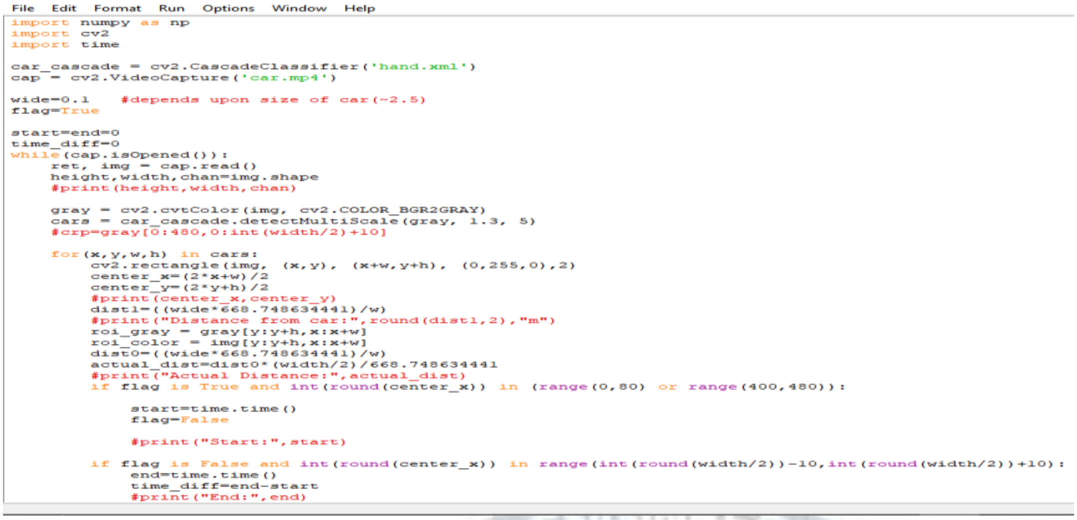
Figure. Profile Diagram

CHAPTER 6: PROJECT CODING

6.1. Code Templates

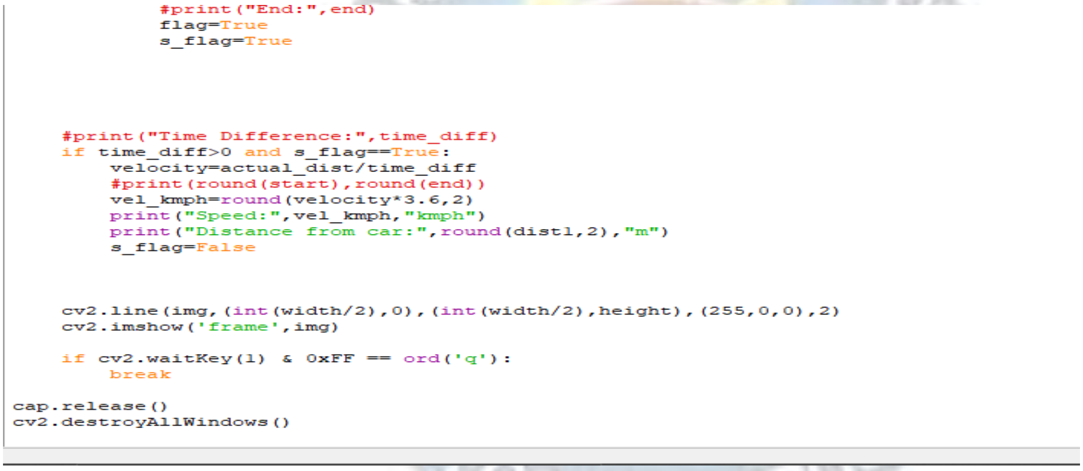
speeddetect.py

```
import numpy as np
import cv2
import time
car_cascade = cv2.CascadeClassifier('hand.xml')
cap = cv2.VideoCapture('car.mp4')
wide=0.1   #depends upon size of car(~2.5)
flag=True
start=end=0
time_diff=0
while(cap.isOpened()):
    ret, img = cap.read()
    height,width,chan=img.shape
    #print(height,width,chan)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    cars = car_cascade.detectMultiScale(gray, 1.3, 5)
    #crp=gray[0:480,0:int(width/2)+10]
    for(x,y,w,h) in cars:
        cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0),2)
        center_x=(2*x+w)/2
        center_y=(2*y+h)/2
        #print(center_x,center_y)
        dist1=((wide*668.748634441)/w)
        #print("Distance from car:",round(dist1,2),"m")
        roi_gray = gray[y:y+h,x:x+w]
        roi_color = img[y:y+h,x:x+w]
        dist0=((wide*668.748634441)/w)
        actual_dist=dist0*(width/2)/668.748634441
        #print("Actual Distance:",actual_dist)
        if flag is True and int(round(center_x)) in (range(0,80) or range(400,480)):
            start=time.time()
            flag=False
            #print("Start:",start)
        if flag is False and int(round(center_x)) in range(int(round(width/2))-10,int(round(width/2))+10):
            end=time.time()
            time_diff=end-start
            #print("End:",end)
            flag=True
            s_flag=True
    #print("Time Difference:",time_diff)
    if time_diff>0 and s_flag==True:
        velocity=actual_dist/time_diff
        #print(round(start),round(end))
        vel_kmph=round(velocity*3.6,2)
        print("Speed:",vel_kmph,"kmph")
        print("Distance from car:",round(dist1,2),"m")
        s_flag=False
    cv2.line(img,(int(width/2),0),(int(width/2),height),(255,0,0),2)
    cv2.imshow('frame',img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```



```
import numpy as np
import cv2
import time

car_cascade = cv2.CascadeClassifier('hand.xml')
cap = cv2.VideoCapture('car.mp4')

wide=0.1   #depends upon size of car(~2.5)
flag=True

start=end=0
time_diff=0
while(cap.isOpened()):
    ret, img = cap.read()
    height,width,chan=img.shape
    #print(height,width,chan)

    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    cars = car_cascade.detectMultiScale(gray, 1.3, 5)
    #crp=gray[0:480,0:int(width/2)+10]

    for(x,y,w,h) in cars:
        cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0),2)
        center_x=(2*x+w)/2
        center_y=(2*y+h)/2
        #print(center_x,center_y)
        dist1=((wide*668.748634441)/w)
        #print("Distance from car:",round(dist1,2),"m")
        roi_gray = gray[y:y+h,x:x+w]
        roi_color = img[y:y+h,x:x+w]
        dist0=((wide*668.748634441)/w)
        actual_dist=dist0*(width/2)/668.748634441
        #print("Actual Distance:",actual_dist)
        if flag is True and int(round(center_x)) in (range(0,80) or range(400,480)):

            start=time.time()
            flag=False

            #print("Start:",start)

        if flag is False and int(round(center_x)) in range(int(round(width/2))-10,int(round(width/2))+10):
            end=time.time()
            time_diff=end-start
            #print("End:",end)
```

Figure. Picture of code implementation



```
            #print("End:",end)
            flag=True
            s_flag=True



    #print("Time Difference:",time_diff)
    if time_diff>0 and s_flag==True:
        velocity=actual_dist/time_diff
        #print(round(start),round(end))
        vel_kmph=round(velocity*3.6,2)
        print("Speed:",vel_kmph,"kmph")
        print("Distance from car:",round(dist1,2),"m")
        s_flag=False


    cv2.line(img,(int(width/2),0),(int(width/2),height),(255,0,0),2)
    cv2.imshow('frame',img)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

Figure. Picture of Code Implementation(continue.)

hand.xml

```
<?xml version="1.0"?>
<opencv_storage>
<A_gest type_id="opencv-haar-classifier">
  <size>
    24 24</size>
  <stages>
    <_>
      <!-- stage 0 -->
      <trees>
        <_>
      <!-- tree 0 -->
      <_>
        <!-- root node -->
        <feature>
          <rects>
            <_>
              3 3 9 16 -1.</_>
            <_>
              3 7 9 8 2.</_></rects>
          <tilted>0</tilted></feature>
        <threshold>-0.0223442204296589</threshold>
        <left_val>0.7737345099449158</left_val>
        <right_val>-0.9436557292938232</right_val></_></_>
    <_>
      <!-- tree 1 -->
      <_>
        <!-- root node -->
        <feature>
          <rects>
            <_>
              0 9 12 5 -1.</_>
            <_>
              6 9 6 5 2.</_></rects>
          <tilted>0</tilted></feature>
        <threshold>-9.3714958056807518e-003</threshold>
        <left_val>0.5525149106979370</left_val>
        <right_val>-0.9004204869270325</right_val></_></_></trees>
  <stage_threshold>-0.3911409080028534</stage_threshold>
  <parent>-1</parent>
  <next>-1</next></_>
<_>
  <!-- stage 1 -->
  <trees>
```

```
<_>
  <!-- tree 0 -->
  <_>
    <!-- root node -->
    <feature>
      <rects>
        <_>
          12 14 12 10 -1.</_>
        <_>
          12 14 6 5 2.</_>
        <_>
          18 19 6 5 2.</_></rects>
      <tilted>0</tilted></feature>
    <threshold>0.0127444602549076</threshold>
    <left_val>-0.7241874933242798</left_val>
    <right_val>0.5557708144187927</right_val></_></_>
<_>
  <!-- tree 1 -->
  <_>
    <!-- root node -->
    <feature>
      <rects>
        <_>
          2 4 16 8 -1.</_>
        <_>
          2 8 16 4 2.</_></rects>
      <tilted>0</tilted></feature>
    <threshold>-0.0203973893076181</threshold>
    <left_val>0.3255875110626221</left_val>
    <right_val>-0.9134256243705750</right_val></_></_>
<_>
  <!-- tree 2 -->
  <_>
```

```
    <!-- root node -->
    <feature>
      <rects>
        <_>
          9 6 15 14 -1.</_>
        <_>
          9 13 15 7 2.</_></rects>
      <tilted>0</tilted></feature>
    <threshold>1.5015050303190947e-003</threshold>
    <left_val>-0.8422530293464661</left_val>
    <right_val>0.2950277030467987</right_val></_></_>
<_>
  <!-- tree 3 -->
  <_>
    <!-- root node -->
    <feature>
      <rects>
        <_>
          0 10 10 5 -1.</_>
        <_>
          5 10 5 5 2.</_></rects>
      <tilted>0</tilted></feature>
    <threshold>-9.5540005713701248e-003</threshold>
    <left_val>0.2949278056621552</left_val>
    <right_val>-0.8186870813369751</right_val></_></_>
<_>
  <!-- tree 4 -->
  <_>
    <!-- root node -->
    <feature>
      <rects>
        <_>
          8 0 16 6 -1.</_>
```

```
            <_>
              8 0 16 3 2.</_></rects>
          <tilted>1</tilted></feature>
        <threshold>-9.0454015880823135e-003</threshold>
        <left_val>-0.9253956079483032</left_val>
        <right_val>0.2449316978454590</right_val></_></_></trees>
  <stage_threshold>-0.8027257919311523</stage_threshold>
  <parent>0</parent>
  <next>-1</next></_>
<_>
  <!-- stage 2 -->
  <trees>
    <_>
      <!-- tree 0 -->
      <_>
        <!-- root node -->
        <feature>
          <rects>
            <_>
              11 9 9 6 -1.</_>
            <_>
              14 12 3 6 3.</_></rects>
          <tilted>1</tilted></feature>
        <threshold>0.0339135192334652</threshold>
        <left_val>-0.6010565757751465</left_val>
        <right_val>0.5952491760253906</right_val></_></_>
        <threshold>1.5015050303190947e-003</threshold>
        <left_val>-0.8422530293464661</left_val>
        <right_val>0.2950277030467987</right_val></_></_>
    <_>
      <!-- tree 1 -->
```

6.2 Outline of Various Files

1. speeddetect.py:

This python extension file is where the whole code of our module exists, and the operations performed on a video camera to detect vehicles is shown in form of code. Every operation in the code is under the module of computer vision Haar Cascade Library. Haar Cascade is an Object Detection algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features proposed by Viola and Jones in their research paper "Rapid Object Detection using a Boosted Cascade of Simple Features" published in 2001. This module will run until a vehicle is detected to the camera and preprocessing is done on that video sequence image.

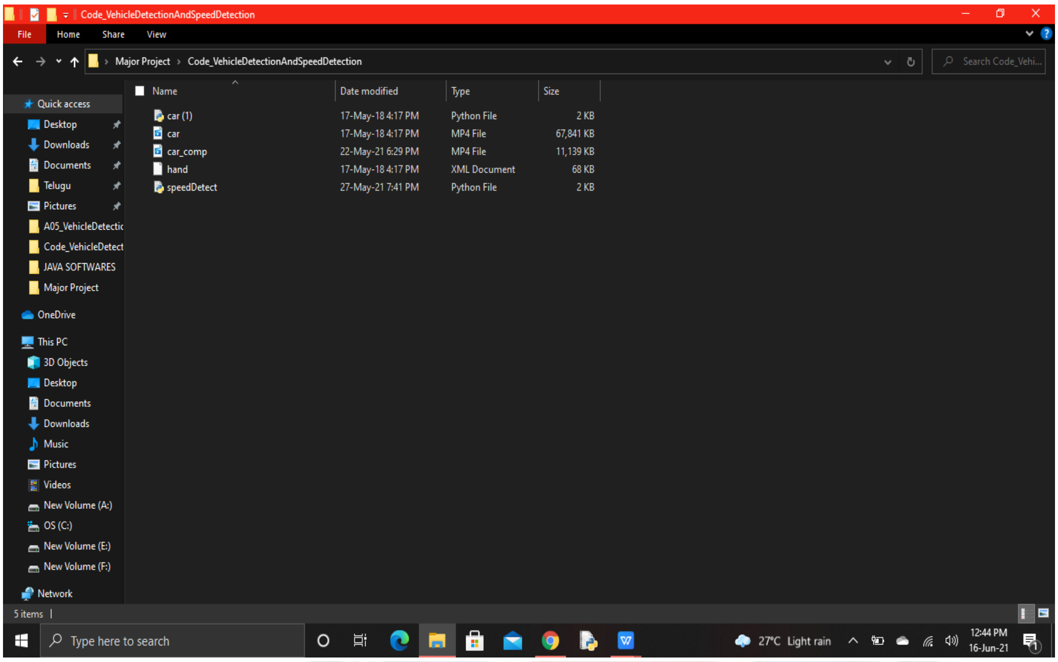
2. Hand.xml:

This XML file used in our code is extracted from xml extension where all the reference points from camera are chosen. Basically, this xml file is used to calculate the distance of reference point from camera. This file also operates the axis of a vehicle and calculates the threshold values of each vehicle detected. These values are used to calculate height, distance and velocity of vehicle.

3. Car.mp4:

This video file is given as input to our source code and the objects detected are depicted on this video file. This is one of the main files used for our project as it acts as a input to our project. Training is done on this video file using inbuilt operations of open computer vision libraries. This is the video file we have uploaded to our code basically to perform haar cascade operations on that video file. Haar Cascade is an Object Detection algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features proposed by Viola and Jones in their research paper "Rapid Object Detection using a Boosted Cascade of Simple Features" published in 2001. This module will run until a vehicle is detected to the camera and preprocessing is done on that video sequence image.

The below picture depicts the outline of our files used in our project. The files that are shown in the image are used in the code and inbuilt operations are performed on the video.



Name	Date modified	Type	Size
car (1)	17-May-18 4:17 PM	Python File	2 KB
car	17-May-18 4:17 PM	MP4 File	67,841 KB
car_comp	22-May-21 6:29 PM	MP4 File	11,139 KB
hand	17-May-18 4:17 PM	XML Document	68 KB
speedDetect	27-May-21 7:41 PM	Python File	2 KB

Figure. Outline of Files

CHAPTER 7: PROJECT 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 tests. Each test type addresses a specific testing requirement.

7.1 Various Test Cases

The various types of testing methodologies involved in this project are;

7.1.1 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. Testing is event driven and is more concerned with the basic outcome of screens or fields. 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. 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.

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.
- Duplicates within pages are omitted.

7.1.2 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. This is a structural testing, that relies on knowledge of its construction and is invasive. This is basically known for knowing the bugs in our code and output. 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. 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.

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

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

7.1.3 Functional Testing

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.

7.1.4 System Testing

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.

7.1.5 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. 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.

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

7.2 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.

The Black-Box can be any software system you want to test. For Example, an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation.

Here are the generic steps followed to carry out any type of Black Box Testing:

- Initially, the requirements and specifications of the system are examined.
- Tester chooses valid inputs (positive test scenario) to check whether cascade classifier processes them correctly. Also, some invalid inputs (negative test scenario) are chosen to verify that the cascade classifier can detect them.
- Tester determines expected outputs for all those inputs.
- Software tester constructs test cases with the selected inputs.
- The test cases are executed.
- Software tester compares the actual outputs with the expected outputs.
- Defects if any are fixed and re-tested.

Types of black box testing are:

- Functional testing - This black box testing type is related to the functional requirements of a system; it is done by software testers.
- Non-functional testing - This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.

- Regression testing - Regression Testing is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

7.3 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.

It is one of two parts of the Box Testing approach to software testing. Its counterpart, Blackbox testing, involves testing from an external or end-user type perspective. On the other hand, White box testing in software engineering is based on the inner workings of an application and revolves around internal testing.

The term "White Box" was used because of the see-through box concept. The clear box or White Box name symbolizes the ability to see through the software's outer shell into its inner workings. Likewise, the "black box" in "Black Box Testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested.

White box testing involves the testing of the software code for the following:

- Internal security holes.
- Broken or poorly structured paths in the coding processes.
- The flow of specific inputs through the code.
- Expected output.
- The functionality of conditional loops.
- Testing of each statement, object, and function on an individual basis.

The testing can be done at system, integration and unit levels of software development. One of the basic goals of white box testing is to verify a working flow for an application. It involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output, you have encountered a bug.

A major White box testing technique is Code Coverage analysis. Code

Coverage analysis eliminates gaps in a Test Case suite. It identifies areas of a program that are not exercised by a set of test cases. Once gaps are identified, you create test cases to verify untested parts of the code, thereby increasing the quality of the software product.

There are automated tools available to perform Code coverage analysis. Below are a few coverage analysis techniques a box tester can use:

Statement Coverage: - This technique requires every possible statement in the code to be tested at least once during the testing process of software engineering.

Branch Coverage: - This technique checks every possible path (if-else and other conditional loops) of a software application.

Apart from above, there are numerous coverage types such as Condition Coverage, Multiple Condition Coverage, Path Coverage, Function Coverage etc. Each technique has its own merits and attempts to test (cover) all parts of software code. Using Statement and Branch coverage you generally attain 80-90% code coverage which is sufficient.

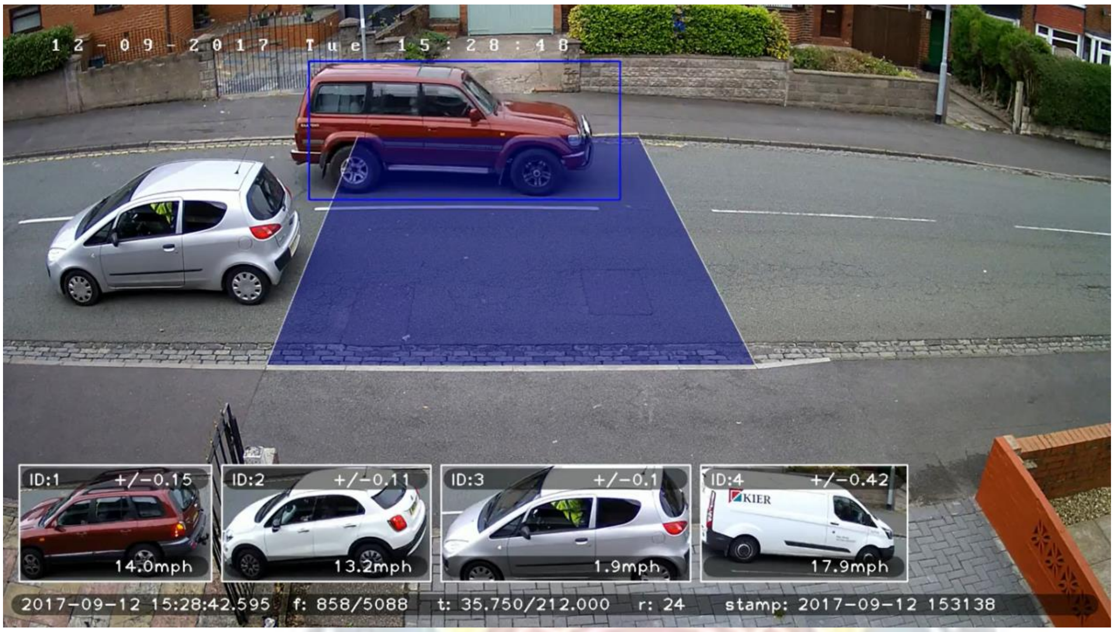
Advantages of White Box Testing

- Code optimization by finding hidden errors.
- White box tests cases can be easily automated.
- Testing is more thorough as all code paths are usually covered.
- Testing can start early in SDLC even if GUI is not available.

Disadvantages of White Box Testing

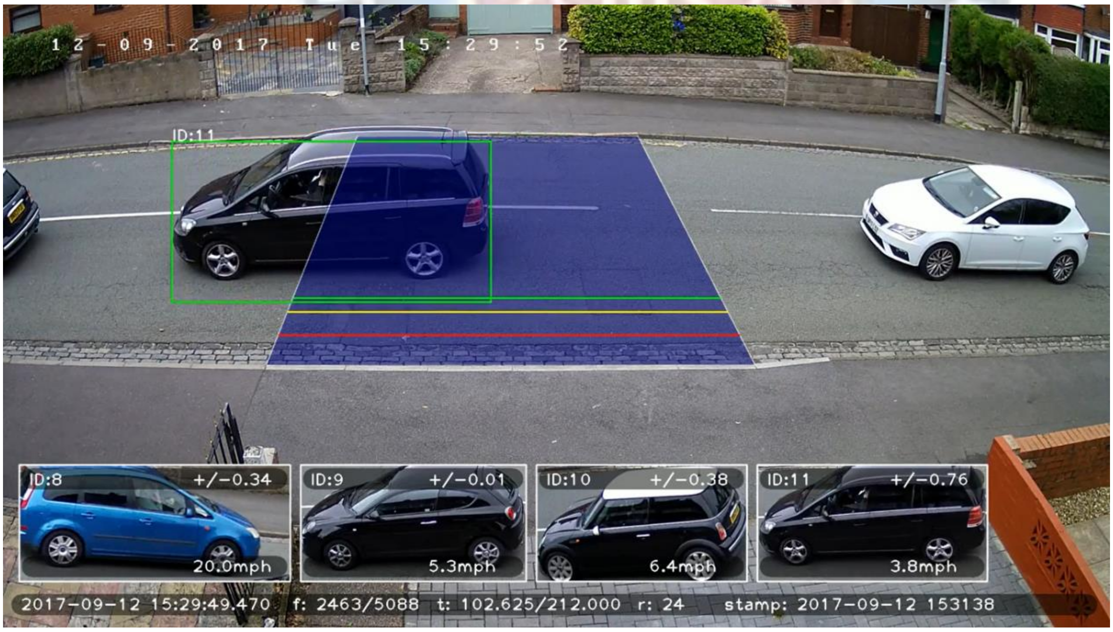
- White box testing can be quite complex and expensive.
- Developers who usually execute white box test cases detest it. The white box testing by developers is not detailed can lead to production errors.
- White box testing requires professional resources, with a detailed understanding of programming and implementation.
- White-box testing is time-consuming, bigger programming applications take the time to test fully.

CHAPTER 8: OUTPUT SCREENS



Vehicle entering reference point.

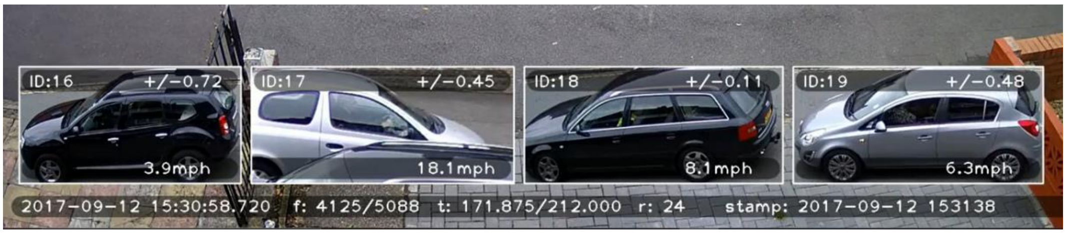
The Blue Line Around Car depicts that it is entering the reference point and the velocity and time of covering the blue block is shown in output below.



Speed Detected of vehicle is shown.

The green line around the car detected is showing that the car is leaving the reference point. The speed, height and distance are depicted in the below image of output.

There is a number called ID number which shows the count of vehicles detected by our algorithm.

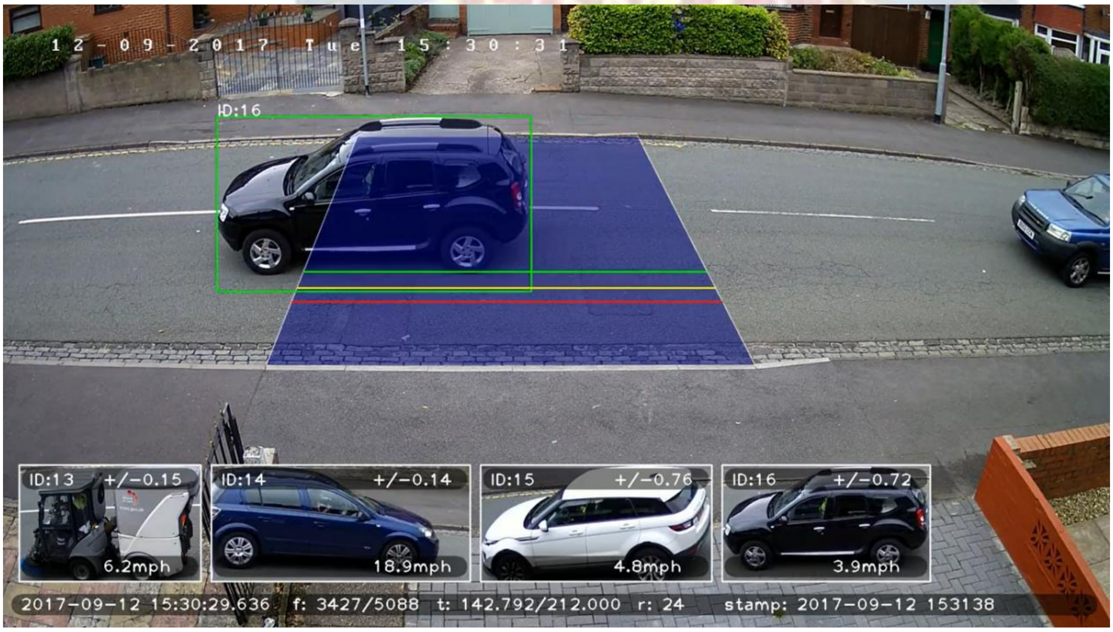

ID:16
+/−0.72
3.9mph
ID:17
+/−0.45
18.1mph
ID:18
+/−0.11
8.1mph
ID:19
+/−0.48
6.3mph
2017−09−12 15:30:58.720 f: 4125/5088 t: 171.875/212.000 r: 24 stamp: 2017−09−12 153138

Speed, Distance, count of vehicle detected.

In above image the threshold values which are basically known as axis for the vehicle is also shown below the detected image of vehicle.

CHAPTER 9: EXPERIMENTAL RESULTS

The approach proposed in this project is tested on four different videos. The average detection accuracy achieved by proposed approach is 87.7%. The proposed approach uses cropping operation on extracted images to minimize the false detection of vehicle on the road. The average false positive detection in the proposed approach is lower than average false positive detection in leading approaches such as RADAR guns. Maximum tracking accuracy achieved by the proposed technique is up to 98.3% in the afternoon session, which is more achieved compared to radar gun reach. The accuracy of the system in frequency measurement was tested using a function generator and an oscilloscope. The input to the amplifier was connected to the output of the function generator and an oscilloscope was connected for reference. Small sinusoidal signal was applied to the circuit and the frequency varied. The table below summarizes the result obtained from the test.



Experimental result of output

Our algorithm was able to track most of the vehicles but due to the unmatched threshold values some vehicles are not able to capture the speed, height and distance from reference point.

Using the process of testing the results obtained were able to show some drastic changes to the predicted output. We also detected the threshold preprocessed images through which the result of each detected image is shown in the output. There are many inputs that we have worked with to build the project. We collected those inputs from a website called Kaggle where we can broadly get any input for a machine learning project.

CHAPTER 10: CONCLUSION AND FUTURE ENHANCEMENT

Conclusion

This project proposes an approach to detect and track the moving vehicles and estimation of their speeds. The innovation of the approach lies in the selection of the Region of Interest for the vehicle detection.

In the proposed method, detection and tracking of the moving vehicles utilizes parameters such as position, height and width of vehicle instead of features extraction. This requires lesser computation and memory.

Using the surveillance cameras, we can detect the vehicle and speed of vehicle. This project also detects the distance travelled by the vehicle from a point of view. The average false positive detection in the proposed approach is lower than average false positive detection in leading approaches such as STA12. Maximum tracking accuracy achieved by the proposed technique is up to 98.3% in the afternoon session, but the average tracking accuracy of the proposed approach is about 92.2% that is improvement to other methods. In the proposed method, detection and tracking of the moving vehicles utilizes parameters such as position, height and width of vehicle instead of features extraction. This requires lesser computation and memory. The proposed approach stores vehicles parameters estimated speed of the detected vehicles in the database. The proposed system can be adopted easily in existing traffic management system.

The proposed method gives better results as compared to previous techniques. Background subtraction is robust against illumination changes in real world. Also, by extracting the noise immunity is improved. As the distance is mapped on the image by calculating it from real world. So, the calculated speed is approximated to actual speed.

The designed speed detection system was capable of continuously monitoring the speed of the approaching vehicle. The system was more accurate in identifying the vehicles that was written in the style similar to the one used in our template.

Future Enhancement

A novel algorithm which takes advantage of the two-color based characteristics and combines them for object extraction is introduced. This approach is more robust against misdetections and the problem of the merging or splitting of vehicles and finally, in the third step, the vehicle speed is determined. The approach used is not affected by weather changes. This requires lesser computation and memory. Vehicle extraction and speed detection had been implemented using the Python software.

The major objective of this work is to track the moving vehicles on the road. The various concepts of deep learning and computer vision have been utilized for this purpose. Track by detection framework was applied for real-time vehicle tracking.

CHAPTER 11: REFERENCES

1. Chen, Hsinchun, et al. "Crime data mining: a general framework and some examples." computer 37.4 Authorized licensed use limited to: University of Southern Queensland. Downloaded on August 02,2020 at 02:07:40 UTC from IEEE Xplore. Restrictions apply.

2. Ektefa, Mohammadreza, et al. "Intrusion detection using data mining techniques." Information Retrieval & Knowledge Management, (CAMP), 2010 International Conference on. IEEE, 2010.

3. Clifton, Chris, and Gary Gengo. "Developing custom intrusion detection filters using data mining." MILCOM 2000. 21st Century Military Communications Conference Proceedings. Vol. 1. IEEE, 2000.

4. Dickerson, John E., and Julie A. Dickerson. "Fuzzy network profiling for intrusion detection." Fuzzy Information Processing Society, 2000. NAFIPS. 19th International Conference of the North American. IEEE, 2000.

5. Siraj, Ambareen, Susan M. Bridges, and Rayford B. Vaughn. "Fuzzy cognitive maps for decision support in an intelligent intrusion detection system." IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th. Vol. 4. IEEE, 2001.

6. Nath, Shyam Varan. "Crime pattern detection using data mining." Web intelligence and intelligent agent technology workshops, 2006. wi-iat 2006 workshops. 2006 ieee/wic/acm international conference on. IEEE, 2006.

7. Florez, German, S. A. Bridges, and Rayford B. Vaughn. "An improved algorithm for fuzzy data mining for intrusion detection." Fuzzy Information Processing Society, 2002. Proceedings. NAFIPS. 2002 Annual Meeting of the North American. IEEE, 2002.

8. Panda, Mrutyunjaya, and Manas Ranjan Patra. "A comparative study of data mining algorithms for network intrusion detection." Emerging Trends in Engineering and Technology, 2008. ICETET'08. First International Conference on. IEEE, 2008.

9. Vaidya, Jaideep, and Chris Clifton. "Privacy-preserving data mining: Why, how, and when." IEEE Security & Privacy 2.6 (2004): 19-27.

10. Mukkamala, Srinivas, Guadalupe Janoski, and Andrew Sung. "Intrusion detection using neural networks and support vector machines." Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on. Vol. 2. IEEE, 2002.

A

Project Report

On

EFFICIENT AND DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS

Submitted by

Ms. Saloni Dayama (17K81A1246)

Mr. K. Harish Goud (17K81A1218)

Mr. Irfan Ali (17K81A1216)

Mr. V. Vinay Reddy (17K81A1253)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Mr. A. Veera Babu

Assistant Professor

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN'S ENGINEERING COLLEGE
An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **EFFICIENT AND DEPLOYMENT CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS**, is being submitted by **SALONI DAYAMA (17K81A1246), K. HARISH GOUD (17K81A1218), IRFAN ALI (17K81A1216) V. VINAY REDDY (17K81A1253)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

	Head of the Department
A. VEERA BABU	Dr. R. NAGARAJU
Department of Information Technology	Department of Information Technology

Internal Examiner

External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of **Information Technology**, session: <2017 – 2021>, St. Martin’s Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **Efficient And Deployable Click Fraud Detection For Mobile Applications** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

Saloni Dayama	**17K81A1246**
K. Harish Goud	**17K81A1218**
Irfan Ali	**17K81A1216**
V. Vinay Reddy	**17K81A1253**

LASYA
INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **IRFAN ALI** WITH ROLL NO.**17K81A1216**, **K.HARISH GOUD** WITH ROLL NO.**17K81A1218**, **SALONI DAYAMA** WITH ROLL NO.**17K81A1246**, **V.VINAY REDDY** WITH ROLL NO.**17K81A1253**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**EFFICIENT AND DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd. ★ Kompally-500014 ★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R. NAGARAJU**, Head of the Department, **Information Technology**, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, in Information Technology, for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **A. VEERA BABU**, Assistant Professor, Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

SALONI DAYAMA	**17K81A1246**
K. HARISH GOUD	**17K81A1218**
IRFAN ALI	**17K81A1216**
V. VINAY REDDY	**17K81A1253**

TABLE OF CONTENTS

Title Page **No**

+

ABSTRACT

Mobile advertising plays a vital role in the mobile app ecosystem. A major threat to the sustainability of this ecosystem is click fraud, i.e., ad clicks performed by malicious code or automatic bot problems. Existing click fraud detection approaches focus on analyzing the ad requests at the server side. However, such approaches may suffer from high false negatives since the detection can be easily circumvented, e.g., when the clicks are behind proxies or globally distributed. In this paper, we present AdSherlock, an efficient and deployable click fraud detection approach at the client side (inside the application) for mobile apps. AdSherlock splits the computation-intensive operations of click request identification into an offline procedure and an online procedure. In the offline procedure, AdSherlock generates both exact patterns and probabilistic patterns based on URL (Uniform Resource Locator) tokenization. These patterns are used in the online procedure for click request identification and further used for click fraud detection together with an ad request tree model. We implement a prototype of AdSherlock and evaluate its performance using real apps. The online detector is injected into the app executable archive through binary instrumentation. Results show that AdSherlock achieves higher click fraud detection accuracy compared with state of the art, with negligible runtime overhead.

LIST OF FIGURES

1. INTRODUCTION

1.1 Project Overview:

Mobile advertising plays a vital role in the mobile app ecosystem. A recent report shows that mobile advertising expenditure worldwide is projected to reach $247.4 billion in 2020 [1]. To embed ads in an app, the app developer typically includes ad libraries provided by a third-party mobile ad provider such as AdMob [2]. When a mobile user is using the app, the embedded ad library fetches ad content from the network and displays ads to the user. The most common charging model is PPC (Pay-Per-Click) [3], where the developer and the ad provider get paid from the advertiser when a user clicks on the ad.

A major threat to the sustainability of this ecosystem is click fraud [4], i.e., clicks (i.e., touch events on mobile devices) on ads which are usually performed by malicious code programmatically or by automatic bot problems. There are many different click fraud tactics which can typically be characterized into two types: in-app frauds insert malicious code into the app to generate forged ad clicks; bots-driven frauds employ bot programs (e.g., a fraudulent application) to click on advertisements automatically. To quantify the inapp ad fraud in real apps, a recent work MAdFraud [5] conducts a large scale measurement about ad fraud in real world apps.

In a dataset including about 130K Android apps, MAdFraud reports that about 30% of apps make ad requests while running in the background. Focusing on bots-driven click fraud, another recent work uses an automated click generation tool ClickDroid [4] to empirically evaluate eight popular advertising networks by performing real click fraud attacks on them. Results [4] show that six advertising networks out of eight are vulnerable to these attacks. Aiming at detecting click frauds in mobile apps, a straightforward approach is a threshold-based detection at the server-side. If an ad server is receiving a high number of clicks with the same device identifier (e.g., IP address) in a short period, these clicks can be considered as fraud. This straightforward approach, however, may suffer from high false negatives since the detection can be easily circumvented when the clicks are behind proxies or globally distributed. In the literature, there are also more sophisticated approaches [6], [7] focusing on detecting click frauds at the server-side.

1.2 Project Objectives:

The precisions of these server-side approaches, however, are not sufficient enough for the click fraud problem. For example, in a recent mobile ad fraud competition [6], the best three approaches achieve only a precision of 46.15% to 51.55% using various machine learning techniques. Given the insufficient precision of server-side approaches, a natural question comes up: how about client-side approaches? In fact, compared with the server-side approaches, it is easier to tell whether there is an actual user input at the client side. However, the attacker of the click fraud could be the app developers themselves, since the developers will get paid for those fraudulent ad clicks. Due to this conflict-of-interest problem, we cannot assume the existence of coordination from developers in designing a client-side approach for click fraud detection, e.g., a click fraud detection SDK. Therefore, in this paper, we focus on designing a client-side approach to detect click frauds in mobile apps, without coordination from developers.

There are two major challenges in designing such a system. First, for a mobile client, its resources are constrained in terms of computation, memory, and energy. Therefore, the proposed approach must perform the complete fraud detection process. See efficiently, without causing significant overhead. This means that we need to design new algorithms to detect click frauds since existing machine-learning algorithms used by server-side approaches are not suitable for the client side. Second, the click fraud detection should be able to execute under practical user scenarios, instead of a controlled environment dedicated to fraud detection. In MAdFraud [5], a controlled environment (i.e., only one app is running and the HTTP requests are collected for offline analysis) is used to measure the ad fraud behavior of a vast number of apps. However, in our case, the click fraud detection should happen inside the mobile client without outside support, i.e., be deployable in real-world scenarios.

1.3 Scope of the Project:

In this paper, we propose AdSherlock, an efficient and deployable click fraud detection approach for mobile apps at the client side. Note that as a client-side approach, AdSherlock is orthogonal to existing server-side approaches. AdSherlock is designed to be used by app stores to ensure a healthy mobile app ecosystem. AdSherlock's high accuracy helps market operators to fight both in-app frauds and bots-driven frauds. Note that, AdSherlock can also be used by any third parties to detect in-app frauds. For example, ad providers can employ AdSherlock to check whether apps embedding their libraries have in-app fraudulent behaviors.

To achieve these goals, AdSherlock relies on an accurate offline pattern extractor and a lightweight online fraud detector. AdSherlock works in two stages. At the first stage, the offline pattern extractor automatically executes each app and generates a set of traffic patterns for efficient ad request identification, i.e., extracts common token patterns across different ad requests.

1.4 Organization of Chapters:

1.4.1 Introduction

Specifically, after tokenization of the network requests, AdSherlock generates both exact patterns and probabilistic patterns for robust matching. Using the offline pattern extractor, AdSherlock can perform the computation and I/O intensive pattern generation operations in an offline manner, without degrading the online fraud detection operations. At the second stage, the online fraud detector as well as the generated patterns are instrumented into the app and run with the app in actual user scenarios. Inside the app, AdSherlock uses an ad request tree model to identify click requests accurately and efficiently. Since the online fraud detector runs inside the app, it can obtain the fine-grained user input events which are further employed for click fraud detection.

We implement AdSherlock and evaluate its performance using real apps. Results show that AdSherlock achieves higher click fraud detection accuracy compared with state of the art, with negligible runtime overhead.

The contributions of this paper are summarized as follows:

- We present the design and implementation of AdSherlock, the first system which can achieve efficient and deployable click fraud detection at the client side.

- We propose a pattern generation mechanism that generates patterns for ad requests and non-ad requests with high accuracy. We also propose an efficient method for online click fraud detection based on an ad request tree model.

- We implement AdSherlock and compare its performance with the state-of-art approach. Results show that Ad- Sherlock achieves higher detection accuracy with lower overhead.

1.4.2 Literature Survey

Research on ad frauds has been extensively carried in the realm of web applications. The relevant literature mostly focuses on click fraud which generally consists of leveraging a single computer or botnets to drive fake or undesirable impressions and clicks. A number of research studies have extensively characterized click frauds [1, 8, 46] and analysed its profit model [43]. Approaches have also been proposed to detect click frauds by analysing network traffic [44, 45] or by mining search engine's query logs [63].

Nevertheless, despite the specificities of mobile development and usage models, the literature on in-app ad frauds is rather limited. One example of work is the DECAF [37] approach for detecting placement frauds: these consist in manipulating visual layouts of ad views (also referred to as elements or controls) to trigger undesirable impressions in Windows Phone apps. DECAF explores the UI states (which refer to snapshots of the UI when the app is running) in order to detect ad placement frauds implemented in the form of hidden ads, the stacking of multiple ads per page, etc. MAdFraud [13], on the other hand, targets Android apps to detect in-app click frauds by analysing network traffic.

Unfortunately, while the community still struggles to properly address well-known, and often trivial, cases of ad frauds, deception techniques used by app developers are even getting more sophisticated, as reported recently in news outlets [24, 31]. Indeed, besides the aforementioned click and placement frauds, many apps implement advanced procedures for tricking users into unintentionally clicking ad views while they are interacting with the app UI elements. In this work, we refer to this type of ad frauds as dynamic interaction frauds.

Figure 1 illustrates the case of the app taijiao music1 where an ad view gets unexpectedly popped up on top of the exit button when the user wants to exit the app: this usually leads to an unintentional ad click. Actually, we performed a user study on this app and found that 9 out of 10 users were tricked into clicking the ad view. To the best of our knowledge, such frauds have not yet been explored in the literature of mobile ad frauds, and are thus not addressed by the state-of-the-art detection approaches.

This paper. We perform an exploratory study of a wide range of new ad fraud types in Android apps and propose an automated approach for detecting them in market apps. To that end, we first provide a taxonomy that characterizes a variety of mobile ad frauds including both static placement frauds and dynamic interaction frauds. While detection of the former can be performed via analysing thestatic information of the layout in a single UI state [37], detection of the latter presents several challenges, notably for:

Dynamically exercising ad views in a UI state, achieving scalability, and ensuring good coverage in transitions between UI states: A UI state is a running page that contains several visual views/elements, also referred to as controls in Android documentation. Because dynamic interaction frauds involve sequences of UI states, a detection scheme must consider the transition between UI states, as well as background resource consumption such as network traffic. For example, in order to detect the ad fraud case presented in Figure 1, one needs to analyse both current and next UI states to identify any ad view that is placed on top of buttons and which could thus entice users to click on ads unexpectedly. Exercising apps to uncover such behaviours can however be timeconsuming: previous work has shown that it takes several hours to traverse the majority UI states of an app based on existing Android automation frameworks [33].

Automatically distinguishing ad views among other views: In contrast with UI on the Windows Phone platform targeted by the state-of-the-art (e.g., DECAF [37]), Android UI models are generic and thus it is challenging to identify ad views in a given

UI state since no explicit labels are provided to distinguish them from other views (e.g., text views). During app development, a view can be added to the Activity, which represents a UI state implementation in Android, by either specifying it in the XML layout [18] or embedding it in the source code. In preliminary investigations, we found that most ad views are actually directly embedded in the code, thus preventing any identification via straightforward XML analysis.

Towards building an approach that achieves accuracy and scalability in Android ad fraud detection, we propose two key techniques aimed at addressing the aforementioned challenges:

Transition graph-based UI exploration. This technique builds a UI transition graph by simulating interaction events associated with user manipulation. We first capture the relationship between UI states through building the transition graphs between them, then identify ad views based on call stack traces and unique features gathered through comparing the ad views and other views in UI states. The scalability of this step is boosted by our proposed ad-first exploration strategy, which leverages probability distributions of the presence of an ad view in a UI state.

Heuristics-supported ad fraud detection. By manually investigating various real-world cases of ad frauds, we devise heuristic rules from the observed characteristics of fraudulent behaviour. Runtime analysis focusing on various behavioural aspects such as view size, bounds, displayed strings or network traffic, is then mapped against the rules to detect ad frauds.

1.4.3 Software & Hardware Requirements

1.4.3.1 Software Requirements

Operating system	:	Windows 10.
Coding Language	:	Python.
Front-End	:	Python.
Designing	:	Html,css,javascript.
Data Base	:	MySQL.

1.4.3.2 Hardware Requirements

System	:	Pentium IV 2.4 GHz.
Hard Disk	:	40 GB.
Floppy Drive	:	1.44 Mb.
Monitor	:	14’ Colour Monitor.
Mouse	:	Optical Mouse.
Ram	:	512 Mb.

1.4.4 Software Development Analysis

1.4.4.1 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.

1.4.4.2 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.

1.4.4.3 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.

1.4.4.4 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.

1.4.4.5 Approaches

As a field, machine learning is closely related to computational statistics, so having a 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.

1.4.4.5.1 K-Nearest Neighbor

The k-nearest neighbor algorithm is a pattern recognition model that can be used for classification as well as regression. Often abbreviated as k-NN, the **k** in k-nearest neighbor is a positive integer, which is typically small. In either classification or regression, the input will consist of the k closest training examples within a space.

We will focus on k-NN classification. In this method, the output is class membership. This will assign a new object to the class most common among its k nearest neighbors. In the case of k = 1, the object is assigned to the class of the single nearest neighbor.

Let’s look at an example of k-nearest neighbor. In the diagram below, there are blue diamond objects and orange star objects. These belong to two separate classes: the diamond class and the star class.

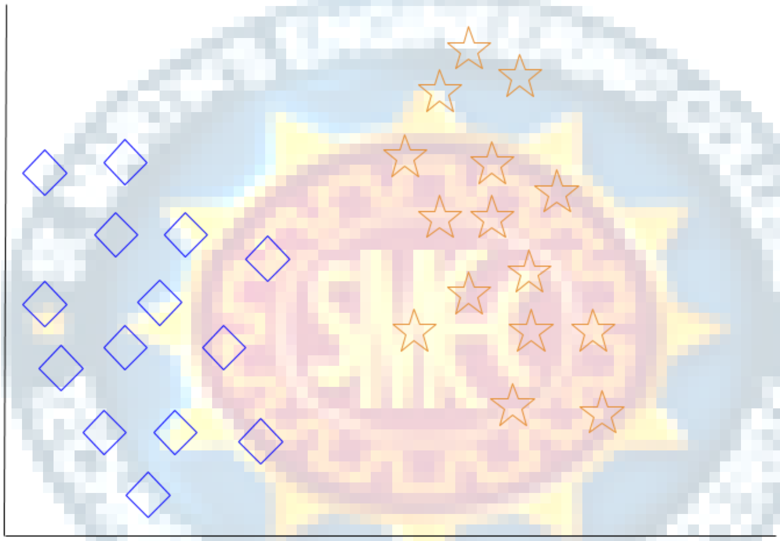


Fig 1.4.4.5.1 K – NN1

When a new object is added to the space — in this case a green heart — we will want the machine learning algorithm to classify the heart to a certain class.

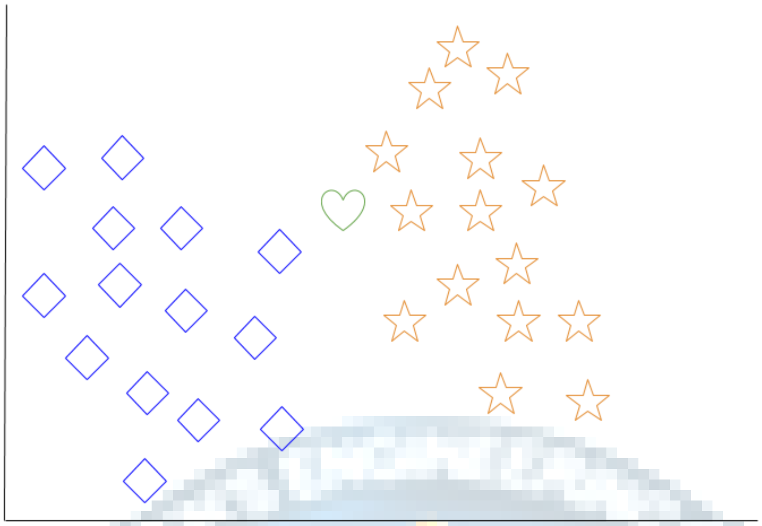


Fig 1.4.4.5.2 K – NN2

When we choose k = 3, the algorithm will find the three nearest neighbors of the green heart in order to classify it to either the diamond class or the star class.In our diagram, the three nearest neighbors of the green heart are one diamond and two stars. Therefore, the algorithm will classify the heart with the star class.

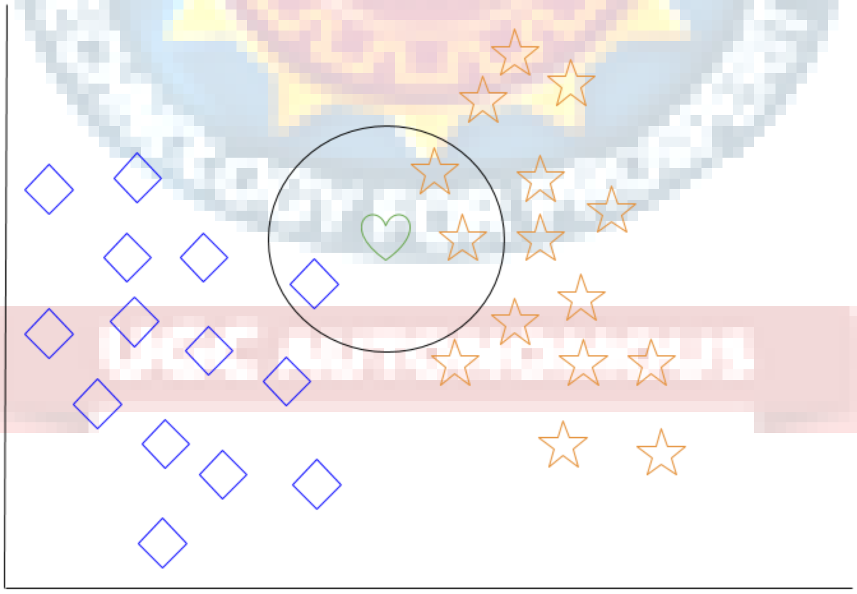


Fig 1.4.4.5.2 K – NN3

Among the most basic of machine learning algorithms, k-nearest neighbor is considered to be a type of “lazy learning” as generalization beyond the training data does not occur until a query is made to the system.

1.4.4.6 Decision Tree Learning

For general use, decision trees are employed to visually represent decisions and show or inform decision making. When working with machine learning and data mining, decision trees are used as a predictive model.

These models map observations about data to conclusions about the data’s target value. The goal of decision tree learning is to create a model that will predict the value of a target based on input variables. In the predictive model, the data’s attributes that are determined through observation are represented by the branches, while the conclusions about the data’s target value are represented in the leaves.When “learning” a tree, the source data is divided into subsets based on an attribute value test, which is repeated on each of the derived subsets recursively. Once the subset at a node has the equivalent value as its target value has, the recursion process will be complete.

Let’s look at an example of various conditions that can determine whether or not someone should go fishing. This includes weather conditions as well as barometric pressure conditions. 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. 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.

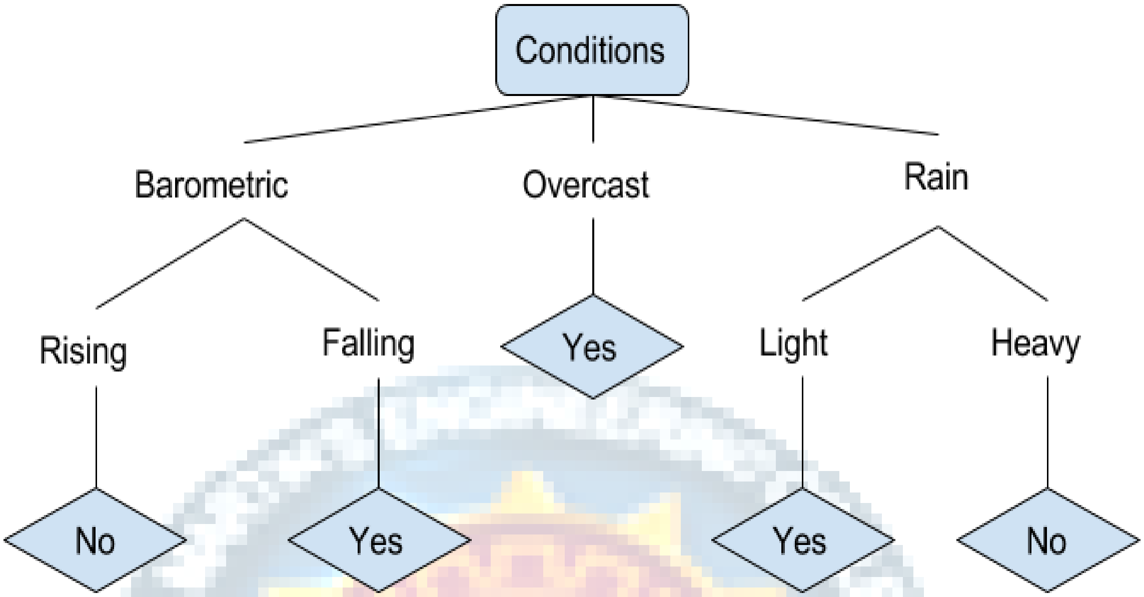


Fig 1.4.4.6.1 Decision Tree

In the simplified decision tree above, an example is classified by sorting it through the tree to the appropriate leaf node. This then returns the classification associated with the particular leaf, which in this case is either a Yes or a No. The tree classifies a day's conditions based on whether or not it is suitable for going fishing.

A true classification tree data set would have a lot more features than what is outlined above, but relationships should be straightforward to determine. When working with decision tree learning, several determinations need to be made, including what features to choose, what conditions to use for splitting, and understanding when the decision tree has reached a clear ending.

1.4.5 Project System Design

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

1.4.5.1 ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto* standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made the choice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needs suddenly change.

Through the ODBC Administrator in Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN.

The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you setup a separate database application, such as SQL Server Client or Visual Basic 4.0. When the ODBC icon is installed in Control Panel, it uses a file called ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand-alone program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and each maintains a separate list of ODBC data sources.

From a programming perspective, the beauty of ODBC is that the application can be written to use the same set of function calls to interface with any data source, regardless of the database vendor. The source code of the application doesn't change whether it talks to Oracle or SQL Server. We only mention these two as an example. There are ODBC drivers available for several dozen popular database systems. Even Excel spreadsheets and plain text files can be turned into data sources. The operating system uses the Registry information written by ODBC Administrator to determine which low-level ODBC drivers are needed to talk to the data source (such as the interface to Oracle or SQL Server). The loading of the ODBC drivers is transparent to the ODBC application program. In a client/server environment, the ODBC API even handles many of the network issues for the application programmer.

The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn't as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow. Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of "plug-in" database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC's framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after.

The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

1.4.6 Project Coding

Global.jsp

```
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%@page import="action.Database"%>
<%
    String name = request.getParameter("name");
    String pass = request.getParameter("pass");

        if(name.equals("global") && pass.equals("global")){

            response.sendRedirect("globalhome.jsp");
        }
%>
```

Userlogin.jsp

```
<%@page import="java.sql.ResultSet"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.Connection"%>

<%@page import="action.Database"%>

<%

    String uname = null;

    String name = request.getParameter("name");

    String pass = request.getParameter("pass");
    Connection con = Database.getConnection();

    Statement st = con.createStatement();

    ResultSet rs = st.executeQuery("select * from user where name='" + name + "'");

    if (rs.next()) {

        uname = rs.getString("name");

        if (rs.getString("name").equals(name) && (rs.getString("pass").equals(pass))) {

            session.setAttribute("n1", uname);

            System.out.println(uname);

            session.setAttribute("v", name);

            System.out.println(name);

            System.out.println("Success");

            response.sendRedirect("userhome.jsp?msg=Login Successfully");

        } else {

            System.out.println("Failed");

            response.sendRedirect("userlogin.jsp?msgg=Incorrect Username or Password");

        }

    } else {

        System.out.println("Not Enter");

        response.sendRedirect("userlogin.jsp?err=User does not exist");

    }

%>
```

1.4.7 Project 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.

1.4.7.1 Types of Testing:

- Unit Testing
- Integration Testing
- Functional Testing
- System Testing
- White box Testing
- Black box Testing
- Acceptance Testing

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.

Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program.

Functional Testing

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

System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results.

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.

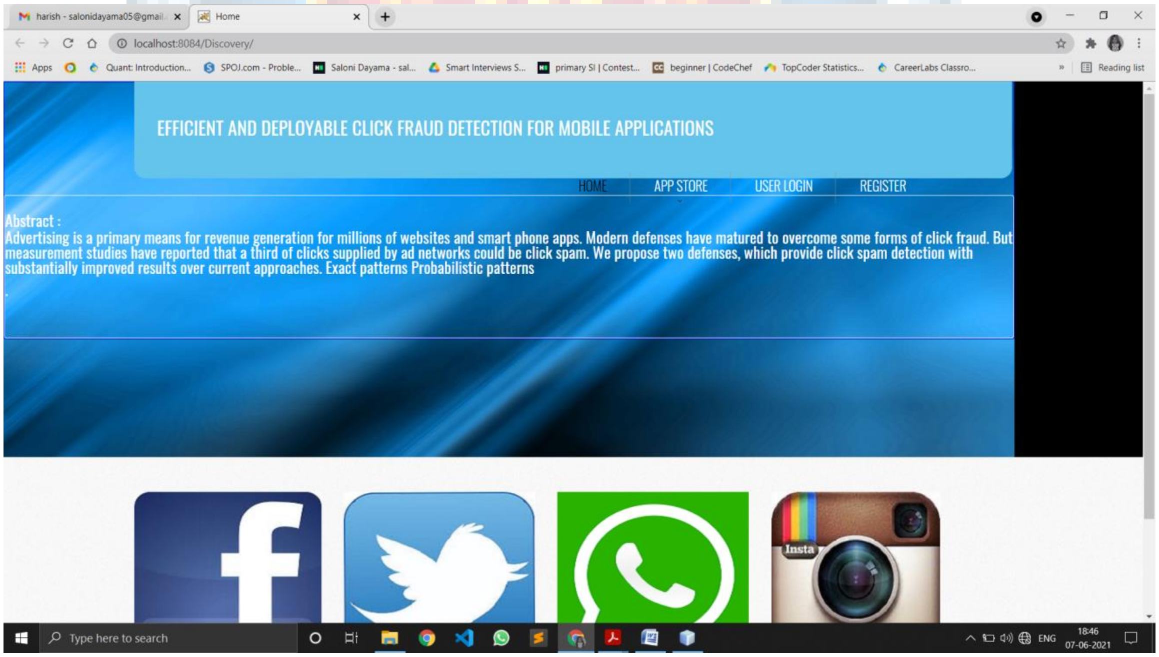
Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested.

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.

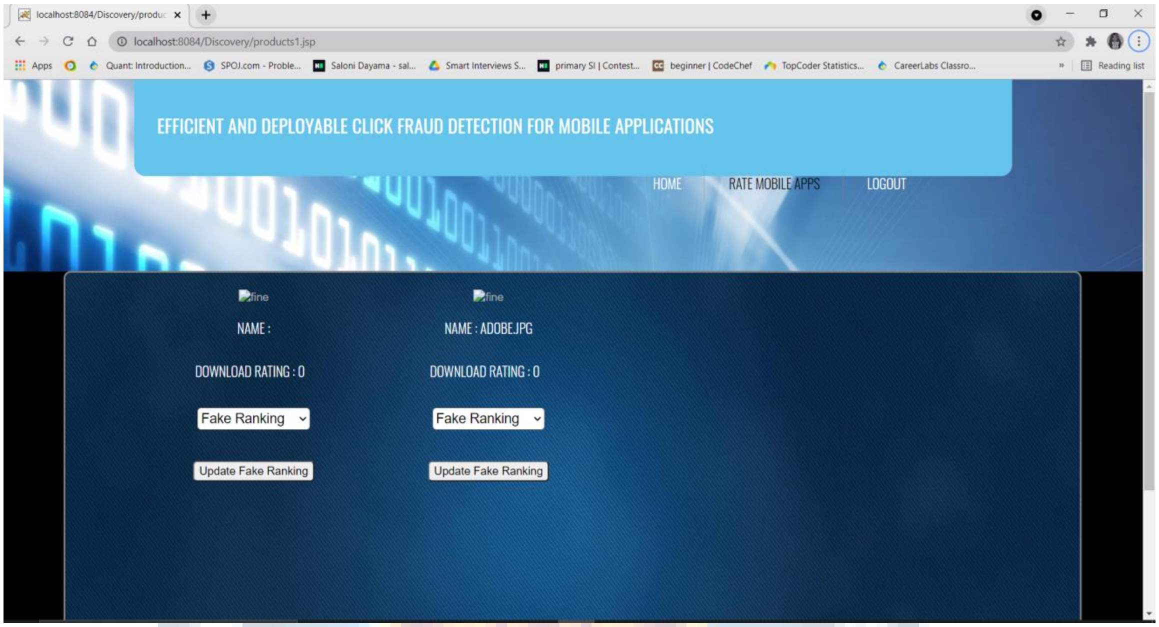
1.4.8 Output Screens



EFFICIENT AND DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS

Abstract :
Advertising is a primary means for revenue generation for millions of websites and smart phone apps. Modern defenses have matured to overcome some forms of click fraud. But measurement studies have reported that a third of clicks supplied by ad networks could be click spam. We propose two defenses, which provide click spam detection with substantially improved results over current approaches. Exact patterns Probabilistic patterns

Fig 1.4.8.1 Home Page



EFFICIENT AND DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS

NAME :

DOWNLOAD RATING : 0

Fake Ranking

Update Fake Ranking

NAME : ADOBE.JPG

DOWNLOAD RATING : 0

Fake Ranking

Update Fake Ranking

Fig 1.4.8.2 Local Anomaly

1.4.9 Conclusion:

we proposed machine learning NLP techniques for classification of restaurant reviews.We removed stop words, applied stemming and applied vectorization technique. We achieved an accuracy of 73% with count vectorization method and naive bayes classification model.

2. LITERATURE SURVEY

Research on ad frauds has been extensively carried in the realm of web applications. The relevant literature mostly focuses on click fraud which generally consists of leveraging a single computer or botnets to drive fake or undesirable impressions and clicks. A number of research studies have extensively characterized click frauds [1, 8, 46] and analysed its profit model [43]. Approaches have also been proposed to detect click frauds by analysing network traffic [44, 45] or by mining search engine's query logs [63].

Nevertheless, despite the specificities of mobile development and usage models, the literature on in-app ad frauds is rather limited. One example of work is the DECAF [37] approach for detecting placement frauds: these consist in manipulating visual layouts of ad views (also referred to as elements or controls) to trigger undesirable impressions in Windows Phone apps. DECAF explores the UI states (which refer to snapshots of the UI when the app is running) in order to detect ad placement frauds implemented in the form of hidden ads, the stacking of multiple ads per page, etc. MAdFraud [13], on the other hand, targets Android apps to detect in-app click frauds by analysing network traffic.

Unfortunately, while the community still struggles to properly address well-known, and often trivial, cases of ad frauds, deception techniques used by app developers are even getting more sophisticated, as reported recently in news outlets [24, 31]. Indeed, besides the aforementioned click and placement frauds, many apps implement advanced procedures for tricking users into unintentionally clicking ad views while they are interacting with the app UI elements. In this work, we refer to this type of ad frauds as dynamic interaction frauds.

Figure 1 illustrates the case of the app taijiao music1 where an ad view gets unexpectedly popped up on top of the exit button when the user wants to exit the app: this usually leads to an unintentional ad click. Actually, we performed a user study on this app and found that 9 out of 10 users were tricked into clicking the ad view. To the best of our knowledge, such frauds have not yet been explored in the literature of mobile ad frauds, and are thus not addressed by the state-of-the-art detection approaches.

This paper. We perform an exploratory study of a wide range of new ad fraud types in Android apps and propose an automated approach for detecting them in market apps. To that end, we first provide a taxonomy that characterizes a variety of mobile ad frauds including both static placement frauds and dynamic interaction frauds. While detection of the former can be performed via analysing thestatic information of the layout in a single UI state [37], detection of the latter presents several challenges, notably for:

Dynamically exercising ad views in a UI state, achieving scalability, and ensuring good coverage in transitions between UI states: A UI state is a running page that contains several visual views/elements, also referred to as controls in Android documentation. Because dynamic interaction frauds involve sequences of UI states, a detection scheme must consider the transition between UI states, as well as background resource consumption such as network traffic. For example, in order to detect the ad fraud case presented in Figure 1, one needs to analyse both current and next UI states to identify any ad view that is placed on top of buttons and which could thus entice users to click on ads unexpectedly. Exercising apps to uncover such behaviours can however be timeconsuming: previous work has shown that it takes several hours to traverse the majority UI states of an app based on existing Android automation frameworks [33].

Automatically distinguishing ad views among other views: In contrast with UI on the Windows Phone platform targeted by the state-of-the-art (e.g., DECAF [37]), Android UI models are generic and thus it is challenging to identify ad views in a given UI state since no explicit labels are provided to distinguish them from other views (e.g., text views). During app development, a view can be added to the Activity, which represents a UI state implementation in Android, by either specifying it in the XML layout [18] or embedding it in the source code. In preliminary investigations, we found that most ad views are actually directly embedded in the code, thus preventing any identification via straightforward XML analysis.

Towards building an approach that achieves accuracy and scalability in Android ad fraud detection, we propose two key techniques aimed at addressing the aforementioned challenges:

Transition graph-based UI exploration. This technique builds a UI transition graph by simulating interaction events associated with user manipulation. We first capture the relationship between UI states through building the transition graphs between them, then identify ad views based on call stack traces and unique features gathered through comparing the ad views and other views in UI states. The scalability of this step is boosted by our proposed ad-first exploration strategy, which leverages probability distributions of the presence of an ad view in a UI state.

Heuristics-supported ad fraud detection. By manually investigating various real-world cases of ad frauds, we devise heuristicrules from the observed characteristics of fraudulent behaviour. Runtime analysis focusing on various behavioural aspects such as view size, bounds, displayed strings or network traffic, is then mapped against the rules to detect ad frauds.

3. SOFTWARE AND HARDWARE REQUIREMENTS

3.1 Software Requirements

Operating system	:	Windows 7 Ultimate.
Coding Language	:	Python.
Front-End	:	Python.
Designing	:	Html,css,javascript.
Data Base	:	MySQL.

3.1.1 Java

Java technology is both a programming language and a platform.

The Java Programming Language

The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called *Java byte codes* —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.

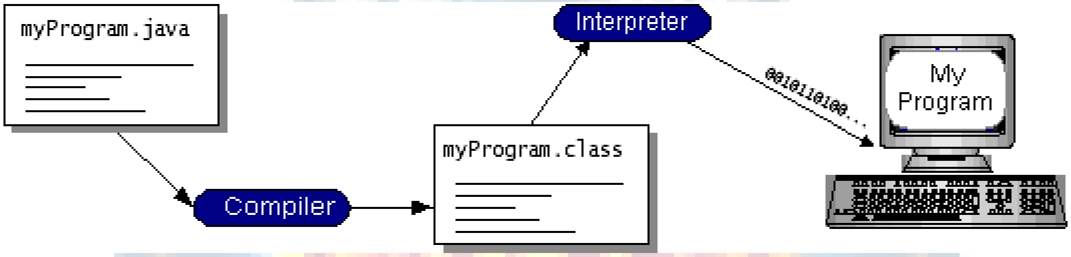
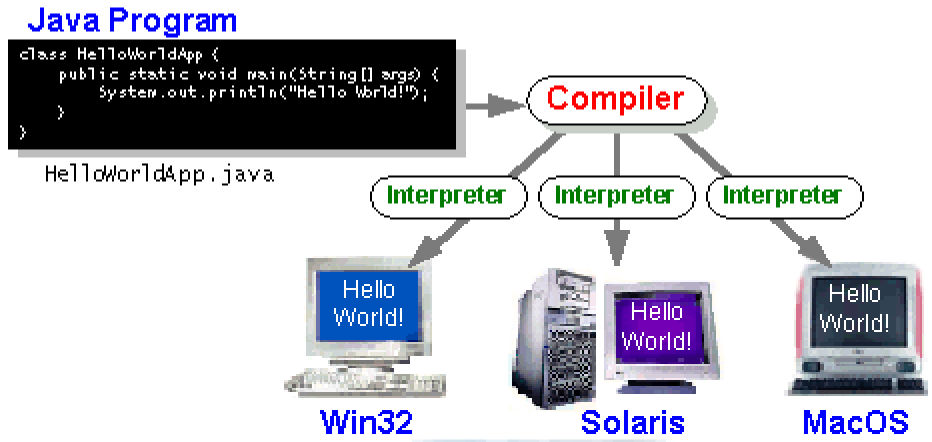


Fig 3.1.1.1 Java Compilation

You can think of Java byte codes as the machine code instructions for the *Java Virtual Machine* (Java VM). Every Java interpreter, whether it’s a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make “write once, run anywhere” possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.



```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

Fig 3.1.1.2 Java Platform

The Java Platform

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

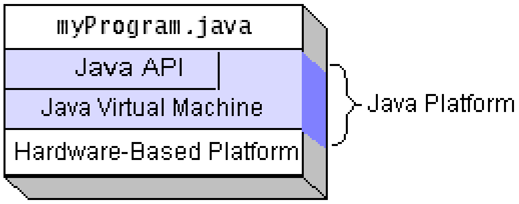
The Java platform has two components:

- The *Java Virtual Machine* (Java VM)
- The *Java Application Programming Interface* (Java API)

You've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as *packages*. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

The following figure depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.



Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

What Can Java Technology Do?

The most common types of programs written in the Java programming language are *applets* and *applications*. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a *server* serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a *servlet*. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

- The essentials: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.

- Applets: The set of conventions used by applets.
- Networking: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
- Internationalization: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
- Security: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
- Software components: Known as JavaBeans^{TM}, can plug into existing component architectures.
- Object serialization: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
- Java Database Connectivity (JDBC^{TM}): Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.

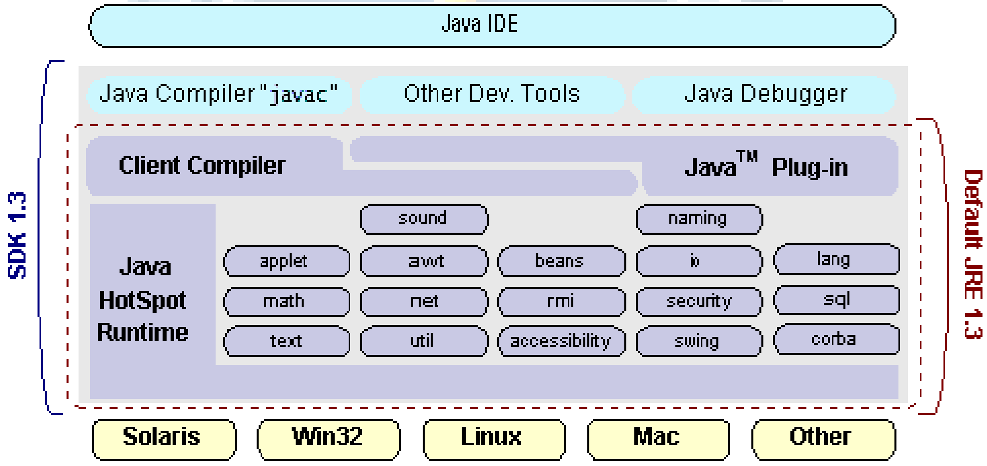


Fig 3.1.1.3 Java SDK

How Will Java Technology Change My Life?

We can’t promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

- Get started quickly: Although the Java programming language is a powerful object-oriented language, it's easy to learn, especially for programmers already familiar with C or C++.
- Write less code: Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.
- Write better code: The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extendible API let you reuse other people's tested code and introduce fewer bugs.
- Develop programs more quickly: Your development time may be as much as twice as fast versus writing the same program in C++. Why? You write fewer lines of code and it is a simpler programming language than C++.
- Avoid platform dependencies with 100% Pure Java: You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure Java^{TM} Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.
- Write once, run anywhere: Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.
- Distribute software more easily: You can upgrade applets easily from a central server. Applets take advantage of the feature of allowing new classes to be loaded "on the fly," without recompiling the entire program.

ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto* standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made the choice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needs suddenly change.

Through the ODBC Administrator in Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN.

The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you setup a separate database application, such as SQL Server Client or Visual Basic 4.0. When the ODBC icon is installed in Control Panel, it uses a file called ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand-alone program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and each maintains a separate list of ODBC data sources.

From a programming perspective, the beauty of ODBC is that the application can be written to use the same set of function calls to interface with any data source, regardless of the database vendor. The source code of the application doesn’t change whether it talks to Oracle or SQL Server. We only mention these two as an example. There are ODBC drivers available for several dozen popular database systems. Even Excel spreadsheets and plain text files can be turned into data sources. The operating system uses the Registry information written by ODBC Administrator to determine which low-level ODBC drivers are needed to talk to the data source (such as the interface to Oracle or SQL Server). The loading of the ODBC drivers is transparent to the ODBC application program. In a client/server environment, the ODBC API even handles many of the network issues for the application programmer.

The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn’t as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow. Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of "plug-in" database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC's framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after.

The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. ***SQL Level API***

The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to "generate" JDBC code and to hide many of JDBC's complexities from the end user.

2. ***SQL Conformance***

SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard functionality in a manner that is suitable for its users.

3. *JDBC must be implemental on top of common database interfaces*

   The JDBC SQL API must "sit" on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

4. *Provide a Java interface that is consistent with the rest of the Java system*

Because of Java's acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

5. *Keep it simple*

This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

6. *Use strong, static typing wherever possible*

Strong typing allows for more error checking to be done at compile time; also, less error appear at runtime.

7. *Keep the common cases simple*

Because more often than not, the usual SQL calls used by the programmer are simple

SELECT’s, INSERT’s, DELETE’s and UPDATE’s, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible.

Finally we decided to proceed the implementation using Java Networking.

And for dynamically updating the cache table we go for MS Access database.

Java ha two things: a programming language and a platform.

Java is a high-level programming language that is all of the following

Simple
Object-oriented
Distributed
Interpreted
Robust
Secure
Architecture-neutral
Portable
High-performance
multithreaded
Dynamic

Java is also unusual in that each Java program is both compiled and interpreted. With a compile you translate a Java program into an intermediate language called Java byte codes the platform-independent code instruction is passed and run on the computer.

Compilation happens just once; interpretation occurs each time the program is executed. The figure illustrates how this works.

3.2 Hardware Requirements

- Processer : Intel i3 or Higher
- Ram : Min 4 GB
- Hard Disk : Min 100 GB

4. SOFTWARE DEVELOPMENT ANALYSIS

4.1 Overview of Problem:

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

4.1.1 Existing System

Since existing machine-learning algorithms used by server-side approaches are not suitable for the client side. Second, the click fraud detection should be able to execute under practical user scenarios, instead of a controlled environment dedicated to fraud detection. In MAdFraud [5], a controlled environment (i.e., only one app is running and the HTTP requests are collected for offline analysis) is used to measure the ad fraud behavior of a vast number of apps. However, in our case, the click fraud detection should happen inside the mobile client without outside support, i.e., be deployable in real-world scenarios. In this paper, we propose AdSherlock, an efficient and deployable click fraud detection approach for mobile apps at the client side. Note that as a client-side approach, AdSherlock is orthogonal to existing server-side approaches. AdSherlock is designed to be used by app stores to ensure a healthy mobile app ecosystem. AdSherlock's high accuracy helps market operators to fight both in-app frauds and bots-driven frauds. Note that, AdSherlock can also be used by any third parties to detect in-app frauds. For example, ad providers can employ AdSherlock to check whether apps embedding their libraries have in-app fraudulent behaviors..

4.1.2 Disadvantages of Existing System

- This type of classification is only done when the classifier has to work on the binary data which is not the case with Restaurant Reviews.

- However, from a practical point of view perhaps the most serious problem with SVMs is the high algorithmic complexity and extensive memory requirements of the required quadratic programming in large-scale tasks.
- If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as “Zero Frequency”.

4.2 Define the Problem :

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

4.2.1 Proposed System

In order to solve the above problem, we propose two pattern classes: exact patterns and probabilistic patterns. Both of them are built from invariant substrings in the HTTP header. We refer to these substrings as tokens. Exact patterns consist of a set of sequential tokens and match an HTTP request if and only if the request contains all tokens in the set with the same ordering. Probabilistic patterns consist of a set of tokens, each of which is associated with an ad score, and a non-ad score. We describe the details of pattern generation in the following sections.

4.2.2 Advantages of Proposed System

- Good at pattern recognition problems
- Data-driven, and performance is high in many problems
- End-to-End training: little or no domain knowledge is needed in system construction
- Learn of representations: cross-modal processing is possible
- Gradient-based learning: learning algorithm is simple
- Mainly supervised learning methods

4.3 Modules Used:

Dynamically exercising ad views in a UI state

Achieving scalability, and ensuring good coverage in transitions between UI states: A UI state is a running page that contains several visual views/elements, also referred to as controls in Android documentation. Because dynamic interaction frauds involve sequences of UI states, a detection scheme must consider the transition between UI states, as well as background resource consumption such as network traffic. For example, in order to detect the ad fraud case presented in Figure 1, one needs to analyse both current and next UI states to identify any ad view that is placed on top of buttons and which could thus entice users to click on ads unexpectedly. Exercising apps to uncover such behaviours can however be timeconsuming: previous work has shown that it takes several hours to traverse the majority UI states of an app based on existing Android automation frameworks [33].

Automatically distinguishing ad views among other views

In contrast with UI on the Windows Phone platform targeted by the state-of-the-art (e.g., DECAF [37]), Android UI models are generic and thus it is challenging to identify ad views in a given UI state since no explicit labels are provided to distinguish them from other views (e.g., text views). During app development, a view can be added to the Activity, which represents a UI state implementation in Android, by either specifying it in the XML layout [18] or embedding it in the source code. In preliminary investigations, we found that most ad views are actually directly embedded in the code, thus preventing any identification via straightforward XML analysis

Ad Frauds

While the literature contains a large body of work on placement frauds in web applications and the Windows Phone platform, very little attention has been paid to such frauds on Android. Furthermore, dynamic interaction frauds have even not been explored to the best of our knowledge.

To build the taxonomy of Android ad frauds, we investigate in this work: (1) the usage policies provided by popular ad libraries [22, 26], (2) the developer policies provided by official Google Play market [49] and popular third-party app markets, including Wandoujia(Alibaba App) Market [60], Huawei App Market [41] and Tencent Myapp Market [42]. (3) the guidelines on ad behaviour drafted by a communication standards association [6], and (4) some real-world ad fraud cases. Figure 3 presents our taxonomy, which summarizes 9 different types of ad frauds, which represents by far the largest number of ad fraud types. Particularly, the five types of dynamic interaction frauds have never been investigated in the literature.

FRAUDDROID

To address ad frauds in the Android ecosystem we design and implement FraudDroid, an approach that combines dynamic analysis on UI state as well as network traffic data to identify fraudulent behaviours. Figure 4 illustrates the overall architecture of FraudDroid. The working process unfolds in two steps: (1) analysis and modelling of UI states, and (2) heuristics-based detection of ad frauds. To efficiently search for ad frauds, one possible step before sending apps to FraudDroid is to focus on such apps that have included ad libraries. To this end, FraudDroid integrates a pre-processing step, which stops the analysis if the input app does not leverage any ad libraries, i.e., there will be no ad frauds in that app. Thus we first propose to filter apps that have no permissions associated with the functioning of ad libraries, namely INTERNET and ACCESS_NETWORK_STATE [34].

4.4 System Architechture

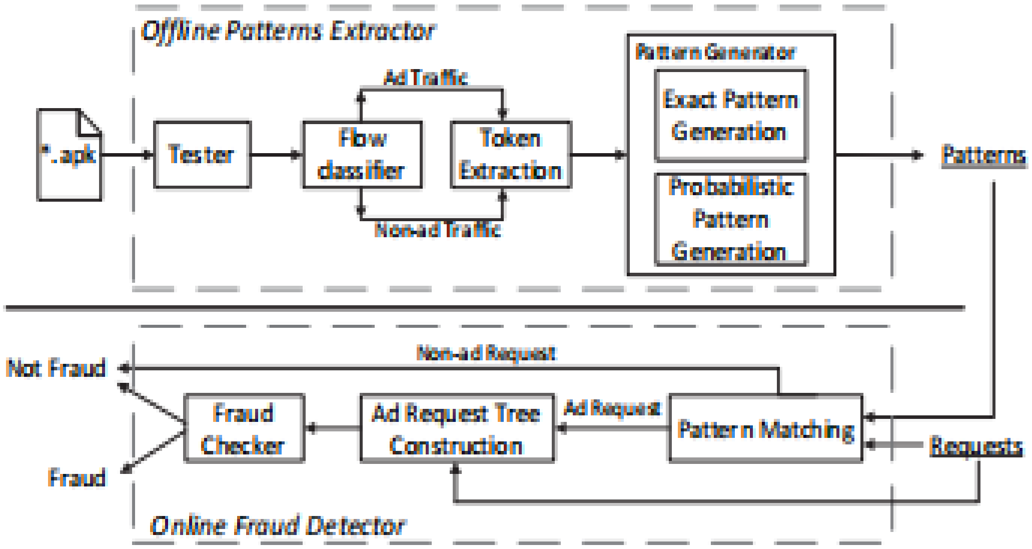


Fig. 1: Overview of AdSherlock.

Fig 4.1.1 Overview of AdSherlock

We propose an inference system, depicted in Figure, which takes two inputs, clicktimestamps from the adnetwork, and an optional seed clickspam input from a bait adfarm. The ad network contains servers which run *click traffic monitors* that store click timestamps.Optionally, to supplement information for click traffic monitors, ourinference system may also receive input from a bait ad farm input to m Clicktok, where the suspicion of click fraud is known with a higher probability.

Our primary focus is to design a generic inference algorithm that is, first, based on the fundamental limitations of automated fake click generation techniques, and second, can address both organic and non-organicclickspam.

Clicktok works on the core observation that both organic and in-organic clickspam cause an increase in redundancy, albeit differently within ad network clickstreams. In the case for organic click fraud,to isolate the source of redundancy, we use a compression functionin combination with a clustering algorithm, to isolate click traffic whose *timing* *patterns* are similar to past timing patterns. A timing pattern is an ordered ascending sequence of time offsets, relative to an absolute start time.

Similarly, we noticed that the same intuition can be leveraged to isolate inorganic click spam. For instance, where malware generates traffic using randomised generators, the traffic with high entropy timing patterns can be clustered together, by exploiting their non-compressibility. Likewise, the injection of small amounts of clickspam per device are evident, when traffic from multiple end-user devices is considered together, thus exploiting the common patterns across infected devices. Adnetworks, or backbone routers, provide us with a advantage point into clicktraffic tainted with clickspam.

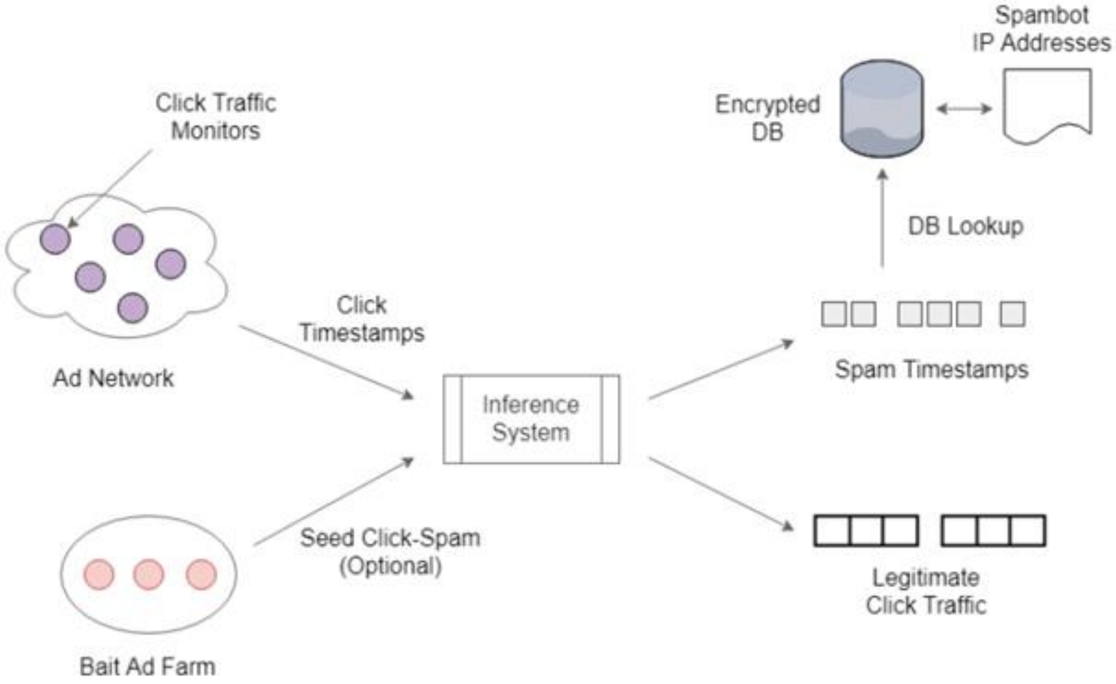


Fig 4.1.2 System Architecture

5. PROJECT SYSTEM DESIGN

5.1 UML Diagrams:

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.

5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

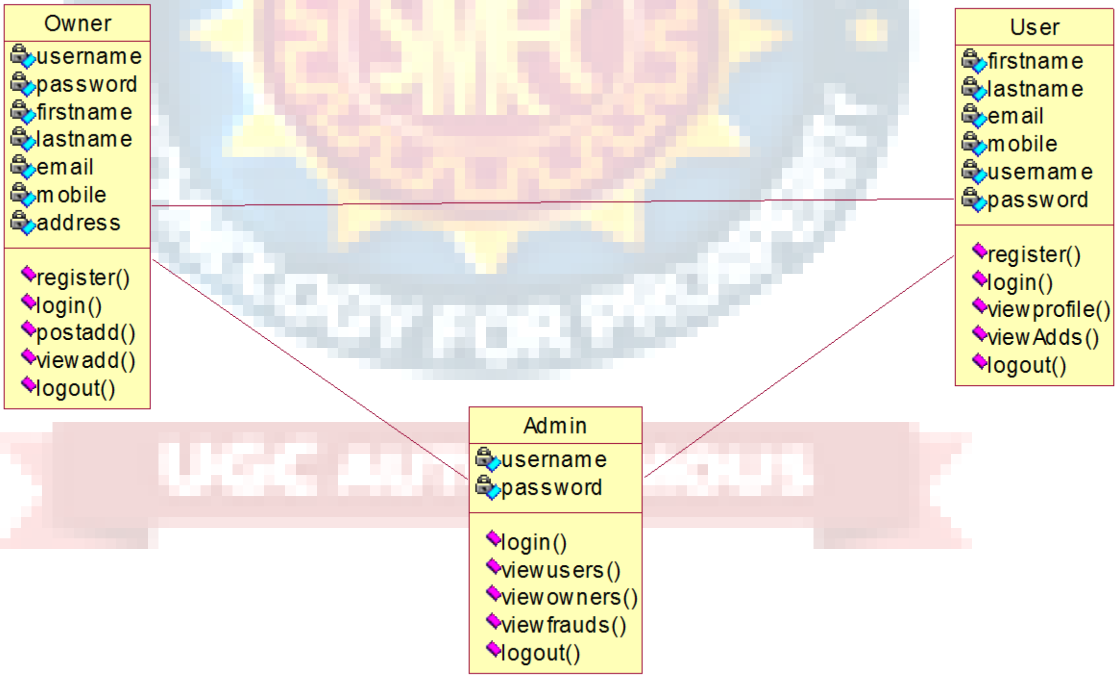


Fig 5.1.1 Class Diagram

USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

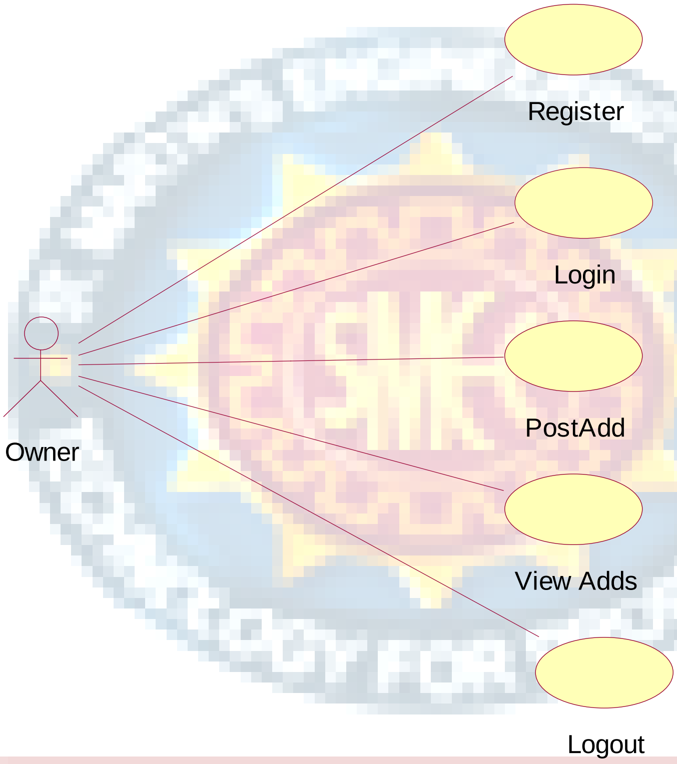


Fig 5.1.2.1 Owner Usecase Diagram

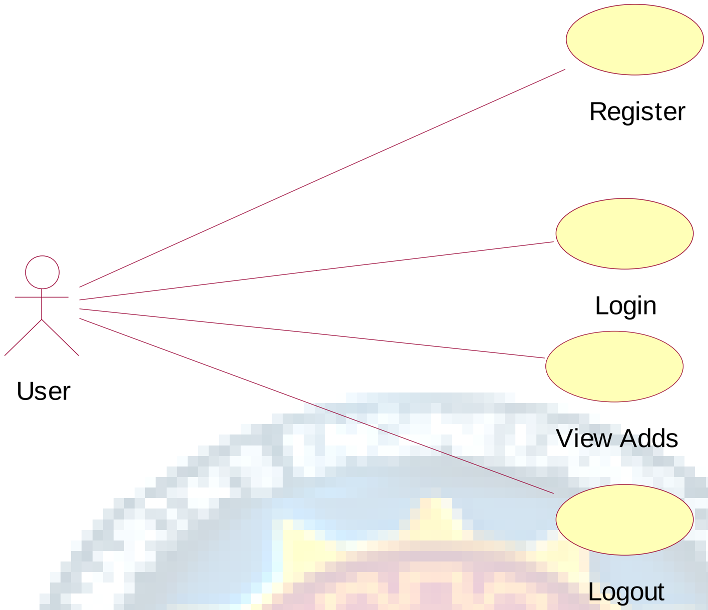


Fig 5.1.2.2 User Usecase Diagram

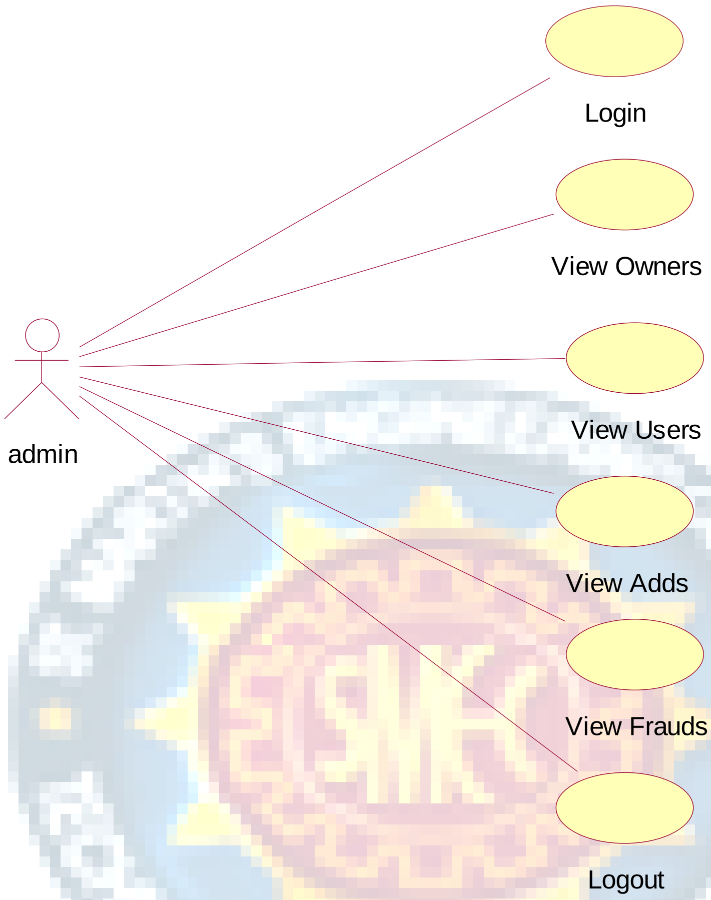


Fig 5.1.2.3 Admin Usecase Diagram

SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

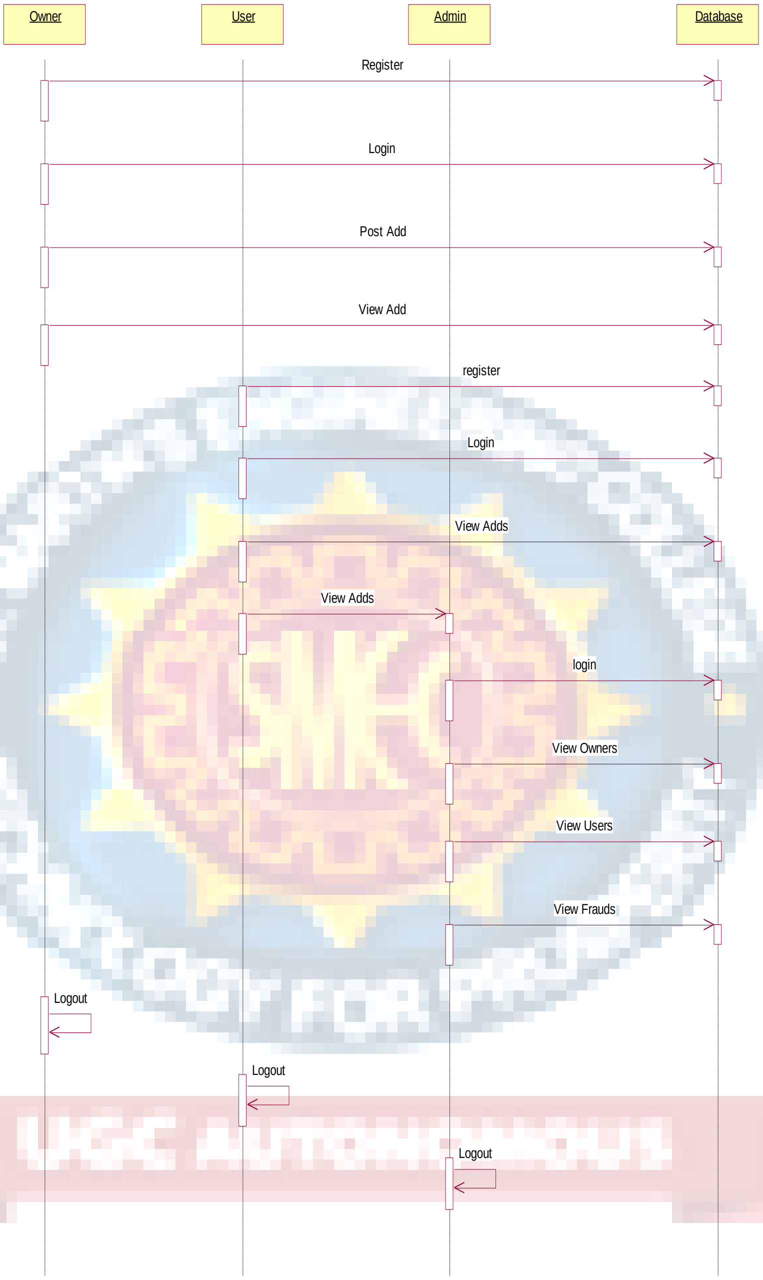


Fig 5.1.3 Sequence Diagram

Collaboration diagram

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling

Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

A Communication diagram models the interactions between objects or parts in terms of sequenced messages. Communication diagrams represent a combination of information taken from Class, Sequence, and Use Case Diagrams describing both the static structure and dynamic behavior of a system.

However, communication diagrams use the free-form arrangement of objects and links as used in Object diagrams. In order to maintain the ordering of messages in such a free-form diagram, messages are labeled with a chronological number and placed near the link the message is sent over. Reading a communication diagram involves starting at message 1.0, and following the messages from object to object.

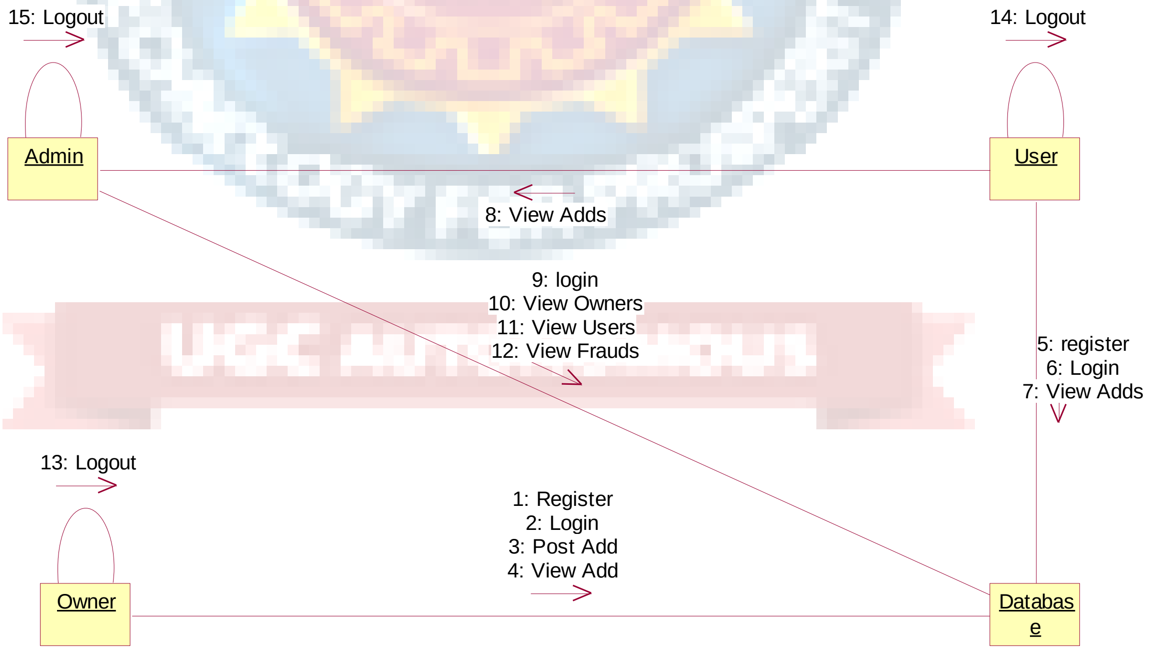


Fig 5.1.4 Collaboration Diagram

Deployment diagram

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.[1] To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have subnodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

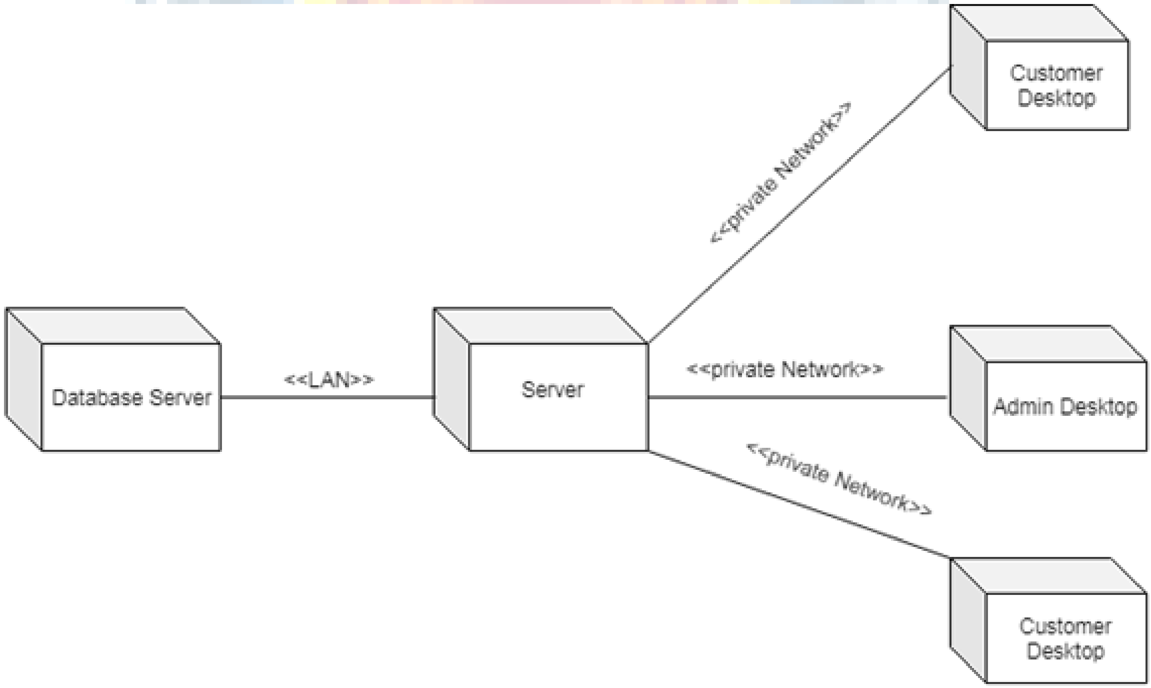


Fig 5.1.5 Deployment Diagram

Package Diagram

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.

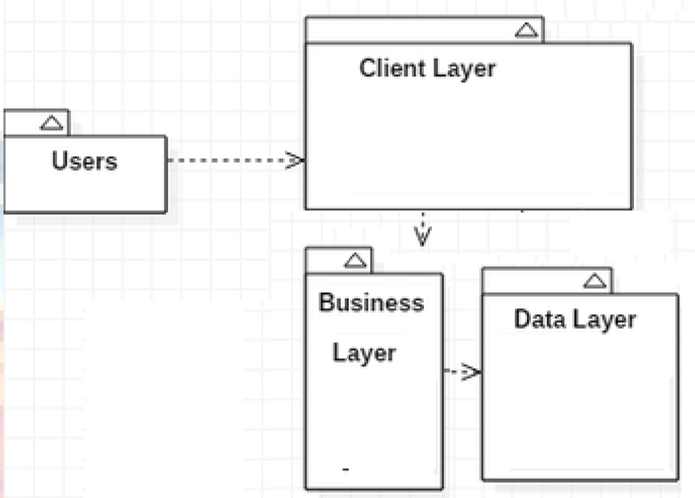


Fig 5.1.6 Package Diagram

Profile Diagram

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connectors and their components.

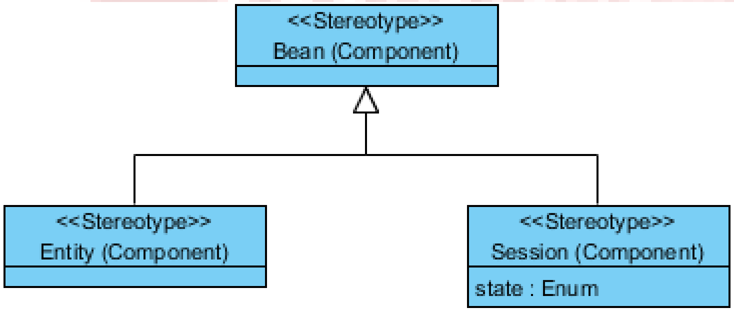


Fig 5.1.7 Profile Diagram

6. PROJECT CODING

6.1 Code Templates

Compare.jsp

```
<%@page import="java.io.OutputStream"%>
<%@page import="java.io.FileOutputStream"%>
<%@page import="java.io.File"%>
<%@page import="action.Database"%>
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%@page import="java.io.InputStream"%>
<!DOCTYPE html>
<html lang="en">
<head>
<!--<title>Products</title>-->
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta name="description" content="Your description">
<meta name="keywords" content="Your keywords">
<meta name="author" content="Your name">
<!--CSS-->
<link rel="stylesheet" href="css/bootstrap.css" >
<link rel="stylesheet" href="css/style.css">
<!--JS-->
<script src="js/jquery.js"></script>
<script src="js/jquery-migrate-1.2.1.js"></script>
<script src="js/superfish.js"></script>
<script src="js/jquery.easing.1.3.js"></script>
```

```
<script src="js/jquery.mobilemenu.js"></script>
<script src="js/jquery.cookie.js"></script>
<script src="js/jquery.equalheights.js"></script>
<script src="js/jquery.ui.totop.js"></script>
</head>
<body>
<!--header-->
<section class="header indent">
  <div class="container">
    <header>
      <h1 class="navbar-brand navbar-brand_ logo"><a href="#">EFFICIENT AND
DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE
APPLICATIONS</a></h1>
      <nav class="navbar navbar-default navbar-static-top my_navbar clearfix"
role="navigation">
        <ul class="nav sf-menu clearfix">
          <li><a href="userhome.jsp">Home</a><em></em></li>
         <li><a href="products.jsp">Mobile Apps</a><em></em></li>
         <li class="active"><a href="compare.jsp">Global
Comparison</a><em></em></li>
         <li><a href="userlogin.jsp">Logout</a></li>
        </ul>
      </nav>
      <em></em></header>
  </div>
</section>
<div class="global">
  <div class="picBox">
    <figure><img src="img/picture1.jpg" alt=""></figure>
  </div>
  <!--content-->
```

```
    <div class="container">
        <div class="row">
            <section class="col-lg-8 col-md-8 col-sm-8">
                <div class="row">
<!--                    <div style="border: solid 1px;border-color: white;border-radius:
2px;height: 680px;width: 1365px;">-->
    <div style="margin-left: -80px;background-image: url('img/banner7.jpg');margin-top: -
60px;border: solid;border-radius: 10px;border-color: gray;height: 480px;width: 1358px">
        <table   style="margin-left: 20px">
                    <tr>&nbsp;&nbsp;&nbsp;
                        <%
                            InputStream n1 = null;
                            String logo = null;
                            String fname = null;
                            int i = 1;
                            String sname = request.getQueryString();
//                            session.setAttribute("n2", sname);
//                            System.out.println("Search " + sname);
                            Connection con = Database.getConnection();
                            Statement st = con.createStatement();
                            ResultSet rs = st.executeQuery("select * from global where file_name =
'"+sname+"'");
                            while (rs.next()) {
                                fname = rs.getString("file_name");
                                String len = rs.getString("image");
                                int len1 = len.length();
                                byte[] b1 = new byte[len1];
                                n1 = rs.getBinaryStream("image");
                                int index = n1.read(b1, 0, len1);
                                System.out.println("Index is printing here : "+index);
                                String putFile =
request.getRealPath(request.getContextPath()).substring(0,
request.getRealPath(request.getContextPath()).lastIndexOf(File.separator) + 1) + "img" +
File.separator;
                                File file = new File(putFile + rs.getString("file_name"));
                                logo = rs.getString("file_name");
                                if (file.exists()) {
                                    file.delete();
                                } else {
                                    OutputStream out1 = new FileOutputStream(file);
                                    while (index != -1) {
                                        out1.write(b1, 0, index);
                                        index = n1.read(b1, 0, len1);
                                    }
                                    out1.close();
                                    i++;
                                }
                        %>
                        <td  padding="0px" style="border: 0px;">
                    <center>
                        <div style="margin-left: 450px">
                            <img src="img/<%=fname%>.jpg" alt="fine" style="width:
100px;height: 100px;margin-left: 120px"></img><br /><br />
                            <h3 style="margin-left: 110px;color: white">Name :
<%=rs.getString("file_name")%></h3><br>
                            <h3 style="margin-left: 110px;color: white">Size :
<%=rs.getString("size")%> KB</h3><br>
                            <h3 style="margin-left: 110px;color: white">Global Ranking :
<%=rs.getString("userrating")%></h3><br>
                            <h3 style="margin-left: 110px;color: white">Local Ranking :
<%=rs.getString("localrating")%></h3><br>
```

```
            </div>



          </center>


            </td>
            <%
              }
            %>
          </tr>
        </table>
</div>
        </div>
      </section>

    </div>
  </div>

</div>
<!--footer-->
<footer hidden>
  <div class="container">
    <div class="row">
      <article class="col-lg-3 col-md-3 col-sm-3 col-lg-offset-1 col-md-offset-1
privacyBox">
        <p>Company Name &copy; 2015 </p>
                        <p class="swty">Web Design: <a
href="http://www.metamorphozis.com" class="bhfy">Free Website Templates</a></p>
      </article>
    </div>
  </div>
```

```
</footer>
<script src="js/bootstrap.min.js"></script>
<script src="js/scripts.js"></script>
</body>
</html>
```

Global.jsp

```
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%@page import="action.Database"%>
<%
    String name = request.getParameter("name");
    String pass = request.getParameter("pass");

        if(name.equals("global") && pass.equals("global")){

            response.sendRedirect("globalhome.jsp");
        }
%>
```

Local.jsp

```

<!DOCTYPE html>
<html lang="en">

<head>
<!--<title>contacts</title>-->
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta name="description" content="Your description">
<meta name="keywords" content="Your keywords">
```

```
<meta name="author" content="Your name">
<meta name = "format-detection" content = "telephone=no" />
<!--CSS-->
<link rel="stylesheet" href="css/bootstrap.css" >
<link rel="stylesheet" href="css/style.css">
<!--JS-->
<script src="js/jquery.js"></script>
<script src="js/jquery-migrate-1.2.1.js"></script>
<script src="js/superfish.js"></script>
<script src="js/jquery.easing.1.3.js"></script>
<script src="js/jquery.mobilemenu.js"></script>
<script src="js/jquery.cookie.js"></script>
<script src="js/forms.js"></script>
<script src="js/jquery.ui.totop.js"></script>
<script src="js/jquery.equalheights.js"></script>

<%
   String user = (String)session.getAttribute("v");
%>
</head>
<body>
<!--header-->
<section class="header indent">
   <div class="container">
      <header>
         <h1 class="navbar-brand navbar-brand_ logo"><a href="#">EFFICIENT AND
DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE
APPLICATIONS</a></h1>
         <nav class="navbar navbar-default navbar-static-top my_navbar clearfix"
role="navigation">
            <ul class="nav sf-menu clearfix">
```

```
                  <li class="active"><a href="local_home.jsp">Home</a><em></em></li>
                 <li><a href="products1.jsp">Rate Mobile Apps</a><em></em></li>
                 <li><a href="locallogin.jsp">Logout</a><em></em></li>
               </ul>
            </nav>
            <em></em>
        </header>
    </div>

</section>
<div class="global">
    <div style="background-image: url('img/picture1.jpg');height: 680px;width: 1365px">
<!--        <figure><img src="img/picture1.jpg" alt=""></figure>-->
<div style="border: solid 1px;border-color: white;border-radius: 2px;height:
680px;width: 1365px;">
    <div style="margin-left: 50px;background-image: url('img/bannerlcl.jpg');margin-top:
190px;border: solid;border-radius: 15px;border-color: gray;height: 380px;width:
1250px">
        <center><br><br>
            <h1 style="color: white">Local Home</p></h1><br>

        </center>
    </div>
</div>
    </div>
    <!--content-->
</div>
<!--footer-->
<footer hidden>
    <div class="container">
        <div class="row">
```

```
            <article class="col-lg-3 col-md-3 col-sm-3 col-lg-offset-1 col-md-offset-1
privacyBox">
                <p>Company Name &copy; 2015 </p>
                                                <p class="swty">Web Design: <a
href="http://www.metamorphozis.com" class="bhfy">Free Website Templates</a></p>
            </article>
        </div>
    </div>
</footer>
<script src="js/bootstrap.min.js"></script>
<script src="js/scripts.js"></script>
</body>
</html>
```

Userlogin.jsp

```
<%@page import="java.sql.ResultSet"%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.Connection"%>

<%@page import="action.Database"%>

<%

    String uname = null;

    String name = request.getParameter("name");

    String pass = request.getParameter("pass");

    Connection con = Database.getConnection();

    Statement st = con.createStatement();

    ResultSet rs = st.executeQuery("select * from user where name='" + name + "'");
```

```
  if (rs.next()) {
    uname = rs.getString("name");
    if (rs.getString("name").equals(name) && (rs.getString("pass").equals(pass))) {
      session.setAttribute("n1", uname);
      System.out.println(uname);
      session.setAttribute("v", name);
      System.out.println(name);
      System.out.println("Success");
      response.sendRedirect("userhome.jsp?msg=Login Successfully");
    } else {
      System.out.println("Failed");
      response.sendRedirect("userlogin.jsp?msgg=Incorrect Username or Password");
    }
  } else {
    System.out.println("Not Enter");
    response.sendRedirect("userlogin.jsp?err=User does not exist");
  }
%>
```

7. PROJECT 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.

7.1 Various Test Cases

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output, often with the help of a test oracle or a previous result that is known to be good, without any knowledge of the test object's internal structure.

7.2 Block Box Testing

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing. The main focus of black box testing is on the validation of your functional requirements. Black box testing gives abstraction from code and focuses on testing effort on the software system behavior. Black box testing facilitates testing communication amongst modules. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation. There are various types of test. Each test type addresses a specific testing requirement.

Input
Black Box
Output

Fig 7.2.1 Black Box Structure

The above Black-Box can be any software system you want to test. For Example, an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation.

Here are the generic steps followed to carry out any type of Black Box Testing.

- Initially, the requirements and specifications of the system are examined.
- Tester chooses valid inputs (positive test scenario) to check whether SUT processes them correctly. Also, some invalid inputs (negative test scenario) are chosen to verify that the SUT is able to detect them.
- Tester determines expected outputs for all those inputs.
- Software tester constructs test cases with the selected inputs.
- The test cases are executed.
- Software tester compares the actual outputs with the expected outputs.
- Defects if any are fixed and re-tested.

7.2.1 Types of Black Box Testing

There are many types of Black Box Testing but the following are the prominent ones -

- **Functional testing** - This black box testing type is related to the functional requirements of a system; it is done by software testers.
- **Non-functional testing** - This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.

- **Regression testing** - Regression Testing is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

7.2.2 Tools used for Black Box Testing:

Tools used for Black box testing largely depends on the type of black box testing you are doing.

- For Functional/ Regression Tests you can use - QTP, Selenium.
- For Non-Functional Tests, you can use - LoadRunner, Jmeter.

7.2.3 Black Box Testing Techniques

Following are the prominent Test Strategy amongst the many used in Black box Testing

- **Equivalence Class Testing:** It is used to minimize the number of possible test cases to an optimum level while maintains reasonable test coverage.
- **Boundary Value Testing:** Boundary value testing is focused on the values at boundaries. This technique determines whether a certain range of values are acceptable by the system or not. It is very useful in reducing the number of test cases. It is most suitable for the systems where an input is within certain ranges.
- **Decision Table Testing**: A decision table puts causes and their effects in a matrix. There is a unique combination in each column.

7.3 White Box Testing

White Box Testing is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing and Glass box testing.

It is one of two parts of the Box Testing approach to software testing. Its counterpart, Blackbox testing, involves testing from an external or end-user type perspective. On the other hand, White box testing in software engineering is based on the inner workings of an application and revolves around internal testing.

The term "WhiteBox" was used because of the see-through box concept. The clear box or WhiteBox name symbolizes the ability to see through the software's outer shell (or "box") into its inner workings. Likewise, the "black box" in "Black Box Testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested. White box testing involves the testing of the software code for the following:

- Internal security holes
- Broken or poorly structured paths in the coding processes
- The flow of specific inputs through the code
- Expected output
- The functionality of conditional loops
- Testing of each statement, object, and function on an individual basis

The testing can be done at system, integration and unit levels of software development. One of the basic goals of whitebox testing is to verify a working flow for an application. It involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output, you have encountered a bug.

7.3.1 Steps in White Box Testing:

We have divided it into **two basic steps**. This is what testers do when testing an application using the white box testing technique:

Step 1) Understand the Source Code

The first thing a tester will often do is learn and understand the source code of the application. Since white box testing involves the testing of the inner workings of an application, the tester must be very knowledgeable in the programming languages used in the applications they are testing. Also, the testing person must be highly aware of secure coding practices. Security is often one of the primary objectives of testing software. The tester should be able to find security issues and prevent attacks from hackers and naive users who might inject malicious code into the application either knowingly or unknowingly.

Step 2) Create Test Cases and Execute

The second basic step to white box testing involves testing the application's source code for proper flow and structure. One way is by writing more code to test the application's source code. The tester will develop little tests for each process or series of processes in the application. This method requires that the tester must have intimate knowledge of the code and is often done by the developer. Other methods include Manual Testing, trial, and error testing and the use of testing tools as we will explain further on in this article.

7.3.2 White Box Testing Techniques

A major White box testing technique is Code Coverage analysis. Code Coverage analysis eliminates gaps in a Test Case suite. It identifies areas of a program that are not exercised by a set of test cases. Once gaps are identified, you create test cases to verify untested parts of the code, thereby increasing the quality of the software product. There are automated tools available to perform Code coverage analysis. Below are a few coverage analysis techniques a box tester can use:

Statement Coverage:- This technique requires every possible statement in the code to be tested at least once during the testing process of software engineering.

Branch Coverage - This technique checks every possible path (if-else and other conditional loops) of a software application.

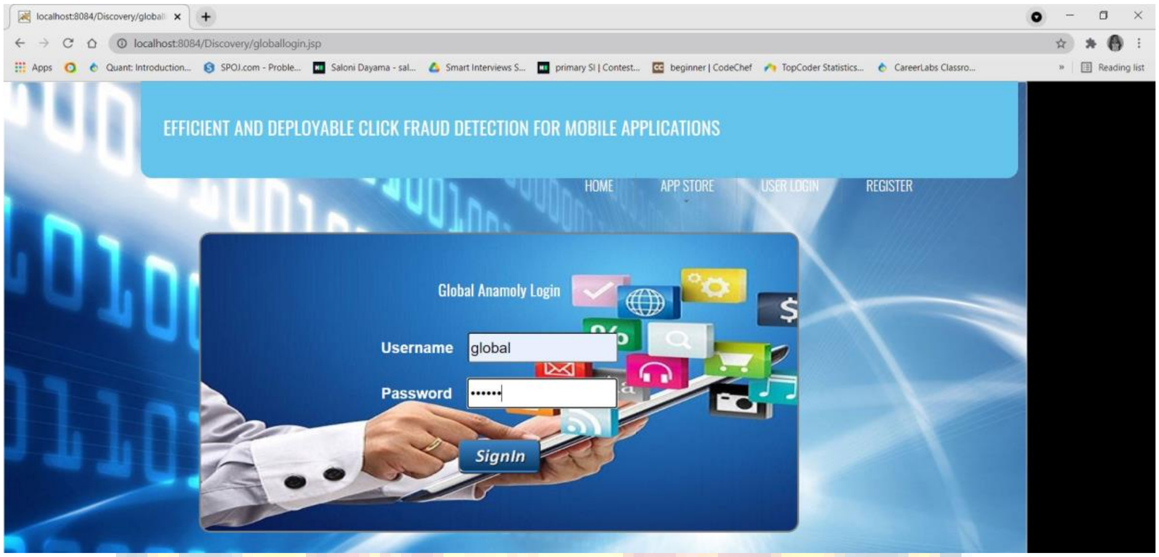
Apart from above, there are numerous coverage types such as Condition Coverage, Multiple Condition Coverage, Path Coverage, Function Coverage etc. Each technique has its own merits and attempts to test (cover) all parts of software code. Using Statement and Branch coverage you generally attain 80-90% code coverage which is sufficient.

Following are important WhiteBox Testing Techniques:

- Statement Coverage
- Decision Coverage
- Branch Coverage
- Condition Coverage
- Multiple Condition Coverage

- Finite State Machine Coverage
- Path Coverage
- Control flow testing
- Data flow testing

8. OUTPUT SCREENS



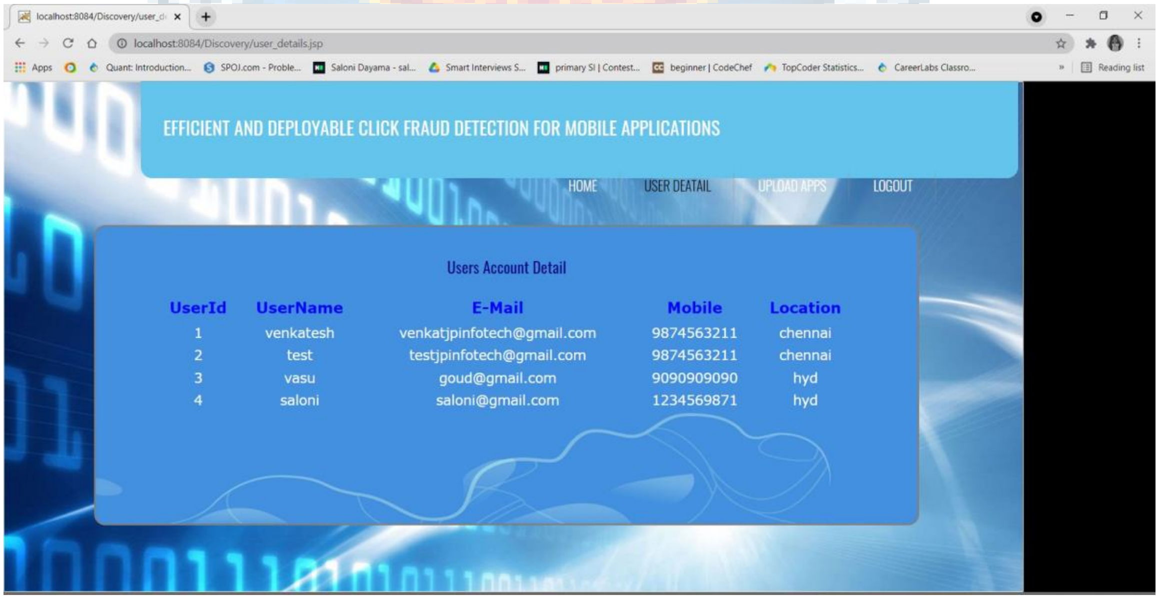
EFFICIENT AND DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS

Global Anamoly Login

Username: global

Password: ••••••

Fig 8.1 Global Anomaly Login



EFFICIENT AND DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS

Users Account Detail

UserId	UserName	E-Mail	Mobile	Location
1	venkatesh	venkatjpinfotech@gmail.com	9874563211	chennai
2	test	testjpinfotech@gmail.com	9874563211	chennai
3	vasu	goud@gmail.com	9090909090	hyd
4	saloni	saloni@gmail.com	1234569871	hyd

Fig 8.2 User Details

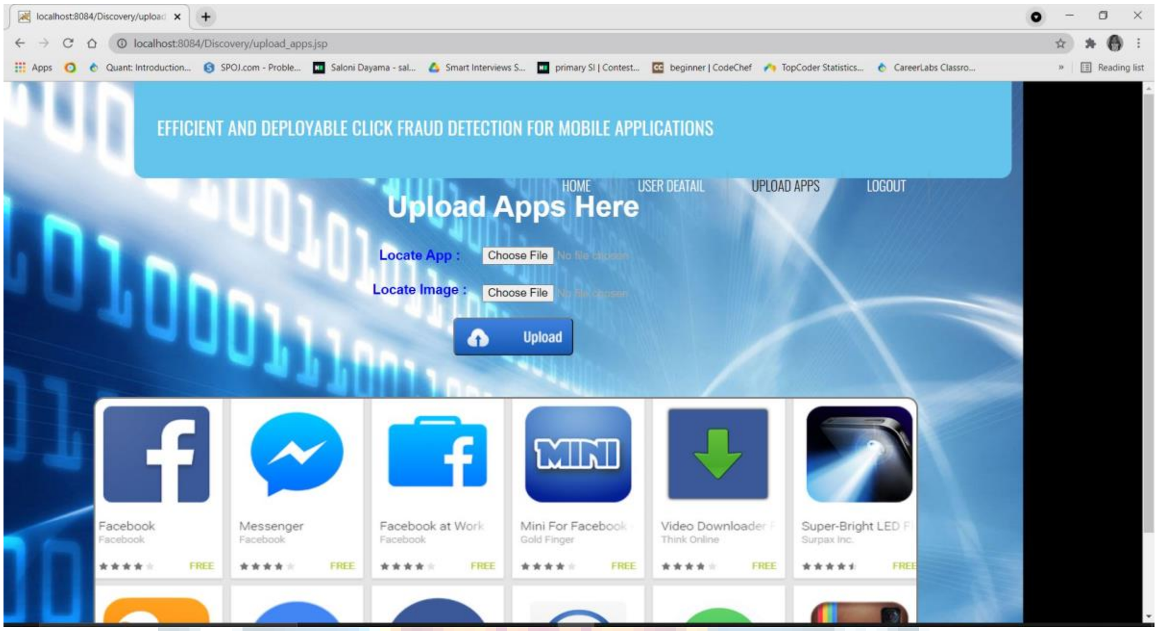
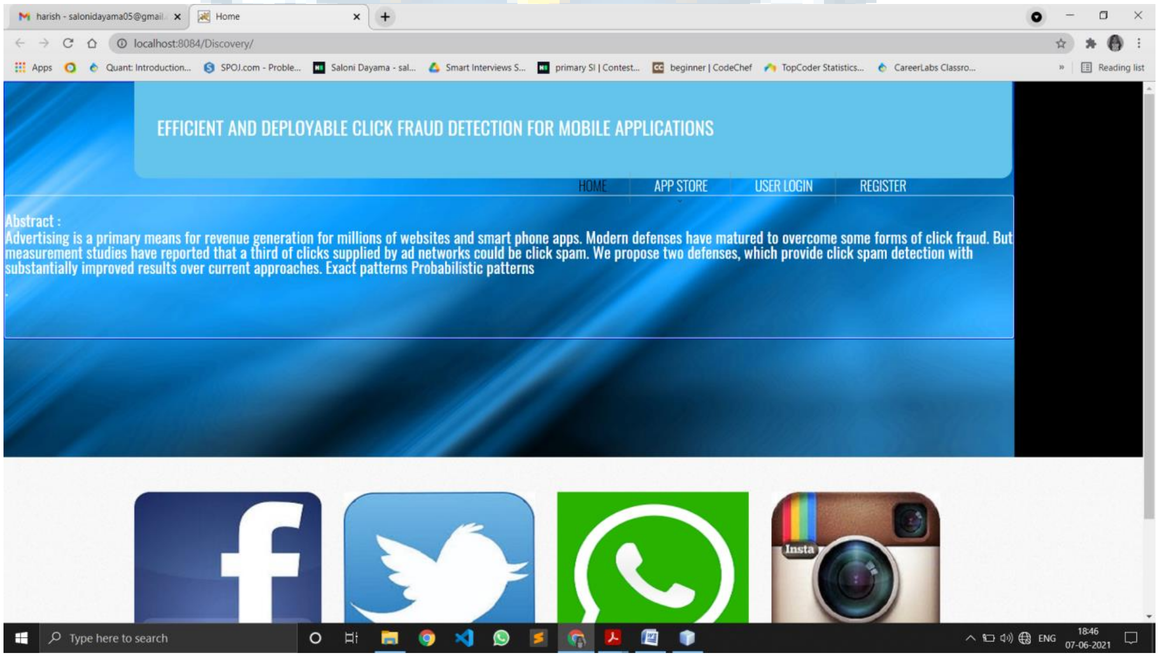


Fig 8.3 Upload Apps


localhost:8084/Discovery/
EFFICIENT AND DEPLOYABLE CLICK FRAUD DETECTION FOR MOBILE APPLICATIONS
HOME
APP STORE
USER LOGIN
REGISTER
Insta

Abstract :
Advertising is a primary means for revenue generation for millions of websites and smart phone apps. Modern defenses have matured to overcome some forms of click fraud. But measurement studies have reported that a third of clicks supplied by ad networks could be click spam. We propose two defenses, which provide click spam detection with substantially improved results over current approaches. Exact patterns Probabilistic patterns

Fig 8.4 Home Page

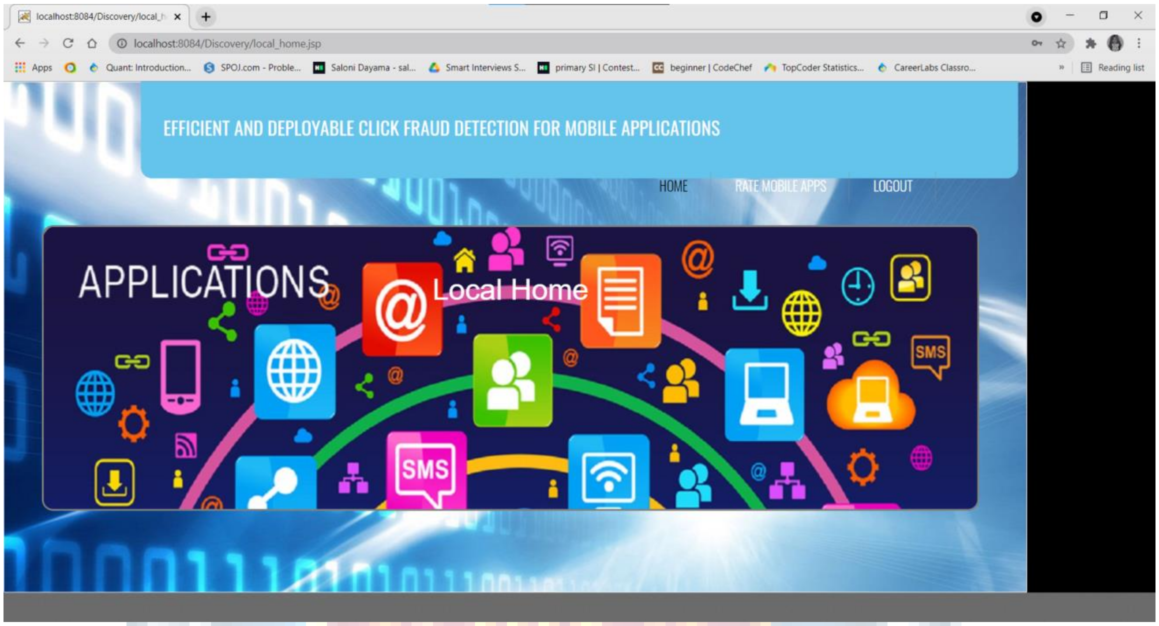
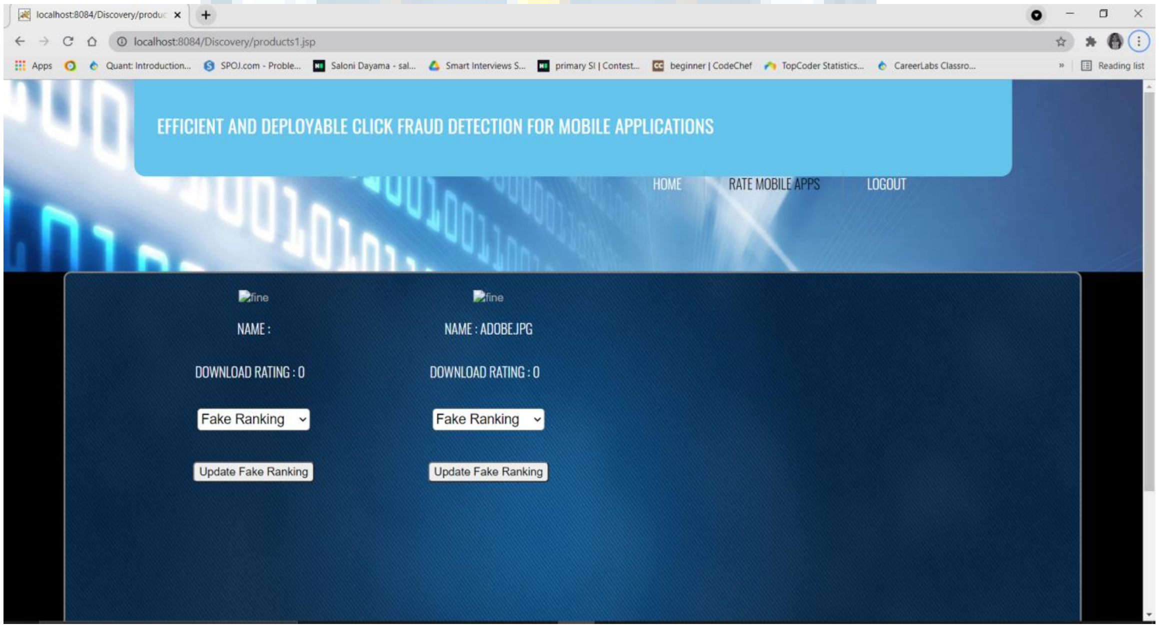


Fig 8.5 Local Home



NAME :

DOWNLOAD RATING : 0

NAME : ADOBE.JPG

DOWNLOAD RATING : 0

Fig 8.6 Products

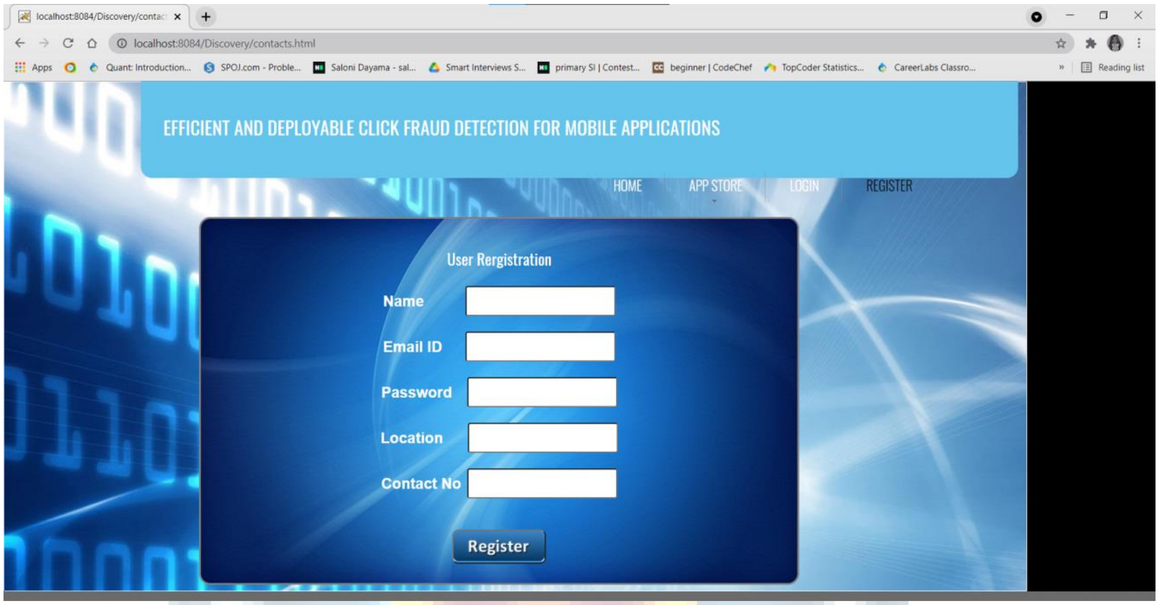


Fig 8.7 User Registration

9. EXPERIMENTAL RESULTS

Active defense improves detection rates by almost 10%.The results of active defenses are documented in Table 3. In both Google and AdSense, active defenses are very successful (> 89%)at detecting fake clicks at all ranges of attack traffic volumes from stealthy to a fire hose, at low FPR of 30–40 per million clicks. The reduction in FPR for low-rate attacks is most improved compared to passive defenses, indicating the importance of considering active attack approaches in fighting click fraud. However, when active defenses are presented with poor context (1-weektrafficset),i.e. applied over only few click sper user, we observe that detection and FPR are similar to passive defenses.

To understand why, we must note that looking for are spon set o injected traffic mainly detects mimicking attacks (as opposed to other variable-rate attacks such as randomly generated fake clicks). In our dataset, a week’s traffic contains between 2 and 43 user clicks per day. At attack rates of two per day, Click to k fails to detect the attack .As attack rates which are still fairly stealthily increases, click fraud attacks are readily detectable; For an increase in fake click rate from 1% to 10% of legit-imate traffic, we observe a reduction in FPR by an order of magnitude for stealth attacks, while detection rate increases from 50% to 70%.For higher attack rates, the click fraud campaigns are mostly randomly generated fake clicks, and these result in modest changes in FPR.

10. CONCLUSION AND FUTURE ENHANCEMENT

10.1 Conclusion

AdSherlock is an efficient and deployable click fraud detection approach for mobile apps at the client side. As a client-side approach, AdSherlock is orthogonal to existing server-side approaches. It splits the computation intensive operations of click request identification into an offline process and an online process. In the offline process, AdSherlock generates both exact patterns and probabilistic patterns based on url tokenization. These patterns are used in the online process for click request identification, and further used for click fraud detection together with an ad request tree model. Evaluation shows that AdSherlock achieves high click fraud detection accuracy with a negligible runtime overhead. In the future, we plan to combine static analysis with the traffic analysis to improve the accuracy of ad request identification and explore attacks designed to evade AdSherlock.

10.2 Future Enhancement

We also evaluated an active defense, where we injected watermarked click traffic into the analysis environment, that works better still. While timing analysis is well studied within the field of information hiding, for its ability to unearth hidden communication, its potential has yet to be fully explored in understanding stealthly click fraud attacks. Our work indicates that timing analysis might indeed be relevant to building better click fraud detection.

11. REFERENCES

[1] "Mobile advertising spending worldwide." [Online]. Available: https://www.statista.com/statistics/280640/mobile-advertisingspending- worldwide/

[2] "Google admob." [Online]. Available: https://apps.admob.com/

[3] M. Mahdian and K. Tomak, "Pay-per-action model for online advertising," in Proc. of ACM ADKDD, 2007.

[4] G. Cho, J. Cho, Y. Song, and H. Kim, "An empirical study of click fraud in mobile advertising networks," in Proc. of ACM ARES, 2015.

[5] J. Crussell, R. Stevens, and H. Chen, "Madfraud: Investigating ad fraud in android applications," in Proc. of ACM MobySys, 2014.

[6] R. Oentaryo, E.-P. Lim, M. Finegold, D. Lo, F. Zhu, C. Phua, E.-Y. Cheu, G.-E. Yap, K. Sim, M. N. Nguyen, K. Perera, B. Neupane, M. Faisal, Z. Aung, W. L. Woon, W. Chen, D. Patel, and D. Berrar, "Detecting click fraud in online advertising: A data mining approach," The Journal of Machine Learning Research, vol. 15, no. 1, pp. 99–140, 2014.

[7] B. Kitts, Y. J. Zhang, G. Wu, W. Brandi, J. Beasley, K. Morrill, J. Ettedgui, S. Siddhartha, H. Yuan, F. Gao, P. Azo, and R. Mahato, Click Fraud Detection: Adversarial Pattern Recognition over 5 Years at Microsoft. Cham: Springer International Publishing, 2015, pp. 181–201.

[8] A. Metwally, D. Agrawal, and A. El Abbadi, "Detectives: detecting coalition hit inflation attacks in advertising networks streams," in Proc. of ACM WWW, 2007.

[9] A. Metwally, D. Agrawal, A. El Abbad, and Q. Zheng, "On hit inflation techniques and detection in streams of web advertising networks," in Proc. of IEEE ICDCS, 2007.

[10] F. Yu, Y. Xie, and Q. Ke, "Sbotminer: large scale search bot detection," in Proc. of ACM WSDM, 2010.

[11] L. Zhang and Y. Guan, "Detecting click fraud in pay-per-click streams of online advertising networks," in Proc. of IEEE ICDCS, 2008.

[12] A. Metwally, D. Agrawal, and A. El Abbadi, "Duplicate detection in click streams," in Proc. of ACM WWW, 2005.

[13] M. S. Iqbal, M. Zulkernine, F. Jaafar, and Y. Gu, "Fcfraud: Fighting click-fraud from the user side," in Proc. of IEEE HASE, 2016.

[14] B. Liu, S. Nath, R. Govindan, and J. Liu, “Decaf: detecting and characterizing ad fraud in mobile apps,” in Proc. of USENIX NSDI, 2014.

[15] G. Cho, J. Cho, Y. Song, D. Choi, and H. Kim, “Combating online fraud attacks in mobile-based advertising,” EURASIP Journal on Information Security, vol. 2016, no. 1, p. 1, 2016.

[16] W. Li, H. Li, H. Chen, and Y. Xia, “Adattester: Secure online mobile advertisement attestation using trustzone,” in Proc. of ACM MobySys, 2015.

[17]“Monkeyrunner.”[Online].Available: http://developer.android.com/studio/test/monkeyrunner/index.html

[18] “Zedge.” [Online]. Available: https://play.google.com/store/apps/

[19] S. Dai, A. Tongaonkar, X. Wang, A. Nucci, and D. Song, “Networkprofiler: Towards automatic fingerprinting of android apps,” in Proc. of IEEE INFOCOM, 2013.

[20] “Mopub.” [Online]. Available: https://www.mopub.com/

A

PROJECT REPORT

On

PREDICTING THE REVIEWS OF THE RESTAURANT USING NATURAL LANGUAGE PROCESSING TECHNIQUE

Submitted by

Mr. L Manish Reddy (17K81A1229)

Mr. K Srikant (17K81A1227)

Ms. M Pranavitha (17K81A1230)

Mr. K Devaraj Reddy (17K81A1226)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Mr. P Manohar

Associate Professor

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN'S ENGINEERING COLLEGE
An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **PREDICTING THE REVIEWS OF THE RESTAURANT USING NATURAL LANGUAGE PROCESSING TECHNIQUE**, is being submitted by **L.Manish Reddy (17K81A1229) ,K.Srikant (17K81A1227), M.Pranavitha (17K81A1227), K.Devaraj Reddy (17K81A1226)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Mr P.Manohar
Department of Information Technology

Head of the Department
Dr. R.Nagaraju
Department of Information Technology

Internal Examiner

External Examiner

Place:

Date:

DECLARATION

We, the students of **Bachelor of Technology** in Department of Information Technology, session: 2017 – 2021, St. Martin’s Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **Predicting The Reviews Of The Restaurant Using Natural Language Processing Technique** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

L.MANISH REDDY	**17K81A1229**
K.SRIKANT	**17K81A1227**
M.PRANAVITHA	**17K81A1230**
K.DEVARAJ REDDY	**17K81A1226**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R.Nagaraju**, Head of the Department, Department of Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D.Babu Rao,** Associate Professor, in Department of Information Technology, for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Mr. P.Manohar**, Associate Professor Department of Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

L.MANISH REDDY	**17K81A1229**
K.SRIKANT	**17K81A1227**
M.PRANAVITHA	**17K81A1230**
K.DEVARAJ REDDY	**17K81A1226**

TABLE OF CONTENTS

Title **Page No**

ABSTRACT

One of the most effective tools any restaurant has is the ability to track food and beverage sales daily. Currently, Recommender systems plays an important role in both academia and industry. These are very helpful for managing information overload. In this paper, we applied machine learning techniques for user reviews and analyze valuable information in the reviews. Reviews are useful for making decisions for both customers and owners. We build a machine learning model with Natural Language Processing techniques that can capture the user's opinions from users' reviews. For experimentation, the python language was used.

In the era of the web, a huge amount of information is now flowing over the network. Since the range of web content covers subjective opinion as well as objective information, it is now common for people to gather information about products and services that they want to buy. However since a considerable amount of information exists as text-fragments without having any kind of numerical scales, it is hard to classify their evaluation efficiently without reading full text. Here we will focus on extracting scored ratings from text fragments on the web and suggests various experiments in order to improve the quality of a classifier.

LIST OF FIGURES

LIST OF SCREENSHOTS

1. INTRODUCTION

1.1 Project Overview:

One of the most effective tools any restaurant has is the ability to track food and beverage sales daily. Currently, Recommender systems plays an important role in both academia and industry. These are very helpful for managing information overload. In this paper, we applied machine learning techniques for user reviews and analyze valuable information in the reviews. Reviews are useful for making decisions for both customers and owners. We build a machine learning model with Natural Language Processing techniques that can capture the user's opinions from users' reviews. For experimentation, the python language was used.

Restaurant customers give their ratings and write reviews based on their satisfaction levels. These ratings and reviews help the other customers to make decision on going to those restaurants. These ratings are also helpful for the restaurant owners to make changes based their reviews for improving their business Restaurant reviews contains textual information. But most of the machine learning algorithms works with numerical data only. Machine learning can be considered as one of the applications of artificial intelligence (AI).ML provides a way to learn the systems without being explicitly programmed and this learning can be used for solving problems.Machine learning takes data as input and it learns some important relations from data to make decisions as per user requirements. The learning process starts with the observations like samples, direct experience and then find patterns in that data to make better decisions to predict or classify new things in the future. For text processing machine learning provides Natural language processing (NLP) capabilities. We can easily analyze our textual datasets through NLP methodologies.NLP provides an opprotunity for data analysts to apply machine learning and deep learning algorithms to our textual datasets. We make use of machine learning algorithms for classifying reviews and recommend the best restaurant. In general, the methods implemented in a recommender system are three types namely Content-based Methods, Collaborative Methods and Hybrid Methods. content-based methods depends on likenesses between the reviews of the users. It prescribes items to a client dependent on recently evaluated most noteworthy things by a similar client. Generally, we need to construct customer-profile data and item-profile data by using the content of shared attribute space. For example, consider a movie, we can represent it with the movie stars in it and the genres. For customer profile, we can do the same thing based on the users likes some movie stars/genres etc. For calculating how good a movie is, we may use cosine similarity. Collaborative techniques are based on user behaviour for recommendation of items. These methods don't need anything else except users' historical preference on a set of items. Because it's based on historical data, the core assumption here is that the users who have agreed in the past tend to also agree in the future. In terms of user preference, it usually expressed by two categories. Hybrid method comprises both the features of content-based methods and collaborative methods.

Generally, we need a procedure for representing text information for the ML algorithm. Bag-of-words is useful to complete this task. This model is simple to implement. It is one of the methods to extract features from the given text for machine learning models. Bag of Words model is used to preprocess the input text by changing it into a bag of words. Now can be represented using a table,which contains the count of words corresponding to the word itself. ML methods need input data to be in number format. But restaurant reviews contain textual information. In this method, each word is also called as "gram". We can also create a vocabulary of two-word pairs. It is called a bigram model. The general model is called as n-gram model. The procedure for changing the text into numbers is called as vectorization in Natural Language Processing models. This vectorization can be done in 3 ways namely count vectorizer, tfidf vectorizer, HashingVectorizer. In count vectorizer, each word count is calculated. After counting the occurrences of words, count vectorizer bulids a parse matrix with x words in the document.

After applying one of the above vectorization models,the entire text data is converted into a sparse matrix form with numeric data. Now, this data is ready for applying machine learning algorithms. Before applying a machine learning algorithm, we divided the given dataset into training and testing data. For this division, we applied 5-fold cross validation technique. In this technique, data is divided into 5 parts also called folds. All the 5 folds can be used as testing sets in one of the iterations. The dataset contains 1000 reviews.As per 5-fold cross validation technique,training set contains 800 reviews and testing set contains 200 reviews. But these reviews are in the form of vectors in numerical notations. we applied several machine learning algorithms for the classification of reviews. To measure accuracy of our model, we used classification accuracy measure,which can be calculated as follows: accuracy= (True Positives +True Negatives)/Count of total samples here, true positives means that actual label is true and model says it is true.True Negatives (TN) means that the actual sample label is false and the algorithm says it is as false. We achieved an accuracy of 73% with count vectorization method and naive bayes classification model

1.2 Project Objectives:

The Internet has opened the new doors for information exchange and the growth of social media has created unprecedented opportunities for citizens to publicly raise their opinions, but it has serious bottlenecks when it comes to doing analysis of these opinions. Even urgency to gain a real time understanding of citizens' concerns has grown very rapidly. Since the viral nature of social media, which is fast and distributed one, some issues get rapidly distributed and unpredictably become important through these word of mouth opinions expressed online which in turn has become known as the sentiments of the users. The decision makers and people do not yet realize how to make sense of this mass communication and interact sensibly with thousands of others with the help of sentiment analysis. To understand thoroughly use of sentiment analysis in today’s business world, this chapter covers the brief about sentiment analysis including introduction of sentiment analysis, early history of sentiment analysis, problems of sentiment analysis, basic concepts of sentiment analysis with mathematical treatment, sentiment and subjectivity classification comprises of opinion mining and summarization, past scenarios of opinion or sentiment collection and their analysis. Methodologies like Sentiment Analysis as Text Classification Problem, Sentiment analysis as Feature Classification with mathematical treatment are explored. Also, Economic consequences of sentiment analysis on individuals, society and organizations with the help of social media sentiment analysis are provided as supporting components.

1.3 Scope of the Project:

The project scope statement is a key document that provides all stakeholders with a clear understanding of why the project was initiated and defines its key goals. Our goal is to predict the Restaurant reviews, to get User opinions by analysing reviews and to know the performance of our trained ML Model. Our proposed system is to apply natural language processing techniques to classify a set of restaurant reviews based on the number of stars that each review received. We develop a maximum entropy classifier to categorize each review from 1-star to 5-stars. We implement a set of features that we believe to be relevant to the sentiment expressed in reviews and analyze their effect on performance, providing insights into what works and why sentiment categorization can be so difficult. We analyze how a review’s conformance to a particular language model can be affected by the sentiment of the review We experiment with different linguistically motivated models of sentiment expression, again using the results to improve the performance of our classifier We examine the effects of part-of-speech tagging on our ability to predict sentiment. We experimented with different methods of preprocessing the data. Because the reviews are unstructured in terms of user input, reviews can look like anything from a paragraph of well-formatted text to a jumble of seemingly unrelated words to a run-on sentence with no apparent regard for grammar or Punctuation.

Our initial pass over the data simply tokenized the reviews based on whitespace and treated each token as a unigram, but we were able to improve performance by removing punctuation in addition to the whitespace and converting all letters to lowercase. In this way, we treat the occurrences of “good”, “Good”, and “good.” all as the same, which gives better predictive power to any test set review containing any of these three forms. Before converting into the unigram stemming was also done which means the various forms (tenses, verbs) of the words were removed and treated as a single word. After the matrix is build the non-frequent words are removed by setting a threshold in order to improve the accuracy. So our matrix includes relevant unigrams as well as bigrams which are occurring more than the threshold times. We examine the effects of part-of-speech tagging on our ability to predict sentiment.

1.4 Organization of Chapters:

1.4.1 Introduction

Businesses often want to know how customers think about the quality of their services in order to improve and make more profits. Restaurant goers may want to learn from others' experience using a variety of criteria such as food quality, service, ambience, discounts and worthiness. Yelp users may post their reviews and ratings on businesses and services or simply express their thoughts on other reviews. Bad (negative) reviews from one's perspective may have an effect on potential customers in making decisions, e.g., a potential customer may cancel a service and persuade other do the same.

There is no lack of studies on the Yelp dataset from different viewpoints that unveil valuable information on a wide range of topics such as the effect of promotion strategies [1], the benefit of retrieving knowledge from implicit user feedback [2], or the important role of local reviewers
[3]. The limitation of most studies is the inability to capture rapid changes in knowledge so that the findings are valid only under a static view of business activities. In the real world, an evolving data source such as Yelp may be better modeled with a dynamic approach to accurately reflect continuous changes in business activities. Incremental learning, an example of such a dynamic approach, has the ability to learn a new concept without retraining on the entire dataset. The question is to quantify how customers and businesses are influenced and how business ratings change in response to recent feedback with an incremental learning approach.

Incremental learning supports two approaches: instance and batch solutions, which differ by batch size. The batch approach requires a full batch of examples to train so that it learns from the most recent data examples as they come in since it has to wait till a sufficient number of examples that can be called a batch becomes available. Among state of-the-art classifiers, our work focuses on ensemble learning where the ability to integrate a new model or to eliminate an old model is an advantage of its design. We use the Random Forests approach, a popular ensemble method, in building our proposed incremental learning model.

Given a decision tree, the ability to induct a new data instance into it relies on how a new split can be processed. Maintaining all relevant examples at each node is the simplest solution, but costly in terms of memory. Without keeping this information, there is no way to make a new split in response to an incoming data instance due to the recursive design.

Incremental learning is inspired by the need to use data mining algorithms on stream data, where training data instances come along a timeline. An early model, incremental induction decision tree [4], reconstructs a decision tree by determining a feasible split after each incoming data instance arrives. The downside of this approach is that it is possible to produce an unstable tree in some rare cases when the splitting feature may be shuffled repeatedly as a result of incoming data. Furthermore, a single decision tree has been known to be outperformed by a forest of decision trees (an ensemble model) that uses consensus opinion.With training data arriving on a timeline, incoming data cannot be used to correct a previous split when a decision at a particular node has been made. To overcome this problem, Domingo et al [5]keep several splitting candidates in every leaf and propose a method, known as Hoeffding bound, to estimate the probability of a good split. The implementation of this method in the form of Hoeffding Trees is reported to have better performance [6]

A non-tree single learner such as Online Naive Bayes classifier proposed by [7] uses updated counters incrementally to compute the probability of the class the new example belongs to. A recently proposed incremental Naive Bayes (in short iNB) algorithm [8] computes the posterior probability of test examples in each class and class conditional probability after an incoming data example arrives, and uses them to adjust the degree of error between classification prediction and observation

Stochastic Gradient Descent [9], an example classifier that works in incremental/online settings, does not make a distribution assumption. A variant of Stochatic Gradient Descent, known as Factorization Machine (FM) [10], uses regularization automatically in training. Using a polynomial kernel, FM is often comparable to Support Vector Machines (SVM) but works well with sparse data. As the performance of FM is stable in this study in all experiments, we use it as a representation of SGD in our study. With training data arriving on a timeline, incoming data cannot be used to correct a previous split when a decision at a particular node has been made With training data arriving on a timeline, Under the online category, several solutions have been suggested including Random Forests (ORF) [11], Hoeffding Tree [12], and online Mondrian Forest [13]. However, none of these approaches have been used for sentiment analysis

1.4.2 Literature Survey

One of the most effective tools any restaurant has is the ability to track food and beverage sales daily. Currently, Recommender systems plays an important role in both academia and industry. These are very helpful for managing information overload. In this paper, we applied machine learning techniques for user reviews and analyze valuable information in the reviews. Reviews are useful for making decisions for both customers and owners. We build a machine learning model with Natural Language Processing techniques that can capture the user's opinions from users' reviews. For experimentation, the python language was used.

Sentiment analysis of customer reviews has a crucial impact on a business's development strategy. Despite the fact that a repository of reviews evolves over time, sentiment analysis often relies on offline solutions where training data is collected before the model is built. If we want to avoid retraining the entire model from time to time, incremental learning becomes the best alternative solution for this task. In this work, we present a variant of online random forests to perform sentiment analysis on customers' reviews. Our model is able to achieve accuracy similar to offline methods and comparable to other online models.

1.4.3 Software & Hardware Requirements

1.4.3.1 Software Requirements

- Operating System : Windows family
- Technology : Python 3.6 or Higher
- IDE : PyCharm

1.4.3.2 Hardware Requirements

- Processer : Intel i3 or Higher
- Ram : Min 4 GB
- Hard Disk : Min 100 GB

1.4.4 Software Development Analysis

1.4.4.1 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.

Sentiment analysis of customer reviews has a crucial impact on a business’s development strategy. Despite the fact that a repository of reviews evolves over time, sentiment analysis often relies on offline solutions where training data is collected before the model.

1.4.4.2 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.

1.4.4.3 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.

1.4.4.4 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.

1.4.4.5 Approaches

As a field, machine learning is closely related to computational statistics, so having a 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. 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.

1.4.4.6 Decision Tree Learning

For general use, decision trees are employed to visually represent decisions and show or inform decision making. When working with machine learning and data mining, decision trees are used as a predictive model. These models map observations about data to conclusions about the data's target value. The goal of decision tree learning is to create a model that will predict the value of a target based on input variables. In the predictive model, the data's attributes that are determined through observation are represented by the branches, while the conclusions about the data's target value are represented in the leaves.

When "learning" a tree, the source data is divided into subsets based on an attribute value test, which is repeated on each of the derived subsets recursively. Once the subset at a node has the equivalent value as its target value has, the recursion process will be complete.

Let's look at an example of various conditions that can determine whether or not someone should go fishing. This includes weather conditions as well as barometric pressure conditions. 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. 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.

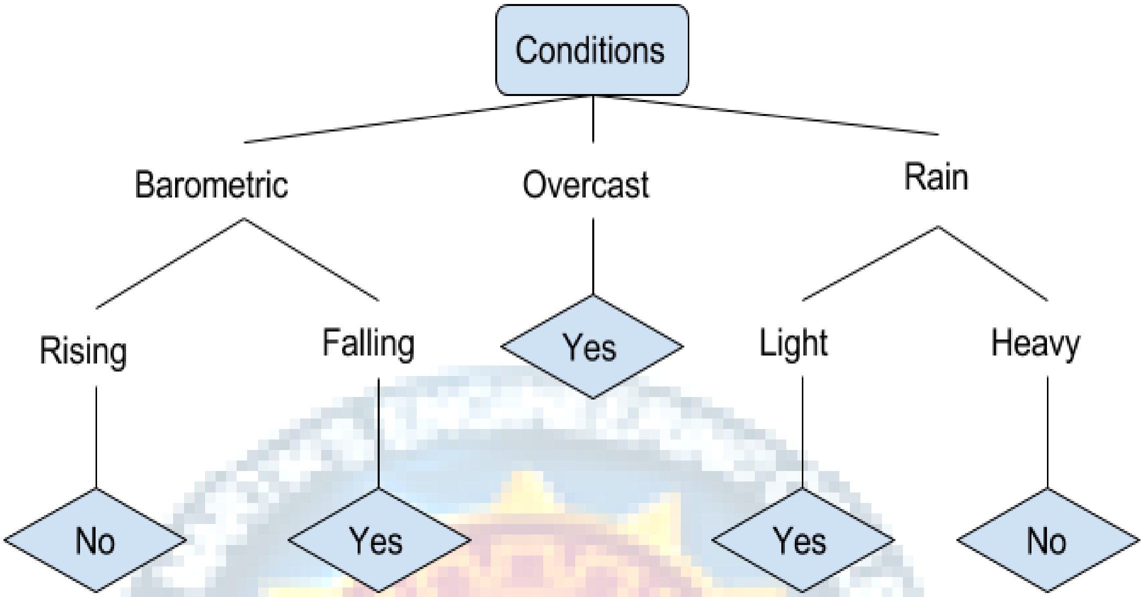


Fig 1.4.4.6.1 Decision Tree

In the simplified decision tree above, an example is classified by sorting it through the tree to the appropriate leaf node. This then returns the classification associated with the particular leaf, which in this case is either a Yes or a No. The tree classifies a day's conditions based on whether or not it is suitable for going fishing. A true classification tree data set would have a lot more features than what is outlined above, but relationships should be straightforward to determine. When working with decision tree learning, several determinations need to be made, including what features to choose, what conditions to use for splitting, and understanding when the decision tree has reached a clear ending.

1.4.5 Project System Design

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

1.4.5.1 NLTK Module

The NLTK module is a massive tool kit, aimed at helping you with the entire Natural Language Processing (NLP) methodology. NLTK will aid you with everything from splitting sentences from paragraphs, splitting up words, recognizing the part of speech of those words, highlighting the main subjects, and then even with helping your machine to understand what the text is all about. In this series, we're going to tackle the field of opinion mining, or sentiment analysis.

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum. NLTK has been called "a wonderful tool for teaching, and working in, computational linguistics using Python," and "an amazing library to play with natural language."

Natural Language Processing with Python provides a practical introduction to programming for language processing. Written by the creators of NLTK, it guides the reader through the fundamentals of writing Python programs, working with corpora, categorizing text, analyzing linguistic structure, and more.

1.4.5.2 Pandas Module

Pandas is an open source library in Python. It provides ready to use high-performance data structures and data analysis tools. Pandas module runs on top of NumPy and it is popularly used for data science and data analytics. NumPy is a low-level data structure that supports multi-dimensional arrays and a wide range of mathematical array operations. Pandas has a higher-level interface. It also provides streamlined alignment of tabular data and powerful time series functionality.DataFrame is the key data structure in Pandas. It allows us to store and manipulate tabular data as a 2-D data structure.Pandas provides a rich feature-set on the DataFrame. For example, data alignment, data statistics, slicing, grouping, merging, concatenating data, etc.

Pandas is a Python package that provides fast, flexible, and expressive data structures designed to make working with structured (tabular, multidimensional, potentially heterogeneous) and time series data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python. Additionally, it has the broader goal of becoming the most powerful and flexible open source data analysis / manipulation tool available in any language. It is already well on its way toward this goal.Pandas is well suited for many different kinds of data:

- Tabular data with heterogeneously-typed columns, as in an SQL table or Excel spreadsheet
- Ordered and unordered (not necessarily fixed-frequency) time series data.
- Arbitrary matrix data (homogeneously typed or heterogeneous) with row and column labels
- Any other form of observational / statistical data sets. The data actually need not be labeled at all to be placed into a pandas data structure

The two primary data structures of pandas, Series (1-dimensional) and DataFrame (2-dimensional), handle the vast majority of typical use cases in finance, statistics, social science, and many areas of engineering. For R users, DataFrame provides everything that R’s data.frame provides and much more. pandas is built on top of NumPy and is intended to integrate well within a scientific computing environment with many other 3rd party libraries.

Here are just a few of the things that pandas does well:

- Easy handling of missing data (represented as NaN) in floating point as well as non-floating point data
- Size mutability: columns can be inserted and deleted from DataFrame and higher dimensional objects
- Automatic and explicit data alignment: objects can be explicitly aligned to a set of labels, or the user can simply ignore the labels and let Series, DataFrame, etc. automatically align the data for you in computations
- Powerful, flexible group by functionality to perform split-apply-combine operations on data sets, for both aggregating and transforming data
- Make it easy to convert ragged, differently-indexed data in other Python and NumPy data structures into DataFrame objects
- Intelligent label-based slicing, fancy indexing, and subsetting of large data sets

- Intuitive merging and joining data sets
- Flexible reshaping and pivoting of data sets
- Hierarchical labeling of axes (possible to have multiple labels per tick)
- Robust IO tools for loading data from flat files (CSV and delimited), Excel files, databases, and saving / loading data from the ultrafast HDF5 format
- Time series-specific functionality: date range generation and frequency conversion, moving window statistics, date shifting and lagging.

1.4.6 Project Coding

```
from django.shortcuts import render
import pandas as  pd
from sklearn import metrics
from django.views.generic import TemplateView
import sklearn
def result(request):
    # Importing the dataset
    #dataset = pd.read_csv('Restaurant_Reviews.tsv', delimiter='\t', quoting=3)
    dataset = pd.read_csv('static/Restaurant_Reviews.tsv',delimiter='\t', quoting=3)

    # Cleaning the texts
    import re
    import nltk
    nltk.download('stopwords')
    from nltk.corpus import stopwords
    from nltk.stem.porter import PorterStemmer
    corpus = []
    for i in range(0, 1000):
        review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][i])
        review = review.lower()
        review = review.split()
        ps = PorterStemmer()
        review = [ps.stem(word) for word in review if not word in
set(stopwords.words('english'))]
        review = ' '.join(review)
        corpus.append(review)

    # Creating the Bag of Words model
    from sklearn.feature_extraction.text import CountVectorizer
    cv = CountVectorizer(max_features=1500)
    X = cv.fit_transform(corpus).toarray()
    y = dataset.iloc[:, 1].values

    # Splitting the dataset into the Training set and Test set
    from sklearn.model_selection import train_test_split
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=0)

    # Fitting Naive Bayes to the Training set
    from sklearn.naive_bayes import GaussianNB
    classifier = GaussianNB()
    classifier.fit(X_train, y_train)

    # Predicting the Test set results
    y_pred = classifier.predict(X_test)

    # Making the Confusion Matrix
    from sklearn.metrics import confusion_matrix
    cm = confusion_matrix(y_test, y_pred)

    from sklearn.metrics import accuracy_score
    accuracy=accuracy_score(y_test, y_pred, normalize=False)

    #d={'i':accuracy,'j':cm}
    d = {'i': metrics.accuracy_score(y_test, y_pred), 'j': metrics.confusion_matrix(y_test,
y_pred)}
    return render(request,'restaurant.html',context=d)
################################################################################
###########
class Home(TemplateView):
    template_name = 'home.html'
```

1.4.7 Project 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.

1.4.7.1 Types of Testing:

- Unit Testing
- Integration Testing
- Functional Testing
- System Testing
- White box Testing
- Black box Testing
- Acceptance Testing

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.

Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program.

Functional Testing

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

System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results.

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.

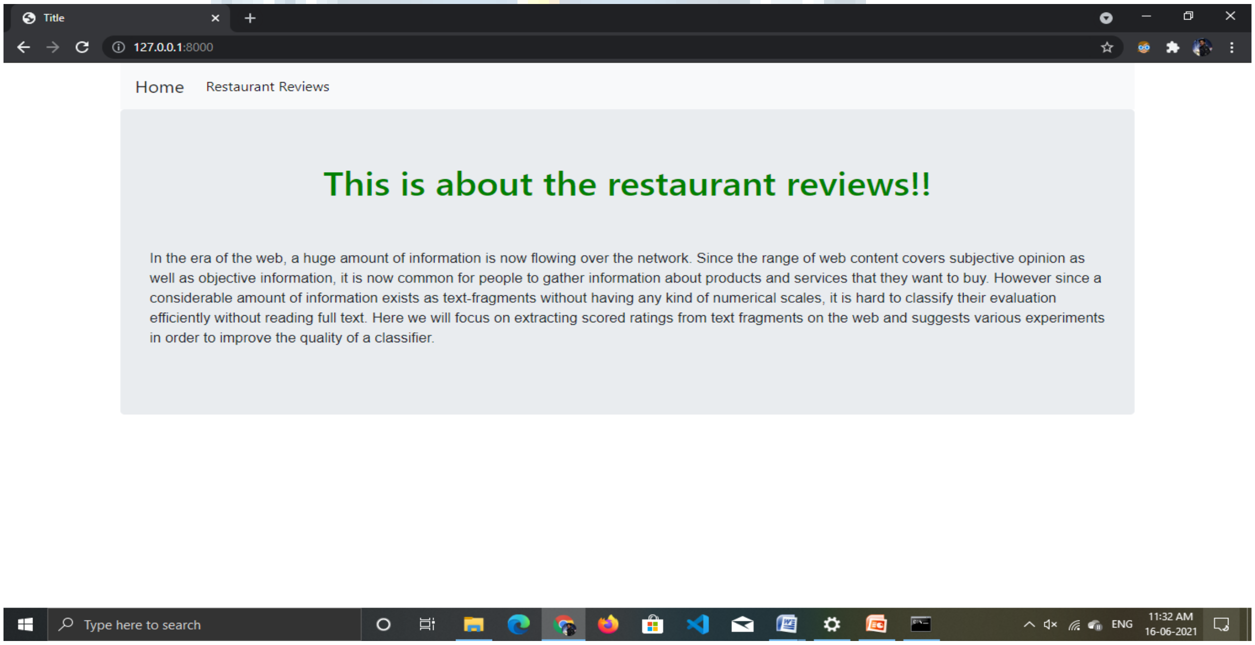
Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested.

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.

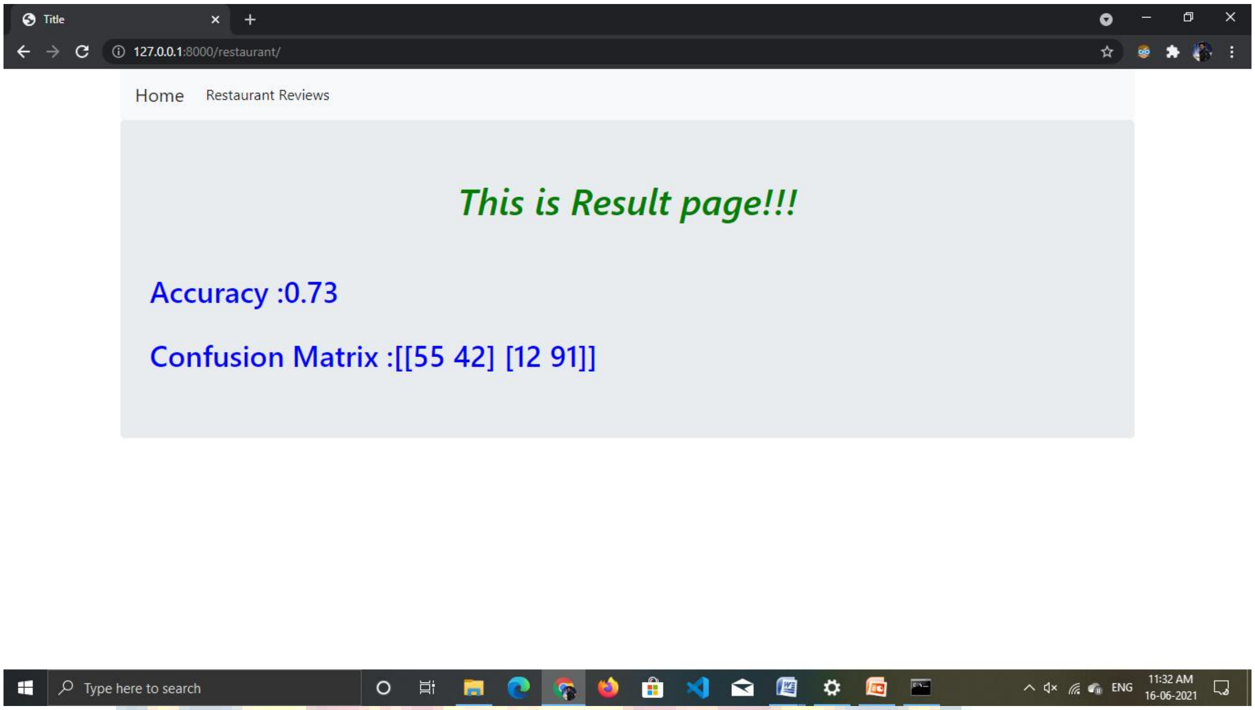
1.4.8 Output Screens



This is about the restaurant reviews!!

In the era of the web, a huge amount of information is now flowing over the network. Since the range of web content covers subjective opinion as well as objective information, it is now common for people to gather information about products and services that they want to buy. However since a considerable amount of information exists as text-fragments without having any kind of numerical scales, it is hard to classify their evaluation efficiently without reading full text. Here we will focus on extracting scored ratings from text fragments on the web and suggests various experiments in order to improve the quality of a classifier.

Screenshot 1.4.8.1 Home Page



This is Result page!!!

Accuracy :0.73

Confusion Matrix :[[55 42] [12 91]]

Screenshot 1.4.8.2 Result Page

1.4.9 Conclusion:

we proposed machine learning NLP techniques for classification of restaurant reviews.We removed stop words, applied stemming and applied vectorization technique. We achieved an accuracy of 73% with count vectorization method and naive bayes classification model.

2. LITERATURE SURVEY

2.1 Survey on Background

2.1.1 Restaurant reviews classification using NLP Techniques

One of the most effective tools any restaurant has is the ability to track food and beverage sales daily. Currently, Recommender systems plays an important role in both academia and industry. These are very helpful for managing information overload. In this paper, we applied machine learning techniques for user reviews and analyze valuable information in the reviews. Reviews are useful for making decisions for both customers and owners. We build a machine learning model with Natural Language Processing techniques that can capture the user's opinions from users' reviews. For experimentation, the python language was used.

2.1.2 Sentiment Analysis of Restaurant Reviews on Yelp with Incremental Learning

Sentiment analysis of customer reviews has a crucial impact on a business's development strategy. Despite the fact that a repository of reviews evolves over time, sentiment analysis often relies on offline solutions where training data is collected before the model is built. If we want to avoid retraining the entire model from time to time, incremental learning becomes the best alternative solution for this task. In this work, we present a variant of online random forests to perform sentiment analysis on customers' reviews. Our model is able to achieve accuracy similar to offline methods and comparable to other online models.

2.1.3 Attribute Sentiment Scoring With Online Text Reviews : Accounting For Language Structure And Attribute Self-Selection

Crowd-sourced online review platforms such as Yelp, TripAdvisor, Amazon and IMDB are increasingly a critical source of scalable, real time feedback for businesses to listen in on their markets. Platforms differ as to which kinds of customer evaluations are presented. While a few of the platforms (e.g., Zagat) show both overall evaluations and attribute level evaluations for each business based on periodic surveys as in Figure 1, most of them (e.g., Yelp, TripAdvisor) choose to provide overall numerical rating (on a 1-5 point scale) and free flowing open ended text describing the product or service experience. On the open-ended system, reviewers can vary in the set of product or service attributes they include and the level of detail on the attributes. Thus, it is not straightforward for consumers and firms to get a quantitative summary of how the product or service performs on different attributes in online reviews, while the overall rating on each business is easy to understand.

2.2 Conclusion on Survey

Machine Learning is not a new technique for text processing. Various researchers applied machine learning techniques for restaurant reviews classification. M. Govindarajan [1] et.al proposed a hybrid classification model for sentiment analysis of restaurant reviews. They proposed an ensemble classifier comprises of support vector machine and Naive Bayes models. With their model, they achieved an accuracy of 90%. Sasikala.P[2] et.al proposed a model for classifying restaurant reviews using sentiments in the words. Their model is based on the score combined with existing text analyzing packages. Most people use 'yelp' for finding a good restaurant. Yelp reviews are very helpful for finding a good restaurant. Boya Yu[3] et.al proposed support vector machines for analyzing Restaurant Features using Sentiment Analysis on Yelp Reviews. Kirange[4] et.al also proposed a Support Vector classifier for Emotion Classification of Restaurant Reviews. They compared their model with Naive Bayes, K-NN and neural network models and shown that SVM achieved good results. Tri Doan [5] et.al proposed a variant of online random forest classifiers for performing sentiment analysis on user reviews. They showed that their model achieved an accuracy similar to offline methods. Ekaterina Pronoza[6] et.al proposed a restaurant information extraction method for the restaurant recommendation system. Veda Waikul [7] et.al proposed an SVM classifier for classifying restaurant reviews. With their model, they achieved an accuracy of 77%.

Internet has opened the new doors for information exchange and the growth of social media has created unprecedented opportunities for citizens to publicly raise their opinions, but it has serious bottlenecks when it comes to do analysis of these opinions. Even urgency to gain a real time understanding of citizens concerns has grown very rapidly. Since, the viral nature of social media which is fast and distributed one, some issues get rapidly distributed and unpredictably become important through this word of mouth opinions expressed online which in turn has known as sentiments of the users. The decision makers and people do not yet realized to make sense of this mass communication and interact sensibly with thousands of others with the help of sentiment analysis. To understand thoroughly use of sentiment analysis in today's business world, this chapter covers the brief about sentiment analysis including introduction of sentiment analysis, early history of sentiment analysis, problems of sentiment analysis, basic concepts of sentiment analysis with mathematical treatment, sentiment and subjectivity classification comprises of opinion mining and summarization, past scenarios of opinion or sentiment collection and their analysis. Methodologies like Sentiment Analysis as Text Classification Problem, Sentiment analysis as Feature Classification with mathematical treatment are explored. Also, Economic consequences of sentiment analysis on individual, society and organization with the help of social media sentiment analysis are provided as supporting component.

3. SOFTWARE AND HARDWARE REQUIREMENTS

3.1 Software Requirements

- Operating System : Windows family
- Technology : Python 3.6 or Higher
- IDE : PyCharm

3.1.1 Python

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. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython 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.

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.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

3.1.2 Purpose

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

3.1.3 Functional Requirements

Graphical User interface with the User.

3.1.4 Non-functional Requirements

- **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.
- **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.
- **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.
- **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.
- **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.
- **Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life.However the allowable wattage can’t be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

3.1.5 Input and Output Design

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

- 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.
- 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.
- 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

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.

3.2 Hardware Requirements

- Processer : Intel i3 or Higher
- Ram : Min 4 GB
- Hard Disk : Min 100 GB

4. SOFTWARE DEVELOPMENT ANALYSIS

4.1 Overview of Problem:

The purpose of this analysis is to build a prediction model to predict whether a review on the restaurant is positive or negative. To do so, we will work on Restaurant Review dataset, we will load it into predictive algorithms Multinomial Naive Bayes, Bernoulli Naive Bayes and Logistic Regression. In the end, we hope to find a "best" model for predicting the review's sentiment.

Dataset: Restaurant_Reviews.tsv is a dataset from Kaggle datasets which consists of 1000 reviews on a restaurant.To build a model to predict if review is positive or negative, following steps are performed.

- Importing Dataset
- Preprocessing Dataset
- Vectorization
- Training and Classification
- Analysis Conclusion

4.1.1 Existing System

Many researchers have done experiments to classify the sentiments of the customers on different datasets earlier. Like Turney (2002) used a semantic orientation algorithm to classify reviews based on the numbers of positively oriented and negatively oriented phrases in each review. Pang et al. (2002) used machine learning tools such as Naïve Bayes, Maximum Entropy and Support Vector Machine (SVM) classifiers to classify movie reviews using a number of simple textual features.

4.1.2 Disadvantages of Existing System

- This type of classification is only done when the classifier has to work on the binary data which is not the case with Restaurant Reviews.
- However, from a practical point of view perhaps the most serious problem with SVMs is the high algorithmic complexity and extensive memory requirements of the required quadratic programming in large-scale tasks.

- If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as “Zero Frequency”.

4.2 Define the Problem :

Let's define the problem in order to think about the solution so as to get an optimal solution.Currently there are many restaurants which are operating all around the world. But hardly restaurants know about what their customers think of the restaurant's service.Restaurants aren't aware of the reasons for their thrusts and troughs , sentiment analysis helps them for the reason for fluctuation of their financial condition.We collect reviews from various restaurants and perform data classification followed by count vectorization and naive bayes Algorithm.Datasets are collected from kaggale.com which provides free datasets.

4.2.1 Proposed System

Our proposed system is to apply natural language processing techniques to classify a set of restaurant reviews based on the number of stars that each review received. We develop a maximum entropy classifier to categorize each review from 1-star to 5-stars. We implement a set of features that we believe to be relevant to the sentiment expressed in reviews and analyze their effect on performance, providing insights into what works and why sentiment categorization can be so difficult. We analyze how a review’s conformance to a particular language model can be affected by the sentiment of the review We experiment with different linguistically motivated models of sentiment expression, again using the results to improve the performance of our classifier We examine the effects of part-of-speech tagging on our ability to predict sentiment. We experimented with different methods of preprocessing the data. Because the reviews are unstructured in terms of user input, reviews can look like anything from a paragraph of well-formatted text to a jumble of seemingly unrelated words to a run-on sentence with no apparent regard for grammar orPunctuation.

Our initial pass over the data simply tokenized the reviews based on whitespace and treated each token as a unigram, but we were able to improve performance by removing punctuation in addition to the whitespace and converting all letters to lowercase. In this way, we treat the occurrences of “good”, “Good”, and “good.” all as the same, which gives better predictive power to any test set review containing any of these three forms. Before converting into the unigram stemming was also done which means the various forms (tenses, verbs) of the words were removed and treated as a single word. After the matrix is build the non-frequent words are removed by setting a threshold in order to improve the accuracy. So our matrix includes relevant unigrams as well as bigrams which are occurring more than the threshold times.

4.2.2 Advantages of Proposed System

- Good at pattern recognition problems
- Data-driven, and performance is high in many problems
- End-to-End training: little or no domain knowledge is needed in system construction
- Learn of representations: cross-modal processing is possible
- Gradient-based learning: learning algorithm is simple
- Mainly supervised learning methods

4.3 Modules Overview :

In Development and analysis of the data set, there is only one single module , which is named as Result Module

4.4 Define the Modules:

We have only one module that is Result Module.What happens in the module is defined below.

4.4.1 Prerequisites

Our dataset will be a collection of 1000 reviews of a restaurant. We’ll use NLP to predict whether a review is positive or negative. This is called ‘Sentiment Analysis’ or ‘Emotional Analysis’ and is extensively used in FinTech.

4.4.2 Training the model

To build the model, we perform the following:

1. Importing the dataset
2. Cleaning the text
3. Creating a ‘Bag of Words’
4. Training and classification

4.4.2.1. Importing the dataset

The dataset is a .tsv (Tab Separated Values) file, with two columns- one with the reviews and another with the review class, i.e., positive (1) or negative (0). We import the dataset with the Pandas library. The parameter delimiter is used to indicate that tab acts as a separator between reviews and their class. Quoting is used to remove the quotes (“) in the review, which may hinder further processing.

4.4.2.2. Cleaning the text

We need to pre-process our data by removing any vague information. For example, we don’t need words such as ‘the,’ ‘and,’ ‘a’ in our text since they do not help in determining whether the review is good or bad. These words are called stopwords. Next, we apply stemming, which is converting all the forms of expression to its root form. For example, ‘loved,’ ‘loving’ to its lemma ‘love.’

4.4.2.3. Creating a ‘Bag of Words’

Next, we apply vectorization to convert the reviews into a numerical format. We create a sparse matrix containing individual reviews as rows and each word of the reviews as columns. We call this the Bag of Words. Our text is now ready for training.

4.4.2.4. Training and classification

The data is split into training and testing sets. The classification models which can be applied to distinguish the reviews are many. But, we use Naive Bayes here, which gives higher accuracy, among others. Naive Bayes is a classifier working on Bayes theorem of probability. It assumes each feature of a dataset as independent ones. It can be extremely faster than other classifiers. You can, of course, try this with other classifiers too.

To see the results of our work, we build a confusion matrix, which shows the number of correct predictions, as well as false positives and false negatives.So, from the matrix, we see that our model has an accuracy of 73%. It has 42 false positives and 12 false negatives. Although the accuracy may seem to be low, it is pretty good for the input of 1000 reviews. With an increase in the number of reviews, the accuracy of the model will increase.

4.4 Module Functionality:

The result module collects the reviews from a .tsv file which contains restaurant reviews and data classification is done and count vectorization is performed and the results are subjected to Naive Bayes Algorithm and obtain the final results. The dataset is imported from kaggle.com which provides free datasets.

5. PROJECT SYSTEM DESIGN

5.1 UML Diagrams:

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML comprises two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML. The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development processes.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of the OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

5.1.1 Usecase Diagram :

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

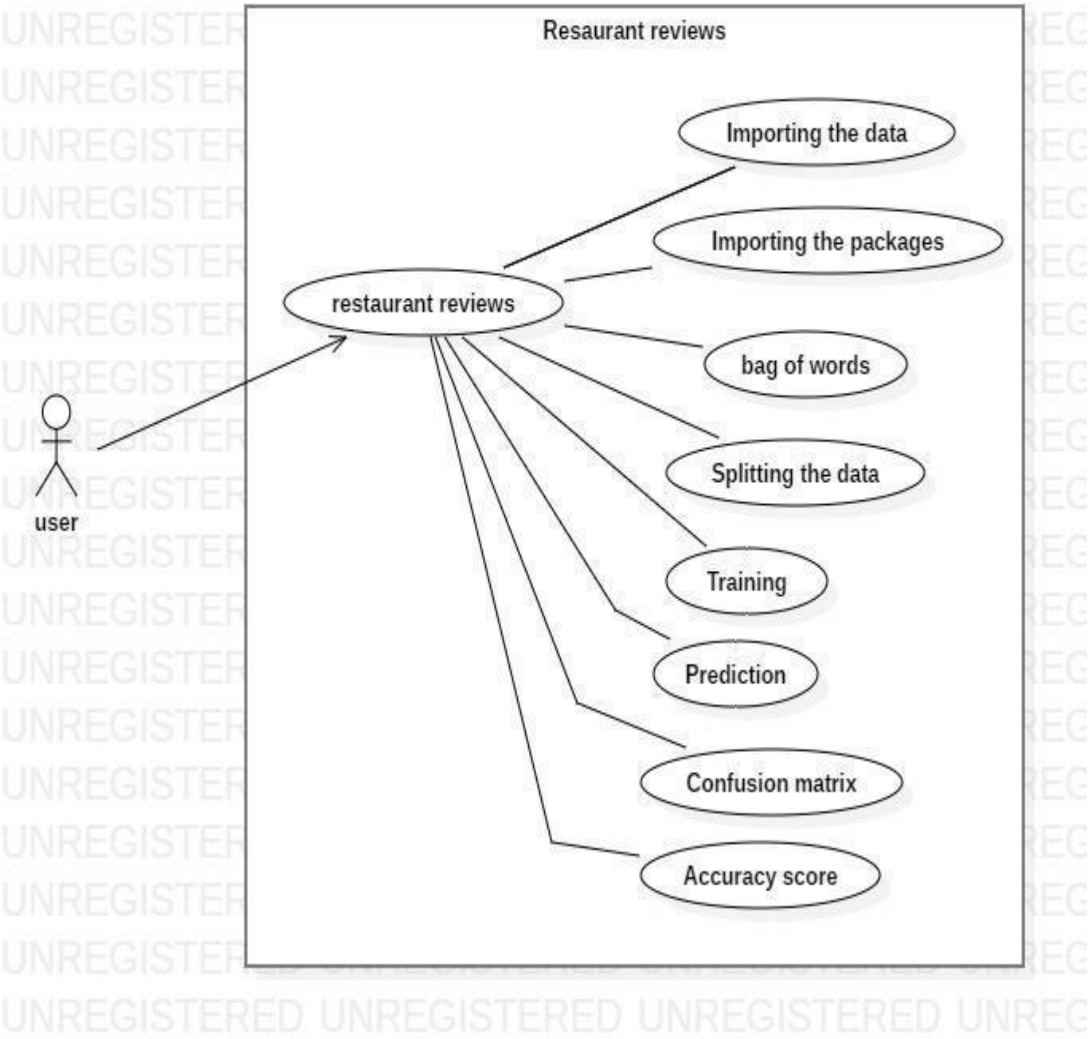


Fig 5.1.1.1 Usecase Diagram

5.1.2 Sequence Diagram :

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagram

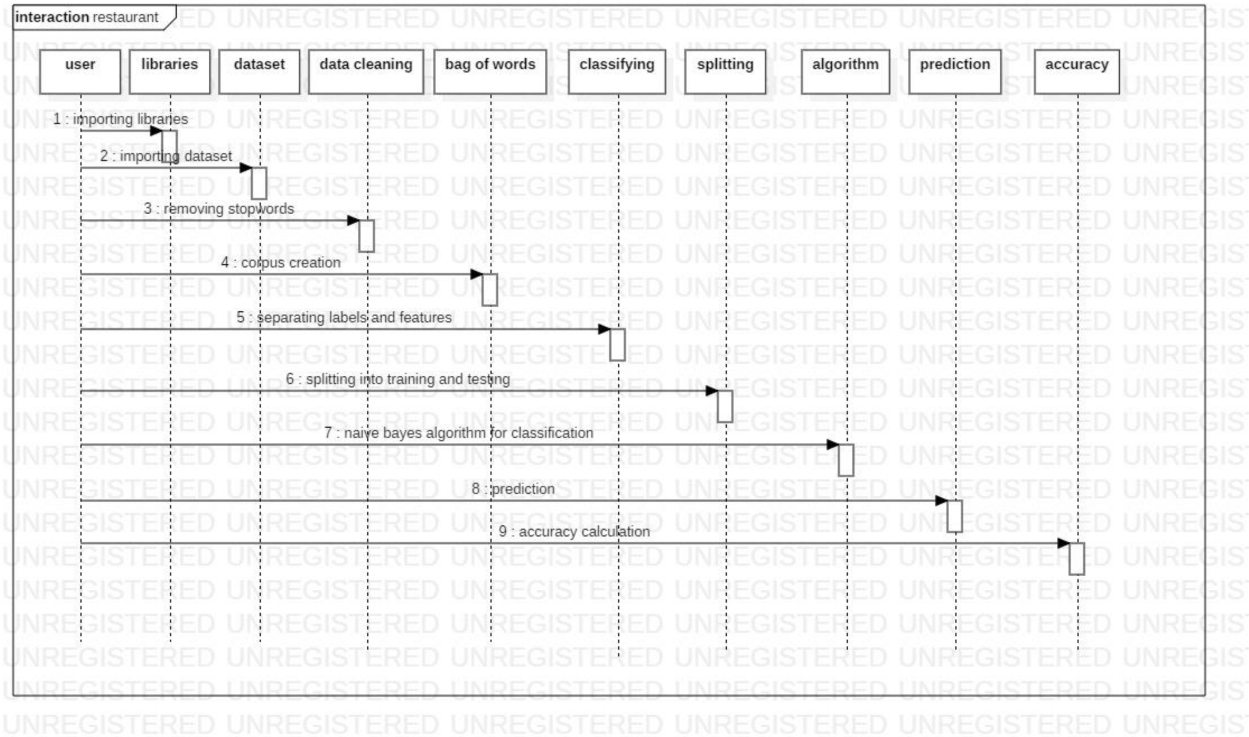


Fig 5.1.2.1 Sequence Diagram

5.1.3 Deployment diagram :

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.[1] To describe a web site, for example, a deployment diagram would show what hardware components (“nodes”) exist (e.g., a web server, an application server, and a database server), what software components (“artifacts”) run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have subnodes, which appear as nested boxes. Asingle node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

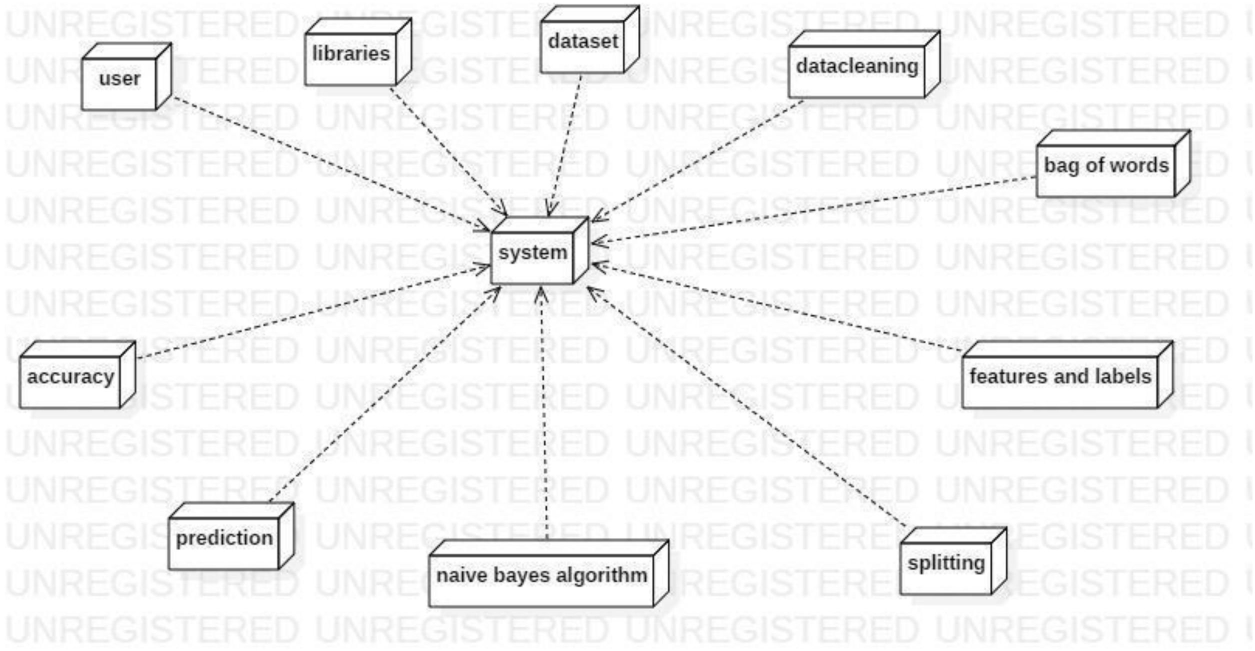


Fig 5.1.3.1 Deployment Diagram

5.1.4 Component Diagram:

The component diagram extends the information given in a component notation element. One way of illustrating the provided and required interfaces by the specified component is in the form of a rectangular compartment attached to the component element.[2] Another accepted way of presenting the interfaces is to use the ball-and-socket graphic convention. A provided dependency from a component to an interface is illustrated with a solid line to the component using the interface from a "lollipop", or ball, labelled with the name of the interface. A required usage dependency from a component to an interface is illustrated by a half-circle, or socket, labelled with the name of the interface, attached by a solid line to the component that requires this interface. Inherited interfaces may be shown with a lollipop, preceding the name label with a caret symbol. To illustrate dependencies between the two, use a solid line with a plain arrowhead joining the socket to the lollipop.[3]

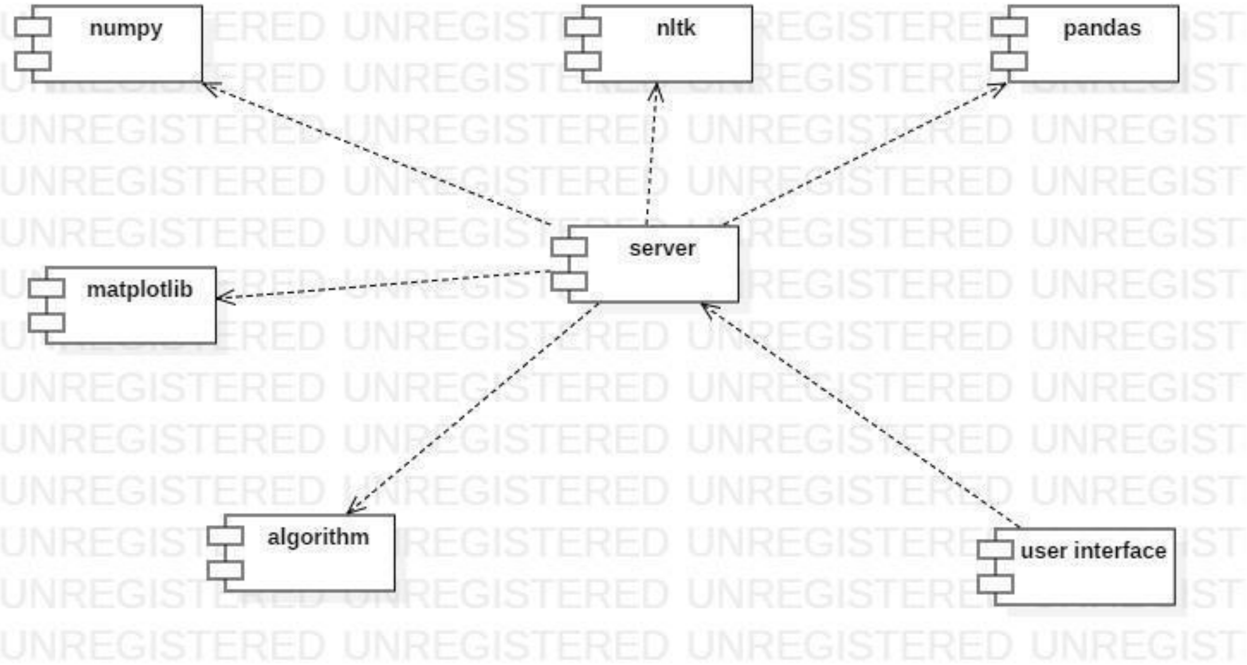


Fig 5.1.4.1 Component Diagram

5.1.5 Activity diagrams:

Activity diagrams are graphical representations of workflows of stepwise activities and actions[1] with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities.[2][3] Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.[citation needed]Activity diagrams are graphical representations of workflows of stepwise activities and actions[1] with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities.[2][3] Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.[citation needed]

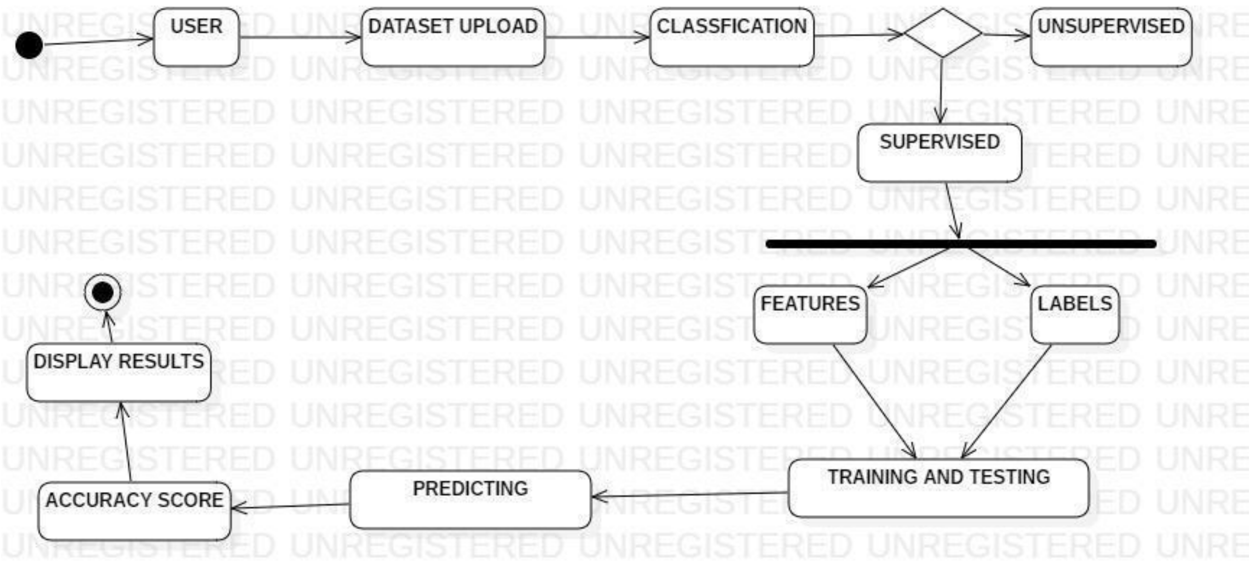


Fig 5.1.5.1 Activity Diagram

5.1.6 Package Diagram:

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.

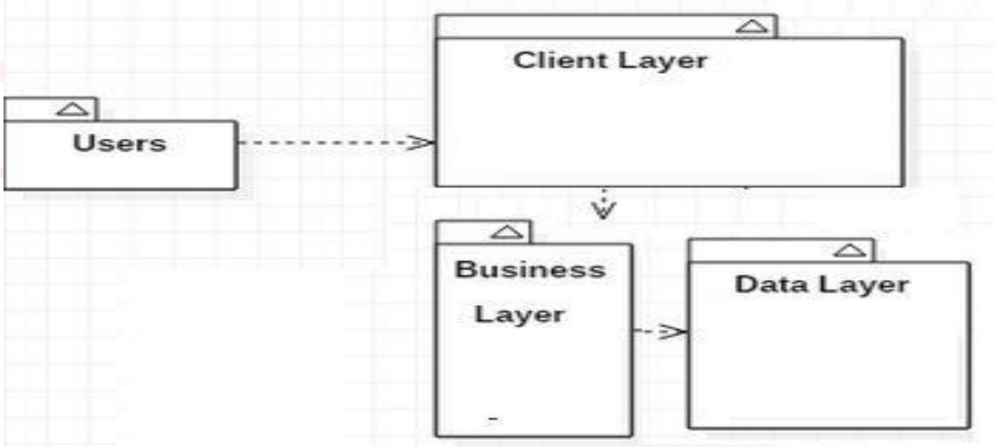


Fig 5.1.6.1 Package Diagram

5.1.7 Profile Diagram :

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connectors and their components.

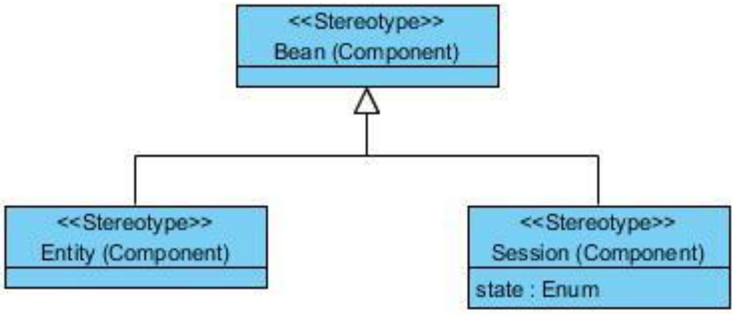


Fig 5.1.7.1 Profile Diagram

6. PROJECT CODING

6.1 Code Templates

app1\views.py

```
from django.shortcuts import render
import pandas as  pd
from sklearn import metrics
from django.views.generic import TemplateView
import sklearn
def result(request):
    # Importing the dataset
    #dataset = pd.read_csv('Restaurant_Reviews.tsv', delimiter='\t', quoting=3)
    dataset = pd.read_csv('static/Restaurant_Reviews.tsv',delimiter='\t', quoting=3)

    # Cleaning the texts
    import re
    import nltk
    nltk.download('stopwords')
    from nltk.corpus import stopwords
    from nltk.stem.porter import PorterStemmer
    corpus = []
    for i in range(0, 1000):
        review = re.sub('[^a-zA-Z]', ' ', dataset['Review'][i])
        review = review.lower()
        review = review.split()
        ps = PorterStemmer()
        review = [ps.stem(word) for word in review if not word in
set(stopwords.words('english'))]
        review = ' '.join(review)
        corpus.append(review)
    # Creating the Bag of Words model
    from sklearn.feature_extraction.text import CountVectorizer
    cv = CountVectorizer(max_features=1500)
    X = cv.fit_transform(corpus).toarray()
    y = dataset.iloc[:, 1].values

    # Splitting the dataset into the Training set and Test set
    from sklearn.model_selection import train_test_split
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=0)

    # Fitting Naive Bayes to the Training set
    from sklearn.naive_bayes import GaussianNB
    classifier = GaussianNB()
    classifier.fit(X_train, y_train)

    # Predicting the Test set results
    y_pred = classifier.predict(X_test)

    # Making the Confusion Matrix
    from sklearn.metrics import confusion_matrix
    cm = confusion_matrix(y_test, y_pred)

    from sklearn.metrics import accuracy_score
    accuracy=accuracy_score(y_test, y_pred, normalize=False)

    #d={'i':accuracy,'j':cm}
    d = {'i': metrics.accuracy_score(y_test, y_pred), 'j': metrics.confusion_matrix(y_test, y_pred)}
    return render(request,'restaurant.html',context=d)
########################################################################
```

```
###########
class Home(TemplateView):
    template_name = 'home.html'
```

app1\urls.py

```
from django.contrib import admin
from django.urls import path
from app1 import views
app_name='app1'
urlpatterns = [
    path('', views.Home.as_view(),name='home'),
    path('restaurant/', views.result,name='result'),
]
```

restaurant\settings.py

```
"""
Django settings for restaurant project.

Generated by 'django-admin startproject' using Django 2.1.4.

For more information on this file, see
https://docs.djangoproject.com/en/2.1/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/2.1/ref/settings/
"""

import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
```

```
# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = '(&3cvt8joy*u-_eril5*wp0%pc@b781nhf3(oe^mehuf)eaf7c'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []



# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'app1'
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'restaurant.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [os.path.join(BASE_DIR, 'templates')]
        ,
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'restaurant.wsgi.application'


# Database
# https://docs.djangoproject.com/en/2.1/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
    }
}



# Password validation
# https://docs.djangoproject.com/en/2.1/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]



# Internationalization
# https://docs.djangoproject.com/en/2.1/topics/i18n/

LANGUAGE_CODE = 'en-us'
```

```
TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/2.1/howto/static-files/

STATIC_URL = '/static/'
STATIC_ROOT=os.path.join(BASE_DIR,'static')
```

restaurant\urls.py

```
from django.contrib import admin
from django.urls import path,include
from app1 import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('app1.urls')),
]
```

restaurant\wsgi.py

```
import os
from django.core.wsgi import get_wsgi_application
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'restaurant.settings')
application = get_wsgi_application()
```

templates\base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPM
O" crossorigin="anonymous">
</head>
<body>
<div class="container">
<nav class="navbar navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="{% url 'app1:home' %}">Home</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
      <li class="nav-item active">
        <a class="nav-link" href="{% url 'app1:result'%}">Restaurant Reviews<span
class="sr-only">(current)</span></a>
      </li>
    </ul>

  </div>
```

```
</nav>
</div>
{% block body_block %}
{% endblock %}
</body>
</html>
```

templates\home.html

```
{% extends 'base.html' %}
{% block body_block %}
   <title>Restaurant Reviews</title>
<style>

h1{
   color: green;
   text-align: center;
}
h2{
   color: blue;
}
p{
   font-family: Arial;
}
</style>

<div class="container">
<div class="jumbotron">
<h1>This is about the restaurant reviews!!</h1><br><br>
<p>

   In the era of the web, a huge amount of information is now flowing over the network.
```

```
Since the range of web content covers
    subjective opinion as well as objective information, it is now common for people to
gather information about products and
    services that they want to buy. However since a considerable amount of information
exists as text-fragments without having
    any kind of numerical scales, it is hard to classify their evaluation efficiently without
reading full text. Here we will
    focus on extracting scored ratings from text fragments on the web and suggests various
experiments in order to improve the
    quality of a classifier.


</p>
</div>
</div>
{% endblock %}
```

templates\restaurant.html

```
<!DOCTYPE html>
{% extends 'base.html' %}
{%  block body_block %}
     <title>Restaurant Reviews</title>
    <style>


h1{
    color: green;
    text-align: center;
}
h2{
    color: blue;
}
p{
   font-family: Arial;
}
</style>
<div class="container">
   <div class="jumbotron">
<h1><em>This is Result page!!!</em></h1><br><br>
<h2>Accuracy :{{ i }}</h2><br>
<h2>Confusion Matrix :{{ j }}</h2>
   </div>
   </div>
{% endblock %}
```

manage.py

```
#!/usr/bin/env python
import os
import sys

if __name__ == '__main__':
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'restaurant.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)
```

6.2 Outline for Various Files

app1\views.py

This is the important file of our project. First the required modules are imported. Then the reviews from the .tsv file are read. We need to pre-process our data by removing any vague information. For example, we don’t need words such as ‘the,’ ‘and,’ ‘a’ in our text since they do not help in determining whether the review is good or bad. These words are called stopwords. Next, we apply stemming, which is converting all the forms of expression to its root form. For example, ‘loved,’ ‘loving’ to its lemma ‘love.’ Next, we apply vectorization to convert the reviews into a numerical format. We create a sparse matrix containing individual reviews as rows and each word of the reviews as columns. We call this the Bag of Words. Our text is now ready for training.

The data is split into training and testing sets. The classification models which can be applied to distinguish the reviews are many. But, we use Naive Bayes here, which gives higher accuracy, among others.Naive Bayes is a classifier working on Bayes theorem of probability. It assumes each feature of a dataset as independent ones. It can be extremely faster than other classifiers. To see the results of our work, we build a confusion matrix, which shows the number of correct predictions, as well as false positives and false negatives.

urls.py

Every page on the Internet needs its own URL. This way your application knows what it should show to a user who opens that URL. In Django, we use something called URLconf (URL configuration). URLconf is a set of patterns that Django will try to match the requested URL to find the correct view.This happens in urls.py file.

settings.py

settings.py is a core file in Django projects. It holds all the configuration values that your web app needs to work; database settings, logging configuration, where to find static files, API keys if you work with external APIs, and a bunch of other stuff.

wsgi.py

Django's primary deployment platform is WSGI, the Python standard for web servers and applications. Django's startproject management command sets up a minimal default WSGI configuration for you, which you can tweak as needed for your project, and direct any WSGI-compliant application server to use.

base.html , home.html and restaurant.html

An HTML file contains Hypertext Markup Language (HTML), which is used to format the structure of a webpage. It is stored in a standard text format and contains tags that define the page layout and content of the webpage, including the text, tables, images, and hyperlinks displayed on the webpage.

manage.py

A command-line utility that lets you interact with this **Django** project in various ways. The **manage.py** script is used to create applications, work with databases, and start the development web server.

7. PROJECT 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.

7.1 Various Testcases

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, non-functional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output, often with the help of a test oracle or a previous result that is known to be good, without any knowledge of the test object's internal structure.

7.2 Block Box Testing

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing. The main focus of black box testing is on the validation of your functional requirements. Black box testing gives abstraction from code and focuses on testing effort on the software system behavior. Black box testing facilitates testing communication amongst modules. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation. There are various types of test. Each test type addresses a specific testing requirement.

Input
Black Box
Output

Fig 7.2.1 Black Box Structure

The above Black-Box can be any software system you want to test. For Example, an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation.

Here are the generic steps followed to carry out any type of Black Box Testing.

- Initially, the requirements and specifications of the system are examined.
- Tester chooses valid inputs (positive test scenario) to check whether SUT processes them correctly. Also, some invalid inputs (negative test scenario) are chosen to verify that the SUT is able to detect them.
- Tester determines expected outputs for all those inputs.
- Software tester constructs test cases with the selected inputs.
- The test cases are executed.
- Software tester compares the actual outputs with the expected outputs.
- Defects if any are fixed and re-tested.

7.2.1 Types of Black Box Testing

There are many types of Black Box Testing but the following are the prominent ones -

- **Functional testing** - This black box testing type is related to the functional requirements of a system; it is done by software testers.
- **Non-functional testing** - This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.

- **Regression testing** - Regression Testing is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

7.2.2 Tools used for Black Box Testing:

Tools used for Black box testing largely depends on the type of black box testing you are doing.

- For Functional/ Regression Tests you can use - QTP, Selenium.
- For Non-Functional Tests, you can use - LoadRunner, Jmeter.

7.2.3 Black Box Testing Techniques

Following are the prominent Test Strategy amongst the many used in Black box Testing

- **Equivalence Class Testing:** It is used to minimize the number of possible test cases to an optimum level while maintains reasonable test coverage.
- **Boundary Value Testing:** Boundary value testing is focused on the values at boundaries. This technique determines whether a certain range of values are acceptable by the system or not. It is very useful in reducing the number of test cases. It is most suitable for the systems where an input is within certain ranges.
- **Decision Table Testing**: A decision table puts causes and their effects in a matrix. There is a unique combination in each column.

7.3 White Box Testing

White Box Testing is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing and Glass box testing.

It is one of two parts of the Box Testing approach to software testing. Its counterpart, Blackbox testing, involves testing from an external or end-user type perspective. On the other hand, White box testing in software engineering is based on the inner workings of an application and revolves around internal testing.

The term "WhiteBox" was used because of the see-through box concept. The clear box or WhiteBox name symbolizes the ability to see through the software's outer shell (or "box") into its inner workings. Likewise, the "black box" in "Black Box Testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested. White box testing involves the testing of the software code for the following:

- Internal security holes
- Broken or poorly structured paths in the coding processes
- The flow of specific inputs through the code
- Expected output
- The functionality of conditional loops
- Testing of each statement, object, and function on an individual basis

The testing can be done at system, integration and unit levels of software development. One of the basic goals of whitebox testing is to verify a working flow for an application. It involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output, you have encountered a bug.

7.3.1 Steps in White Box Testing:

we have divided it into **two basic steps**. This is what testers do when testing an application using the white box testing technique:

Step 1) Understand the Source Code

The first thing a tester will often do is learn and understand the source code of the application. Since white box testing involves the testing of the inner workings of an application, the tester must be very knowledgeable in the programming languages used in the applications they are testing. Also, the testing person must be highly aware of secure coding practices. Security is often one of the primary objectives of testing software. The tester should be able to find security issues and prevent attacks from hackers and naive users who might inject malicious code into the application either knowingly or unknowingly.

Step 2) Create Test Cases and Execute

The second basic step to white box testing involves testing the application's source code for proper flow and structure. One way is by writing more code to test the application's source code. The tester will develop little tests for each process or series of processes in the application. This method requires that the tester must have intimate knowledge of the code and is often done by the developer. Other methods include Manual Testing, trial, and error testing and the use of testing tools as we will explain further on in this article.

7.3.2 White Box Testing Techniques

A major White box testing technique is Code Coverage analysis. Code Coverage analysis eliminates gaps in a Test Case suite. It identifies areas of a program that are not exercised by a set of test cases. Once gaps are identified, you create test cases to verify untested parts of the code, thereby increasing the quality of the software product. There are automated tools available to perform Code coverage analysis. Below are a few coverage analysis techniques a box tester can use:

Statement Coverage:- This technique requires every possible statement in the code to be tested at least once during the testing process of software engineering.

Branch Coverage - This technique checks every possible path (if-else and other conditional loops) of a software application.

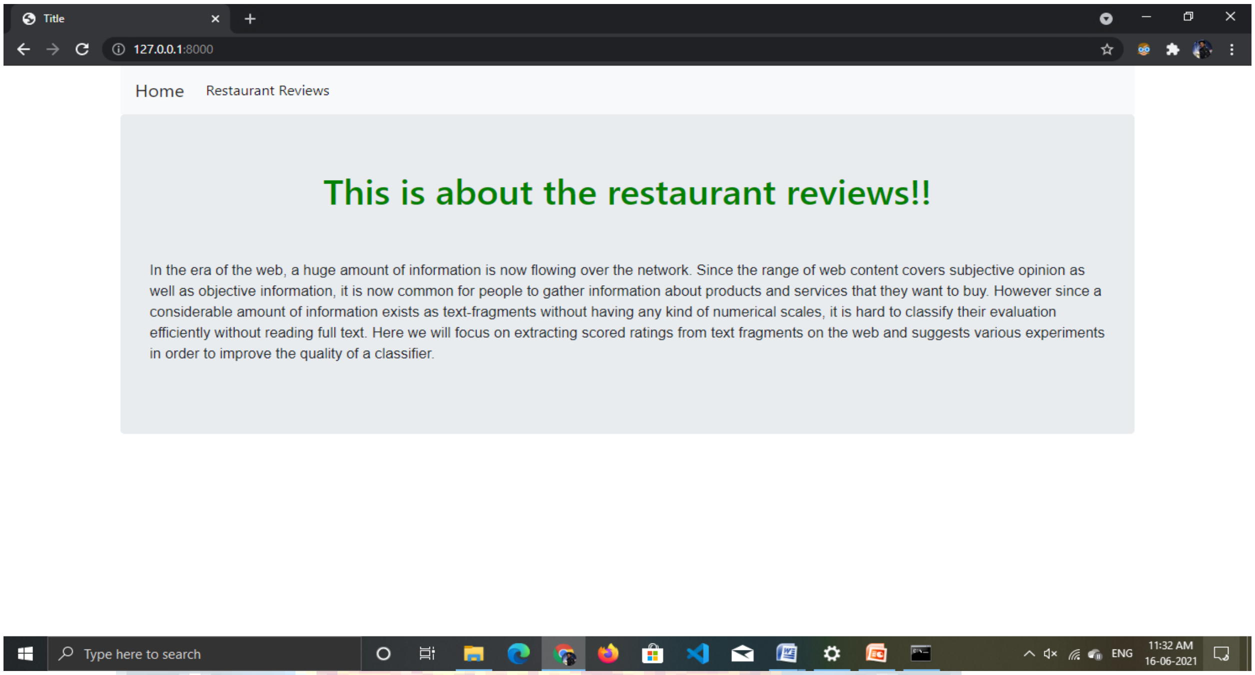
Apart from above, there are numerous coverage types such as Condition Coverage, Multiple Condition Coverage, Path Coverage, Function Coverage etc. Each technique has its own merits and attempts to test (cover) all parts of software code. Using Statement and Branch coverage you generally attain 80-90% code coverage which is sufficient.

Following are important WhiteBox Testing Techniques:

- Statement Coverage
- Decision Coverage
- Branch Coverage
- Condition Coverage
- Multiple Condition Coverage
- Finite State Machine Coverage

- Path Coverage
- Control flow testing
- Data flow testing

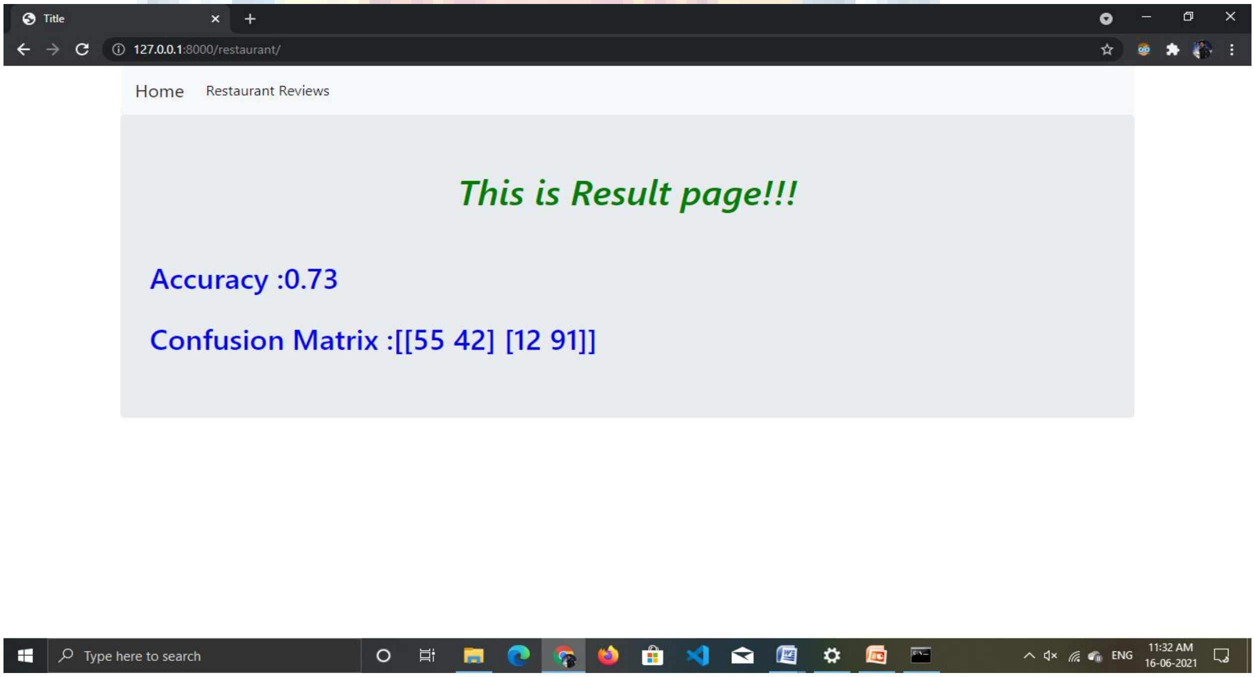
8. OUTPUT SCREENS



This is about the restaurant reviews!!

In the era of the web, a huge amount of information is now flowing over the network. Since the range of web content covers subjective opinion as well as objective information, it is now common for people to gather information about products and services that they want to buy. However since a considerable amount of information exists as text-fragments without having any kind of numerical scales, it is hard to classify their evaluation efficiently without reading full text. Here we will focus on extracting scored ratings from text fragments on the web and suggests various experiments in order to improve the quality of a classifier.

Screenshot 8.1 Home Page



This is Result page!!!

Accuracy :0.73

Confusion Matrix :[[55 42] [12 91]]

Screenshot 8.2 Result Page

9. EXPERIMENTAL RESULTS

9.1 Confusion Matrix:

In predictive analytics, a table of confusion (sometimes also called a confusion matrix) is a table with two rows and two columns that reports the number of false positives, false negatives, true positives, and true negatives. This allows more detailed analysis than mere proportion of correct classifications (accuracy). Accuracy will yield misleading results if the data set is unbalanced; that is, when the numbers of observations in different classes vary greatly. For example, if there were 95 cats and only 5 dogs in the data, a particular classifier might classify all the observations as cats. The overall accuracy would be 95%, but in more detail the classifier would have a 100% recognition rate (sensitivity) for the cat class but a 0% recognition rate for the dog class. F1 score is even more unreliable in such cases, and here would yield over 97.4%, whereas informedness removes such bias and yields 0 as the probability of an informed decision for any form of guessing (here always guessing cat). Confusion matrix is not limited to binary classification and can be used in multi-class classifiers as well. According to Davide Chicco and Giuseppe Jurman, the most informative metric to evaluate a confusion matrix is the Matthews correlation coefficient (MCC).

True Class

55	42
12	91

Predicted Class

Fig 9.1.1 Confusion Matrix

55: True Positive 42: False Positive

12: False negative 91: True Negative

- **True positives (TP):** These are cases in which we predicted yes (Reviews are positive), and prediction is correct.
- **True negatives (TN):** Mode predicted as negative review, and reviews are negative.
- **False positives (FP):** Model predicted positive, but those are negative . (Also known as a "Type I error.").
- **False negatives (FN):** Model predicted negative, but reviews are positive . (Also known as a "Type II error.")

10. CONCLUSION AND FUTURE ENHANCEMENT

10.1 Conclusion

we proposed machine learning NLP techniques for classification of restaurant reviews. We removed stop words, applied stemming and applied vectorization technique. We achieved an accuracy of 73% with count vectorization method and naive bayes classification model.

10.2 Future Enhancement

Automated systems can go through huge quantities of data so this helps to classify data easily. We can extend the Project by taking reviews from user visiting each restaurant.We can display rating of restaurant from user reviews.

Humans are the "Gold Standard" of sentiment analysis yet there is always disagreement within a group of raters on sentiment. Humans generally only agree about 80% of the time. Automatic sentiment analysis can strive towards this level but, obviously, can not exceed it. People and automatic systems both have a place in the process. The Automated systems can go through huge quantities of data while humans can do a higher quality job on a smaller sample.Saying "People are no good because they are not scalable" is probably just as silly as saying "Automatic systems are no good because they are not as accurate".Focus on and use the strengths of each as needed for your particular situation.It will have a lot to do with social forums/platforms where people express free opinion.

Presently tweets are one such open medium, then if facebook at some point chooses to make the timeline updates/status messages open to search (I think it will someday do that through a minuscule sounding update in "privacy policy") it will be gold mine of real-time sentiments. Present Sentiments hold a key to the future events. To make it sound a bit technical, you can say that the sentiments represent the "present value of future events". Now this value can have deep social, political and monetary significance. It can be "Expression of opinion about a public figure", "opinions expressed through tweets before elections", or "the buzz before a movie release", all these can be great cues for things to come. Therefore when people comment about present news stories, the sentiment analysis can actually offer a key to predict the future outcomes or atleast anticipate them better!

11. REFERENCES

1] Ariyasriwatana, W., Buente, W., Oshiro, M., & Streveler, D. (2014). Categorizing health-related cues to action: using Yelp reviews of restaurants in Hawaii. New Review of Hypermedia and Multimedia, 20(4), 317-340.

[2] Byers, J. W., Mitzenmacher, M., & Zervas, G. (2012, June). The groupon effect on yelp ratings: a root cause analysis. In Proceedings of the 13th ACM conference on electronic commerce (pp. 248-265). ACM.

[3] Hicks, A., Comp, S., Horovitz, J., Hovarter, M., Miki, M., & Bevan, J. L. (2012). Why people use Yelp. com: An exploration of uses and gratifications. Computers in Human Behavior, 28(6), 2274-2279.

[4] Mukherjee, A., Venkataraman, V., Liu, B., & Glance, N. S. (2013, July). What yelp fake review filter might be doing?. In ICWSM. 6

[5] dos Santos, C. N., & Gatti, M. (2014). Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts. In COLING (pp. 69-78).

[6] Mullen, T., & Collier, N. (2004, July). Sentiment Analysis using Support Vector Machines with Diverse Information Sources. In EMNLP (Vol. 4, pp. 412-418).

[7] Kiritchenko, S., Zhu, X., Cherry, C., & Mohammad, S. (2014, August). NRC-Canada-2014: Detecting aspects and sentiment in customer reviews. In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014) (pp. 437-442). Dublin, Ireland: Association for Computational Linguistics and Dublin City University.

[8] Huang, J., Rogers, S., & Joo, E. (2014). Improving restaurants by extracting subtopics from yelp reviews. iConference 2014 (Social Media Expo).

[9] Shalev-Shwartz, S., Singer, Y., Srebro, N., & Cotter, A. (2011). Pegasos: Primal estimated sub-gradient solver for svm. Mathematical programming, 127(1), 3-30.

[10] Saif, Hassan, et al. "On stopwords, filtering and data sparsity for sentiment analysis of Twitter." (2014): 810-817.

A

Project report

On

ONLINE DEPRESSION DETECTION APPLICATION

Submitted by

Ms. S.BHARGAVI	**(17K81A1247)**
Mr. M.V.S.SAI KRISHNA	**(17K81A1237)**
Mr. P.SAI CHARAN	**(17K81A1240)**
Ms. D.SHIVANI	**(17K81A1209)**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Dr. R.NAGARAJU

HOD & PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN’S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **“ONLINE DEPRESSION DETECTION APPLICATION”**, is being submitted by **S. BHARGAVI (17K81A1247), M. V. S. SAI KRISHNA (17K81A1237), P. SAI CHARAN (17K81A1240), D. SHIVANI (17K81A1209)** in partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology in Information Technology** is recorded of bonafide work carried out by them. The result embodiedin this report have been verified and found satisfactory.

	Head of the Department
Dr. R. NAGARAJU	Dr.R.NAGARAJU
Department of Information Technology	Department of Information Technology

Internal Examiner

External Examiner

Place:

Date:

LASYA INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **D.SHIVANI** WITH ROLL NO.**17K81A1209, M.V.S.SAI KRISHNA** WITH ROLL NO.**17K81A1237**, **P.SAI CHARAN** WITH ROLL NO.**17K81A1240**, **S.BHARGAVI** WITH ROLL NO.**17K81A1247**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN’S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED “**ONLINE DEPRESSION DETECTION APPLICATION**” AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd.
★ Kompally-500014. ★

ORUGANTI VENKAT

DIRECTOR

TRAININGS & PLACEMENTS LASYA IT
SOLUTIONS PVT LTD.

DECLARATION

We, the student of **Bachelor of Technology in Department of INFORMATION TECHNOLOGY**, session: <2017 – 2021>, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **ONLINE DEPRESSION DETECTION APPLICATION** is the outcome of our own bonafide work and is correctto the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submittedin any university for award of any degree.

S.BHARGAVI	**17K81A1247**
M.V.S.SAI KRISHNA	**17K81A1237**
P.SAI CHARAN	**17K81A1240**
D.SHIVANI	**17K81A1209**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R. NAGARAJU**, Head of the Department, Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Dr. R. Nagaraju**, Professor & Head of the Department, Department of Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

S.BHARGAVI	**17K81A1247**
M.V.S.SAI KRISHNA	**17K81A1237**
P.SAI CHARAN	**17K81A1240**
D.SHIVANI	**17K81A1209**

TABLE OF CONTENTS

Title **Page No**

ABSTRACT

Depression is viewed as the largest contributor to global disability and a major reason for suicide. It has an impact on the language usage reflected in the written text. The key objective of our study is to examine Reddit users' posts to detect any factors that may reveal the depression attitudes of relevant online users. For such purpose, we employ the Natural Language Processing (NLP) techniques and machine learning approaches to train the data and evaluate the efficiency of our proposed method. We identify a lexicon of terms that are more common among depressed accounts. The results show that our proposed method can significantly improve performance accuracy. The best single feature is bigram with the Support Vector Machine (SVM) classifier to detect depression with 80% accuracy and 0.80 F1 scores. The strength and effectiveness of the combined features (LIWC+LDA+bigram) are most successfully demonstrated with the Multilayer Perception (MLP) classifier resulting in the top performance for depression detection reaching 91% accuracy and 0.93 F1 scores. According to our study, better performance improvement can be achieved by proper feature selections and their multiple feature combinations.

LIST OF FIGURES

1. INTRODUCTION

Depression as a common mental health disorder has long been defined as a single disease with a set of diagnostic criteria. It often co-occurs with anxiety or other psychological and physical disorders; and has an impact on feelings and behavior of the affected individuals. According to the WHO study, there are 322 million people estimated to suffer from depression, equivalent to 4.4% of the global population. Nearly half of the in-risk individuals live in the South-East Asia (27%) and Western Pacific region (27%) including China and India. In many countries depression is still under-diagnosed and left without any adequate treatment which can lead into a serious self-perception and at its worst, to suicide. In addition, the social stigma surrounding depression prevents many affected individuals from seeking an appropriate professional assistance.

As a result, they turn to less formal resources such as social media. With the development of Internet usage, people have started to share their experiences and challenges with mental health disorders through online forums, micro-blogs or tweets. Their online activities inspired many researchers to introduce new forms of potential health care solutions and methods for early depression detection systems. Using different Natural Language Processing (NLP) techniques and text classification approaches, they tried to succeed in a higher performance improvement. Some studies use single set features, such as bag of words (BOW), N-grams, LIWC or LDA, to identify depression in their posts. Some other papers compare the performance of individual features with various machine learning classifiers. Recent studies examine the power of single features and their combinations such as N-grams + LIWC or BOW+LDA and TF-IDF+LDA to improve the accuracy results.

Our study has four specific contributions: first, to examine the relationship between depression and user’s language usage; second, to design three LIWC features for our specific research problem; third, to evaluate the power of N-grams probabilities, LIWC and LDA as single features for performance accuracy; fourth, to show the predictive power of both single and combined features with proposed classification approaches to achieve a higher performance in depression identification tasks.

1.1 PROJECT OVERVIEW

Depression is viewed as the largest contributor to global disability and a major reason for suicide. It has an impact on the language usage reflected in the written text. The key objective of our study is to examine Reddit users' posts to detect any factors that may reveal the depression attitudes of relevant online users. For such purpose, we employ the Natural Language Processing (NLP) techniques and machine learning approaches to train the data and evaluate the efficiency of our proposed method. We identify a lexicon of terms that are more common among depressed accounts. The results show that our proposed method can significantly improve performance accuracy. The best single feature is bigram with the Support Vector Machine (SVM) classifier to detect depression with 80% accuracy and 0.80 F1 scores. The strength and effectiveness of the combined features (LIWC+LDA+bigram) are most successfully demonstrated with the Multilayer Perceptron (MLP) classifier resulting in the top performance for depression detection reaching 91% accuracy and 0.93 F1 scores. According to our study, better performance improvement can be achieved by proper feature selections and their multiple feature combinations.

1.2 PROJECT OBJECTIVES

Stress has become an embedded part of our life, being stressed by our financial worries, our job, etc. Stress causes physical illnesses, such as heart attacks, arthritis, and chronic headaches or psychological diseases like mental illness, anger, anxiety, and depression. There are several research works coming up to resolve the limitations on measuring, analyzing and identifying the human stress levels Amongst the many stress monitoring methods the more reliable method by sharing stress related post. This project aim analysing users post this application can detect depression using SVM (support vector machine) algorithm which analyse users post and give result as negative or positive.

1.3 SCOPE OF THE PROJECT

The user expresses different kind of expressions in the post. Among those posts we will detect only the depressed posts and user can review any post among all the posts. Admin will be able to classify the posts with the help of SVM algorithm like depressed, positive and negative posts. Our application is implemented in order to understand the every user mind and detect depression based on their posts.

1.4 ORGANIZATION OF CHAPTERS

1.4.1 INTRODUTION

Depression as a common mental health disorder has long been defined as a single disease with a set of diagnostic criteria. It often co-occurs with anxiety or other psychological and physical disorders; and has an impact on feelings and behavior of the affected individuals. According to the WHO study, there are 322 million people estimated to suffer from depression, equivalent to 4.4% of the global population. Nearly half of the in-risk individuals live in the South-East Asia (27%) and Western Pacific region (27%) including China and India. In many countries depression is still under-diagnosed and left without any adequate treatment which can lead into a serious self-perception and at its worst, to suicide. In addition, the social stigma surrounding depression prevents many affected individuals from seeking an appropriate professional assistance.

1.4.2 LITERATURE SURVEY

Sentimental Analysis is reference to the task of Natural Language Processing to determine whether a text contains subjective information and what information it expresses i.e., whether the attitude behind the text is positive, negative or neutral. This paper focuses on the several machine learning techniques which are used in analyzing the sentiments and in opinion mining. Sentimental analysis with the blend of machine learning could be useful in predicting the product reviews and consumer attitude towards to newly launched product. This paper presents a detail survey of various machine learning techniques and then compared with their accuracy, advantages and limitations of each technique. On comparing we get 85% of accuracy by using supervised machine learning technique which is higher than that of unsupervised learning techniques.

1.4.3 SOFTWARE AND HARDWARE REQUIREMENTS

SOFTWARE REQUIREMENTS

❖ Operating system	:	Windows 7 Ultimate.
❖ Coding Language	:	Python.
❖ Designing	:	Html, css, javascript.
❖ Data Base	:	MySQL (XAMPP Server).

HARDWARE REQUIREMENTS

❖ Processor	:	i3 Processor
❖ Hard Disk	:	1 TB
❖ Ram	:	4 GB

1.4.4 SOFTWARE DEVELOPMENT ANALYSIS

PYTHON

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.

ARTIFICIAL INTELLIGENCE

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

1.4.5 PROJECT SYSTEM DESIGN

CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

1.4.6 PROJECT CODING

```
{% extends 'RUser/design.html' %}
{% block userblock %}
<link rel="icon" href="images/icon.png" type="image/x-icon" />
  <link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
    <link href="https://fonts.googleapis.com/css?family=Righteous" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Fredoka+One" rel="stylesheet">
    <style>
      body {background-color:#FFFFFF;}
      .container-fluid {padding:50px;}
      .container{background-color:white;padding:50px;}
      #title{font-family: 'Fredoka One', cursive;
}
      .text-uppercase{
      font-family: 'Righteous', cursive;
      }
      input,textarea{
      font-family:Aldrich;
      font-size:15px;
      }
    .style1 {
  color: #FF0000;
  font-weight: bold;
}
.style4 {color: #FFFF00; font-weight: bold; }
      </style>
  <body>
  <div class="container-fluid">
    <div class="container">
      <div class="row">
        <div class="col-md-5">
          <form role="form" method="POST" >
            {% csrf_token %}
            <fieldset>
              <p class="text-uppercase pull-center style1">CREATE YOUR OWN
TWEET !!! </p>
                   <hr>
                      {% csrf_token %}
                      <table>
                  <tr>
                    <td bgcolor="#FF0000"><span class="style4">User
Name</span></td>
                           <td><input type="text" name="uname" value="{{objc}}"
```

```
readonly></td>
                        </tr>
                  <tr>
                    <td bgcolor="#FF0000"><span class="style4">Tweet
Name</span></td>
                          <td><input type="text" name="tname" required></td>
                        </tr>
                  <tr>
                    <td bgcolor="#FF0000"><span class="style4">Uses</span></td>
                          <td><input type="text" name="uses" required></td>
                        </tr>
                        <tr>
                    <td bgcolor="#FF0000"><span
class="style4">Description</span></td>
                          <td><textarea name="tdesc" cols="40"
rows="4"></textarea></td>
                        </tr>
                        <tr>
                          <td><input type="submit" class="style1"></td>
                        </tr>
                 </table>
             </fieldset>
           </form>
         </div>
         <div class="col-md-2">
           <!-------null------>
         </div>
         </div>
      </div>
   </div>
{% endblock %}
```

1.4.7 PROJECT 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.

1.4.8 INPUT SCREENS

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.

1.4.9 OUTPUT SCREENS

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.

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.

1.4.10 CONCLUSIONS

To investigate the effect of depression detection, we propose machine learning technique as an efficient and scalable method. To detect depression we are using SVM (support vector machine) algorithm which analyse users post and give result as negative or positive. If users express depression words in post then SVM detect it as a negative post else positive post. We tried to identify the presence of depression in Reddit social media; and searched for affective performance increase solutions of depression detection. We characterized a closer connection between depression and a language usage by applying NLP and text classification techniques. We identified a lexicon of words more common among the depressed accounts. According to our findings, the language predictors of depression contained the words related to preoccupation with themselves, feelings of sadness, anxiety, anger, hostility or suicidal thoughts, with a greater emphasis on the present and future.

2. LITERATURE SURVEY

2.1 SURVEY ON BACKGROUND

Conducting large health population studies is expensive. For instance, collecting field information about the efficacy of health campaigns or the impact of a disease may require the involvement of many health providers over an extended period of time and sometimes may not reach the target population. In fact, due to the aforementioned difficulties, health-related population statistics may be unavailable or lag by several years. Recently, social media networks have emerged as a source of sensory data for various aspects of social behavior. This source of information is used to drive marketing campaigns, conduct threat analysis and profile groups of individuals among numerous other applications. However, these applications are usually limited to specific case studies and do not provide a systematic approach to translating social media data into knowledge. In this paper, we propose a framework that can extract knowledge from social media networks in support of large scale health studies. The framework consists of an automated workflow designed to collect data from social media platforms, filter the data based on geographical criteria, and extract information relevant to a target hypothesis. The framework is demonstrated in the case of mortality and incidence of three chronic diseases, namely asthma, cancer, and diabetes. Twitter data is extracted over the period 2010 to 2015 for each target geographical region and classified based on its relevance to each of the aforementioned diseases. Due to the large number of extracted records, a simple random sampling approach is used to support the supervised training and testing of the classifier in the framework. Despite the limited number of records used for the training of the classifiers as a result of this approach, high classification accuracies are achieved for all three diseases.

2.2 CONCLUSIONS ON SURVEY

Sentimental Analysis is reference to the task of Natural Language Processing to determine whether a text contains subjective information and what information it expresses i.e., whether the attitude behind the text is positive, negative or neutral. This paper focuses on the several machine learning techniques which are used in analyzing the sentiments and in opinion mining. Sentimental analysis with the blend of machine learning could be useful in predicting the product reviews and consumer attitude towards to newly launched product. This paper presents a detail survey of various machine learning techniques and then compared with their accuracy, advantages and limitations of each technique. On comparing we get 85% of accuracy by using supervised machine learning technique which is higher than that of unsupervised learning techniques.

Micro blogging websites like Twitter and Face book, in this new era, is loaded with opinions and data. One of the most widely used micro-blogging site, Twitter, is where people share their ideas in the form of tweets and therefore it becomes one of the best sources for sentimental analysis. Opinions can be widely grouped into three categories good for positive, bad for negative and neutral and the process of analyzing differences of opinions and grouping them in all these categories is known as Sentiment Analysis. Data mining is basically used to uncover relevant information from web pages especially from the social networking sites. Merging data mining with other fields like text mining, NLP and computational intelligence we are able to classify tweets as good, bad or neutral. The main emphasis of this research is on the classification of emotions of tweets' data gathered from Twitter. In the past, researchers were using existing machine learning techniques for sentiment analysis but the results showed that existing machine learning techniques were not providing better results of sentiment classification. In order to improve classification results in the domain of sentiment analysis, we are using ensemble machine learning techniques for increasing the efficiency and reliability of proposed approach. For the same, we are merging Support Vector Machine with Decision Tree and experimental results prove that our proposed approach is providing better classification results in terms of f-measure and accuracy in contrast to individual classifiers.

3. SOFTWARE AND HARDWARE REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS

❖ Operating system	:	Windows 7 Ultimate.
❖ Coding Language	:	Python.
❖ Designing	:	Html, css, javascript.
❖ Data Base	:	MySQL (XAMPP Server).

3.2 HARDWARE REQUIREMENTS

❖ Processor	:	i3 Processor
❖ Hard Disk	:	1 TB
❖ Ram	:	4 GB

Functional Requirements

- Graphical User interface with the User.

Non Functional Requirements

- **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.
- **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.
- **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.
- **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.
- **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.
- **Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life.

4. SOFTWARE DEVELOPMENT ANALYSIS

4.1 OVERVIEW OF PROBLEM

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

4.2 DEFINE THE PROBLEM

Stress has become an embedded part of our life, being stressed by our financial worries, our job, etc. Stress causes physical illnesses, such as heart attacks, arthritis, and chronic headaches or psychological diseases like mental illness, anger, anxiety, and depression. There are several research works coming up to resolve the limitations on measuring, analyzing and identifying the human stress levels Amongst the many stress monitoring methods the more reliable method by sharing stress related post.

4.2.1 SYSTEM ARCHITECTURE

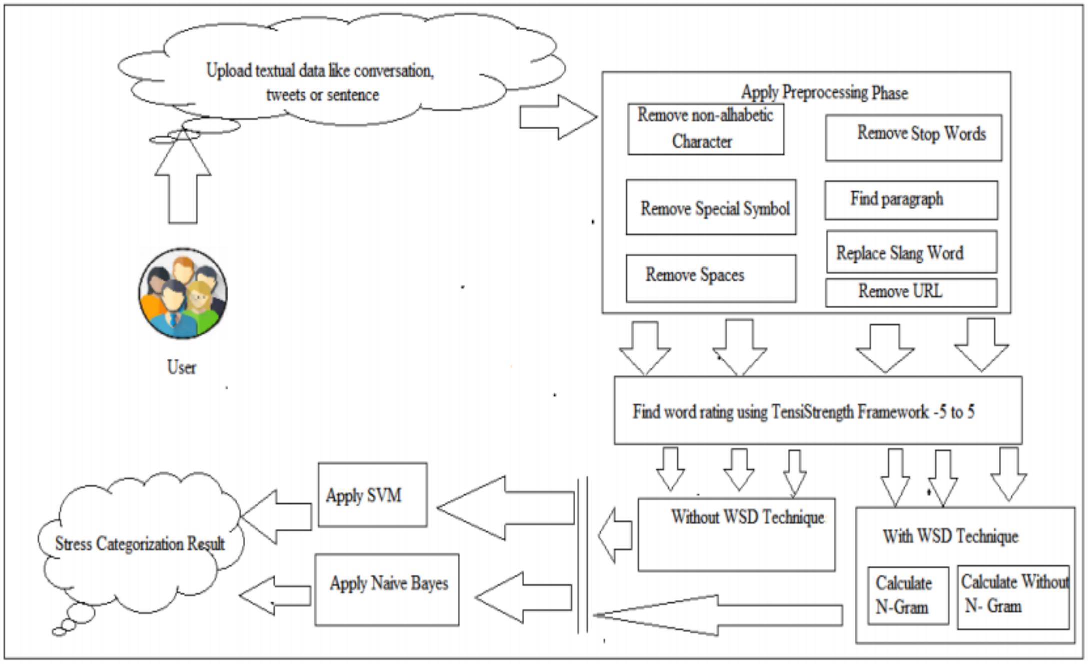

Upload textual data like conversation, tweets or sentence
Apply Preprocessing Phase
Remove non-alhabetic Character
Remove Stop Words
Remove Special Symbol
Find paragraph
Replace Slang Word
Remove Spaces
Remove URL
User
Find word rating using TensiStrength Framework -5 to 5
Without WSD Technique
With WSD Technique
Calculate N-Gram
Calculate Without N- Gram
Apply SVM
Apply Naive Bayes
Stress Categorization Result

Fig 4.1: Architecture 1

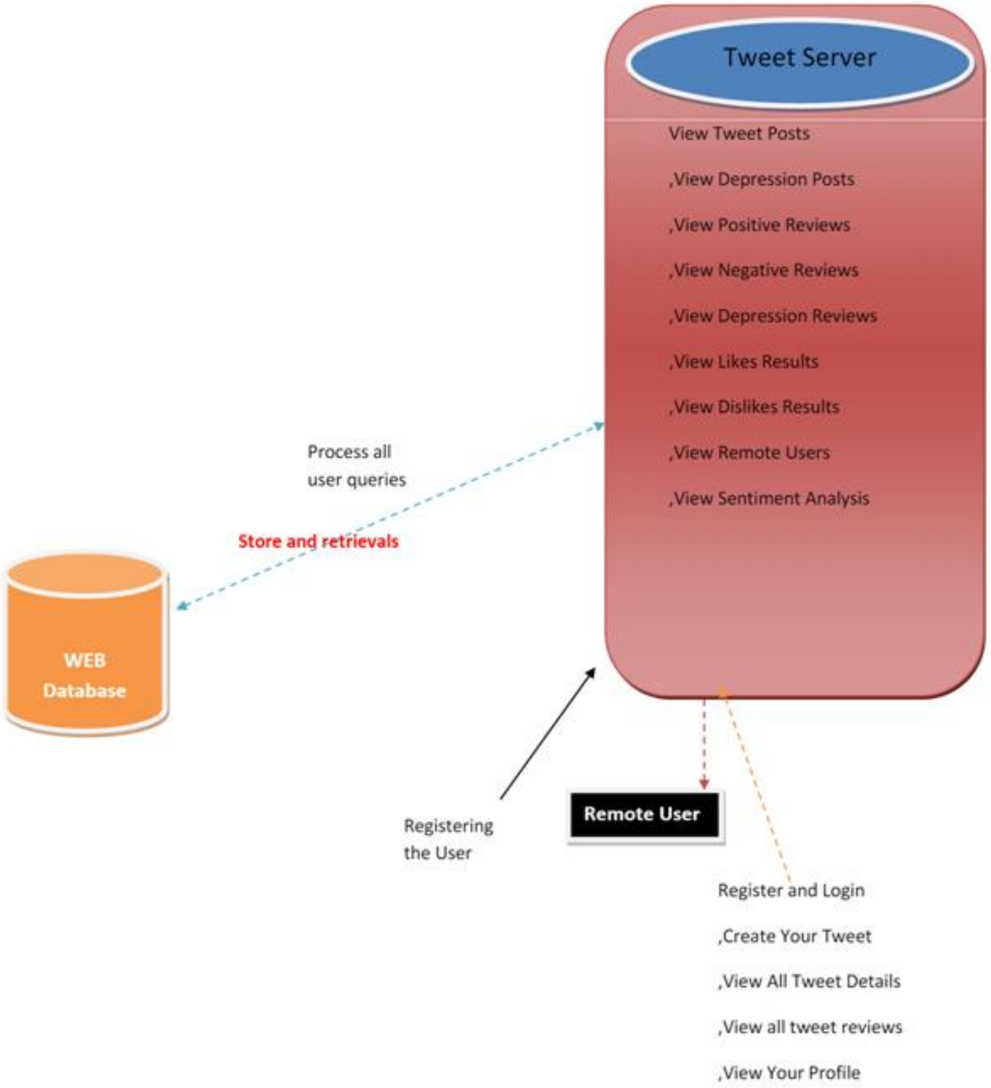


Fig 4.2: Architecture 2

4.3 MODULES OVERVIEW

Admin Module: In this module, the Admin has to login by using valid user name and password. After login successful he can do some operations such View Tweet Posts, View Depression Posts, View Positive Reviews, View Negative Reviews, View Depression Reviews, View Likes Results, View Dislikes Results, View Remote Users, View Sentiment Analysis.

User Module: In this module, there are n numbers of users are present. User should register before doing some operations. After registration successful he has to wait for admin to authorize him and after admin authorized him. He can login by using authorized user name and password. Login successful he will do some operations like Create Your Tweet, View All Tweet Details, View all tweet reviews, View Your Profile.

5. PROJECT SYSTEM DESIGN

5.1 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

5.1.1 CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

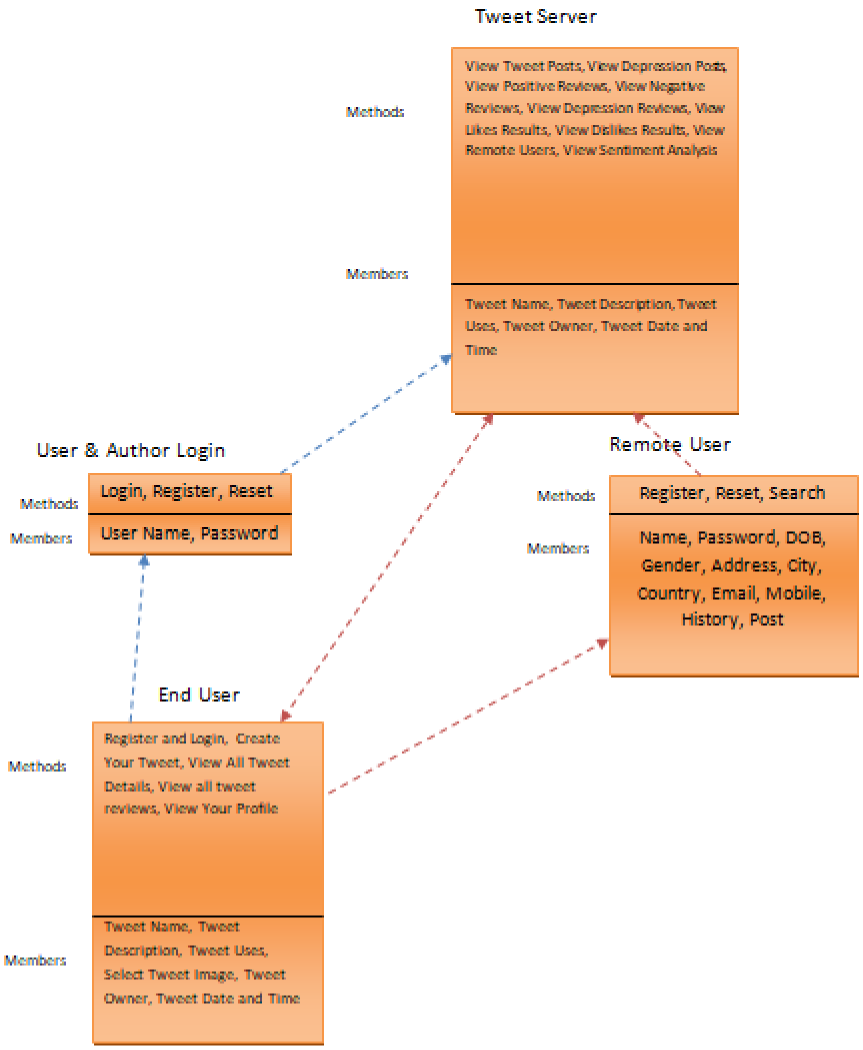

Tweet Server
Methods
View Tweet Posts, View Depression Posts, View Positive Reviews, View Negative Reviews, View Depression Reviews, View Likes Results, View Dislikes Results, View Remote Users, View Sentiment Analysis
Members
Tweet Name, Tweet Description, Tweet Uses, Tweet Owner, Tweet Date and Time
User & Author Login
Methods
Login, Register, Reset
Members
User Name, Password
Remote User
Methods
Register, Reset, Search
Members
Name, Password, DOB, Gender, Address, City, Country, Email, Mobile, History, Post
End User
Methods
Register and Login, Create Your Tweet, View All Tweet Details, View all tweet reviews, View Your Profile
Members
Tweet Name, Tweet Description, Tweet Uses, Select Tweet Image, Tweet Owner, Tweet Date and Time

Fig 5.1.1: Class Diagram

5.1.2 USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

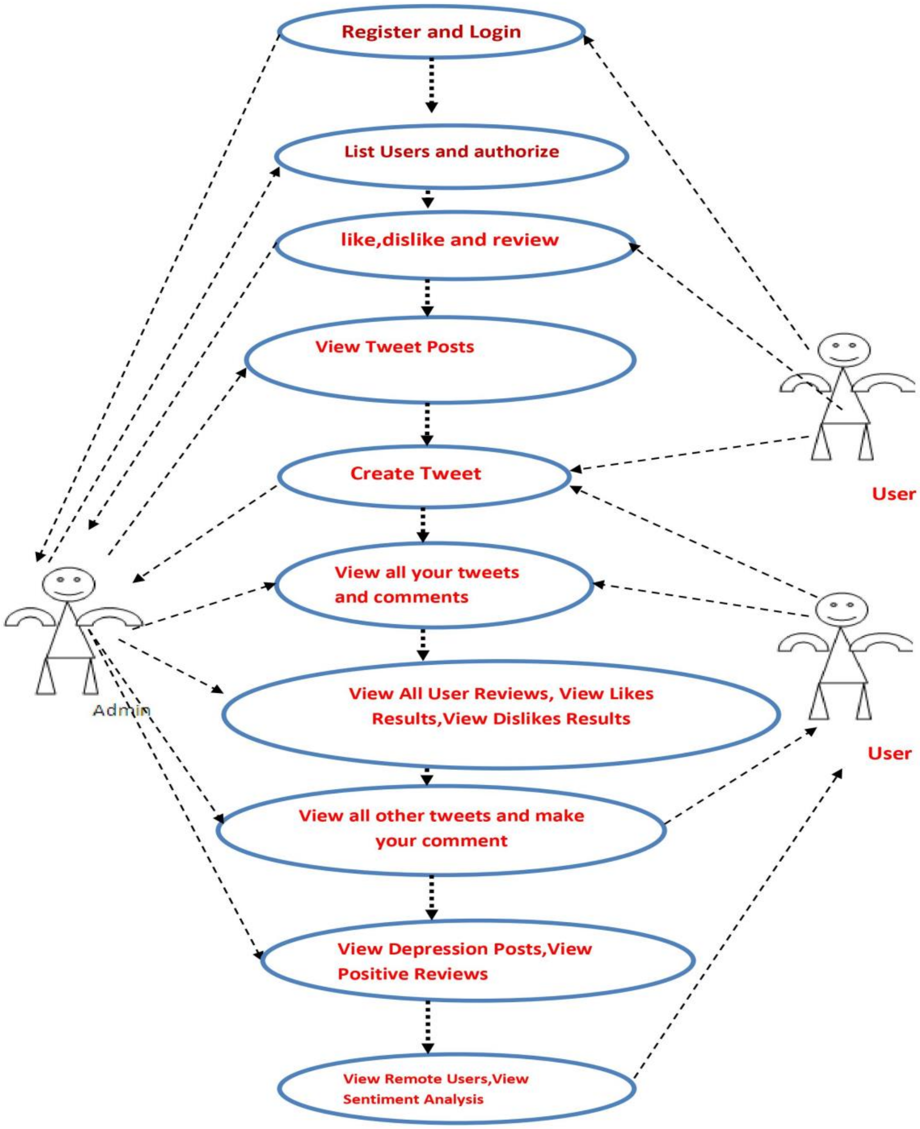


Fig 5.1.2: Use Case Diagram

5.1.3 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

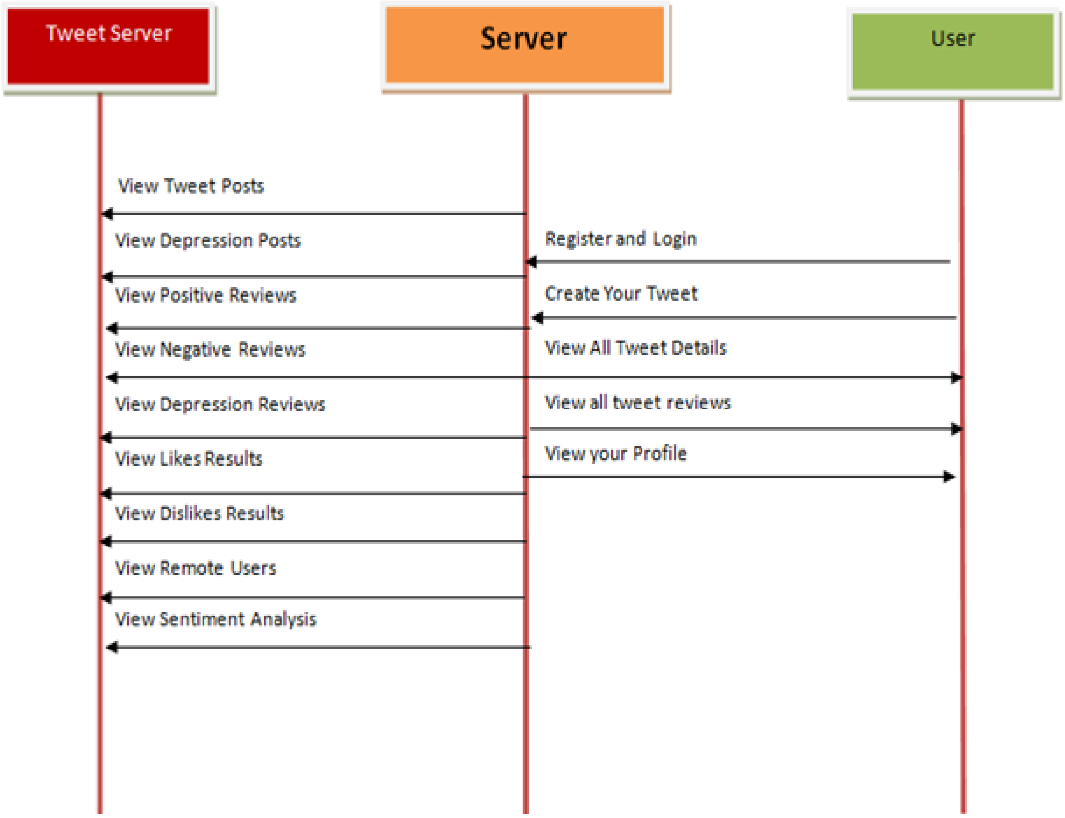


Fig 5.1.3: Sequence Diagram

5.1.4 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

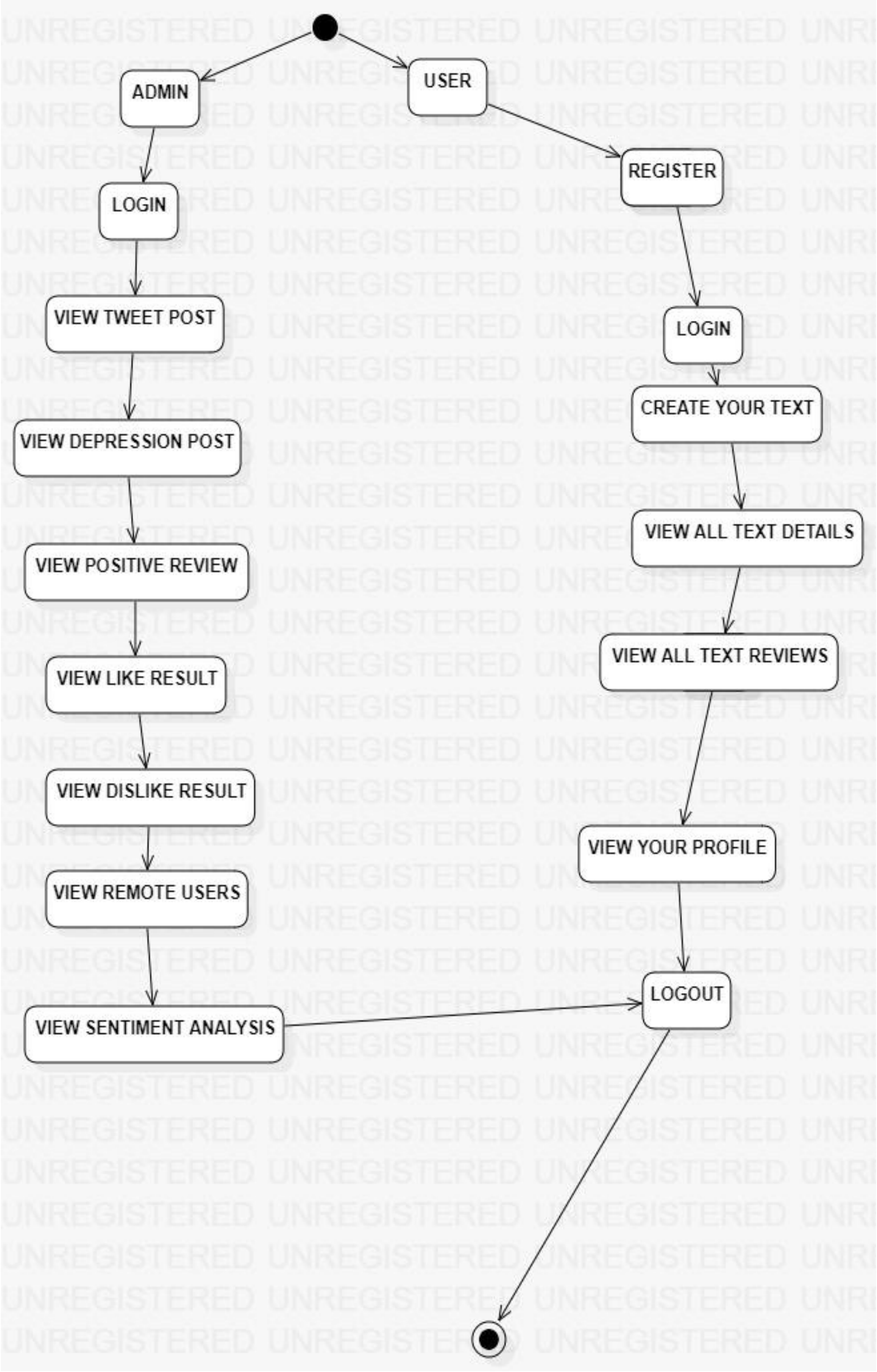


Fig 5.1.4: Activity Diagram

5.1.5 DEPLOYMENT DIAGRAM

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub-nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

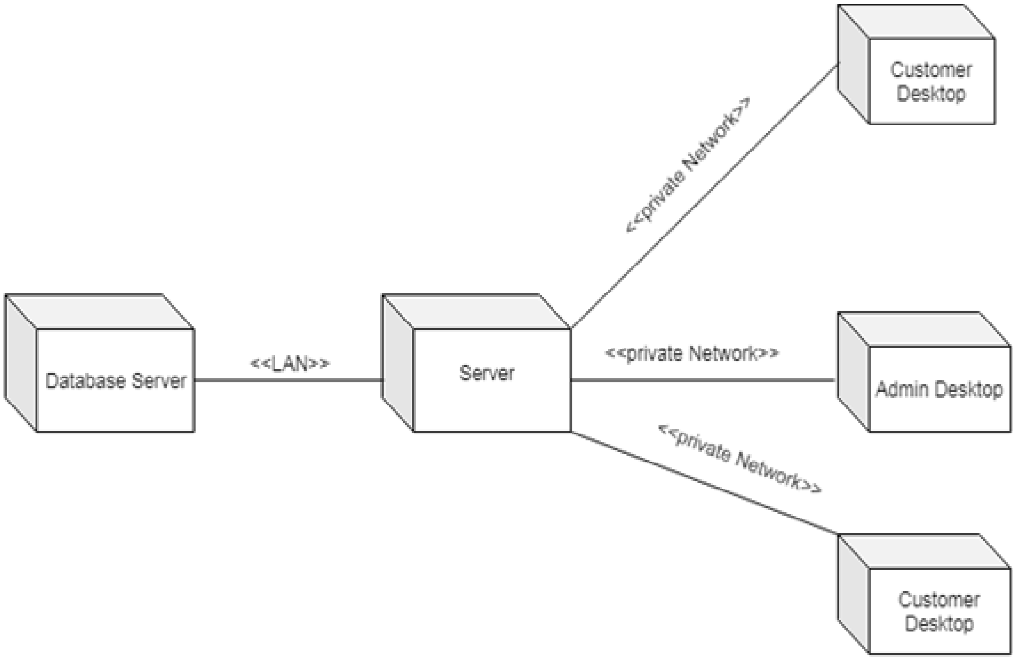


Fig 5.1.5: Deployment Diagram

5.1.6 PACKAGE DIAGRAM

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, package able element, dependency, element import, package import, package merge.

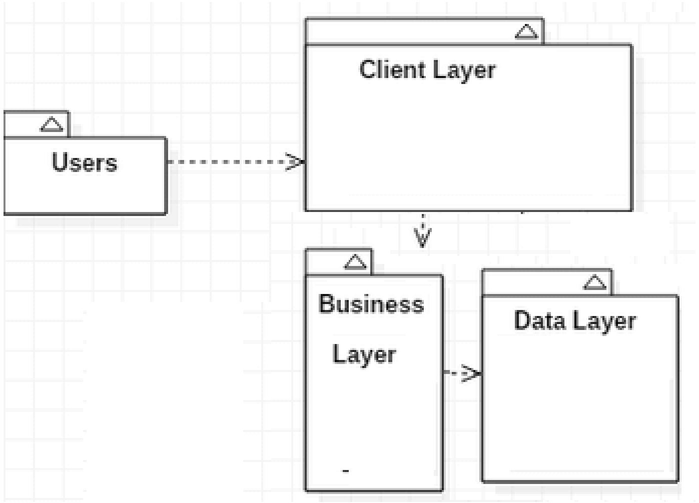


Fig 5.1.6: Package Diagram

5.1.7 PROFILE DIAGRAM

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connectors and their components.

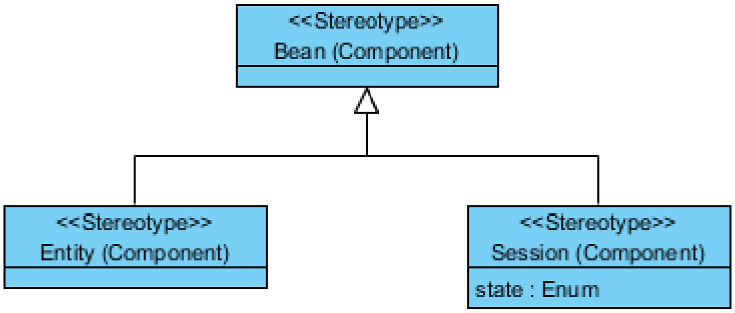


Fig 5.1.7: Profile Diagram

5.1.8 FLOW CHART DIAGRAM

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

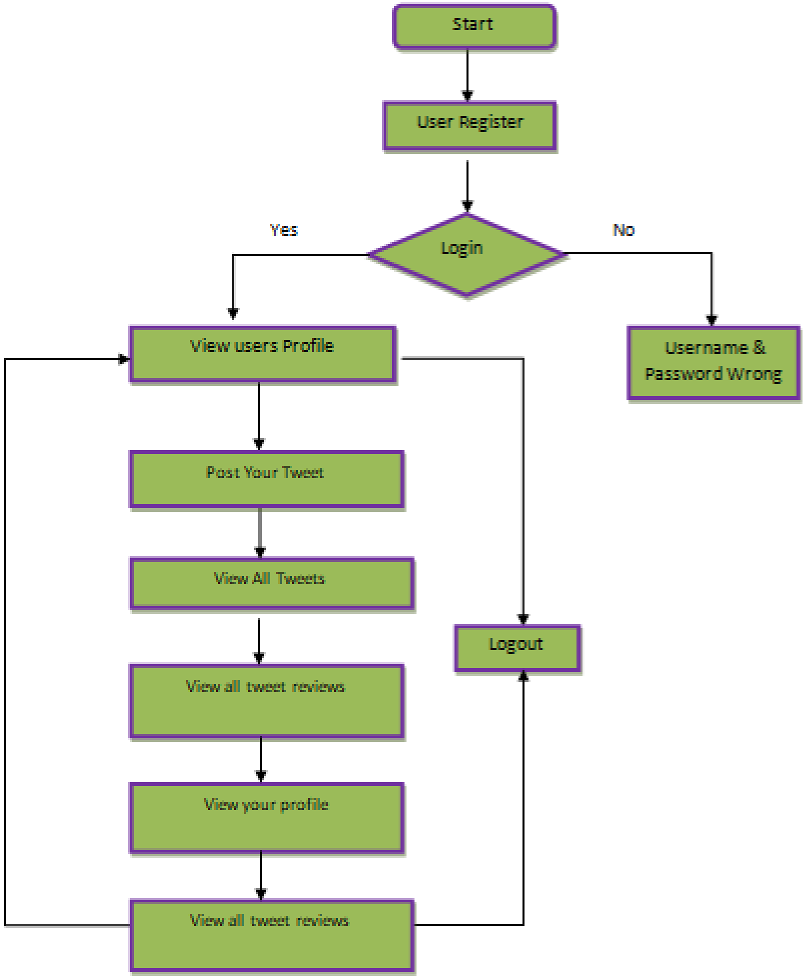


Fig 5.1.8: Flowchart Diagram for User

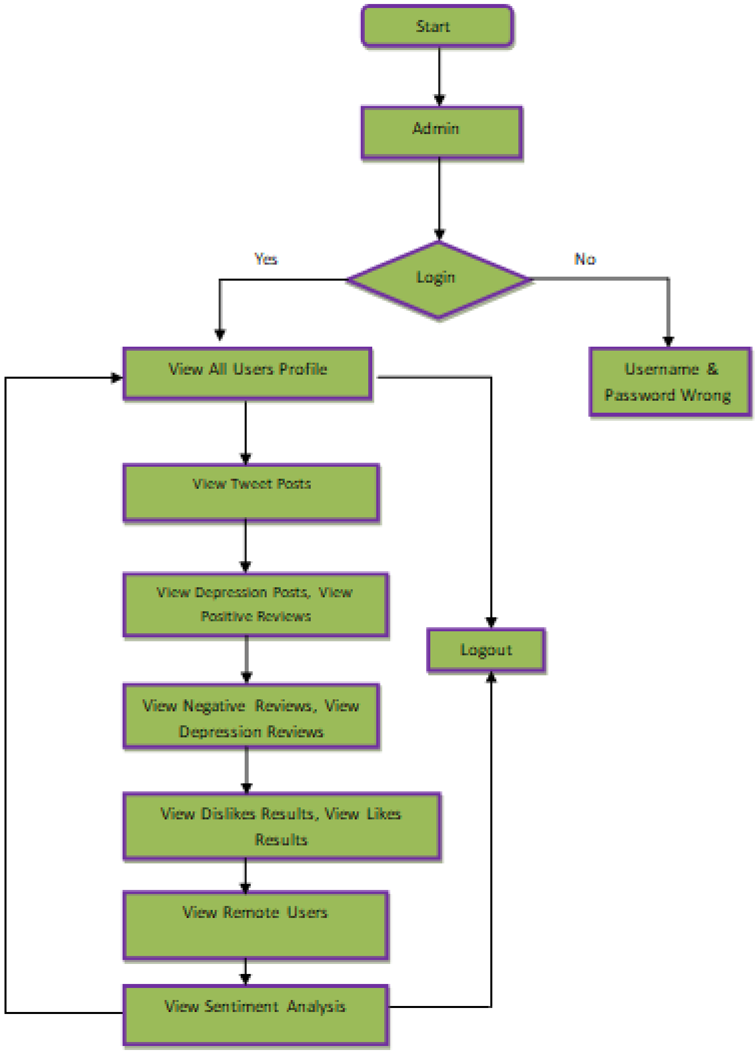

Start
Admin
Login
Yes
No
View All Users Profile
Username & Password Wrong
View Tweet Posts
View Depression Posts, View Positive Reviews
Logout
View Negative Reviews, View Depression Reviews
View Dislikes Results, View Likes Results
View Remote Users
View Sentiment Analysis

Fig 5.1.9: Flowchart Diagram for Admin

6. PROJECT CODING

6.1 TECHNOLOGY

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. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython 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

What is Python

Python is a popular programming language. It was created by Guido van Rossum, 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.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use

curly-brackets for this purpose.

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 Quickstart

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()
```

Virtual Environments and Packages

Introduction

Python applications will often use packages and modules that don't come as part of the standard library. Applications will sometimes need a specific version of a library, because the application may require that a particular bug has been fixed or the application may be written using an obsolete version of the library's interface.

This means it may not be possible for one Python installation to meet the requirements of every application. If application A needs version 1.0 of a particular module but application B needs version 2.0, then the requirements are in conflict and installing either version 1.0 or 2.0 will leave one application unable to run.

The solution for this problem is to create a virtual environment, a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.

Different applications can then use different virtual environments. To resolve the earlier example of conflicting requirements, application A can have its own virtual environment with version 1.0 installed while application B has another virtual environment with version 2.0. If application B requires a library be upgraded to version 3.0, this will not affect application A's environment.

Creating Virtual Environments

The module used to create and manage virtual environments is called venv. venv will usually install the most recent version of Python that you have available. If you have multiple versions of Python on your system, you can select a specific Python version by running python3 or whichever version you want.

To create a virtual environment, decide upon a directory where you want to place it, and run the venv module as a script with the directory path:

```
python3 -m venv tutorial-env
```

This will create the tutorial-env directory if it doesn't exist, and also create directories inside it containing a copy of the Python interpreter, the standard library, and various supporting files.

A common directory location for a virtual environment is .venv. This name keeps the directory typically hidden in your shell and thus out of the way while giving it a name that explains why the directory exists. It also prevents clashing with .env environment variable definition files that some tooling supports.

Once you've created a virtual environment, you may activate it.

On Windows, run:

```
tutorial-env\Scripts\activate.bat
```

On Unix or MacOS, run:

```
source tutorial-env/bin/activate
```

(This script is written for the bash shell. If you use the csh or fish shells, there are alternate activate.csh and activate.fish scripts you should use instead.)

Activating the virtual environment will change your shell’s prompt to show what virtual environment you’re using, and modify the environment so that running python will get you that particular version and installation of Python. For example:

```
$ source ~/envs/tutorial-env/bin/activate

(tutorial-env) $ python

Python 3.5.1 (default, May  6 2016, 10:59:36)

  ...

>>> import sys

>>>sys.path

['', '/usr/local/lib/python35.zip', ...,

'~/envs/tutorial-env/lib/python3.5/site-packages']

>>>
```

Managing Packages with pip

You can install, upgrade, and remove packages using a program called pip. By default pip will install packages from the Python Package Index, <https://pypi.org>. You can browse the Python Package Index by going to it in your web browser, or you can use pip’s limited search feature:

```
(tutorial-env) $ pip search astronomy

skyfield            - Elegant astronomy for Python

gary                - Galactic astronomy and gravitational dynamics.

novas               - The United States Naval Observatory NOVAS astronomy library

astroobs            - Provides astronomy ephemeris to plan telescope observations

PyAstronomy         - A collection of astronomy related tools for Python.
```

Artificial Intelligence

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.

Trending AI Articles:

1. Cheat Sheets for AI, Neural Networks, Machine Learning, Deep Learning & Big Data
2. Data Science Simplified Part 1: Principles and Process
3. Getting Started with Building Realtime API Infrastructure
4. AI & NLP Workshop

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

IBM Watson

“Watson” is an IBM supercomputer that combines Artificial Intelligence (AI) and complex inquisitive programming for ideal execution as a “question answering” machine. The supercomputer is named for IBM’s founder, Thomas J. Watson.

IBM Watson is at the forefront of the new era of computing. At the point when IBM Watson made, IBM communicated that “more than 100 particular techniques are used to inspect perceive sources, find and make theories, find and score affirm, and combination and rank speculations.” recently, the Watson limits have been expanded and the way by which Watson works has been changed to abuse new sending models (Watson on IBM Cloud) and propelled machine learning capacities and upgraded hardware open to architects and authorities. It isn’t any longer completely a request answering figuring system arranged from Q&A joins yet can now ‘see’, ‘hear’, ‘read’, ‘talk’, ‘taste’, ‘translate’, ‘learn’ and ‘endorse’.

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 a 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.

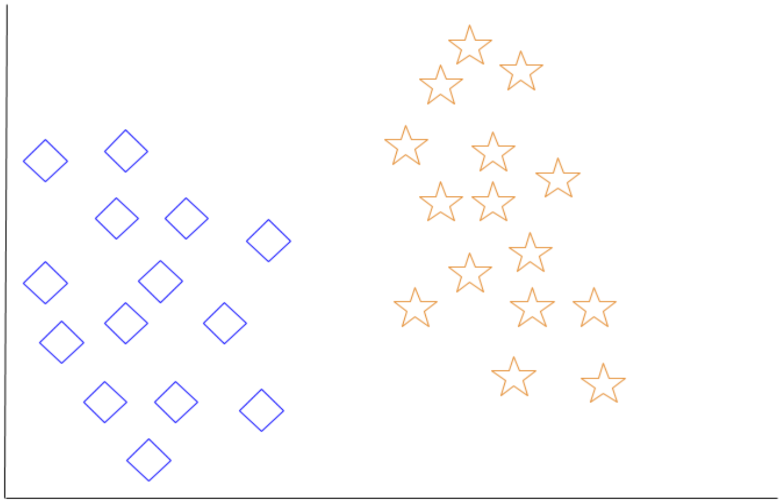
Approaches to machine learning are continuously being developed. For our purposes, we’ll go through a few of the popular approaches that are being used in machine learning at the time of writing.

K-nearest neighbor

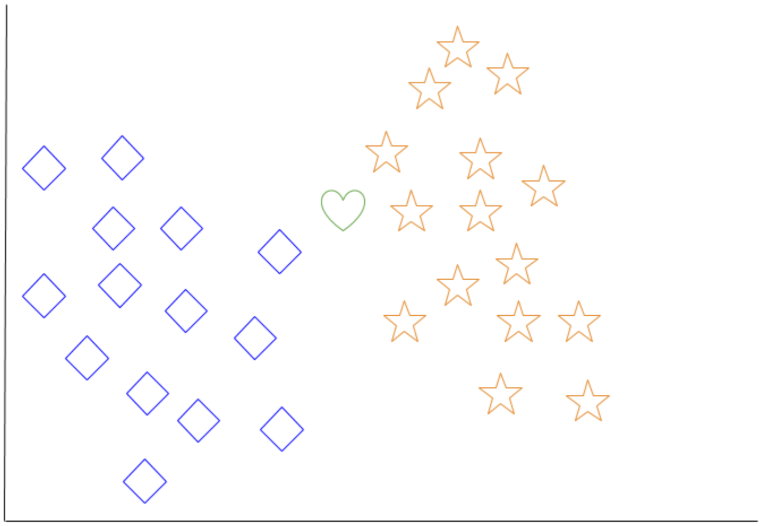
The k-nearest neighbor algorithm is a pattern recognition model that can be used for classification as well as regression. Often abbreviated as k-NN, the **k** in k-nearest neighbor is a positive integer, which is typically small. In either classification or regression, the input will consist of the k closest training examples within a space.

We will focus on K-NN classification. In this method, the output is class membership. This will assign a new object to the class most common among its k nearest neighbors. In the case of k = 1, the object is assigned to the class of the single nearest neighbor.

Let’s look at an example of k-nearest neighbor. In the diagram below, there are blue diamond objects and orange star objects. These belong to two separate classes: the diamond class and the star class.

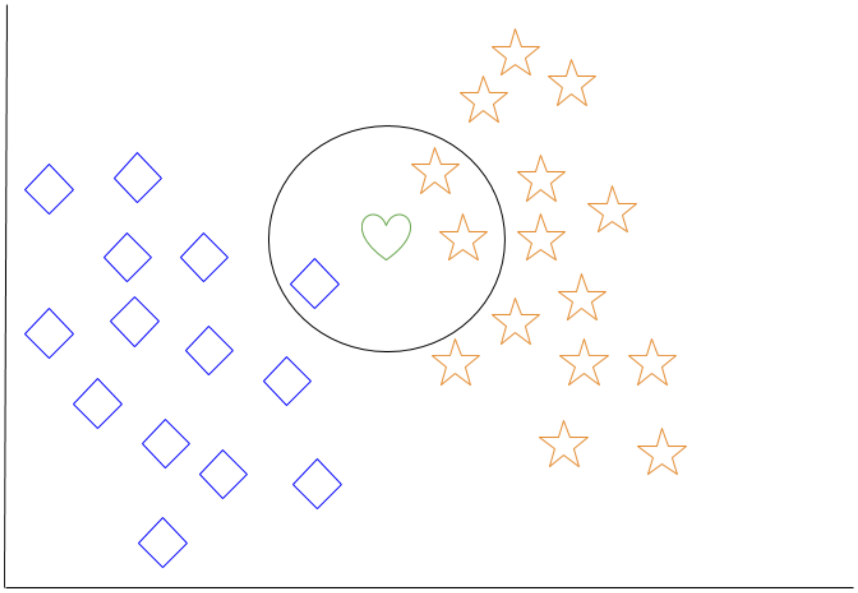


When a new object is added to the space — in this case a green heart — we will want the machine learning algorithm to classify the heart to a certain class.



When we choose k = 3, the algorithm will find the three nearest neighbors of the green heart in order to classify it to either the diamond class or the star class.

In our diagram, the three nearest neighbors of the green heart are one diamond and two stars. Therefore, the algorithm will classify the heart with the star class.



Among the most basic of machine learning algorithms, k-nearest neighbor is considered to be a type of "lazy learning" as generalization beyond the training data does not occur until a query is made to the system.

Decision Tree Learning

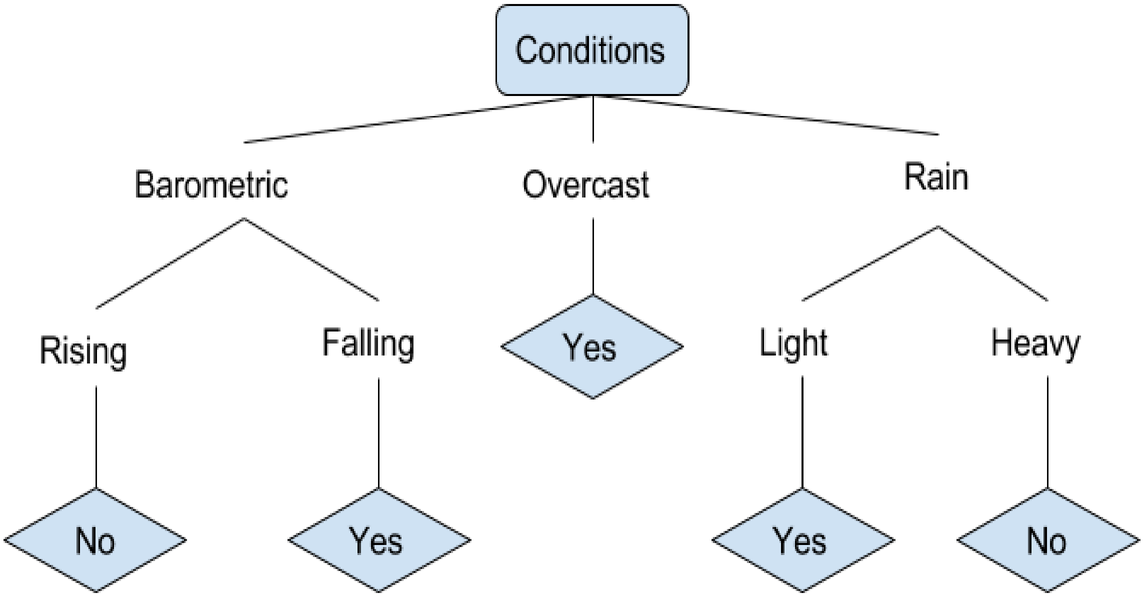
For general use, decision trees are employed to visually represent decisions and show or inform decision making. When working with machine learning and data mining, decision trees are used as a predictive model. These models map observations about data to conclusions about the data's target value.

The goal of decision tree learning is to create a model that will predict the value of a target based on input variables.

In the predictive model, the data's attributes that are determined through observation are represented by the branches, while the conclusions about the data's target value are represented in the leaves.

When "learning" a tree, the source data is divided into subsets based on an attribute value test, which is repeated on each of the derived subsets recursively. Once the subset at a node has the equivalent value as its target value has, the recursion process will be complete.

Let's look at an example of various conditions that can determine whether or not someone should go fishing. This includes weather conditions as well as barometric pressure conditions.



In the simplified decision tree above, an example is classified by sorting it through the tree to the appropriate leaf node. This then returns the classification associated with the particular leaf, which in this case is either a Yes or a No. The tree classifies a day's conditions based on whether or not it is suitable for going fishing.

A true classification tree data set would have a lot more features than what is outlined above, but relationships should be straightforward to determine. When working with decision tree learning, several determinations need to be made, including what features to choose, what conditions to use for splitting, and understanding when the decision tree has reached a clear ending.

Deep Learning

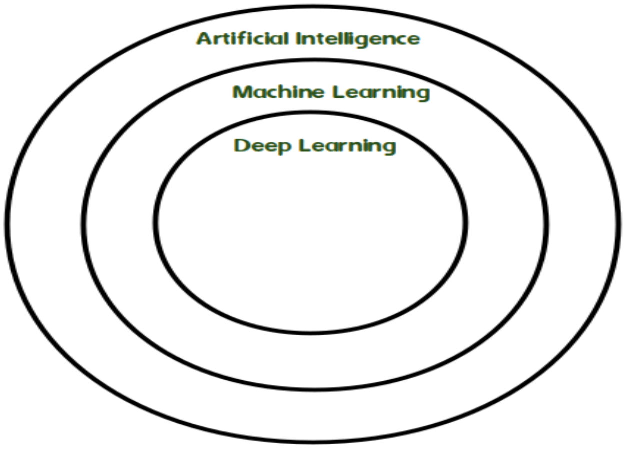
Deep learning is a branch of machine learning which is completely based on artificial neural networks, as neural network is going to mimic the human brain so deep learning is also a kind of mimic of human brain. In deep learning, we don't need to explicitly program everything. The concept of deep learning is not new. It has been around for a couple of years now.

It’s on hype nowadays because earlier we did not have that much processing power and a lot of data. As in the last 20 years, the processing power increases exponentially, deep learning and machine learning came in the picture.
A formal definition of deep learning is- neurons

Deep learning is a particular kind of machine learning that achieves great power and flexibility by learning to represent the world as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations computed in terms of less abstract ones.

In human brain approximately 100 billion neurons all together this is a picture of an individual neuron and each neuron is connected through thousand of their neighbours.

The question here is how do we recreate these neurons in a computer. So, we create an artificial structure called an artificial neural net where we have nodes or neurons. We have some neurons for input value and some for output value and in between, there may be lots of neurons interconnected in the hidden layer.



6.2 CODING

Login.html:

```
<link href="//maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
rel="stylesheet" id="bootstrap-css">
<!DOCTYPE html>
<html lang="en">
  <title>Login</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
  <head>
<link rel="icon" href="images/icon.png" type="image/x-icon" />
  <link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
    <link href="https://fonts.googleapis.com/css?family=Righteous" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Fredoka+One" rel="stylesheet">
    <style>
      body {
      }
      .container-fluid {padding:50px;}
      .container{background-color:white;padding:50px;   }
      #title{font-family: 'Fredoka One', cursive;}
      .text-uppercase{
      font-family: 'Righteous', cursive;
      }
    .style1 {color: #FF0000}
    .style4 {color: #FF0000; font-weight: bold; }
</style>
 </head>
  <body>
       <div class="container-fluid">
    <div class="container">
      <h2 class="style1 text-center" id="title"><strong>Online Depression Detection
Application</strong></h2>
       <p class="text-center">
       <span class="style4"><small id="passwordHelpInline" class="text-
muted">Natural language processing, machine learning, Reddit, social networks,
depression. </small></span> </p>
      <hr>
      <div class="row">
        <div class="col-md-5">
          <form role="form" method="POST" >
            {% csrf_token %}
            <fieldset>
              <p class="text-uppercase pull-center">&nbsp;</p>
           </fieldset>
          </form>
        </div>

        <div class="col-md-2">
          <!-------null------>
        </div>

        <div class="col-md-5">
            <form method="POST" role="form">
 {% csrf_token %}


            <fieldset>
              <p class="text-uppercase"> Login Using Your Account: </p>
              <div class="form-group">
       <input type="text" name="username"  class="form-control input-lg"
placeholder="User Name" required>
```

```
              </div>
       <div class="form-group">
                <input type="password" name="password"  class="form-control input-
lg" placeholder="Password" required>
              </div>
              <div>
                <input type="submit" name="submit1"  class="btn btn-md"
value="sign_in">
              </div></br>

              <p class="text-uppercase"> Login Using Your Account: </p>
              <div>
              <button class="btn btn-lg "><a href="{% url 'tweetserverlogin'
%}">TWEET SERVER</a></button>
                     <button class="btn btn-lg "><a href="{% url 'Register1'
%}">REGISTER</a></button>
              </div>
            </fieldset>
        </form>
        </div>
      </div>
    </div>
  </div>
  </body>
</html>
```

User views.py:

```
from django.db.models import Count
from django.shortcuts import render, redirect, get_object_or_404
import datetime

# Create your views here.
```

```
from Remote_User.models import review_Model,ClientRegister_Model,tweets_Model



def login(request):


    if request.method == "POST" and 'submit1' in request.POST:

        username = request.POST.get('username')
        password = request.POST.get('password')
        try:

            enter = ClientRegister_Model.objects.get(username=username,
password=password)
            request.session["userid"] = enter.id
            return redirect('CreateTweet')
        except:
            pass

    return render(request,'RUser/login.html')




def Register1(request):

    if request.method == "POST":
        username = request.POST.get('username')
        email = request.POST.get('email')
        password = request.POST.get('password')
        phoneno = request.POST.get('phoneno')
        country = request.POST.get('country')
        state = request.POST.get('state')
        city = request.POST.get('city')
        ClientRegister_Model.objects.create(username=username, email=email,
password=password, phoneno=phoneno,
                                            country=country, state=state, city=city)

        return render(request, 'RUser/Register1.html')
    else:

        return render(request,'RUser/Register1.html')



def ViewYourProfile(request):
    userid = request.session['userid']
    obj = ClientRegister_Model.objects.get(id= userid)
    return render(request,'RUser/ViewYourProfile.html',{'object':obj})

def Review(request,pk):
    userid = request.session['userid']
    userObj = ClientRegister_Model.objects.get(id=userid)
    username = userObj.username

    objs = tweets_Model.objects.get(id=pk)
    tname = objs.names

    datetime_object = datetime.datetime.now()

    result = ''
    pos = []
    neg = []
    oth = []
    se = 'se'
    if request.method == "POST":
        uname = request.POST.get('uname')
        tname1 = request.POST.get('tname')
        suggestion1 = request.POST.get('suggestion')
        cmd = request.POST.get('review')



        if '#' in cmd:
            startingpoint = cmd.find('#')
            a = cmd[startingpoint:]
            endingPoint = a.find(' ')
            title = a[0:endingPoint]
            result = title[1:]
        # return redirect('')

        for f in cmd.split():
            if f in ('good', 'nice', 'better', 'best', 'excellent', 'extraordinary', 'happy', 'won',
'love', 'great'):
                se = 'Positive'
            elif f in ('worst', 'waste', 'poor', 'error', 'imporve', 'bad'):
                se = 'Negative'

            elif f in ('alone', 'break', 'blame', 'depressed', 'deserveunhappy', 'die', 'escap',
'feelalone', 'feltpain', 'no job', 'tooworried'):
                se = 'Depression'
        review_Model.objects.create(uname=uname ,
ureview=cmd,sanalysis=se,dt=datetime_object,tname=tname1 ,suggestion=suggestion1)

    return render(request,'RUser/Review.html', {'objc':username,'objc1':tname,'result':
result, 'se': se})
```

```
def CreateTweet(request):
    userid = request.session['userid']
    userObj = ClientRegister_Model.objects.get(id=userid)
    userid = userObj.username

    result = ''
    pos = []
    neg = []
    oth = []
    se = 'se'
    uname=''
    if request.method == "POST":
        uname = request.POST.get('uname')
        tname = request.POST.get('tname')
        uses = request.POST.get('uses')
        cmd = request.POST.get('tdesc')



        if '#' in cmd:
            startingpoint = cmd.find('#')
            a = cmd[startingpoint:]
            endingPoint = a.find(' ')
            title = a[0:endingPoint]
            result = title[1:]
        # return redirect('')

        for f in cmd.split():
            if f in ('good', 'nice', 'better', 'best', 'excellent', 'extraordinary', 'happy', 'won',
'love', 'greate'):
                se = 'Positive'
            elif f in ('worst', 'waste', 'poor', 'error', 'imporve', 'bad'):
                se = 'Negative'

            elif f in ('alone', 'break', 'blame', 'depressed', 'deserveunhappy', 'die', 'escap', 'feel
alone', 'felt pain', 'no job', 'too worried'):
                se = 'Depression'



        tweets_Model.objects.create(userId=userObj,uname=uname ,names=tname
,uses=uses, tdesc=cmd, topics=result, sanalysis=se,
                                    senderstatus='process')

    return render(request,'RUser/CreateTweet.html', {'objc':userid,'result': result, 'se': se})

def ViewAllTweets(request):

    obj = tweets_Model.objects.all()
    return render(request, 'RUser/ViewAllTweets.html', {'list_objects': obj})


def Viewreviews(request):

    obj = review_Model.objects.all()

    return render(request,'RUser/Viewreviews.html',{'list_objects': obj})




def ratings(request,pk):
    vott1, vott, neg = 0, 0, 0
    objs = tweets_Model.objects.get(id=pk)
    unid = objs.id
    vot_count = tweets_Model.objects.all().filter(id=unid)
    for t in vot_count:
        vott = t.ratings
        vott1 = vott + 1
        obj = get_object_or_404(tweets_Model, id=unid)
        obj.ratings = vott1
        obj.save(update_fields=["ratings"])
        return redirect('ViewAllTweets')

    return render(request,'RUser/ratings.html',{'objs':vott1})



def dislikes(request,pk):
    vott1, vott, neg = 0, 0, 0
    objs = tweets_Model.objects.get(id=pk)
    unid = objs.id
    vot_count = tweets_Model.objects.all().filter(id=unid)
    for t in vot_count:
        vott = t.dislikes
        vott1 = vott - 1
        obj = get_object_or_404(tweets_Model, id=unid)
        obj.dislikes = vott1
        obj.save(update_fields=["dislikes"])
        return redirect('ViewAllTweets')
    return render(request,'RUser/dislikes.html',{'objs':vott1})
```

Admin views.py:

```
from django.db.models import  Count, Avg
from django.shortcuts import render, redirect
from django.db.models import Count

# Create your views here.
from Remote_User.models import tweets_Model,ClientRegister_Model,review_Model



def tweetserverlogin(request):
    if request.method  == "POST":
        admin = request.POST.get('admin')
        password = request.POST.get('password')
        if admin == "Server" and password =="Server":
            return redirect('viewallusers')



    return render(request,'TServer/tweetserverlogin.html')

def viewtreandingquestions(request,chart_type):
    dd = {}
    pos,neu,neg =0,0,0
    poss=None
    topic =
tweets_Model.objects.values('ratings').annotate(dcount=Count('ratings')).order_by('-
dcount')
    for t in topic:
        topics=t['ratings']

pos_count=tweets_Model.objects.filter(topics=topics).values('names').annotate(topiccou
nt=Count('ratings'))
        poss=pos_count
        for pp in pos_count:
            senti= pp['names']
            if senti == 'positive':
                pos= pp['topiccount']
            elif senti == 'negative':
                neg = pp['topiccount']
            elif senti == 'nutral':
                neu = pp['topiccount']
        dd[topics]=[pos,neg,neu]
    return
render(request,'TServer/viewtreandingquestions.html',{'object':topic,'dd':dd,'chart_type':
chart_type})

def View_Positive_reviews(request):

    rtype='Positive'
    #obj = review_Model.objects.all()

    obj = review_Model.objects.all().filter(sanalysis=rtype)

    return render(request,'TServer/View_Positive_reviews.html',{'list_objects': obj})

def View_Negative_reviews(request):

    rtype='Negative'
    #obj = review_Model.objects.all()

    obj = review_Model.objects.all().filter(sanalysis=rtype)

    return render(request,'TServer/View_Negative_reviews.html',{'list_objects': obj})

def View_Depression_reviews(request):
    rtype='Depression'
    #obj = review_Model.objects.all()

    obj = review_Model.objects.all().filter(sanalysis=rtype)

    return render(request,'TServer/View_Depression_reviews.html',{'list_objects': obj})



def viewallusers(request):
    obj=ClientRegister_Model.objects.all()
    return render(request,'TServer/viewallusers.html',{'objects':obj})

def negativechart(request,chart_type):
    dd = {}
    pos, neu, neg = 0, 0, 0
    poss = None
    topic =
tweets_Model.objects.values('ratings').annotate(dcount=Count('ratings')).order_by('-
dcount')
    for t in topic:
        topics = t['ratings']
        pos_count =
tweets_Model.objects.filter(topics=topics).values('names').annotate(topiccount=Count('r
atings'))
        poss = pos_count
        for pp in pos_count:
            senti = pp['names']
            if senti == 'positive':
                pos = pp['topiccount']
            elif senti == 'negative':
                neg = pp['topiccount']
            elif senti == 'nutral':
                neu = pp['topiccount']
        dd[topics] = [pos, neg, neu]
    return
render(request,'TServer/negativechart.html',{'object':topic,'dd':dd,'chart_type':chart_type
})



def charts(request,chart_type):
    chart1 = tweets_Model.objects.values('names').annotate(dcount=Avg('ratings'))
    return render(request,"TServer/charts.html", {'form':chart1, 'chart_type':chart_type})

def dislikeschart(request,dislike_chart):
    charts = tweets_Model.objects.values('names').annotate(dcount=Avg('dislikes'))
    return render(request,"TServer/dislikeschart.html", {'form':charts,
'dislike_chart':dislike_chart})

def View_All_User_Tweets(request):
    chart =
tweets_Model.objects.values('names','uname','ratings','dislikes','uses','sanalysis','tdesc').a
nnotate(dcount=Avg('usefulcounts'))
    return render(request,'TServer/View_All_User_Tweets.html',{'objects':chart})

def View_Sentiment_Analysis(request):

    if request.method == "POST":
        kword = request.POST.get('type')
        obj = tweets_Model.objects.all().filter(sanalysis=kword)
        return render(request, 'TServer/View_Sentiment_Analysis.html', {'objs': obj})
    return render(request, 'TServer/View_Sentiment_Analysis.html')
```

```
def View_Depression_Posts(request):

    kword = 'Depression'
    obj = tweets_Model.objects.all().filter(sanalysis=kword)
    return render(request, 'TServer/View_Depression_Posts.html', {'objs': obj})
```

7. PROJECT 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.

7.1 TYPES OF TESTS

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.

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.

Functional testing

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.

System Testing

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.

7.2 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.

7.3 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.

7.4 Various Test Cases

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

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.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

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

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.

8. OUTPUT SCREENS

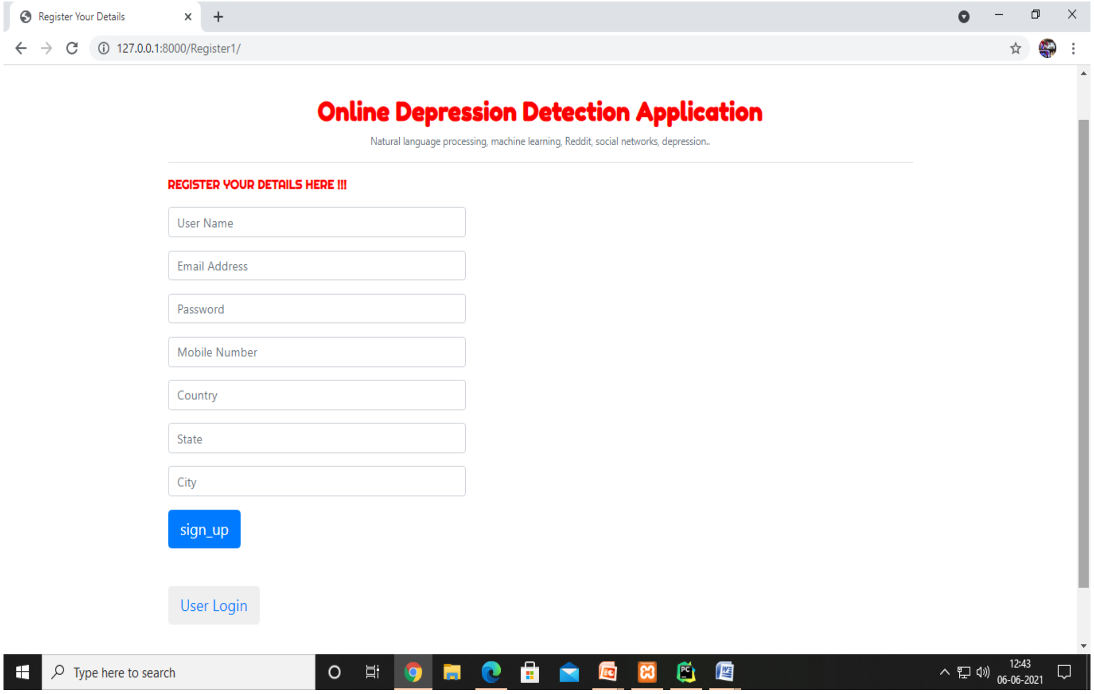

Register Your Details
127.0.0.1:8000/Register1/
Online Depression Detection Application
Natural language processing, machine learning, Reddit, social networks, depression..
REGISTER YOUR DETAILS HERE !!!
User Name
Email Address
Password
Mobile Number
Country
State
City
sign_up
User Login
Type here to search
12:43
06-06-2021

Fig: User Registration Page

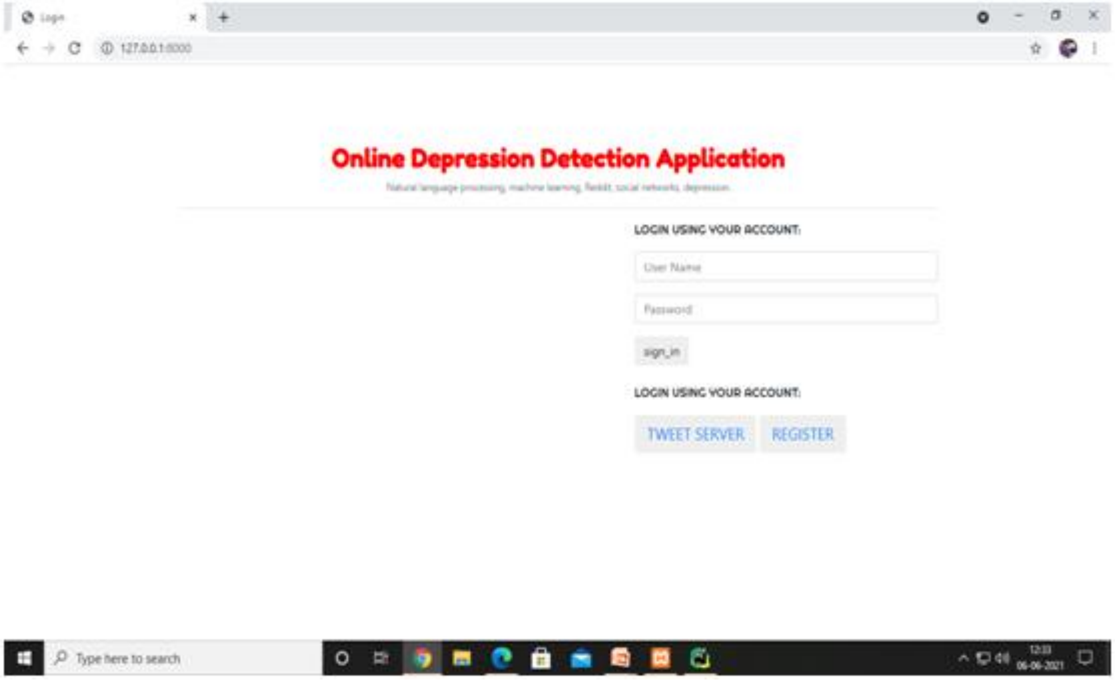
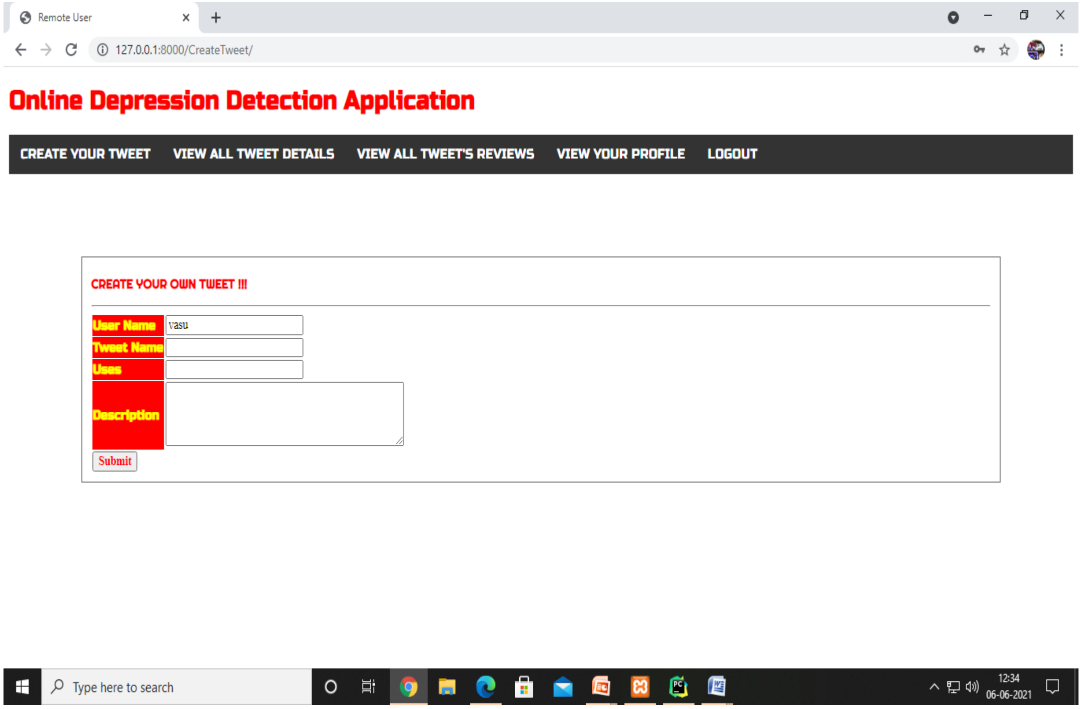

Login
127.0.0.1:8000
Online Depression Detection Application
Natural language processing, machine learning, Reddit, social networks, depression.
LOGIN USING YOUR ACCOUNT:
User Name
Password
sign_in
LOGIN USING YOUR ACCOUNT:
TWEET SERVER
REGISTER
Type here to search
12:33
06-06-2021

Fig: User Login Page

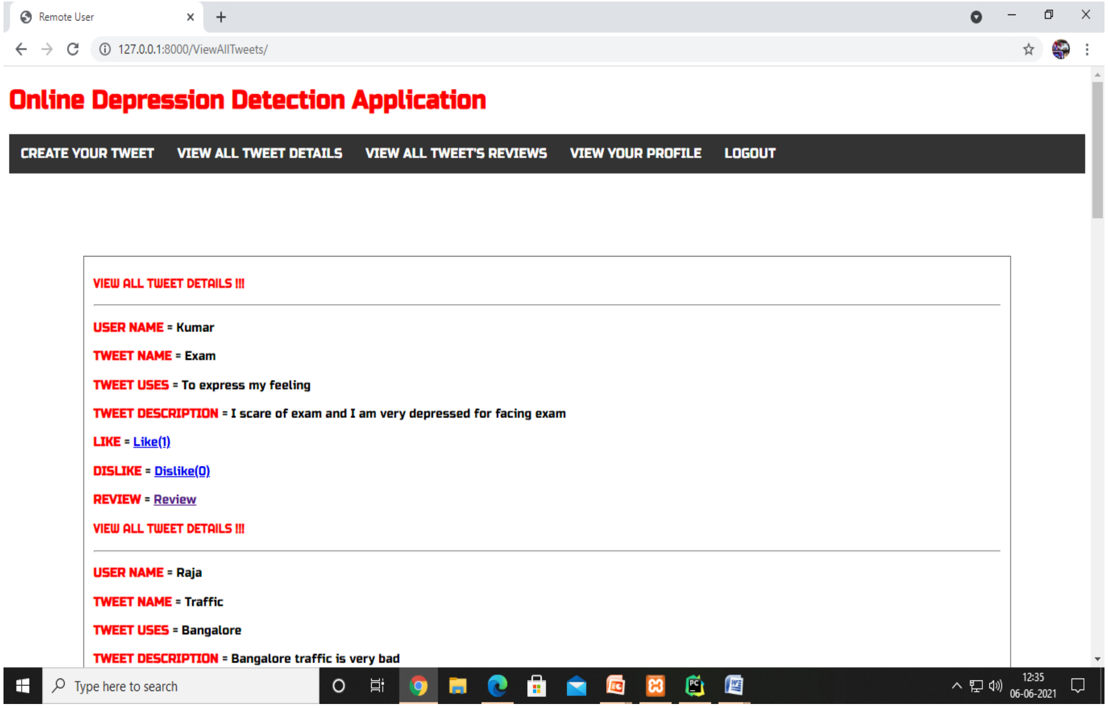


CREATE YOUR OWN TWEET !!!

User Name	vasu
Tweet Name	
Uses	
Description	

Submit

Fig: Upload Tweet



VIEW ALL TWEET DETAILS !!!

USER NAME = Kumar

TWEET NAME = Exam

TWEET USES = To express my feeling

TWEET DESCRIPTION = I scare of exam and I am very depressed for facing exam

LIKE = Like(1)

DISLIKE = Dislike(0)

REVIEW = Review

VIEW ALL TWEET DETAILS !!!

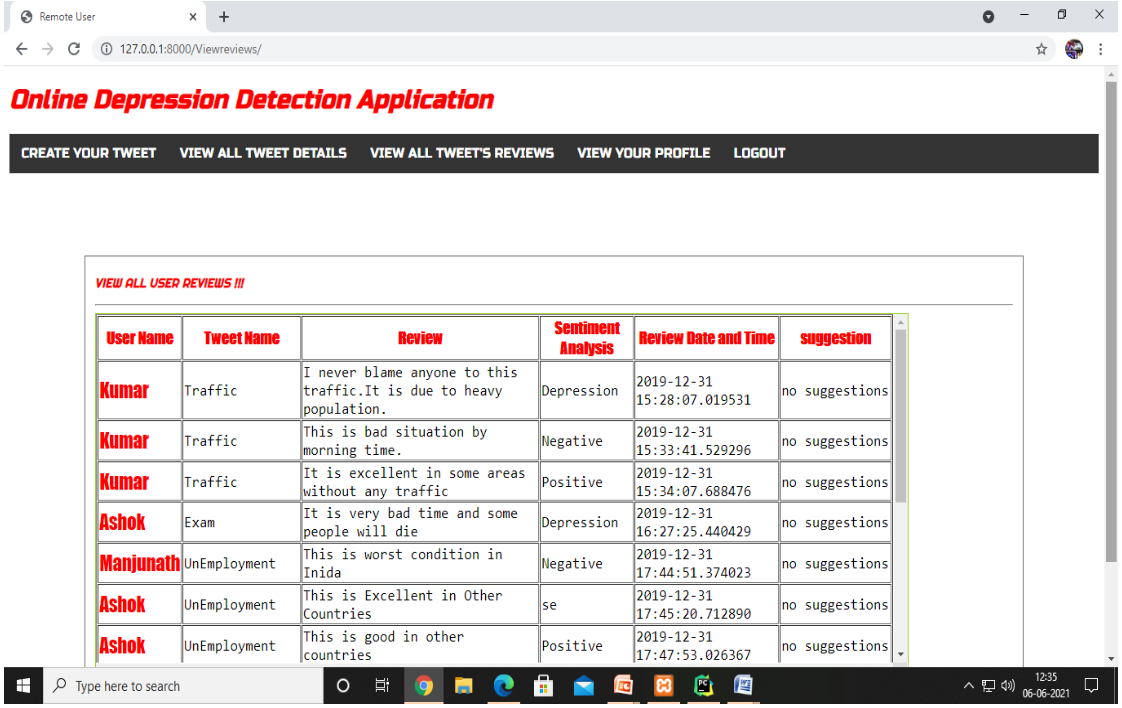
USER NAME = Raja

TWEET NAME = Traffic

TWEET USES = Bangalore

TWEET DESCRIPTION = Bangalore traffic is very bad

Fig: View All Tweets

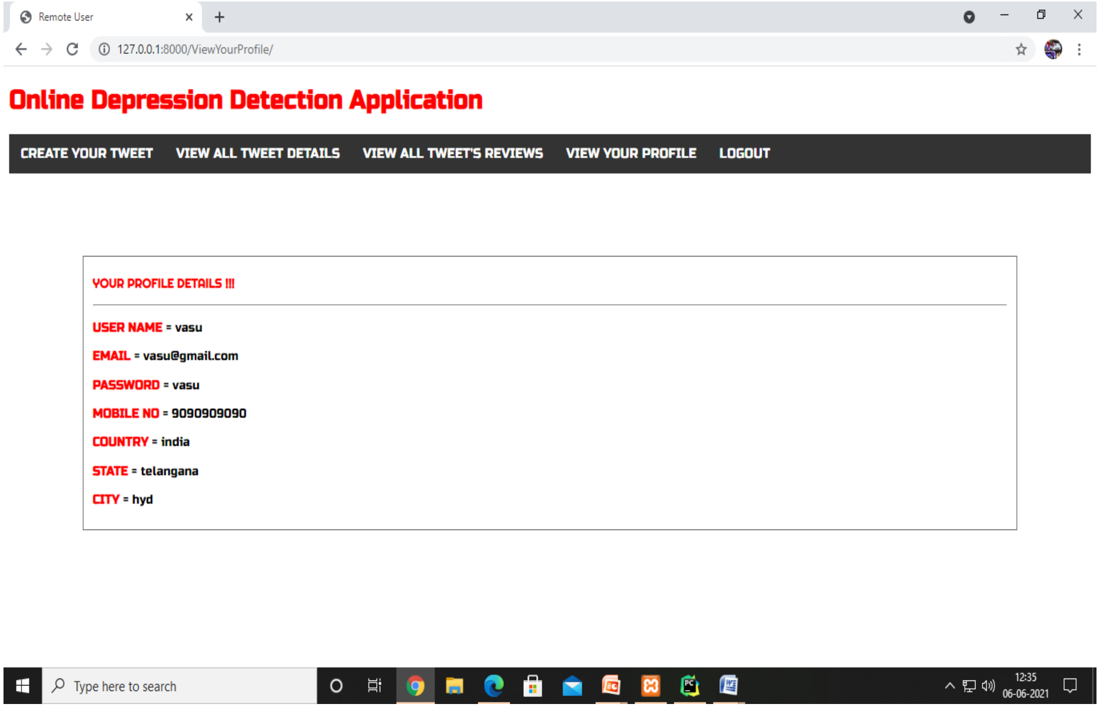


Online Depression Detection Application

VIEW ALL USER REVIEWS !!!

User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	suggestion
Kumar	Traffic	I never blame anyone to this traffic.It is due to heavy population.	Depression	2019-12-31 15:28:07.019531	no suggestions
Kumar	Traffic	This is bad situation by morning time.	Negative	2019-12-31 15:33:41.529296	no suggestions
Kumar	Traffic	It is excellent in some areas without any traffic	Positive	2019-12-31 15:34:07.688476	no suggestions
Ashok	Exam	It is very bad time and some people will die	Depression	2019-12-31 16:27:25.440429	no suggestions
Manjunath	UnEmployment	This is worst condition in Inida	Negative	2019-12-31 17:44:51.374023	no suggestions
Ashok	UnEmployment	This is Excellent in Other Countries	se	2019-12-31 17:45:20.712890	no suggestions
Ashok	UnEmployment	This is good in other countries	Positive	2019-12-31 17:47:53.026367	no suggestions

Fig: View all User Reviews



Online Depression Detection Application

YOUR PROFILE DETAILS !!!

USER NAME = vasu

EMAIL = vasu@gmail.com

PASSWORD = vasu

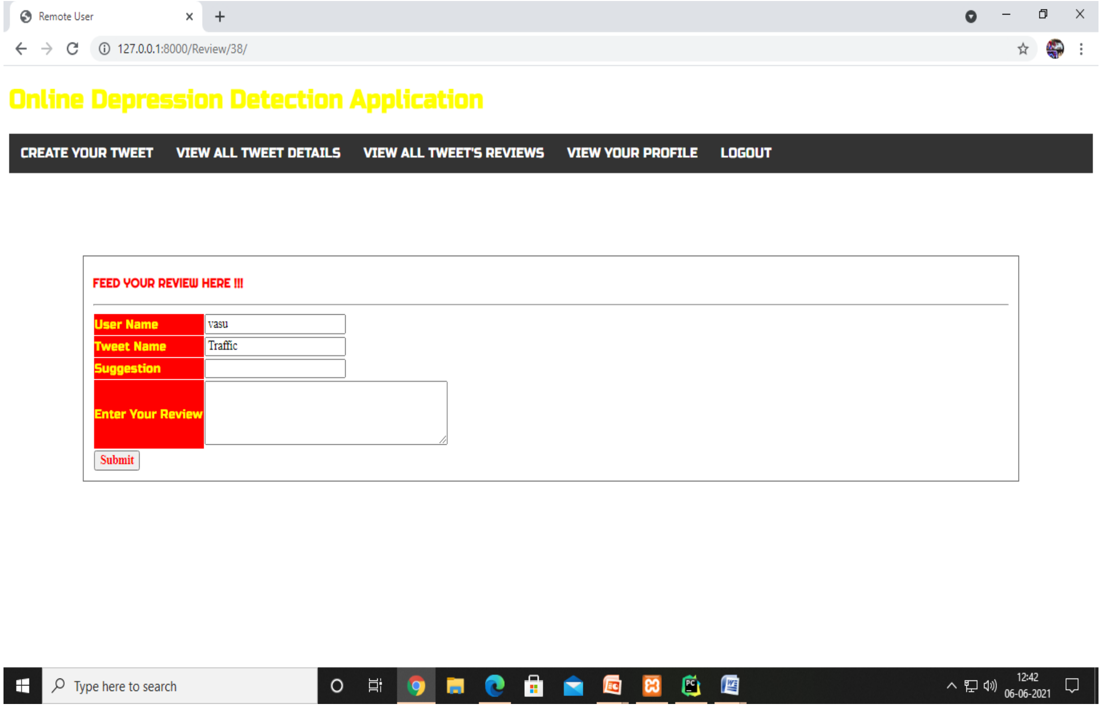
MOBILE NO = 9090909090

COUNTRY = india

STATE = telangana

CITY = hyd

Fig: View User Profile

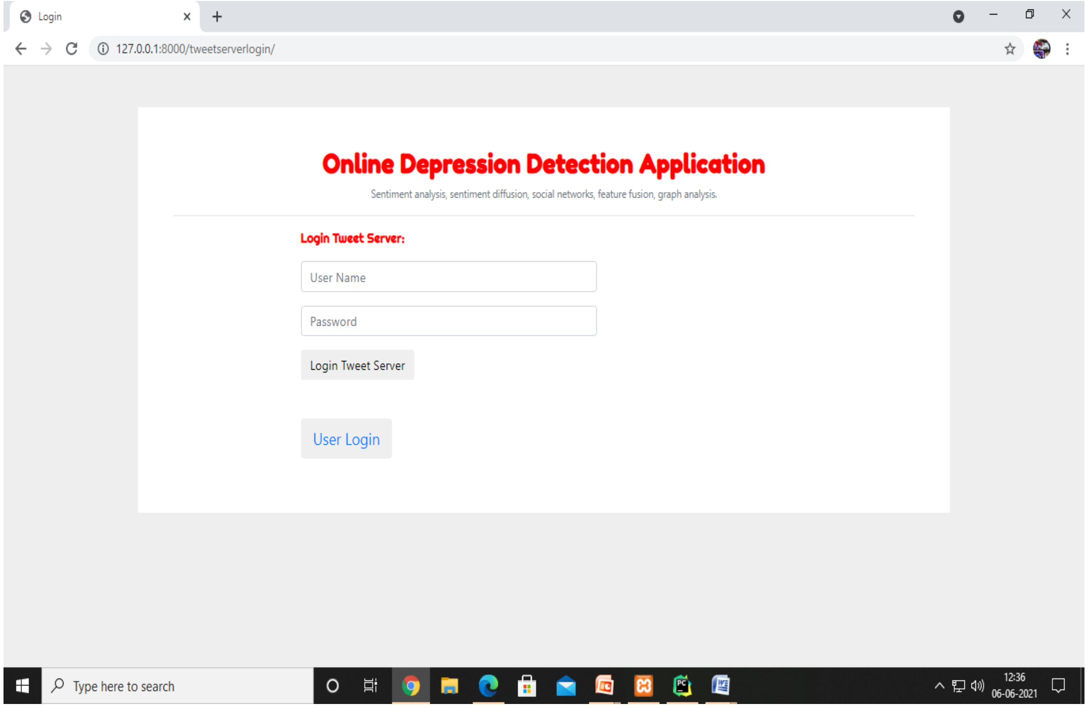


Online Depression Detection Application

FEED YOUR REVIEW HERE !!!

User Name	vasu
Tweet Name	Traffic
Suggestion	
Enter Your Review	

Fig: Upload Review

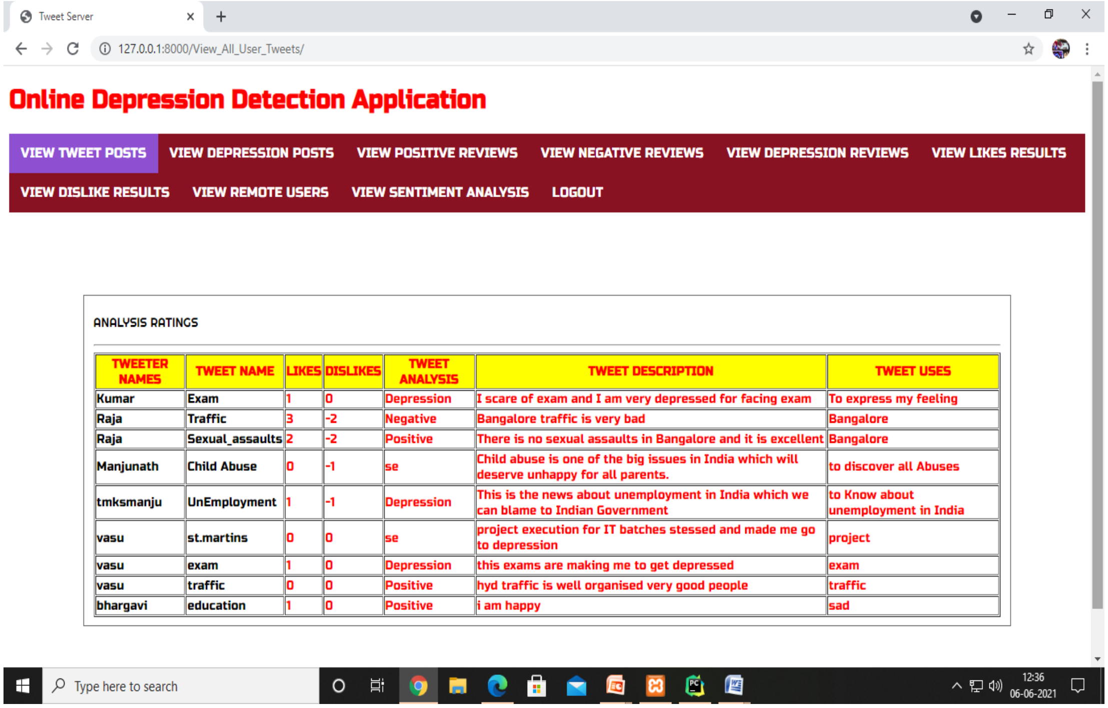


Online Depression Detection Application

Sentiment analysis, sentiment diffusion, social networks, feature fusion, graph analysis.

Login Tweet Server:

Fig: Admin Login

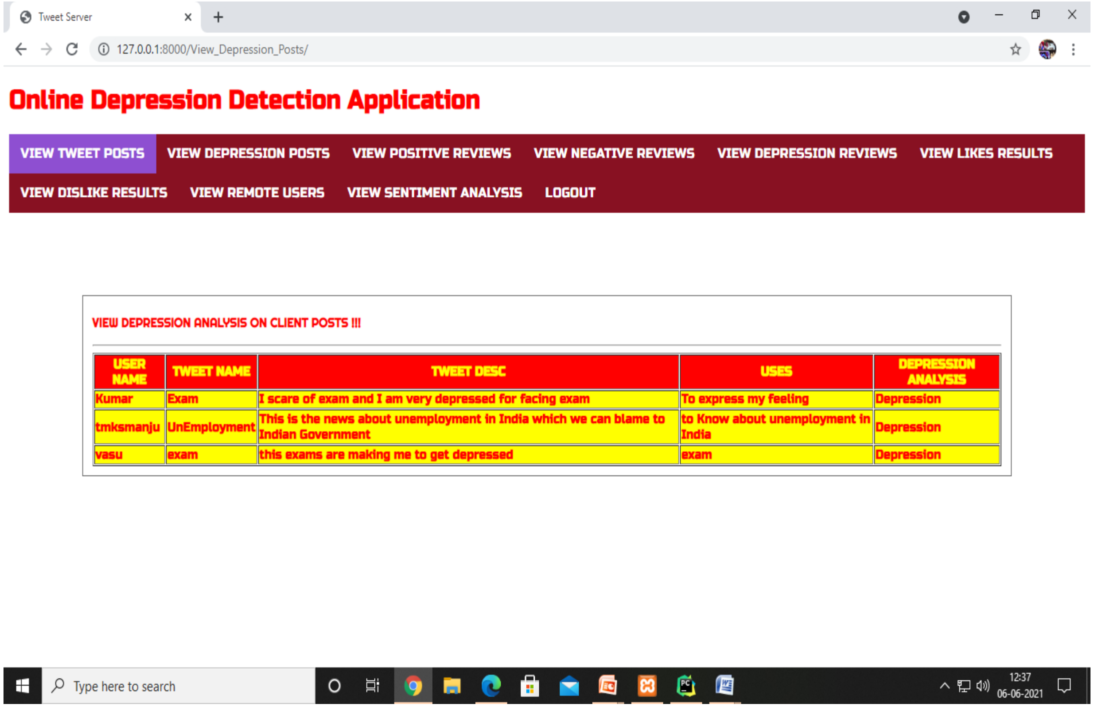


Online Depression Detection Application

ANALYSIS RATINGS

TWEETER NAMES	TWEET NAME	LIKES	DISLIKES	TWEET ANALYSIS	TWEET DESCRIPTION	TWEET USES
Kumar	Exam	1	0	Depression	I scare of exam and I am very depressed for facing exam	To express my feeling
Raja	Traffic	3	-2	Negative	Bangalore traffic is very bad	Bangalore
Raja	Sexual_assaults	2	-2	Positive	There is no sexual assaults in Bangalore and it is excellent	Bangalore
Manjunath	Child Abuse	0	-1	se	Child abuse is one of the big issues in India which will deserve unhappy for all parents.	to discover all Abuses
tmksmanju	UnEmployment	1	-1	Depression	This is the news about unemployment in India which we can blame to Indian Government	to Know about unemployment in India
vasu	st.martins	0	0	se	project execution for IT batches stessed and made me go to depression	project
vasu	exam	1	0	Depression	this exams are making me to get depressed	exam
vasu	traffic	0	0	Positive	hyd traffic is well organised very good people	traffic
bhargavi	education	1	0	Positive	i am happy	sad

Fig: View Tweet Post

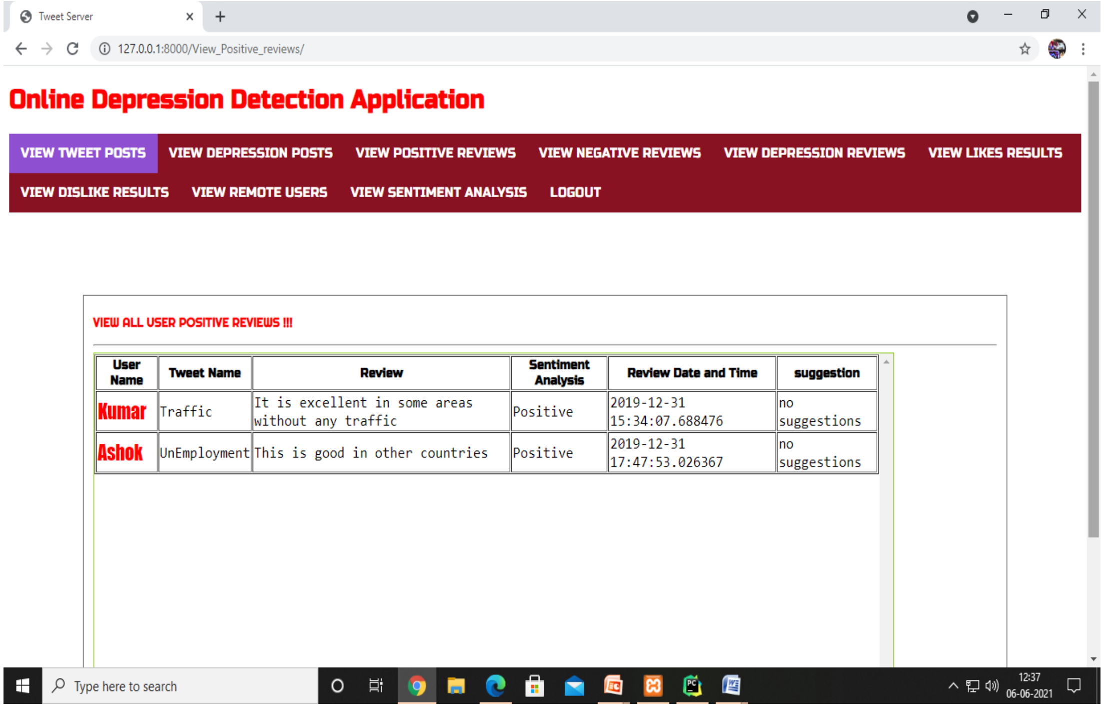


Online Depression Detection Application

VIEW DEPRESSION ANALYSIS ON CLIENT POSTS !!!

USER NAME	TWEET NAME	TWEET DESC	USES	DEPRESSION ANALYSIS
Kumar	Exam	I scare of exam and I am very depressed for facing exam	To express my feeling	Depression
tmksmanju	UnEmployment	This is the news about unemployment in India which we can blame to Indian Government	to Know about unemployment in India	Depression
vasu	exam	this exams are making me to get depressed	exam	Depression

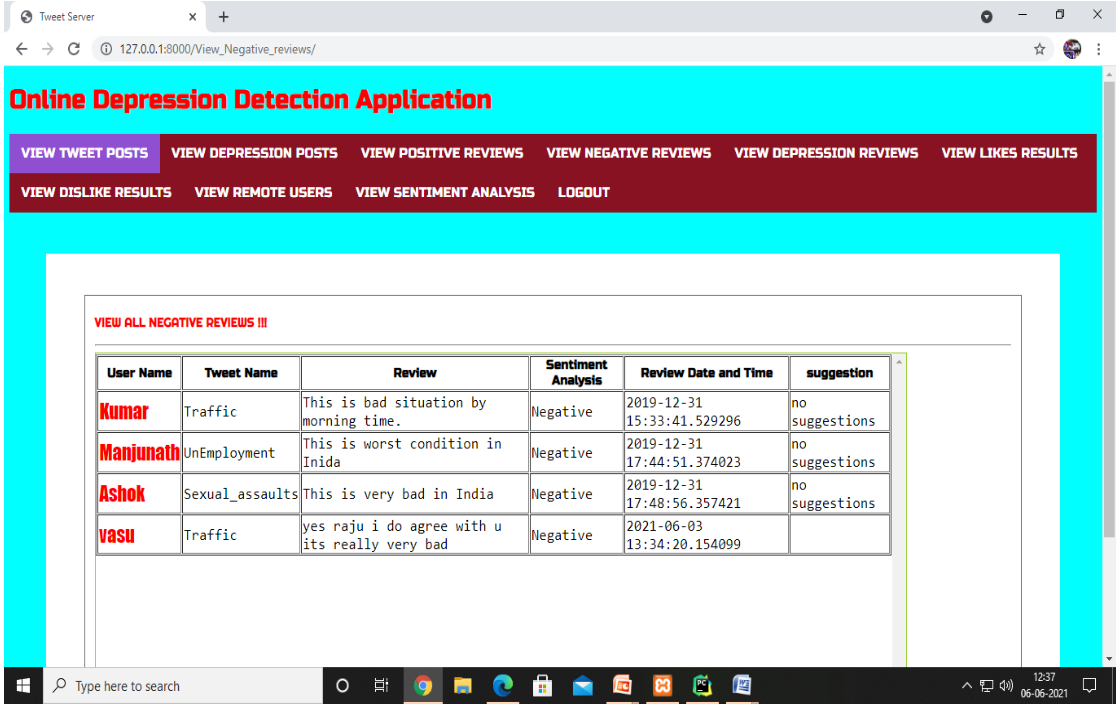
Fig: View Depression Posts



VIEW ALL USER POSITIVE REVIEWS !!!

User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	suggestion
Kumar	Traffic	It is excellent in some areas without any traffic	Positive	2019-12-31 15:34:07.688476	no suggestions
Ashok	UnEmployment	This is good in other countries	Positive	2019-12-31 17:47:53.026367	no suggestions

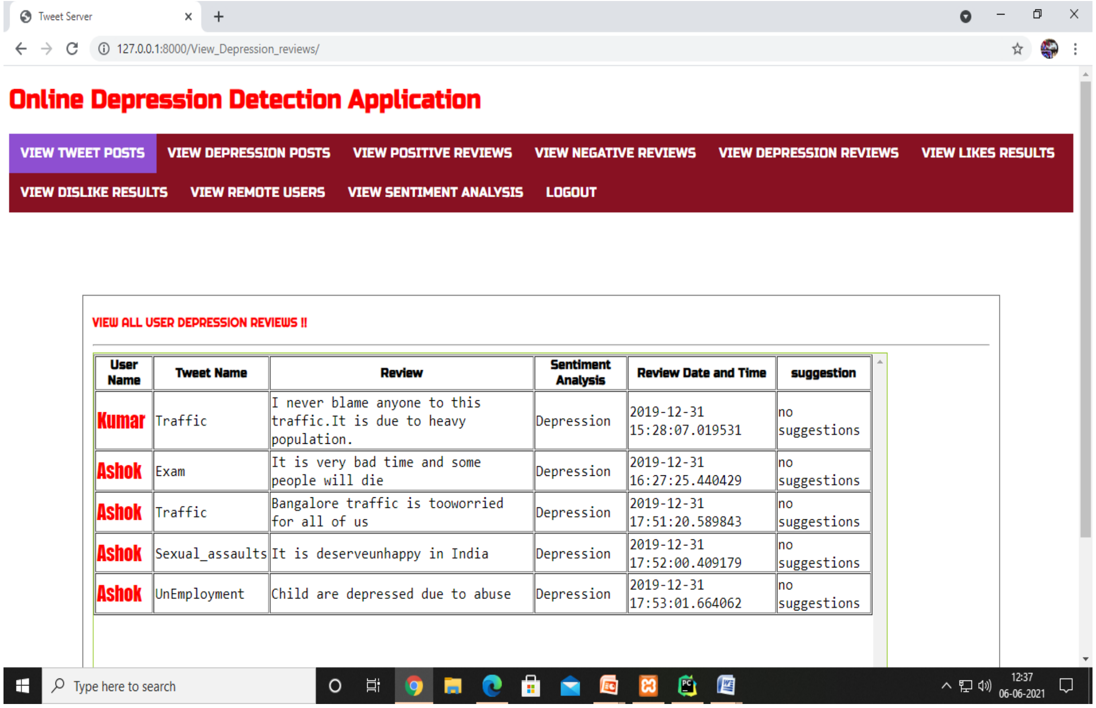
Fig: View Positive Reviews



VIEW ALL NEGATIVE REVIEWS !!!

User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	suggestion
Kumar	Traffic	This is bad situation by morning time.	Negative	2019-12-31 15:33:41.529296	no suggestions
Manjunath	UnEmployment	This is worst condition in Inida	Negative	2019-12-31 17:44:51.374023	no suggestions
Ashok	Sexual_assaults	This is very bad in India	Negative	2019-12-31 17:48:56.357421	no suggestions
vasu	Traffic	yes raju i do agree with u its really very bad	Negative	2021-06-03 13:34:20.154099	

Fig: View Negative Posts



Online Depression Detection Application

VIEW ALL USER DEPRESSION REVIEWS !!

User Name	Tweet Name	Review	Sentiment Analysis	Review Date and Time	suggestion
Kumar	Traffic	I never blame anyone to this traffic.It is due to heavy population.	Depression	2019-12-31 15:28:07.019531	no suggestions
Ashok	Exam	It is very bad time and some people will die	Depression	2019-12-31 16:27:25.440429	no suggestions
Ashok	Traffic	Bangalore traffic is tooworried for all of us	Depression	2019-12-31 17:51:20.589843	no suggestions
Ashok	Sexual_assaults	It is deserveunhappy in India	Depression	2019-12-31 17:52:00.409179	no suggestions
Ashok	UnEmployment	Child are depressed due to abuse	Depression	2019-12-31 17:53:01.664062	no suggestions

Fig: View Depression Reviews

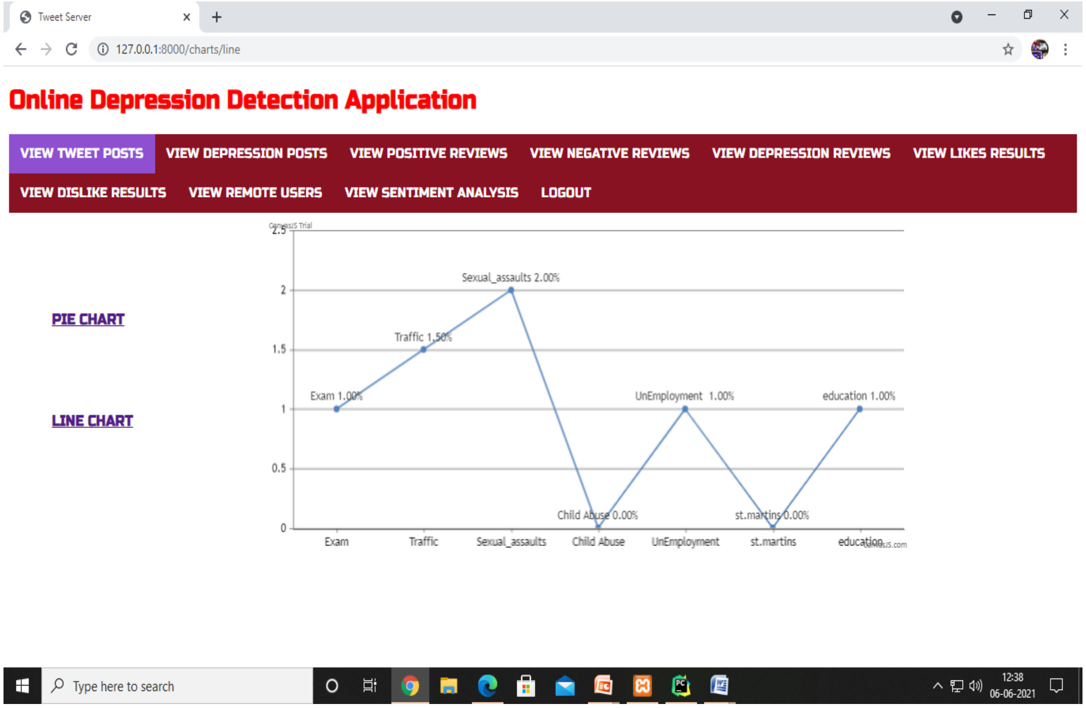


Fig: Line Chart

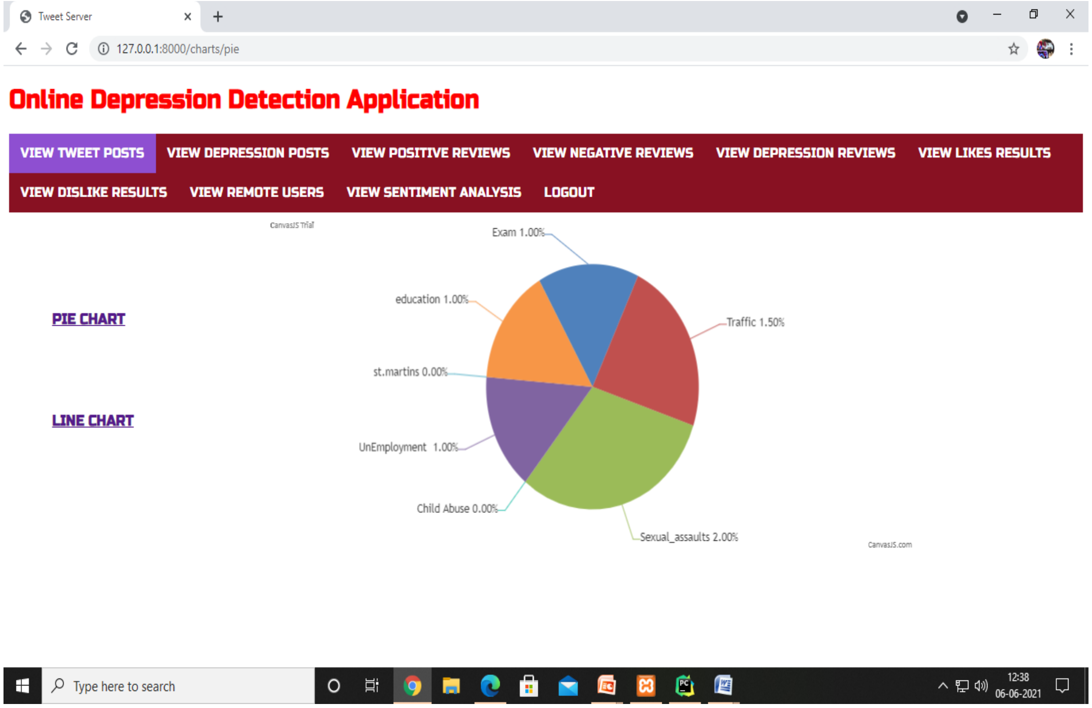


Fig: Pie Chart

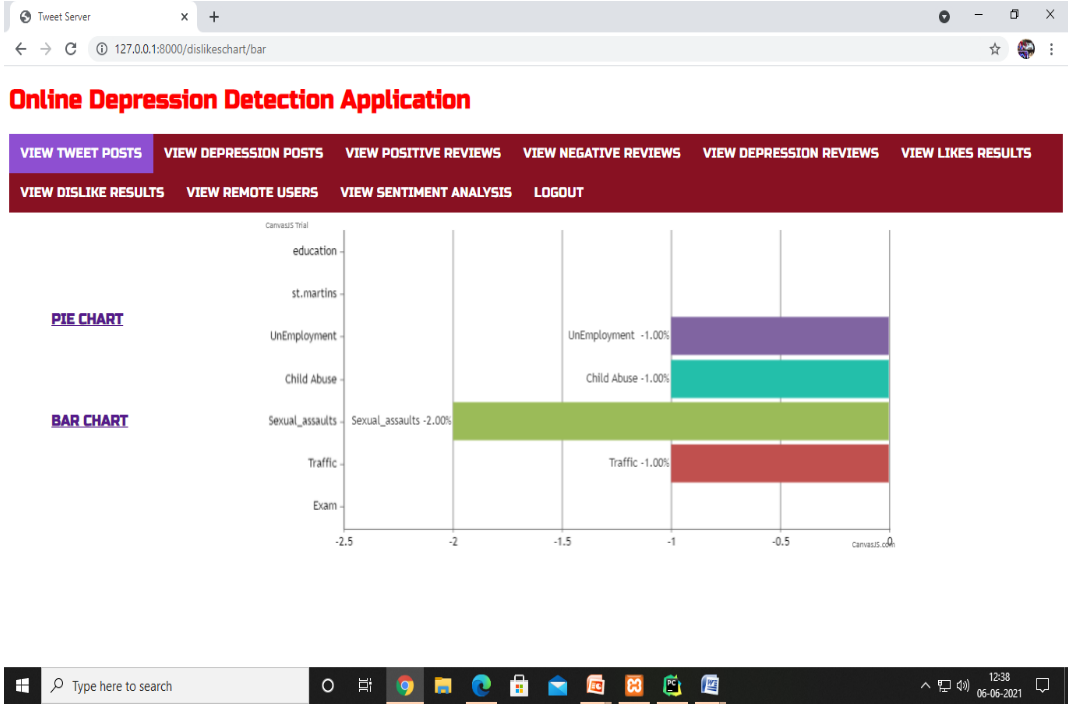
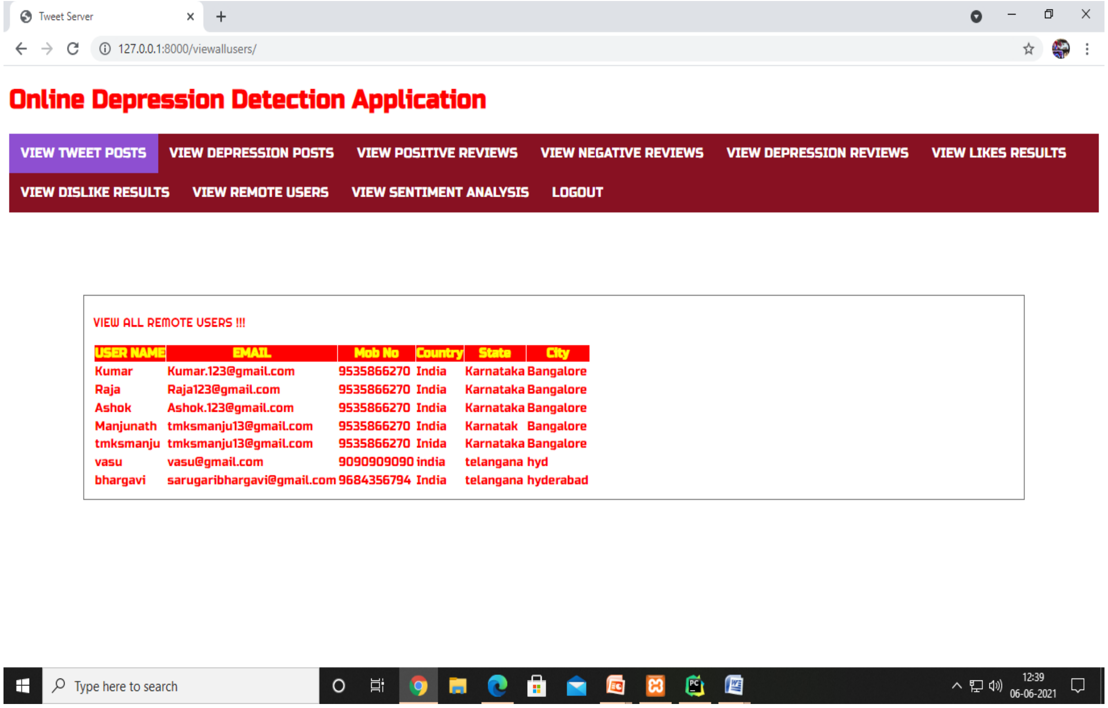


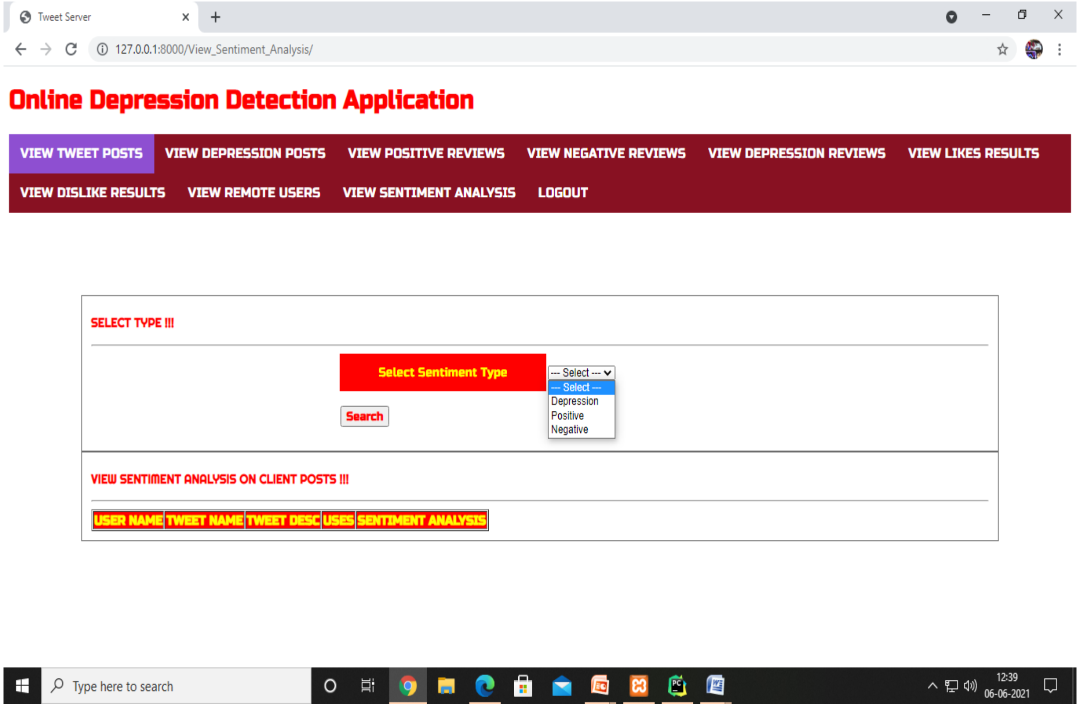
Fig: Bar chart



VIEW ALL REMOTE USERS !!!

USER NAME	EMAIL	Mob No	Country	State	City
Kumar	Kumar.123@gmail.com	9535866270	India	Karnataka	Bangalore
Raja	Raja123@gmail.com	9535866270	India	Karnataka	Bangalore
Ashok	Ashok.123@gmail.com	9535866270	India	Karnataka	Bangalore
Manjunath	tmksmanju13@gmail.com	9535866270	India	Karnatak	Bangalore
tmksmanju	tmksmanju13@gmail.com	9535866270	Inida	Karnataka	Bangalore
vasu	vasu@gmail.com	9090909090	india	telangana	hyd
bhargavi	sarugaribhargavi@gmail.com	9684356794	India	telangana	hyderabad

Fig: View All Users



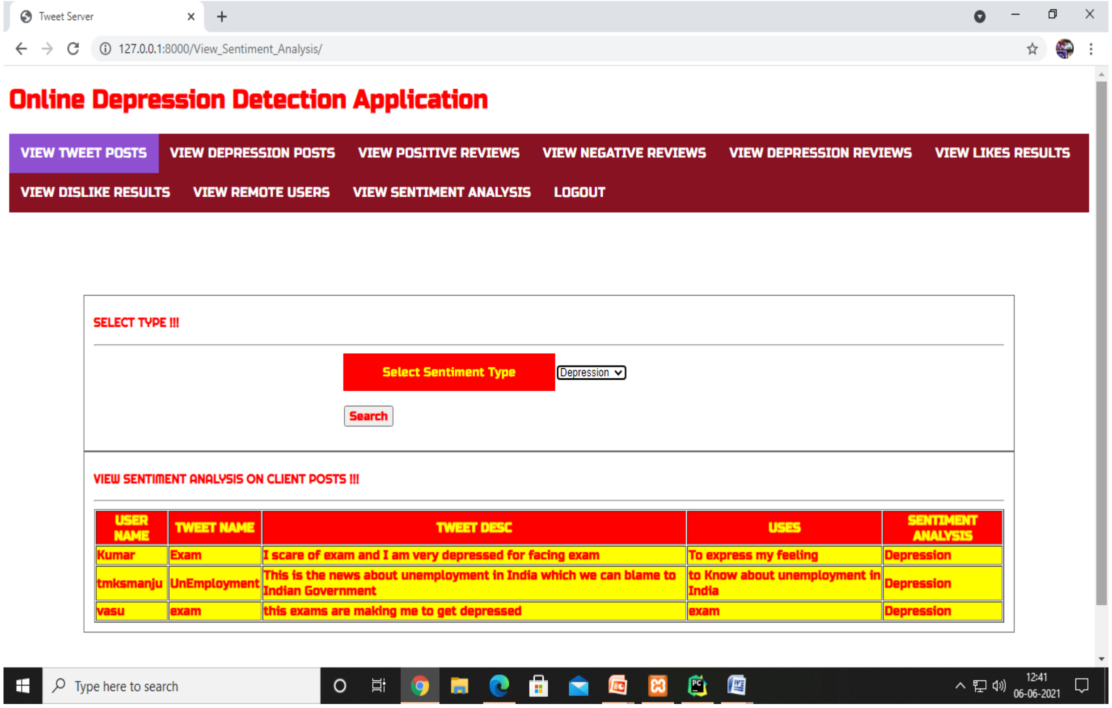
SELECT TYPE !!!

Select Sentiment Type

VIEW SENTIMENT ANALYSIS ON CLIENT POSTS !!!

USER NAME	TWEET NAME	TWEET DESC	USES	SENTIMENT ANALYSIS

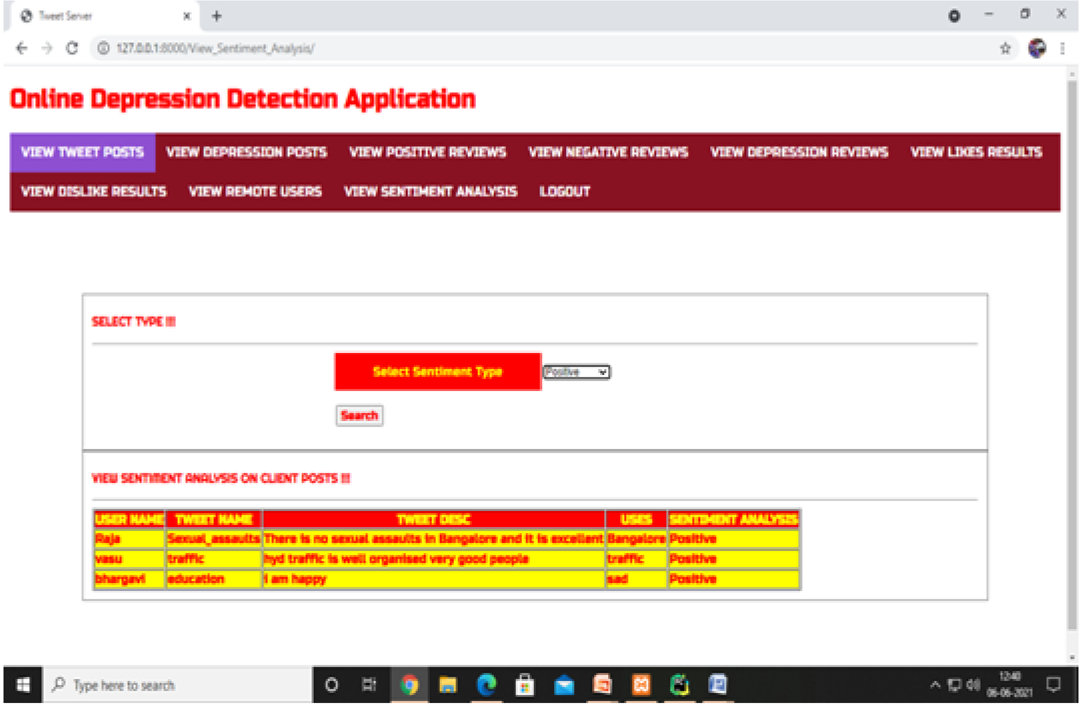
Fig: View Sentiment Analysis



VIEW SENTIMENT ANALYSIS ON CLIENT POSTS !!!

USER NAME	TWEET NAME	TWEET DESC	USES	SENTIMENT ANALYSIS
Kumar	Exam	I scare of exam and I am very depressed for facing exam	To express my feeling	Depression
tmksmanju	UnEmployment	This is the news about unemployment in India which we can blame to Indian Government	to Know about unemployment in India	Depression
vasu	exam	this exams are making me to get depressed	exam	Depression

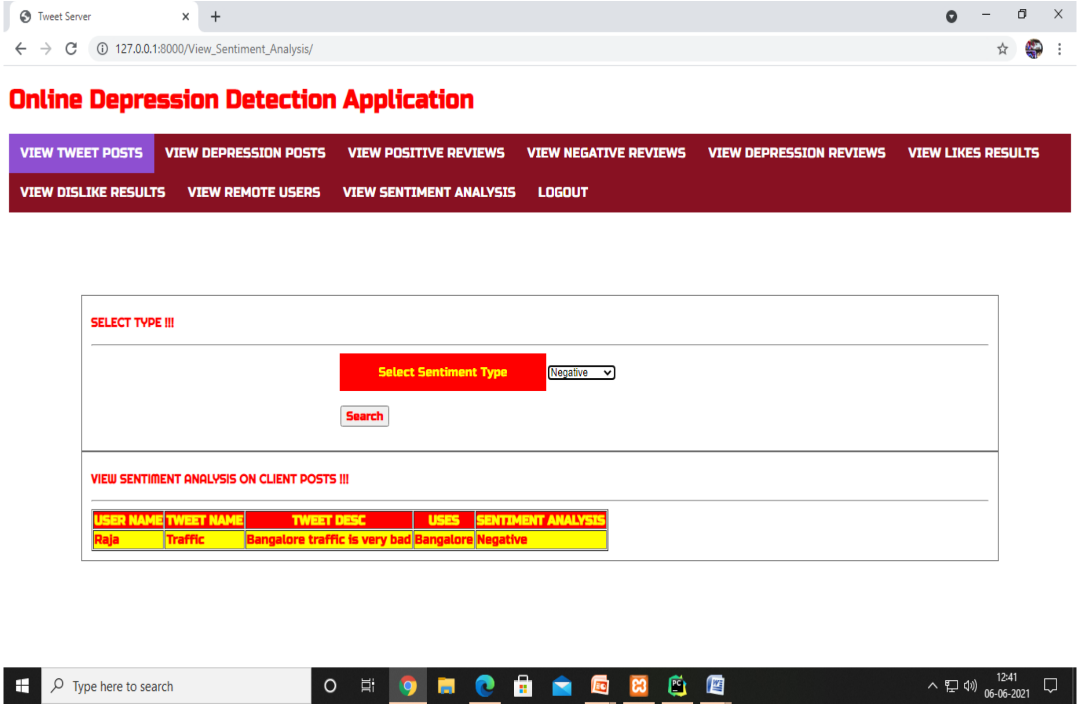
Fig: Sentiment Analysis for Depression



VIEW SENTIMENT ANALYSIS ON CLIENT POSTS !!!

USER NAME	TWEET NAME	TWEET DESC	USES	SENTIMENT ANALYSIS
Raja	Sexual_assaults	There is no sexual assaults in Bangalore and it is excellent	Bangalore	Positive
vasu	traffic	hyd traffic is well organised very good people	traffic	Positive
bhargavi	education	i am happy	sad	Positive

Fig: Sentiment Analysis for Positive



Select Sentiment Type: Negative

USER NAME	TWEET NAME	TWEET DESC	USES	SENTIMENT ANALYSIS
Raja	Traffic	Bangalore traffic is very bad	Bangalore	Negative

Fig: Sentiment Analysis for Negative

9. CONCLUSION AND FUTURE ENHANCEMENT

To investigate the effect of depression detection, we propose machine learning technique as an efficient and scalable method. To detect depression we are using SVM (support vector machine) algorithm which analyse users post and give result as negative or positive. If users express depression words in post then SVM detect it as a negative post else positive post. We tried to identify the presence of depression in Reddit social media; and searched for affective performance increase solutions of depression detection. We characterized a closer connection between depression and a language usage by applying NLP and text classification techniques. We identified a lexicon of words more common among the depressed accounts. According to our findings, the language predictors of depression contained the words related to preoccupation with themselves, feelings of sadness, anxiety, anger, hostility or suicidal thoughts, with a greater emphasis on the present and future.

To measure the signs of depression, we examined the performance of both single feature and combined feature sets using various text classifying methods. Our results show that a higher predictive performance is hidden in proper features selection and their multiple feature combinations. The strength and effectiveness of combined features are demonstrated with the MLP classifier reaching 91% accuracy and 0.93 F1 score achieving the highest performance degree for detecting the presence of depression in Reddit social media conducted in our study. Additionally, the best feature among the single feature sets is bigram; with SVM classifier it can detect depression with 80% accuracy and 0.79 F1 score. Considering LIWC and LDA features, LIWC outperformed topic models generated by LDA. Although our experiment shows that the performances of applied methodologies are reasonably good, the absolute values of the metrics indicate that this is a challenging task and worthy of further exploration. We believe this experiment could further underline the infrastructure for new mechanisms applied in different areas of healthcare to estimate depression and related variables. It can be beneficial for the individuals suffering from mental health disorders to be more proactive towards their fast recovery.

In future work, we plan to use another technique to extract paraphrases from more types of emotional features. Also, we plan to use more dataset to verify our techniques efficiency and effectiveness. We in agreement with the existing body of literature that suggests that more focused studies in depression analysis are needed. We will try to examine the relationship between the users' personality and their depression-related behavior reflected in social media.

10. REFERENCES

1. Hong-Han Shuai, Chih-Ya Shen, De-Nian Yang, Yi-Feng Carol Lan, Wang-Chien Lee, Philip S. Yu, Ming-Syan Chen, “A Comprehensive Study on Social Network Mental Disorders Detection via Online Social Media Mining”, IEEE Transactions on Knowledge and Data Engineering, Pages: 1212 – 1225, Year: 2018, Volume: 30, Issue: 7, Journal Article, Publisher: IEEE

2. Megha Rathi, Aditya Malik, Daksh Varshney, Rachita Sharma, Sarthak Mendiratta, “Sentiment Analysis of Tweets Using Machine Learning Approach”, 2018 Eleventh International Conference on Contemporary Computing (IC3), Pages:1-3, Year: 2018, Conference Paper, Publisher: IEEE

3. Mohammed H. Abd El-Jawad, Rania Hodhod, Yasser M. K. Omar, “Sentiment Analysis of Social Media Networks Using Machine Learning”, 2018 14th International Computer Engineering Conference (ICENCO), Pages:174-176, Year: 2018, Conference Paper, Publisher: IEEE

4. Wajdi Zaghouani, “A Large-Scale Social Media Corpus for the Detection of Youth Depression (Project Note)”, Procedia Computer Science, Volume 142, 2018, Pages: 347-351

5. Alvaro Esperanca, Zina Ben Miled, Malika Mahoui, “Social Media Sensing Framework for Population Health”, 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Pages:0298-0304, Year: 2019, Conference Paper, Publisher: IEEE

6. Hoong-Cheng Soong, Norazira Binti A Jalil, Ramesh Kumar Ayyasamy, Rehan Akbar, “The Essential of Sentiment Analysis and Opinion Mining in Social Media: Introduction and Survey of Recent Approaches and Techniques”, 2019 IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE), Pages:272-277, Year: 2019, Conference Paper, Publisher: IEEE

7. B. K. Bhavitha, A. P. Rodrigues, N. N. Chiplunkar, “Comparative study of machine learning techniques in sentimental analysis”, 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, 2017,pp.216- 221.

8. Liqiang Nie, Yi-Liang Zhao, Mohammad Akbari, Jialie Shen, TatSeng Chua,

"Bridging the Vocabulary Gap between Health Seekers and Healthcare Knowledge", IEEE Transactions on Knowledge and Data Engineering, Pages:396 – 409, Year: 2015. Volume: 27, Issue: 2, Journal Article, Publisher: IEEE

9. Huijie Lin, Jia Jia, Quan Guo, Yuanyuan Xue, Jie Huang, Lianhong Cai, Ling Feng, "Psychological stress detection from cross-media microblog data using Deep Sparse Neural Network", 2014 IEEE International Conference on Multimedia and Expo (ICME), Pages:1- 6, Year: 2014, Conference Paper, Publisher: IEEE.

10. R. Parikh and M. Movassate, "Sentiment Analysis of User-Generated Twitter Updates using Various Classification Techniques".

11. H. A. Schwartz, J. Eichstaedt, M. L. Kern, G. Park, M. Sap, D. Stillwell, M. Kosinski, and L. Ungar, "Towards assessing changes in degree of depression through facebook," in Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, 2014, pp. 118–125.

12. S. Tsugawa, Y. Kikuchi, F. Kishino, K. Nakajima, Y. Itoh, and H. Ohsaki, "Recognizing depression from twitter activity," in Proceedings of the 33rd annual ACM conference on human factors in computing systems. ACM, 2015, pp. 3187–3196.

13. J. Wolohan, M. Hiraga, A. Mukherjee, Z. A. Sayyed, and M. Millard, "Detecting linguistic traces of depression in topic-restricted text: Attending to self-stigmatized depression with nlp," in Proceedings of the First International Workshop on Language Cognition and Computational
Models, 2018, pp. 11–21.

14. Y. Tyshchenko, "Depression and anxiety detection from blog posts data." University of Tartu Institute of Computer Science Computer Science Curriculum, 2018.

15. S. C. Guntuku, D. B. Yaden, M. L. Kern, L. H. Ungar, and J. C. Eichstaedt, "Detecting depression and mental illness on social media: an integrative review," Current Opinion in Behavioral Sciences, vol. 18, pp. 43–49, 2017.

16. R. A. Calvo, D. N. Milne, M. S. Hussain, and H. Christensen, "Natural language processing in mental health applications using non-clinical texts," Natural Language Engineering, vol. 23, no. 5, pp. 649–685, 2017.

17. Á. Hernández-Castañeda and H. Calvo, "Deceptive text detection using continuous semantic space models," Intelligent Data Analysis, vol. 21, no. 3, pp. 679–695, 2017.

18. S. Freud, “The psychopathology of everyday life. se, 6,” London: Hogarth, 1901.

19. A. T. Beck, Depression: Clinical, experimental, and theoretical aspects. University of Pennsylvania Press, 1967.

20. T. Pyszczynski and J. Greenberg, “Self-regulatory perseveration and the depressive self-focusing style: a self-awareness theory of reactive depression.” Psychological bulletin, vol. 102, no. 1, p. 122, 1987.

A

PROJECT REPORT

On

SUPERMARKET BILLING SYSTEM USING WEBCAM

Submitted by

Ms. G. AKANKSHA	(17K81A1213)
Mr. Ch. SAI KRISHNA PRASAD	(17K81A1206)
Mr. K. AJIT KUMAR REDDY	(17K81A1224)
Ms. D. SREENIDHI	(17K81A1207)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Mr. P. GANESH KUMAR

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN'S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **SUPERMARKET BILLING SYSTEM USING WEBCAM**, is being submitted by **G.AKANKSHA (17K81A1213), Ch. SAI KRISHNA PRASAD (17K81A1206), K. AJIT KUMAR REDDY (17K81A1224), D. SREENIDHI (17K81A1207)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN** INFORMATION TECHNOLOGY is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Project Guide	Head of the Department
P. GANESH KUMAR	**Dr. R. NAGARAJU**
Department of Information Technology	Department of Information Technology

Internal Examiner External Examiner

Place:

Date:

LASYA
INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **CH.SAI KRISHNA PRASAD** WITH ROLL NO.**17K81A1206, D.SREENIDHI** WITH ROLL NO.**17K81A1207**, **G.AKANKSHA** WITH ROLL NO.**17K81A1213**, **K.AJIT KUMAR REDDY** WITH ROLL NO.**17K81A1224**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**SUPERMARKET BILLING SYSTEM USING WEBCAM**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd.
★ Kompally-500014 ★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

DECLARATION

We, the students of **Bachelor of Technology** in Department of Information Technology, session: 2017 – 2021, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled SUPERMARKET BILLING SYSTEM USING WEBCAM is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

G. AKANKSHA	**17K81A1213**
Ch. SAI KRISHNA PRASAD	**17K81A1206**
K. AJIT KUMAR REDDY	**17K81A1224**
D. SREENIDHI	**17K81A1207**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin’s Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R. Nagaraju**, Head of the Department, Information Technology, St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Mr. P. GANESH KUMAR**, Assistant Professor, Department of InformationTechnology, St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

G. AKANKSHA	**17K81A1213**
Ch. SAI KRISHNA PRASAD	**17K81A1206**
K. AJIT KUMAR REDDY	**17K81A1224**
D. SREENIDHI	**17K81A1207**

INDEX

Title **Page No**

ABSTRACT

Supermarket is the place where customers come to purchase their daily essential products and pay for them through traditional billing. Nowadays, if a consumer would like to buy something at a shopping mall, consumers need to take the particular items from the display shelf and then queue up and wait for their turn to make payment. Problem will surely arise when the size of a shopping mall is relatively huge and sometimes consumers don’t even know where certain items are placed. Besides, consumers also need to queue for a long time at the cashier to wait for turn to make payment, We need to calculate how many products are sold and generate the bill for the customer. We propose a automated billing system that also helps the customer to have a hassle-free shopping experience. We have 2 users in the system. First one is the administrator who will decide the taxes and commissions on the products and can see the report of any product. He is the one who will decide the products available for customers. The second one is the customer or the billing manager who can purchase the items available or can make the bill for the customers. This can also be used for online purchasing as the customer can access it easily. To develop a supermarket basket that assists the customer to locate and select products & inform them on the products details in the shopping arena. Additionally, with each product identified uniquely and support billing and inventory updates. We develop smart shopping system for the customer that assists the customer to locate the shelves where the product. Also by using the concept of market basket analysis we can solve the problems of the customers to find the items related to that product.

LIST OF FIGURES

1. INTRODUCTION

Supermarket is the place where customers come to purchase their daily essential products and pay for them through traditional billing. We need to calculate how many products are sold and generate the bill for the customer. We have 2 users in the system. First one is the administrator who will decide the taxes and commissions on the products and can see the report of any product. He is the one who will decide the products available for customers. The second one is the customer or the billing manager who can purchase the items available or can make the bill for the customers. This can also be used for online purchasing as the customer can access it easily. Nowadays, if a consumer would like to buy something at a shopping mall, consumers need to take the particular items from the display shelf and then queue up and wait for their turn to make payment. Problem will surely arise when the size of a shopping mall is relatively huge and sometimes consumers dont even know where certain items are placed. Besides, consumers also need to queue for a long time at the cashier to wait for turn to make payment, The time taken for consumers to wait for the consumers in front of the queue to scan every single item and then followed by making payment will definitely take plenty of time. This condition will surely become worst during the season of big sales or if the shopping mall still uses the conventional way to key in the price of every item by hand to the cash register. On the other hand, consumers often have to worry about plenty of things when going to the shopping mall. While doing survey we found that most of the people prefer to leave the shopping mall instead of waiting in long queues to buy a few products. People find it difficult to locate the product they wanted to buy, after selecting product they need to stand in a long queue for billing and payment. To try to solve the problems previously identified, recent years have seen the appearance of several technological solutions for hypermarket assistance. All such solutions share the same objectives to save consumers. In the present scenario, it is essential to have an automatic billing system for shopping malls, supermarket and other wholesale & retail stores. Numerous billing systems like barcode scanning mechanism-based systems or tag-based systems are available in the market. It is important to replace such existing system with better and robust systems so hereby we proposed “Supermarket billing system using webcam". In this system, the basic fundamental is barcode scanning for products, but we replace the conventional barcode scanner for faster and better results.

1.1 PROJECT OVERVIEW

The project is employed for automating the billing system in supermarkets, the database of this project will consists of some predefines shapes. The camera will capture the image of goods, it will find the objects which are predefined then it compares with the database, the software part will calculate the amount of bill. Now there will be two sorts of customers registered and non-registered, if the customer is registered then the amount of bill can directly be debited from his account. Barcodes are widely used in many grocery supermarkets like Hypermarket, D-mart, etc.. In our prototype, the android phone is being used as a barcode scanner for simple, better and portable barcode scanner. This scanner is connected wirelessly to MCU via Bluetooth module. MCU is also connected to PC/Laptop for creating the database of all customers, their products, and bills. This database also tracks the total sale and number of goods sold per day. In addition, RFID technology is implemented in this system for payment through card-based system. Simulation and hardware-based results are proposed in this paper. Unstaffed retail shop has been emerging in the past few years and significantly affected conventional shopping styles. In this field unmanned retail container plays an important role, it can highly influence the user shopping experience, the traditional method on weighing sensors cannot sense what the customer is taking. This paper proposes a smart unstaffed retail shop scheme based on image processing & open CV python which aiming at exploring the feasibility of implementing the unstaffed retail shopping style. The merits of this project are that it use open CV python which gives more than 98% accuracy & It is better than manual counting while the demerits are employability will be decreased because one machine can do work of many persons. To try to solve the problems previously identified, recent years have seen the appearance of several technological solutions for hypermarket assistance. All such solutions share the same objectives to save consumers.

1.2 PROJECT OBJECTIVES

The objectives for the smart shopping cart system project is to make the shopping easy for the customer in the supermarket and can save the time of the customer waiting in the queue as the bill is already made in the customer's screen by individually scanning their product and add into their cart. We always see that in a big Shoppe the customer fond to be hard to find the products they need to ask for the helper or the owner of the Shoppe and also, they need hold up in the line in the billing counter. Sometimes might be finding products is easy than waiting in the billing queue because it consumes more time of the customer. So now by taking the motivation of this scenario which was regularly done in all the Shoppe we are designing this system which can be benefited for the customer. To provide faster service at the checkouts this in the advantage for shop owners is that they will require fewer cashiers, which will result in a huge reduction in their cost. To develop a system which allows customer to pre decided budget and only buys the essential commodities actually needed by him, also the system aids. To remove the long queues at the billing counter. To develop the profitable system for the shopping centers this reduces the number of billing counters and in turn will help in reducing employee costs significantly.

1.3 SCOPE OF THE PROJECT

To develop a supermarket basket that assists the customer to locate and select products & inform them on the products details in the shopping arena. Additionally, with each product identified uniquely and support billing and inventory updates. We develop smart shopping system for the customer that assists the customer to locate the shelves where the product. Also by using the concept of market basket analysis we can solve the problems of the customers to find the items related to that product. The best and most useful example of this market basket analysis is that if a customer purchases bread then he will also purchases the related items that is better using these concepts we can make customer to purchase the related products.

The scope of the project is described as follows.

- Calculate the bill.
- Give the bill to the customers.
- Store how many products are sold.
- Store products and their prices with the information.
- Set the rate of taxes and commission on products.
- Can see the report of the product in a fixed period of time.
- Change the Graphical User Interface of the system.

1.4 ORGANISATION OF CHAPTERS

1.4.1 INTRODUCTION

Nowadays, if a consumer would like to buy something at a shopping mall, consumers need to take the particular items from the display shelf and then queue up and wait for their turn to make payment. Problem will surely arise when the size of a shopping mall is relatively huge and sometimes consumers dont even know where certain items are placed. Besides, consumers also need to queue for a long time at the cashier to wait for turn to make payment, The time taken for consumers to wait for the consumers in front of the queue to scan every single item and then followed by making payment will definitely take plenty of time. This condition will surely become worst during the season of big sales or if the shopping mall still uses the conventional way to key in the price of every item by hand to the cash register. On the other hand, consumers often have to worry about plenty of things when going to the shopping mall. While doing survey we found that most of the people prefer to leave the shopping mall instead of waiting in long queues to buy a few products. People find it difficult to locate the product they wanted to buy, after selecting product they need to stand in a long queue for billing and payment. To try to solve the problems previously identified, recent years have seen the appearance of several technological solutions for hypermarket assistance. All such solutions share the same objectives to save consumers.

1.4.2 LITERATURE SURVEY

A number of methods are proposed by researchers in this domain. B. Ananthabarathi proposed High Speed Billing System in which RF detector is placed inside the shopping cart which is linked to the server for billing [3]. R.Rajeshkumar, R.Mohanraj, M.Varatharaj proposed Smart Trolley in which they have used RFID cards for each product and RFID reader with MCU on each trolley for calculating the bills while shopping [4]. P. Chandrasekar, T. Sangeetha have proposed Smart Shopping Cart with Zigbee and RFID in which they utilize RFID cards for each product along with Product Identification Device (PID) for the trolley which is used for calculation of products and bill. This approach used Zigbee for transmitting the billing details to central billing system [1]. Few more researchers have proposed system for billing management but most of the methods are similar in nature and used MCU plus communication module based system for each trolley [5], [6], [7].

1.4.3 REQUIREMENTS SPECIFICATION

SOFTWARE REQUIREMENTS

❖ Operating system	:	Windows Family.
❖ Coding Language	:	Python.
❖ Front End	:	Python.
❖ Designing	:	HTML, CSS, Java Script
❖ IDE	:	PyCharm.
❖ Data Base	:	MySQL.

HARDWARE REQUIREMENTS

❖ Processor	:	Any Update Processor.
❖ RAM	:	Min. 4GB.
❖ HARD DISK	:	Min. 100GB.

1.4.4 SOFTWARE DEVELOPMENT ANALYSIS

Machine Learning

Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data. Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here. At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into classificatio*n* tasks and regression tasks: in classification, the labels are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section. Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as clustering and dimensionality reduction. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

1.4.5 PROJECT SYSTEM DESIGN

- **Add Product Details:** To build project I used some sample products image to train product identification models
- **Train Model:** In this Module screen train model generated with 100% accuracy and now show product to web cam.
- **Add/Remove Product from basket**: To allow application to identify product image and then show in text area and if we again show same product then application will remove from text area

1.4.6 PROJECT CODING

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. 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. Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time. Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc. The biggest strength of Python is huge collection of standard library which can be used for the following-

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like Opencv, Pillow)
- Web scraping (like Scrapy, BeautifulSoup, Selenium)
- Test frameworks
- Multimedia

The libraries used in the project are:

TKinter: Tkinter is a standard GUI (graphical user interface) package. Tkinter is Python's default GUI module and also the most common way that is used for GUI programming in Python.

Matplotlib: Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.

Numpy: Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

TensorFlow: TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks.

1.4.7 PROJECT 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.

Types of tests

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.

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.

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

System testing: System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results.

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.

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.

Unit testing: Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Integration testing: Integration testing is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

User Acceptance testing: 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.

1.4.8 INPUT SCREENS

The Input Screen allows users to search for transactions using three search options, Quick Search, Passenger Search, and Transaction Search. Common Search Options are applied to these search options to refine the search results.

Objectives:

- 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.
- 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.
- 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

1.4.9 OUTPUT SCREENS

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

Objectives:

- 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.
- Select methods for presenting information.
- Create document, report, or other formats that contain information produced by the system.
- 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.

1.4.10 CONCLUSIONS

In this Python project, the users are also provided an option to purchase items from the supermarket. The user can view items and then purchase the items which they need. To buy an item, the user needs to enter the product name and then click enter to confirm. This system then displays a message saying the user to pay the price of the item in the counter. In the modern era, people have more income to spend and lesser time to spend, so they typically opt for supermarkets for grocery. Truly the client is in a position & absolves to opt for product from large on the market varieties which attract the large customers mainly in big cities thus therefore long queues of shoppers are seen at these stores. In several cases, the barcode is either broken or there is also downside in reading barcode because of lighting effects, low resolution etc.

2. LITERATURE SURVEY

2.1 SURVEY ON BACKGROUND

"Image Processing System for Automatic Segmentation and Yield Prediction of fruits using Open CV." (2018)

This paper proposes an image processing system for automatic segmentation and yield prediction of fruits is proposed on the basis of color and shape features is being performed. Initially the preprocessing is done on input fruit tree images. Then it is converted from RGB to HSV color space to detect the fruit region from its background. Color thresholding is used to mask the desired colors. Gaussian filter is used to remove noise. The contour of the image is taken. Then these images are processed by image processing algorithm. Color and shape based counting of fruit is presented at the output. The edge detection and combination of a circular fitting algorithm is applied for the automatic segmentation and automatic counting of fruits in the image. Different types of fruits (orange/tangerine, pomegranate, apple, lemon, mango, cherry) are used for automatic counting. Open CV Python software is used to perform the required image processing operations.

"Object detection and recognition of intelligent service robot based on deep learning." (2018)

This study aims at the accuracy and real-time performance of object detection and recognition of service robot in complex scenes, an end to end object detection and recognition algorithm based on deep learning is proposed. Firstly, the local multi branch deep convolution neural network is adopted to enhance the feature representation capability of the model by enhancing the convolution module function. Then, combining the anchor point mechanism, the object class and position regression prediction is carried out on the multi-layer feature map. When the local features and the global features are fully fused, the natural multi-scale detection and recognition is realized on multiple receptive fields.

"Object Detection and Recognition for Assistive Robots." (2017)

This study presents a vision system for assistive robots that is able to detect and recognize objects from a visual input in ordinary environments in real time. The system computes color, motion, and shape cues, combining them in a probabilistic manner to accurately achieve object detection and recognition, taking some inspiration from vision science. In addition, with the purpose of processing the input visual data in real time, a graphical processing unit (GPU) has been employed. The presented approach has been implemented and evaluated on a humanoid robot torso located at realistic scenarios. For further experimental validation, a public image repository for object recognition has been used, allowing a quantitative comparison with respect to other state-of-the-art techniques when realworld scenes are considered.

"New Object Detection, Tracking, and Recognition Approaches for Video Surveillance Over Camera Network." (2015)

This paper proposes a framework for achieving these tasks in a non-overlapping multiple camera network. A new object detection algorithm using mean shift (MS) segmentation is introduced, and occluded objects are further separated with the help of depth information derived from stereo vision. The detected objects are then tracked by a new object tracking algorithm using a novel Bayesian Kalman filter with simplified Gaussian mixture (BKF-SGM). It employs a Gaussian mixture (GM) representation of the state and noise densities and a novel direct density simplifying algorithm for avoiding the exponential complexity growth of conventional Kalman filters (KFs) using GM.

"Fast and Lightweight Object Detection Network: Detection and recognition on resource constrained devices." (2018)

The intrinsic ability of humans to rapidly detect, differentiate, and classify objects allows us to make quick decisions in regards to what we see. Several appliances can make use of fast and lightweight automated object detection for images or videos. Throughout the last five years, the technology industry has constantly introduced computational and hardware solutions, such as devices with impressive processing and storage capabilities. However, object detection methods usually require either high processing power or large storage availability, making it hard for resource constrained devices to perform the detection in real-time without a connection to a powerful server. The model presented in this paper requires only 95 megabytes of storage and took 113 ms in average per image running on a laptop CPU, making it suitable for standalone devices that can be used on the go.

“High Speed Billing System in Departmental Stores.” (2012)

The aim of this study was to make vending of goods in shops completely automatic billing in order to save time. The main idea of the project is to create a complete self functioning rapid billing and dispatch system in a super-market. The system basically consists of a cart fitted with an RF-detector linked to a billing server. When goods in departmental stores are added to the cart, the RF detector automatically detects the item type, quantity and sends the information to the billing server. The billing server simultaneously starts billing for the particular cart. In case the customer removes an item from the cart, the server immediately recognizes it and aligns the billing accordingly. Once the customer acknowledges the server to provide the bill, the server provides the gross amount. The customer on paying the bill is provided an acknowledgement by the server and the goods are dispatched to the customer by the time.

2.2 CONCLUSIONS ON SURVEY

The study is aimed to understand a system employed for automating the billing system in supermarkets, the database will consists of some predefines shapes. The study revealed that Python is the most effective and relevant technology to develop the intended aUtomated system along with support of it’s wide range of libraries like OpenCv and Haar Cascade. The challenge is to program the system using OpenCv and Haar Cascade algorithms which enable us to perform image processing, object detection and image tracing functions and to use to camera will capture the image of goods, it will find the objects which are predefined then it compares with the database, the software part will calculate the amount of bill. The most feasible option is to use a android phone, which will be used as a barcode scanner for simple, better and portable barcode scanner. RFID technology is the most effective and efficient mechanism to be used in our proposed system. Unstaffed retail shop has been emerging in the past few years and significantly affected conventional shopping styles. In this field unmanned retail container plays an important role, it can highly influence the user shopping experience, the traditional method on weighing sensors cannot sense what the customer is taking. This study also helped us to conclude that a smart unstaffed retail shop scheme based on image processing & open CV python which aiming at exploring the feasibility of implementing the unstaffed retail shopping style. The merits using open CV python which gives more than 98% accuracy & It is better than manual counting while the demerits are employability will be decreased because one machine can do work of many persons. To try to solve the problems previously identified, recent years have seen the appearance of several technological solutions for hypermarket assistance.

3. SOFTWARE AND HARDWARE REQUIREMENTS

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

3.1 SOFTWARE REQUIREMENTS

❖ Operating system	:	Windows Family.
❖ Coding Language	:	Python.
❖ Front End	:	Python.
❖ Designing	:	HTML, CSS, Java Script
❖ IDE	:	PyCharm.
❖ Data Base	:	MySQL.

3.2 HARDWARE REQUIREMENTS

❖ Processor	:	Any Update Processor.
❖ RAM	:	Min. 4GB.
❖ Hard disk	:	Min. 100GB.

Functional Requirements

- Graphical User interface with the User.

Debugger and Emulator

- Any Browser (Particularly Chrome)

4. SOFTWARE DEVELOPMENT ANALYSIS

4.1 OVERVIEW OF THE PROBLEM

The existing system is a traditional billing system, the billing is done by barcode scanner we need to detect every barcode attached to every item in purchased item list. When all the items get scanned the price and quantity of items is automatically get into the system and then the bill is get generated. Customers can pay bill through credit/debit cards or by cash. But it is a time consuming process for the billing purpose, so that the waiting time to pay the bill is increased. To overcome the time consuming process the RFID based smart trolley is proposed, along with the help of aurdino and python technologies.

4.2 DEFINE THE PROBLEM

The aim of the proposed system is to To develop a supermarket basket that assists the customer to locate and select products &inform them on the products details in the shopping arena. Additionally, with each product identified uniquely and support billing and inventory updates. We develop smart shopping system for the customer that assists the customer to locate the shelves where the product. Also by using the concept of market basket analysis we can solve the problems of the customers to find the items related to that product. The best and most useful example of this market basket analysis is that if a customer purchases bread then he will also purchases the related items that is better than using these traditional billing concepts and we can provide the customer with better purchase experience and help him find the related products. We have two users- admin and customer, and there are two types of customers registered and non-registered customers. Registered customers are provided with a membership card which can be used to access the application and pay the bill instead of credit or debit card. The proposed system is developed using python technology, using the relevant libraries and algorithms like OpenCV and Haar cascade.

4.3 MODULES OVERVIEW

The project we built contains three modules, each module having unique and essential functionality. Our project is to develop a automatic supermarket billing system which uses a web cam to perform the task of image processing, object detection, training the system to detect the added products and displaying the details of the product added to the basket. The three modules included in the system performs the intended operations. The first module, “Add Product Details” is the first module which is used to add a product details to the database, the system opens the camera when you click on this module. The second module, “Train Model” is used to train the system to check the accuracy of the image we added using the webcam, the image must be scanned and added to the module at least 8times to reach the intended 100% accuracy. The final module, “Add/ Remove Product from Basket” enables us to update the list of products, we either add or remove the products according to the availability in stock and personal preference of the customer.

4.4 MODULE DEFINITIONS

- **Add Product Details:** To build the project we used some sample products image to train product identification models. This module enables us to details of the detected product image.
- **Train Model:** In this Module screen train model generated with 100% accuracy and then show the product to web cam.
- **Add/Remove Product from basket**: To allow application to identify product image and then show in text area and if we again show same product then application will remove from text area.

4.5 MODULE FUNCTIONALITIES

The project is employed for automating the billing system in supermarkets, the database of this project will consists of some predefines shapes. The camera will capture the image of goods, it will find the objects which are predefined then it compares with the database, the software part will calculate the amount of bill.

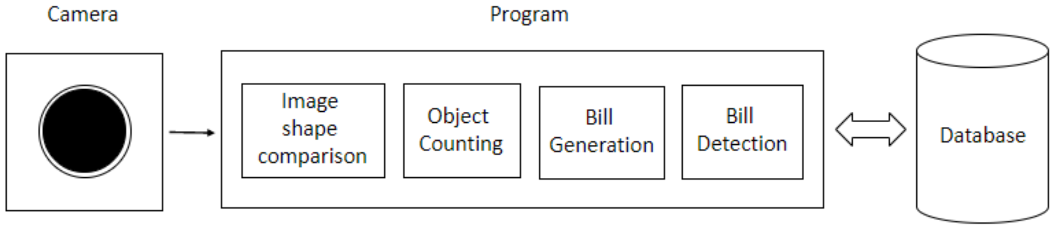


FIG. 4.2.1 SYSTEM ARCHITECTURE

Now there will be two sorts of customers registered and non-registered, if the customer is registered then the amount of bill can directly be debited from his account. Barcodes are widely used in many grocery supermarkets like Hypermarket , D-mart, etc.. In our prototype, the android phone is being used as a barcode scanner for simple, better and portable barcode scanner. This scanner is connected wirelessly to MCU via Bluetooth module. MCU is also connected to PC/Laptop for creating the database of all customers, their products, and bills. This database also tracks the total sale and number of goods sold per day. In addition, RFID technology is implemented in this system for payment through card-based system.

5. PROJECT SYSTEM DESIGN

5.1 DATA FLOW DIAGRAMS

Also known as DFD, Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation. Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow.

GOALS:

- To graphically represent the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system.
- The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams. DFD has often been used due to the following reasons:
- To represent logical information flow of the system
- Helps in determination of physical system construction requirements
- Simplicity of notation
- To establish manual and automated systems requirements

5.1.1 LEVEL 0 DFD

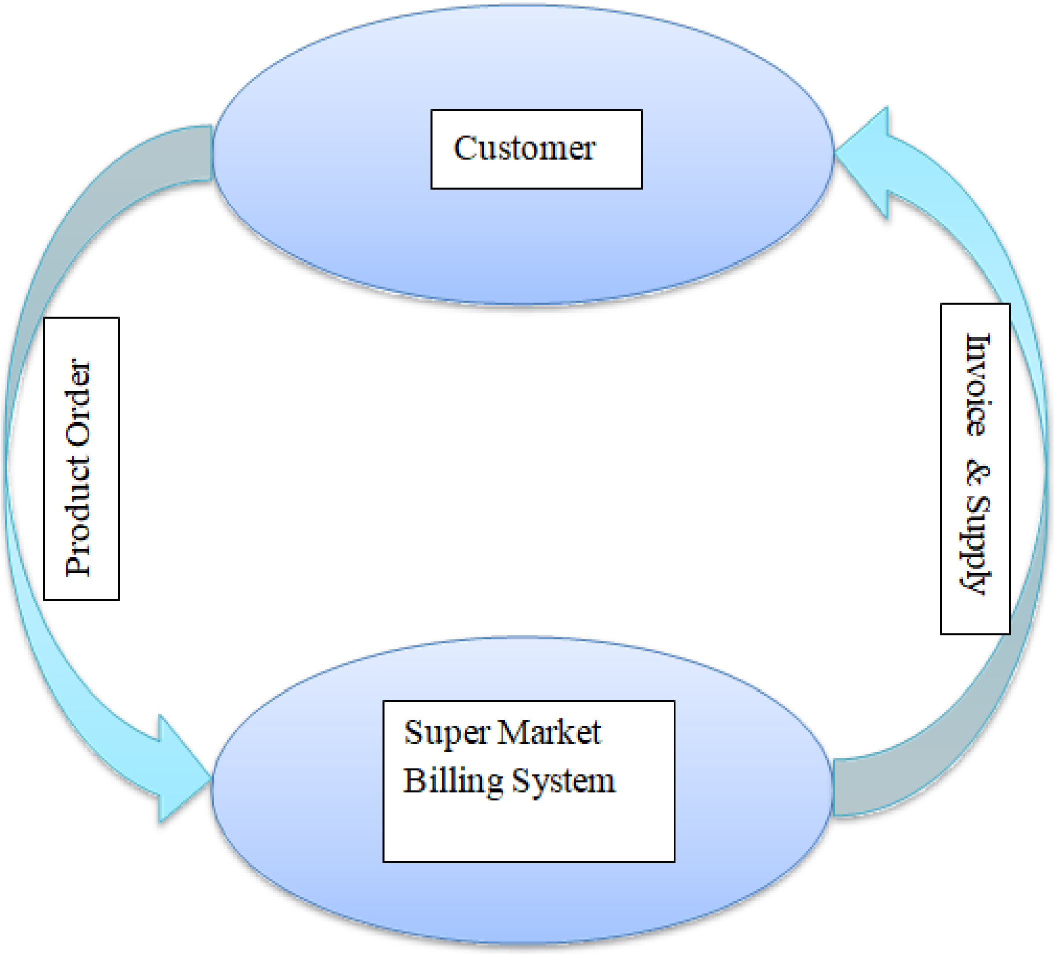


FIG. 5.1.1 LEVEL 0 DFD

5.1.2 LEVEL 1 DFD

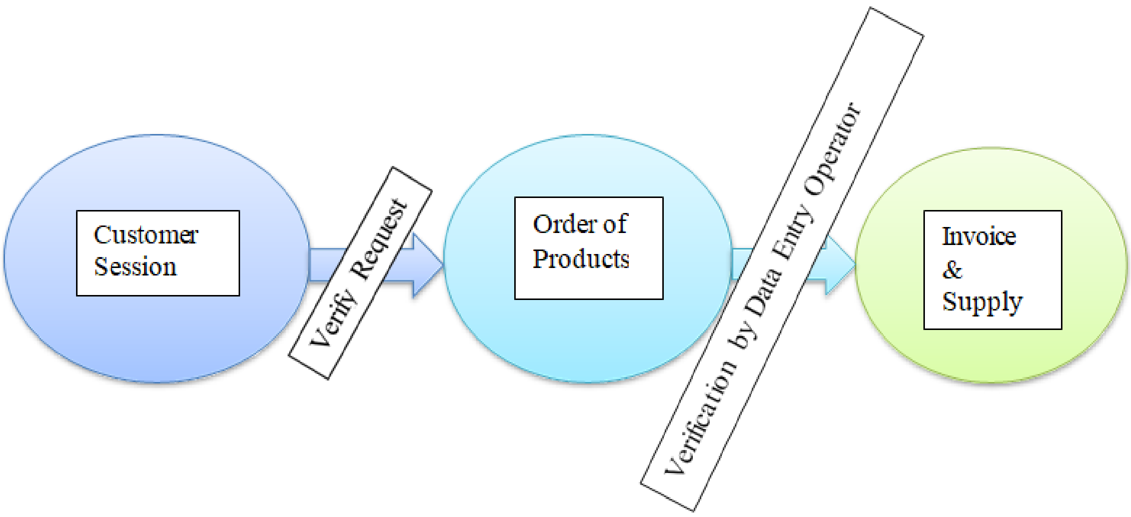


FIG. 5.1.2 LEVEL 1 DFD

5.1.3 LEVEL 2 DFD

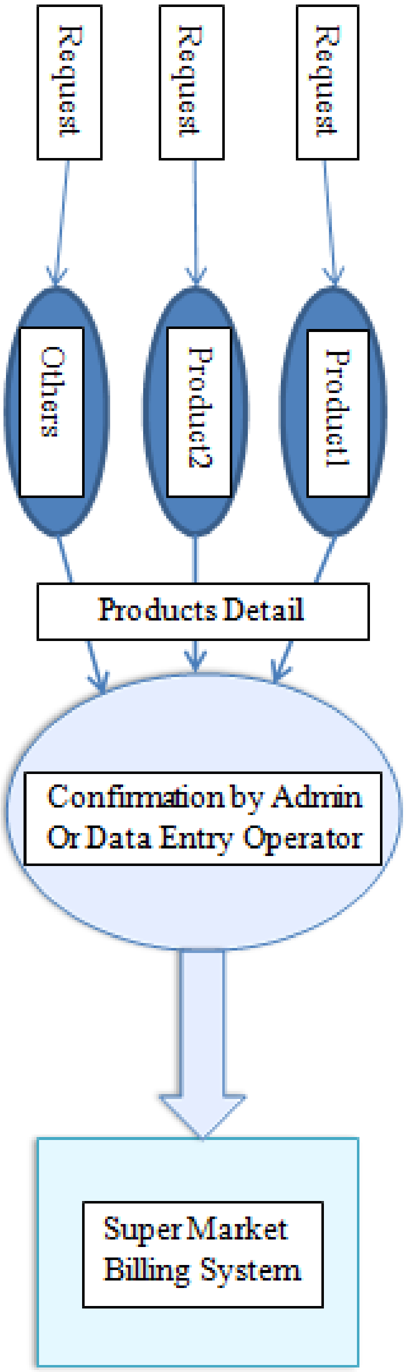


FIG. 5.1.3 LEVEL 2 DFD

5.2 E-R DIAGRAMS

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships. ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.

The primary reasons for using ER diagrams- Helps you to define terms related to entity relationship modeling Provide a preview of how all your tables should connect, what fields are going to be on each table. Helps to describe entities, attributes, relationships. ER diagrams are translatable into relational tables which allows you to build databases quickly. ER diagrams can be used by database designers as a blueprint for implementing data in specific software applications. The database designer gains a better understanding of the information to be contained in the database with the help of ERP diagram. ERD Diagram allows you to communicate with the logical structure of the database to users

Entity Relationship Diagram Symbols & Notations:

It mainly contains three basic symbols which are rectangle, oval and diamond to represent relationships between elements, entities and attributes. There are some sub-elements which are based on main elements in ERD Diagram. ER Diagram is a visual representation of data that describes how data is related to each other using different ERD Symbols and Notations.

Following are the main components and its symbols in ER Diagrams:

- **Rectangles:** This Entity Relationship Diagram symbol represents entity types
- **Ellipses :** Symbol represent attributes
- **Diamonds:** This symbol represents relationship types

- **Lines:** It links attributes to entity types and entity types with other relationship types
- **Primary key:** attributes are underlined
- **Double Ellipses:** Represent multi-valued attributes

Components of ER diagrams:

This model is based on three basic components:

- Entities
- Attributes
- Relationships

GOALS:

- Systematically analyze data requirements to produce a well-designed database.
- Representing real-world entities and the relationships between them.
- Creating an ER Model in DBMS is considered as a best practice before implementing your database.
- Analyze data requirements systematically to produce a well-designed database. So, it is considered a best practice to complete ER modeling before implementing your database.

5.2.1 E-R DIAGRAM

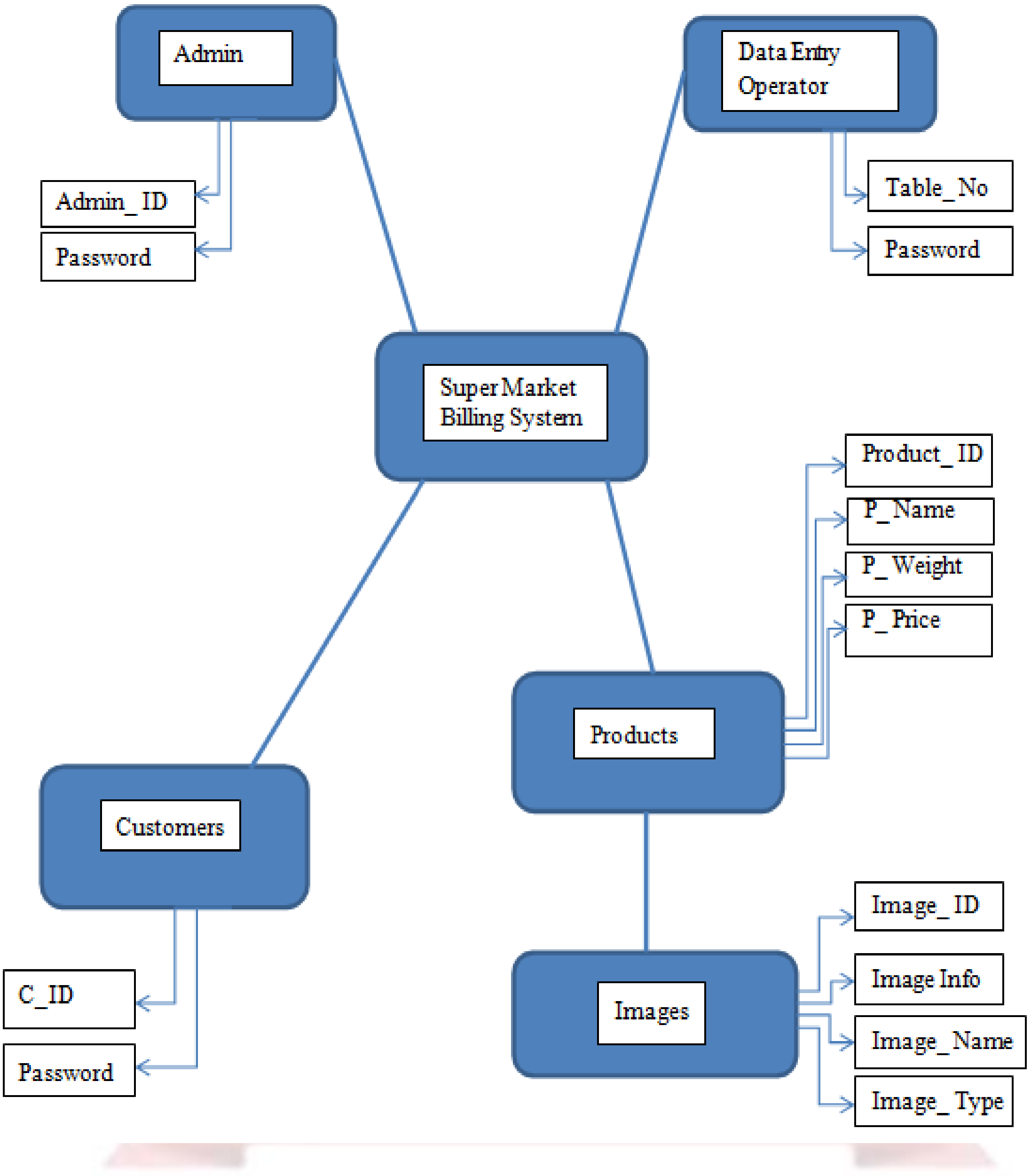


FIG. 5.2.1 E-R DIAGRAM

5.3 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modeling language.
- Encourage the growth of OO tools market.
- Support higher level development concepts such as collaborations, frameworks, patterns and components.
- Integrate best practices.

5.3.1 USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

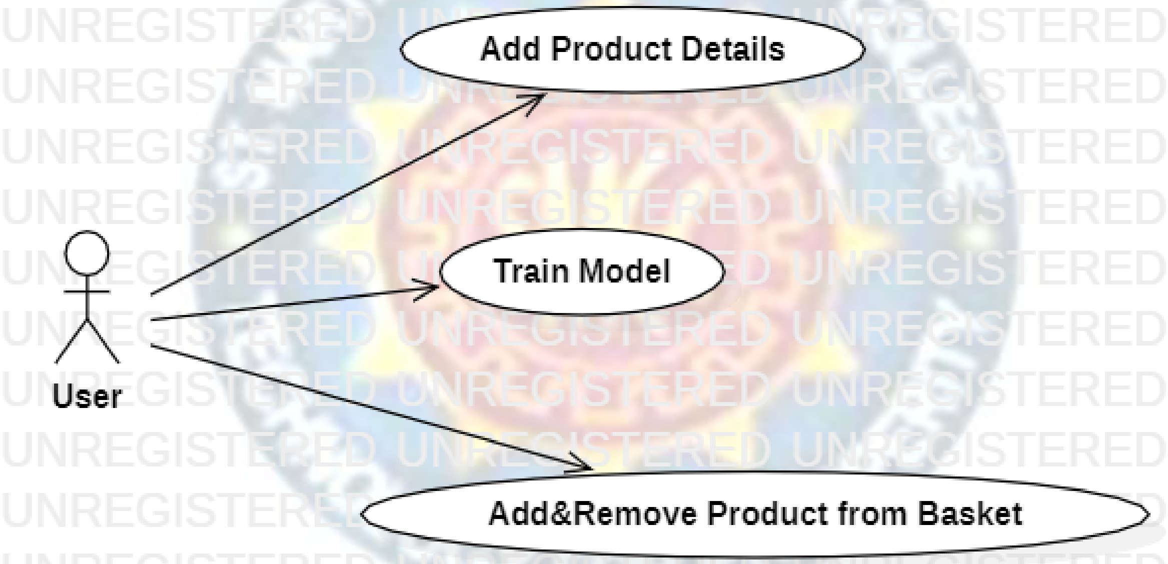


FIG. 5.3.1 USE CASE DIAGRAM

5.3.2 CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

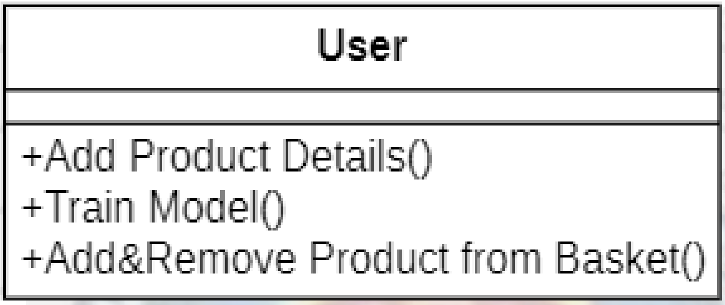


FIG. 5.3.2 CLASS DIAGRAM

5.3.3 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

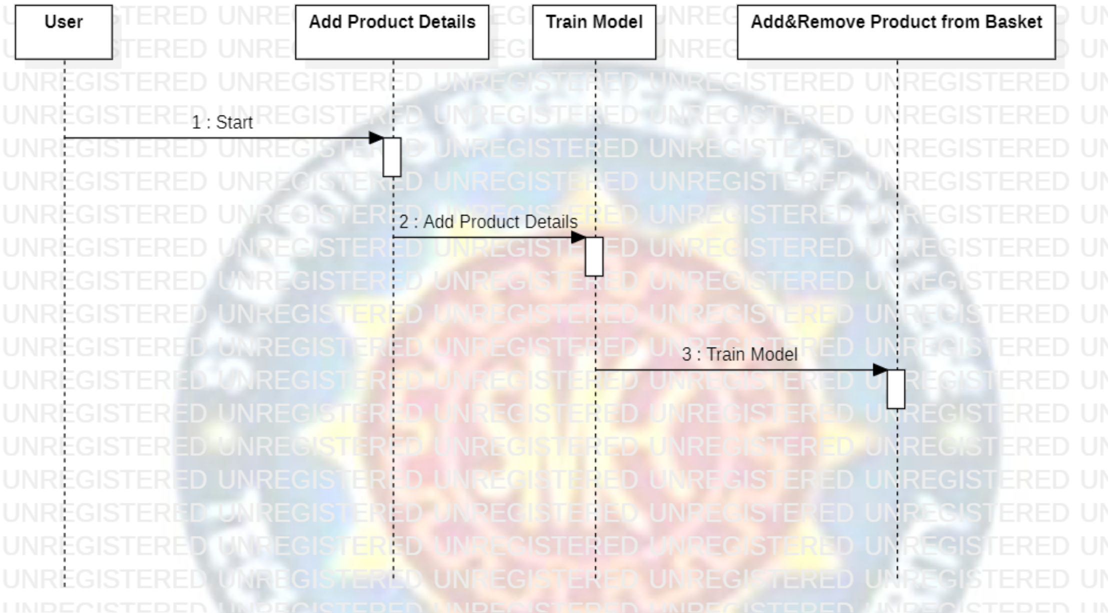


FIG. 5.3.3 SEQUENCE DIAGRAM

5.3.4 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

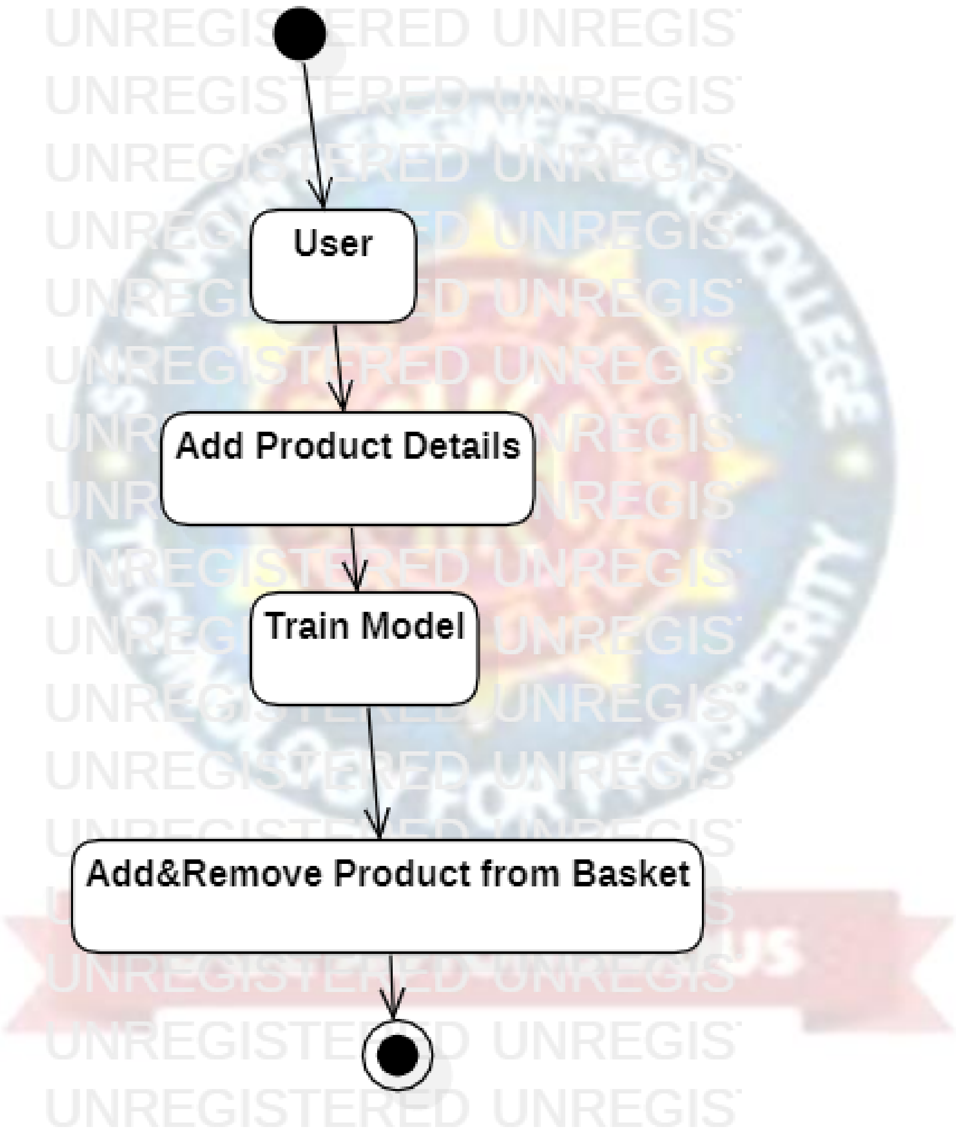


FIG. 5.3.4 ACTIVITY DIAGARAM

6. PROJECT CODING

6.1 TECHNOLOGIES

PYTHON

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. C 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. C Python is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management.

Python is currently the most widely used multi-purpose, high-level programming language.

Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.

Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.

Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.

The biggest strength of Python is huge collection of standard library which can be used for the following:

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like Opencv, Pillow)
- Web scraping (like Scrapy, BeautifulSoup, Selenium)
- Test frameworks
- Multimedia

Advantages of Python:

Let’s see how Python dominates over other languages.

1. Extensive Libraries

Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don’t have to write the complete code for that manually.

2. Extensible

As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add scripting capabilities to our code in the other language.

4. Improved Productivity

The language’s simplicity and extensive libraries render programmers more productive than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

5. IOT Opportunities

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet Of Things. This is a way to connect the language with the real world.

6. Simple and Easy

When working with Java, you may have to create a class to print ‘Hello World’. But in Python, just a print statement will do. It is also quite easy to learn, understand, and code. This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.

7. Readable

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and indentation is mandatory. This further aids the readability of the code.

8. Object-Oriented

This language supports both the procedural and object-oriented programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the encapsulation of data and functions into one.

9. Free and Open-Source

Like we said earlier, Python is freely available. But not only can you download Python for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

10. Portable

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn’t the same with Python. Here, you need to code only once, and you can run it anywhere. This is called Write Once Run Anywhere (WORA). However, you need to be careful enough not to include any system-dependent features.

11. Interpreted

Lastly, we will say that it is an interpreted language. Since statements are executed one by one, debugging is easier than in compiled languages.

Advantages of Python Over Other Languages

1. Less Coding

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don't have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

2. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

The 2019 Github annual survey showed us that Python has overtaken Java in the most popular programming language category.

3. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and machine learning, automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

Disadvantages of Python

So far, we've seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let's now see the downsides of choosing Python over another language.

1. Speed Limitations

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in slow execution. This, however, isn't a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

2. Weak in Mobile Computing and Browsers

While it serves as an excellent server-side language, Python is much rarely seen on the client-side. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called Carbonnelle.
The reason it is not so famous despite the existence of Brython is that it isn't that secure.

3. Design Restrictions

As you know, Python is dynamically-typed. This means that you don't need to declare the type of variable while writing the code. It uses duck-typing. But wait, what's that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can raise run-time errors.

4. Underdeveloped Database Access Layers

Compared to more widely used technologies like JDBC (Java DataBase Connectivity) and ODBC (Open DataBase Connectivity), Python's database access layers are a bit underdeveloped. Consequently, it is less often applied in huge enterprises.

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.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Python Installation

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 Quickstart

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":

Virtual Environments and Packages

Introduction

Python applications will often use packages and modules that don’t come as part of the standard library. Applications will sometimes need a specific version of a library, because the application may require that a particular bug has been fixed or the application may be written using an obsolete version of the library’s interface.

This means it may not be possible for one Python installation to meet the requirements of every application. If application A needs version 1.0 of a particular module but application B needs version 2.0, then the requirements are in conflict and installing either version 1.0 or 2.0 will leave one application unable to run.

The solution for this problem is to create a virtual environment, a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.

Different applications can then use different virtual environments. To resolve the earlier example of conflicting requirements, application A can have its own virtual environment with version 1.0 installed while application B has another virtual environment with version 2.0. If application B requires a library be upgraded to version 3.0, this will not affect application A’s environment.

Creating Virtual Environments

The module used to create and manage virtual environments is called venv. venv will usually install the most recent version of Python that you have available. If you have multiple versions of Python on your system, you can select a specific Python version by running python3 or whichever version you want.

To create a virtual environment, decide upon a directory where you want to place it, and run the venv module as a script with the directory path:

```
python3 -m venv tutorial-env
```

This will create the tutorial-env directory if it doesn't exist, and also create directories inside it containing a copy of the Python interpreter, the standard library, and various supporting files.

A common directory location for a virtual environment is .venv. This name keeps the directory typically hidden in your shell and thus out of the way while giving it a name that explains why the directory exists. It also prevents clashing with .env environment variable definition files that some tooling supports.

Once you've created a virtual environment, you may activate it.

On Windows, run:

```
tutorial-env\Scripts\activate.bat
```

On Unix or MacOS, run:

```
source tutorial-env/bin/activate
```

(This script is written for the bash shell. If you use the csh or fish shells, there are alternate activate.csh and activate.fish scripts you should use instead.)

Activating the virtual environment will change your shell's prompt to show what virtual environment you're using, and modify the environment so that running python will get you that particular version and installation of Python. For example:

```
$ source ~/envs/tutorial-env/bin/activate

(tutorial-env) $ python

Python 3.5.1 (default, May  6 2016, 10:59:36)

  ...

>>> import sys

>>>sys.path
```

```
['', '/usr/local/lib/python35.zip', ...,

'~/envs/tutorial-env/lib/python3.5/site-packages']

>>>
```

MACHINE LEARNING

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.

Challenges in Machines Learning

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are:

Quality of data: Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task: Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons: As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems: Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of Over-fitting & Under-fitting: If the model is over-fitting or under-fitting, it cannot be represented well for the problem.

Curse of dimensionality: Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment: Complexity of the ML model makes it quite difficult to be deployed in real life.

Applications of Machines Learning

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML:

- Emotion analysis
- Sentiment analysis
- Error detection and prevention
- Weather forecasting and prediction
- Stock market analysis and forecasting
- Speech synthesis
- Speech recognition
- Customer segmentation

- Object recognition
- Fraud detection
- Fraud prevention
- Recommendation of products to customer in online shopping

OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene.

Features of OpenCV Library

Using OpenCV library, you can-

- Read and write images
- Capture and save videos
- Process images (filter, transform)
- Perform feature detection
- Detect specific objects such as faces, eyes, cars, in the videos or images.
- Analyze the video, i.e., estimate the motion in it, subtract the background, and track objects in it.

OpenCV was originally developed in C++. In addition to it, Python and Java bindings were provided. OpenCV runs on various Operating Systems such as windows, Linux, OSx, FreeBSD, Net BSD, Open BSD, etc.

OpenCV Library Modules

Following are the main library modules of the OpenCV library.

Core Functionality

This module covers the basic data structures such as Scalar, Point, Range, etc., that are used to build OpenCV applications. In addition to these, it also includes the multidimensional array Mat, which is used to store the images. In the Java library of OpenCV, this module is included as a package with the name org.opencv.core.

Image Processing

This module covers various image processing operations such as image filtering, geometrical image transformations, color space conversion, histograms, etc. In the Java library of OpenCV, this module is included as a package with the name org.opencv.imgproc.

Video

This module covers the video analysis concepts such as motion estimation, background subtraction, and object tracking. In the Java library of OpenCV, this module is included as a package with the name org.opencv.video.

Video I/O

This module explains the video capturing and video codecs using OpenCV library. In the Java library of OpenCV, this module is included as a package with the name org.opencv.videoio.

calib3d

This module includes algorithms regarding basic multiple-view geometry algorithms, single and stereo camera calibration, object pose estimation, stereo correspondence and elements of 3D reconstruction. In the Java library of OpenCV, this module is included as a package with the name org.opencv.calib3d.

features2d

This module includes the concepts of feature detection and description. In the Java library of OpenCV, this module is included as a package with the name org.opencv.features2d.

Objdetect

This module includes the detection of objects and instances of the predefined classes such as faces, eyes, mugs, people, cars, etc. In the Java library of OpenCV, this module is included as a package with the name org.opencv.objdetect.

Highgui

This is an easy-to-use interface with simple UI capabilities. In the Java library of OpenCV, the features of this module is included in two different packages namely, org.opencv.imgcodecs and org.opencv.videoio.

6.2 CODING

SUPERMARKET.PY

```
import tkinter
import cv2
import PIL.Image, PIL.ImageTk
from tkinter import simpledialog
import time
from tkinter import messagebox
import os
from keras.utils.np_utils import to_categorical
import numpy as np
from keras.layers import  MaxPooling2D
from keras.layers import Dense, Dropout, Activation, Flatten
from keras.layers import Convolution2D
from keras.models import Sequential
from keras.models import model_from_json
```

```
import pickle
from tkinter import *
import random

class App:
    global classifier
    global labels
    global X_train
    global Y_train
    global prices
    global cart
    global text
    global person_id
    global img_canvas
    global cascPath
    global faceCascade
    global pid

    def __init__(self, window, window_title, video_source=0):
        global cart
        global text
        cart = []
        self.window = window
        self.window.title(window_title)
        self.window.geometry("1300x1200")
        self.video_source = video_source
        self.vid = MyVideoCapture(self.video_source)
        self.canvas = tkinter.Canvas(window, width = self.vid.width, height = 
self.vid.height)
        self.canvas.pack()
        self.font1 = ('times', 13, 'bold')
        self.btn_snapshot=tkinter.Button(window, text="Add Product Details", 
command=self.snapshot)
        self.btn_snapshot.place(x=10,y=50)
        self.btn_snapshot.config(font=self.font1)
        self.btn_train=tkinter.Button(window, text="Train Model",
command=self.trainmodel)
        self.btn_train.place(x=10,y=100)
        self.btn_train.config(font=self.font1)
        self.btn_predict=tkinter.Button(window, text="Add/Remove Product from
Basket", command=self.predict)
        self.btn_predict.place(x=10,y=150)
        self.btn_predict.config(font=self.font1)

        self.btn_person=tkinter.Button(window, text="Capture Person",
command=self.capturePerson)
        self.btn_person.place(x=10,y=200)
        self.btn_person.config(font=self.font1)

        self.img_canvas = tkinter.Canvas(window, width = 200, height = 200)
        self.img_canvas.place(x=10,y=250)

        self.text=Text(window,height=35,width=45)
        scroll=Scrollbar(self.text)
        self.text.configure(yscrollcommand=scroll.set)
        self.text.place(x=1000,y=50)
        self.text.config(font=self.font1)

        self.cascPath = "haarcascade_frontalface_default.xml"
        self.faceCascade = cv2.CascadeClassifier(self.cascPath)

        self.delay = 15
        self.update()
        self.window.mainloop()

    def getID(self,name):
        index = 0
        for i in range(len(labels)):
            if labels[i] == name:
                index = i
                break
        return index

    def capturePerson(self):
        option = 0
        ret, frame = self.vid.get_frame()
        img = frame

        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        faces = self.faceCascade.detectMultiScale(gray,1.3,5)
        print("Found {0} faces!".format(len(faces)))
        for (x, y, w, h) in faces:
            cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 2)
            img = frame[y:y + h, x:x + w]
            img = cv2.resize(img,(500,500))
            option = 1
        if option == 1:
            self.pid = random.randint(1000, 100000)
            cv2.imwrite("images/"+str(self.pid)+".jpg",img);
            cv2.imshow("Person ID : "+str(self.pid)+".jpg",img)
            c2.waitKey(0)
        else:
            messagebox.showinfo("Face or person not detected","Face or person not
detected")

    def snapshot(self):
        ret, frame = self.vid.get_frame()
        if ret:
            img = cv2.cvtColor(frame, cv2.COLOR_RGB2BGR)
            pname = simpledialog.askstring("Enter Product Name", "Enter Product
Name",parent=self.window)
```

```
            price = simpledialog.askfloat("Enter Product Price", "Enter Product Price",
parent=self.window, minvalue=1.0, maxvalue=100000.0)
            if not os.path.exists('Product/'+pname):
                os.makedirs('Product/'+pname)
            img_name = time.strftime("%d-%m-%Y-%H-%M-%S") + ".jpg"
            cv2.imwrite('Product/'+pname+'/'+img_name,img)
            f = open("details.txt", "a+")
            f.write(pname+","+str(price)+","+img_name+"\n")
            f.close()
            messagebox.showinfo("Product details saved","Product details saved")

    def trainmodel(self):
        global labels
        global X_train
        global Y_train
        global classifier
        global prices
        labels = []
        X_train = []
        Y_train = []
        prices = []
        path = 'Product'
        for root, dirs, directory in os.walk(path):
            for j in range(len(directory)):
                name = os.path.basename(root)
                if name not in labels:
                    labels.append(name)

        for i in range(len(labels)):
            cost = '0'
            with open("details.txt", "r") as file:
                for line in file:
                    line = line.strip('\n')
                    line = line.strip()
                    arr = line.split(",")
                    if arr[0] == labels[i] and cost == '0':
                        cost = arr[1]
            file.close()
            prices.append(cost)

        for root, dirs, directory in os.walk(path):
            for j in range(len(directory)):
                name = os.path.basename(root)
                img = cv2.imread(root+"/"+directory[j])

                img = cv2.resize(img, (256,256))
                im2arr = np.array(img)
                im2arr = im2arr.reshape(256,256,3)
                X_train.append(im2arr)
                Y_train.append(self.getID(name))
        X_train = np.asarray(X_train)
        Y_train = np.asarray(Y_train)
        print(Y_train)
        print(labels)
        print(prices)
        X_train = X_train.astype('float32')
        X_train = X_train/255

        test = X_train[3]
        cv2.imshow("aa",test)
        cv2.waitKey(0)
        indices = np.arange(X_train.shape[0])
        np.random.shuffle(indices)
        X_train = X_train[indices]
        Y_train = Y_train[indices]
        Y_train = to_categorical(Y_train)

        if os.path.exists('Model/model.json'):
            with open('Model/model.json', "r") as json_file:
                loaded_model_json = json_file.read()
                classifier = model_from_json(loaded_model_json)

            classifier.load_weights("Model/model_weights.h5")
            classifier.make_predict_function()
            print(classifier.summary())
            f = open('Model/history.pckl', 'rb')
            data = pickle.load(f)
            f.close()

            acc = data['accuracy']
            accuracy = acc[9] * 100
            messagebox.showinfo("Training model accuracy","Training Model Accuracy = "+str(accuracy))
        else:
            classifier = Sequential()
            classifier.add(Convolution2D(32, 3, 3, input_shape = (256, 256, 3), activation = 'relu'))
            classifier.add(MaxPooling2D(pool_size = (2, 2)))
            classifier.add(Convolution2D(32, 3, 3, activation = 'relu'))
            classifier.add(MaxPooling2D(pool_size = (2, 2)))
            classifier.add(Flatten())
            classifier.add(Dense(output_dim = 256, activation = 'relu'))
            classifier.add(Dense(output_dim = 4, activation = 'softmax'))
            print(classifier.summary())
            classifier.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = ['accuracy'])
            hist = classifier.fit(X_train, Y_train, batch_size=16, epochs=10, shuffle=True, verbose=2)
            classifier.save_weights('Model/model_weights.h5')
            model_json = classifier.to_json()
            with open("Model/model.json", "w") as json_file:
                json_file.write(model_json)
            f = open('Model/history.pckl', 'wb')
            pickle.dump(hist.history, f)
            f.close()
            f = open('Model/history.pckl', 'rb')
            data = pickle.load(f)
            f.close()
            acc = data['accuracy']
            accuracy = acc[9] * 100
            messagebox.showinfo("Training model accuracy","Training Model Accuracy = "+str(accuracy))

    def predict(self):
        ret, frame = self.vid.get_frame()
        img = cv2.cvtColor(frame, cv2.COLOR_RGB2BGR)
        img = cv2.resize(img, (256,256))
        im2arr = np.array(img)
        im2arr = im2arr.reshape(1,256,256,3)
        image = np.asarray(im2arr)
        image = image.astype('float32')
        image = image/255
        preds = classifier.predict(image)
        predict = np.argmax(preds)
        pname = labels[predict]
        print(str(pname)+" "+str(np.amax(preds)))
        if np.amax(preds) >= 0.85:
            cost = prices[predict]
            if pname in cart:
                cart.remove(pname)
            else:
                cart.append(pname)
            self.text.delete('1.0', END)
            total_amt = 0
            for i in range(len(cart)):
                for k in range(len(labels)):
                    if labels[k] == cart[i]:
                        cost = prices[k]
                        k = len(labels)
                total_amt = total_amt + float(cost)
                self.text.insert(END,"Product Name : "+cart[i]+"\n")
                self.text.insert(END,"Product Cost : "+cost+"\n\n")
            self.text.insert(END,"Total Amount : "+str(total_amt)+"\n\n")
        else:
            messagebox.showinfo("Unable to recognized product","Unable to recognized
product")


    def update(self):
        ret, frame = self.vid.get_frame()
        if ret:
            self.photo = PIL.ImageTk.PhotoImage(image = PIL.Image.fromarray(frame))
            self.canvas.create_image(0, 0, image = self.photo, anchor = tkinter.NW)
            self.window.after(self.delay, self.update)


class MyVideoCapture:
    def __init__(self, video_source=0):

        self.vid = cv2.VideoCapture(video_source)
        if not self.vid.isOpened():
            raise ValueError("Unable to open video source", video_source)
        self.width = self.vid.get(cv2.CAP_PROP_FRAME_WIDTH)
        self.height = self.vid.get(cv2.CAP_PROP_FRAME_HEIGHT)
        self.pid = 0

    def get_frame(self):
        if self.vid.isOpened():
            ret, frame = self.vid.read()
            if ret:
                return (ret, cv2.cvtColor(frame, cv2.COLOR_BGR2RGB))
            else:
                return (ret, None)
        else:
            return (ret, None)

    def __del__(self):
        if self.vid.isOpened():
            self.vid.release()
App(tkinter.Tk(), "Tkinter and OpenCV")
```

TEST.PY

```
import tkinter
import cv2
import PIL.Image, PIL.ImageTk
from tkinter import simpledialog
import time
from tkinter import messagebox
import os
from keras.utils.np_utils import to_categorical
import numpy as np
from keras.layers import  MaxPooling2D
from keras.layers import Dense, Dropout, Activation, Flatten
from keras.layers import Convolution2D
from keras.models import Sequential
from keras.models import model_from_json
import pickle
from tkinter import *
class App:
    global classifier
    global labels
    global X_train
    global Y_train
```

```
  global prices
  global cart
  global text

  def __init__(self, window, window_title, video_source=0):
    global cart
    global text
    cart = []
    self.window = window
    self.window.title(window_title)
    self.window.geometry("1300x1200")
    self.video_source = video_source
    self.vid = MyVideoCapture(self.video_source)
    self.canvas = tkinter.Canvas(window, width = self.vid.width, height =
self.vid.height)
    self.canvas.pack()
    self.font1 = ('times', 13, 'bold')
    self.btn_snapshot=tkinter.Button(window, text="Add Product Details",
command=self.snapshot)
    self.btn_snapshot.place(x=10,y=50)
    self.btn_snapshot.config(font=self.font1)
    self.btn_train=tkinter.Button(window, text="Train Model",
command=self.trainmodel)
    self.btn_train.place(x=10,y=100)
    self.btn_train.config(font=self.font1)
    self.btn_predict=tkinter.Button(window, text="Add/Remove Product from
Basket", command=self.predict)
    self.btn_predict.place(x=10,y=150)
    self.btn_predict.config(font=self.font1)

    self.text=Text(window,height=35,width=45)
    scroll=Scrollbar(self.text)
    self.text.configure(yscrollcommand=scroll.set)
    self.text.place(x=1000,y=50)
```

```
        self.text.config(font=self.font1)

        self.delay = 15
        self.update()
        self.window.mainloop()

    def getID(self,name):
        index = 0
        for i in range(len(labels)):
            if labels[i] == name:
                index = i
                break
        return index

    def snapshot(self):
        ret, frame = self.vid.get_frame()
        if ret:
            img = cv2.cvtColor(frame, cv2.COLOR_RGB2BGR)
            pname = simpledialog.askstring("Enter Product Name", "Enter Product
Name",parent=self.window)
            price = simpledialog.askfloat("Enter Product Price", "Enter Product Price",
parent=self.window, minvalue=1.0, maxvalue=100000.0)
            if not os.path.exists('Product/'+pname):
                os.makedirs('Product/'+pname)
            img_name = time.strftime("%d-%m-%Y-%H-%M-%S") + ".jpg"
            cv2.imwrite('Product/'+pname+'/'+img_name,img)
            f = open("details.txt", "a+")
            f.write(pname+","+str(price)+","+img_name+"\n")
            f.close()
            messagebox.showinfo("Product details saved","Product details saved")

    def trainmodel(self):
        global labels
        global X_train
        global Y_train
        global classifier
        global prices
        labels = []
        X_train = []
        Y_train = []
        prices = []
        path = 'Product'
        for root, dirs, directory in os.walk(path):
            for j in range(len(directory)):
                name = os.path.basename(root)
                if name not in labels:
                    labels.append(name)

        for i in range(len(labels)):
            cost = '0'
            with open("details.txt", "r") as file:
                for line in file:
                    line = line.strip('\n')
                    line = line.strip()
                    arr = line.split(",")
                    if arr[0] == labels[i] and cost == '0':
                        cost = arr[1]
            file.close()
            prices.append(cost)

        for root, dirs, directory in os.walk(path):
            for j in range(len(directory)):
                name = os.path.basename(root)
                img = cv2.imread(root+"/"+directory[j])
                img = cv2.resize(img, (256,256))
                im2arr = np.array(img)
                im2arr = im2arr.reshape(256,256,3)
                X_train.append(im2arr)
```

```
            Y_train.append(self.getID(name))
    X_train = np.asarray(X_train)
    Y_train = np.asarray(Y_train)
    print(Y_train)
    print(labels)
    print(prices)
    X_train = X_train.astype('float32')
    X_train = X_train/255

    test = X_train[3]
    cv2.imshow("aa",test)
    cv2.waitKey(0)
    indices = np.arange(X_train.shape[0])
    np.random.shuffle(indices)
    X_train = X_train[indices]
    Y_train = Y_train[indices]
    Y_train = to_categorical(Y_train)

    if os.path.exists('Model/model.json'):
        with open('Model/model.json', "r") as json_file:
            loaded_model_json = json_file.read()
            classifier = model_from_json(loaded_model_json)

        classifier.load_weights("Model/model_weights.h5")
        classifier._make_predict_function()
        print(classifier.summary())
        f = open('Model/history.pckl', 'rb')
        data = pickle.load(f)
        f.close()
        acc = data['accuracy']
        accuracy = acc[9] * 100
        messagebox.showinfo("Training model accuracy","Training Model Accuracy
= "+str(accuracy))
    else:
        classifier = Sequential()
        classifier.add(Convolution2D(32, 3, 3, input_shape = (256, 256, 3), activation = 'relu'))
        classifier.add(MaxPooling2D(pool_size = (2, 2)))
        classifier.add(Convolution2D(32, 3, 3, activation = 'relu'))
        classifier.add(MaxPooling2D(pool_size = (2, 2)))
        classifier.add(Flatten())
        classifier.add(Dense(output_dim = 256, activation = 'relu'))
        classifier.add(Dense(output_dim = 4, activation = 'softmax'))
        print(classifier.summary())
        classifier.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = ['accuracy'])
        hist = classifier.fit(X_train, Y_train, batch_size=16, epochs=10, shuffle=True, verbose=2)
        classifier.save_weights('Model/model_weights.h5')
        model_json = classifier.to_json()
        with open("Model/model.json", "w") as json_file:
            json_file.write(model_json)
        f = open('Model/history.pckl', 'wb')
        pickle.dump(hist.history, f)
        f.close()
        f = open('Model/history.pckl', 'rb')
        data = pickle.load(f)
        f.close()
        acc = data['accuracy']
        accuracy = acc[9] * 100
        messagebox.showinfo("Training model accuracy","Training Model Accuracy = "+str(accuracy))


def predict(self):
    ret, frame = self.vid.get_frame()
    img = cv2.cvtColor(frame, cv2.COLOR_RGB2BGR)
    img = cv2.resize(img, (256,256))
    im2arr = np.array(img)
    im2arr = im2arr.reshape(1,256,256,3)
    image = np.asarray(im2arr)
    image = image.astype('float32')
    image = image/255
    preds = classifier.predict(image)
    predict = np.argmax(preds)
    pname = labels[predict]
    print(str(pname)+" "+str(np.amax(preds)))
    cost = prices[predict]
    if pname in cart:
        cart.remove(pname)
    else:
        cart.append(pname)
    self.text.delete('1.0', END)
    total_amt = 0
    for i in range(len(cart)):
        for k in range(len(labels)):
            if labels[k] == cart[i]:
                cost = prices[k]
                total_amt = total_amt + float(cost)
                k = len(labels)
        self.text.insert(END,"Product Name : "+cart[i]+"\n")
        self.text.insert(END,"Product Cost : "+cost+"\n\n")
     self.text.insert(END,"\nTotal Amount : "+str(total_amt))
def update(self):
    ret, frame = self.vid.get_frame()
    if ret:
        self.photo = PIL.ImageTk.PhotoImage(image = PIL.Image.fromarray(frame))
        self.canvas.create_image(0, 0, image = self.photo, anchor = tkinter.NW)
        self.window.after(self.delay, self.update)

class MyVideoCapture:
def __init__(self, video_source=0):
    self.vid = cv2.VideoCapture(video_source)
    if not self.vid.isOpened():
        raise ValueError("Unable to open video source", video_source)
    self.width = self.vid.get(cv2.CAP_PROP_FRAME_WIDTH)
    self.height = self.vid.get(cv2.CAP_PROP_FRAME_HEIGHT)

def get_frame(self):
    if self.vid.isOpened():
        ret, frame = self.vid.read()
        if ret:
            return (ret, cv2.cvtColor(frame, cv2.COLOR_BGR2RGB))
        else:
            return (ret, None)
    else:
        return (ret, None)

def __del__(self):
    if self.vid.isOpened():
        self.vid.release()
App(tkinter.Tk(), "Tkinter and OpenCV")
```

TEST1.PY

```
from keras.utils.np_utils import to_categorical
import os
import cv2
import numpy as np
from keras.layers import  MaxPooling2D
from keras.layers import Dense, Dropout, Activation, Flatten
from keras.layers import Convolution2D
from keras.models import Sequential
from keras.models import model_from_json
import pickle
```

```
path = 'Product'
labels = []
X_train = []
Y_train = []
def getID(name):
    index = 0
    for i in range(len(labels)):
        if labels[i] == name:
            index = i
            break
    return index

for root, dirs, directory in os.walk(path):
    for j in range(len(directory)):
        name = os.path.basename(root)
        if name not in labels:
            labels.append(name)

for root, dirs, directory in os.walk(path):
    for j in range(len(directory)):
        name = os.path.basename(root)
        img = cv2.imread(root+"/"+directory[j])
        img = cv2.resize(img, (256,256))
        im2arr = np.array(img)
        im2arr = im2arr.reshape(256,256,3)
        X_train.append(im2arr)
        Y_train.append(getID(name))
X_train = np.asarray(X_train)
Y_train = np.asarray(Y_train)
print(Y_train)

X_train = X_train.astype('float32')
X_train = X_train/255
```

```
test = X_train[3]
cv2.imshow("aa",test)
cv2.waitKey(0)
indices = np.arange(X_train.shape[0])

np.random.shuffle(indices)
X_train = X_train[indices]
Y_train = Y_train[indices]
Y_train = to_categorical(Y_train)



classifier = Sequential()
classifier.add(Convolution2D(32, 3, 3, input_shape = (256, 256, 3), activation =
'relu'))
classifier.add(MaxPooling2D(pool_size = (2, 2)))
classifier.add(Convolution2D(32, 3, 3, activation = 'relu'))
classifier.add(MaxPooling2D(pool_size = (2, 2)))
classifier.add(Flatten())
classifier.add(Dense(output_dim = 256, activation = 'relu'))
classifier.add(Dense(output_dim = 4, activation = 'softmax'))
print(classifier.summary())
classifier.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics =
['accuracy'])
hist = classifier.fit(X_train, Y_train, batch_size=16, epochs=10, shuffle=True,
verbose=2)
classifier.save_weights('Model/model_weights.h5')
model_json = classifier.to_json()
with open("Model/model.json", "w") as json_file:
    json_file.write(model_json)
f = open('Model/history.pckl', 'wb')
pickle.dump(hist.history, f)
f.close()
f = open('Model/history.pckl', 'rb')
data = pickle.load(f)
```

```
f.close()
acc = data['accuracy']
accuracy = acc[9] * 100
print(accuracy)
```

6.3 METHODS

The following methods are used in the project code:

TKinter: Tkinter is a standard GUI (graphical user interface) package. Tkinter is Python's default GUI module and also the most common way that is used for GUI programming in Python.

Matplotlib: Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.

Numpy: Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

TensorFlow: TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks.

7. PROJECT 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.

7.1 VARIOUS TEST CASES

Test Case 1:

We upload the images of the products and add the details of the products

Test Case 2:

We remove the products from the basket

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.

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.

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.

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.

7.2 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.

7.3 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.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

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.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

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

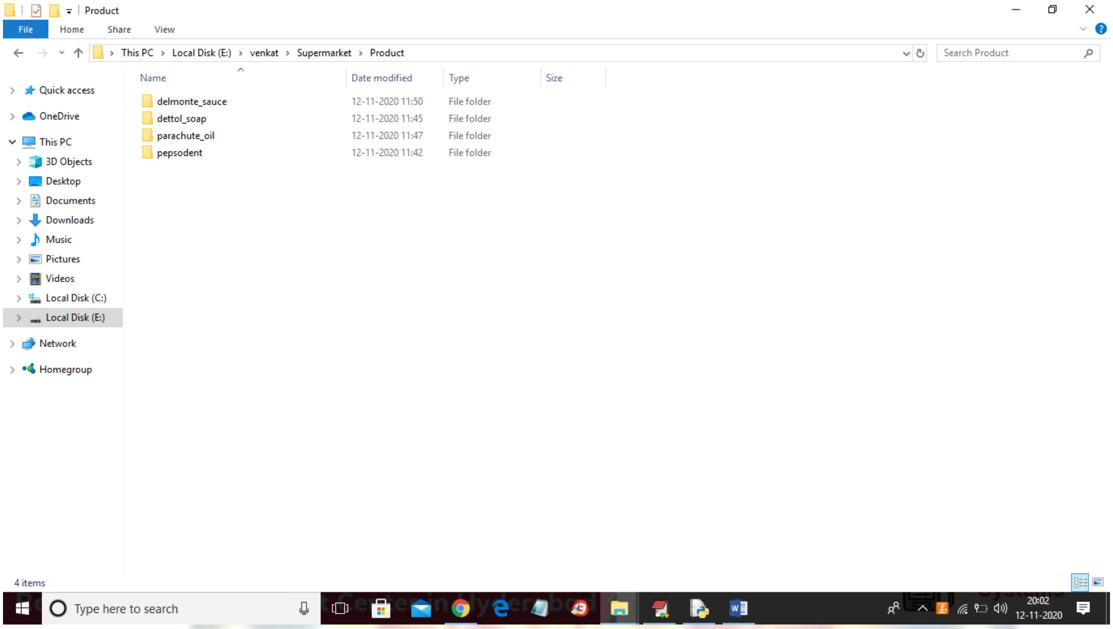
Acceptance Testing

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.

8. OUTPUT SCREENS

To build supermarket basket project we used some sample products image to train product identification models and below are some products details screenshots.



Name	Date modified	Type	Size
delmonte_sauce	12-11-2020 11:50	File folder	
dettol_soap	12-11-2020 11:45	File folder	
parachute_oil	12-11-2020 11:47	File folder	
pepsodent	12-11-2020 11:42	File folder	

FIG. 8.1 SAMPLE PRODUCT IMAGE 1

In above screen I took 4 products folders and each folder contains images of those products. For example below is the images of Dettol_soap folder

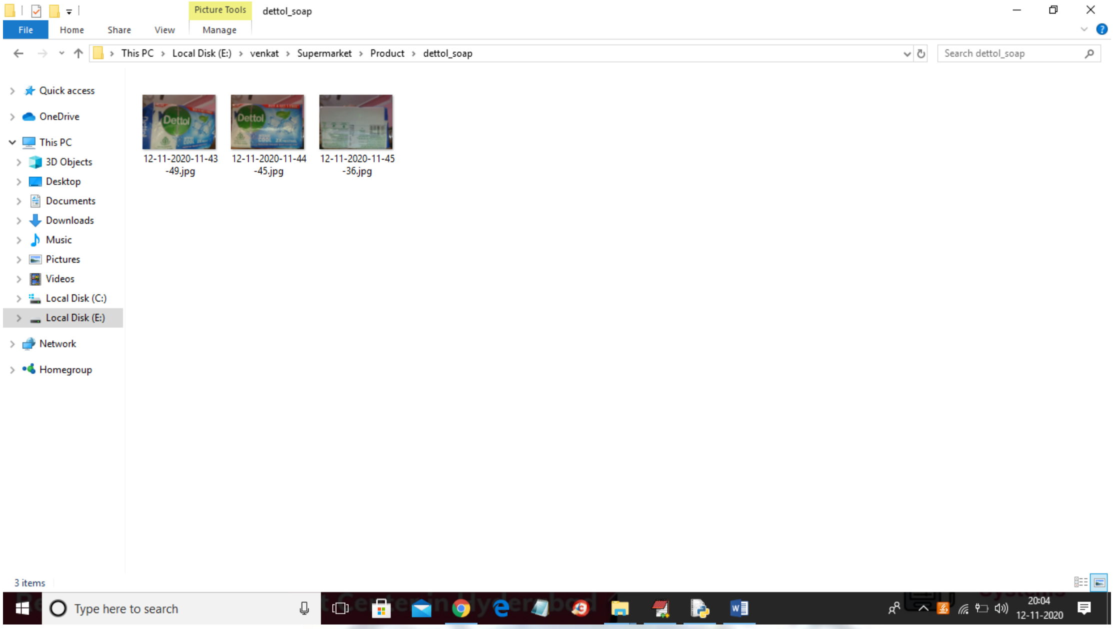


FIG. 8.2 SAMPLE PRODUCT IMAGE 2

In above screens we can see Dettol images and now to identify products run the project by double click on ‘run.bat’ file to get below screen.

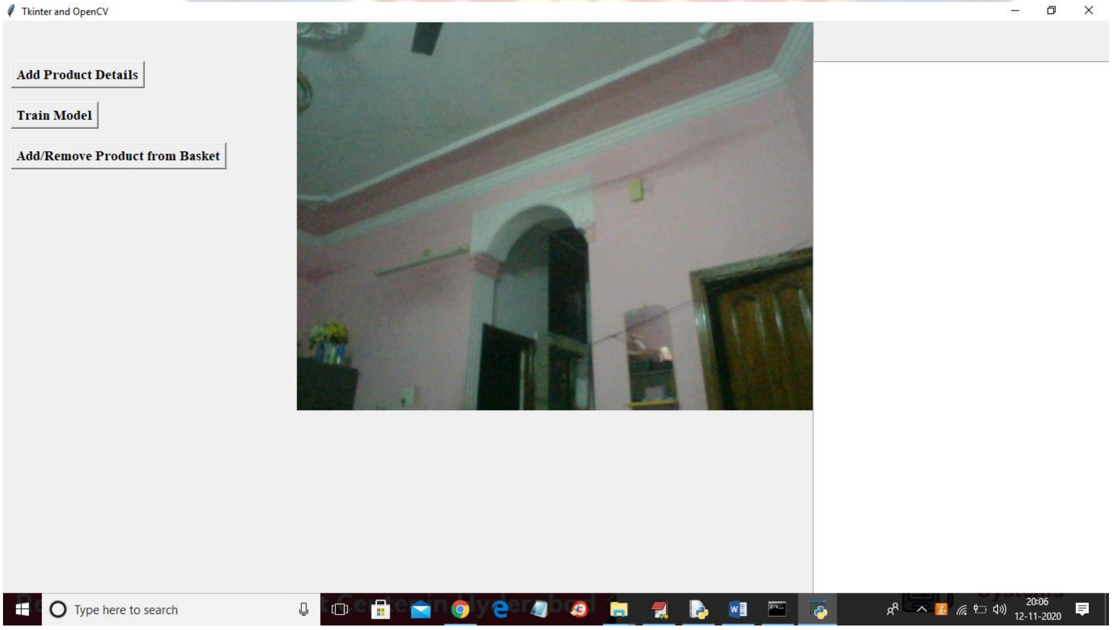
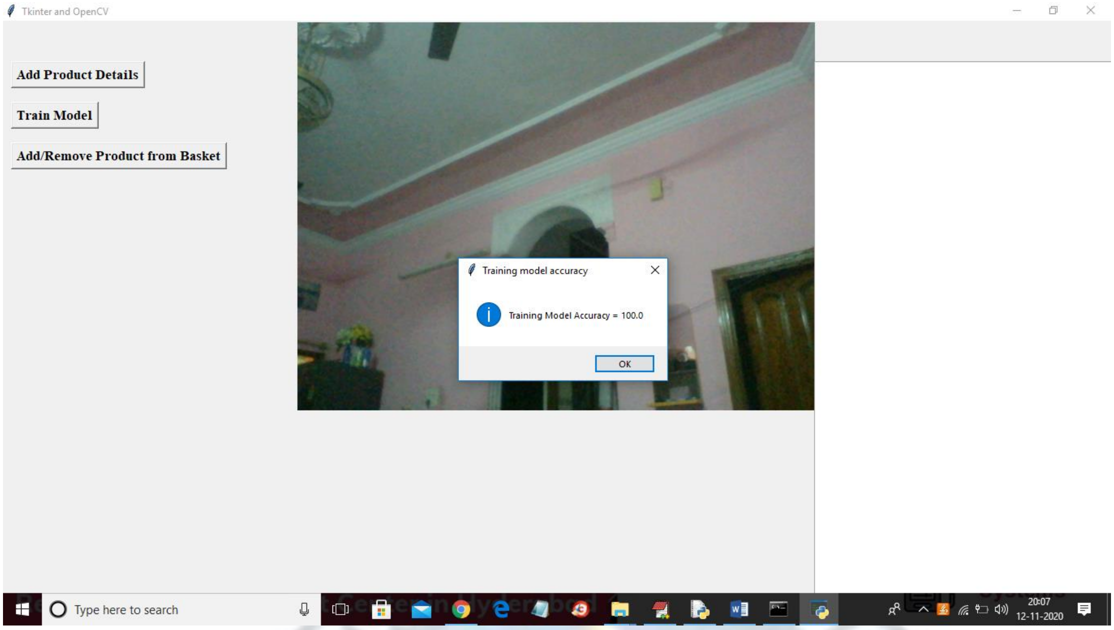


FIG. 8.3 OPENING WEBCAM

In above screen we can see application connected to web cam and now click on ‘Train Model’ button to train model with images.



Training model accuracy

Training Model Accuracy = 100.0

OK

FIG. 8.4 TRAIN MODEL

In above screen train model generated with 100% accuracy and now show product to web cam and click on ‘Add/Remove Product from Basket’ button to allow application to identify product image and then show in text area and if we again show same product then application will remove from text area.

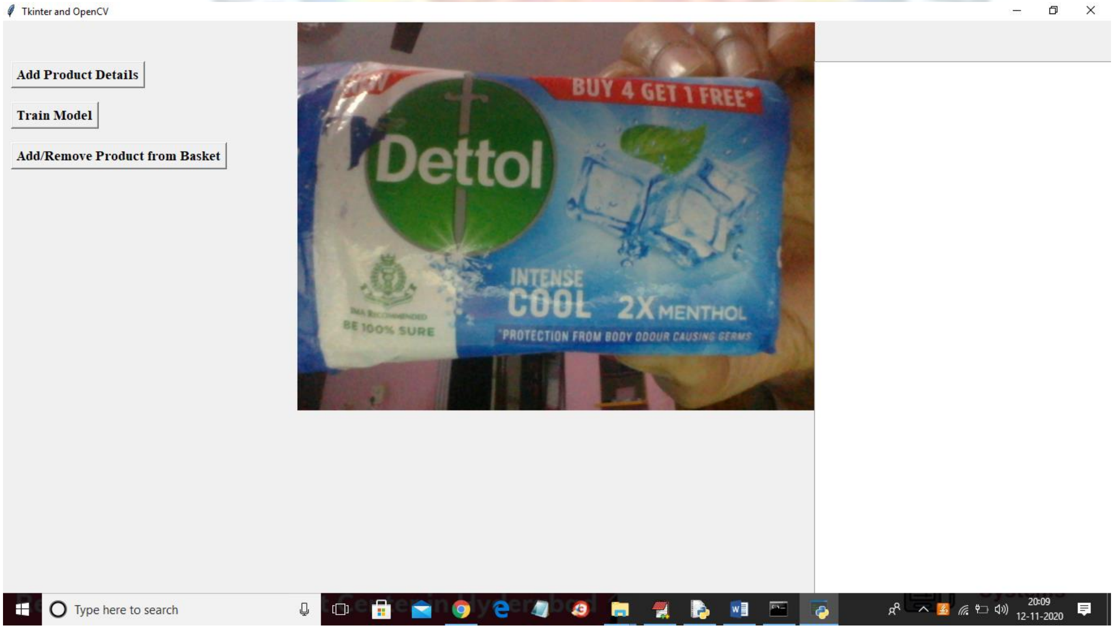
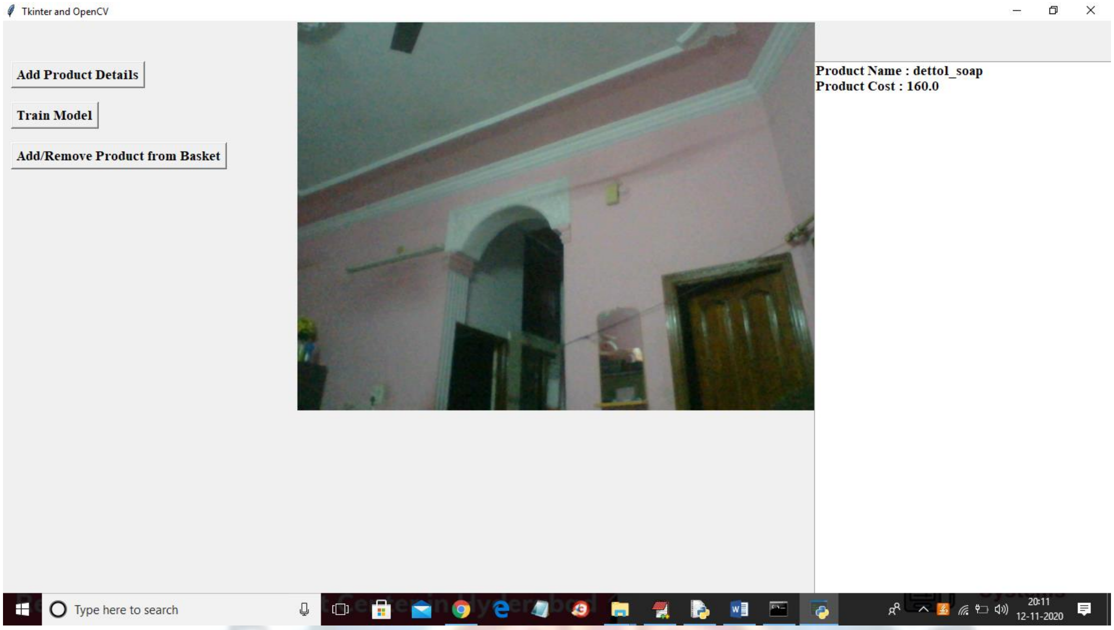


FIG. 8.5 ADDING NEW PRODUCT 1

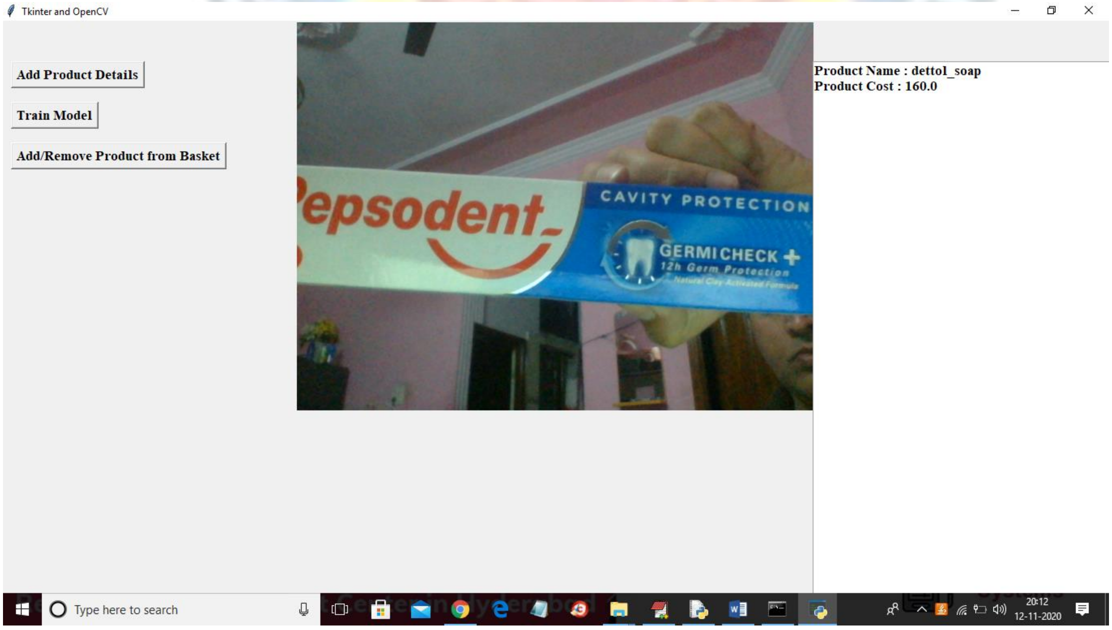
In above screen I am showing one product and after clicking on ‘Add/Remove Product from Basket’ button will get below result.



Product Name : dettol_soap
Product Cost : 160.0

FIG. 8.6 ADDING PRODUCT 1 DETAILS

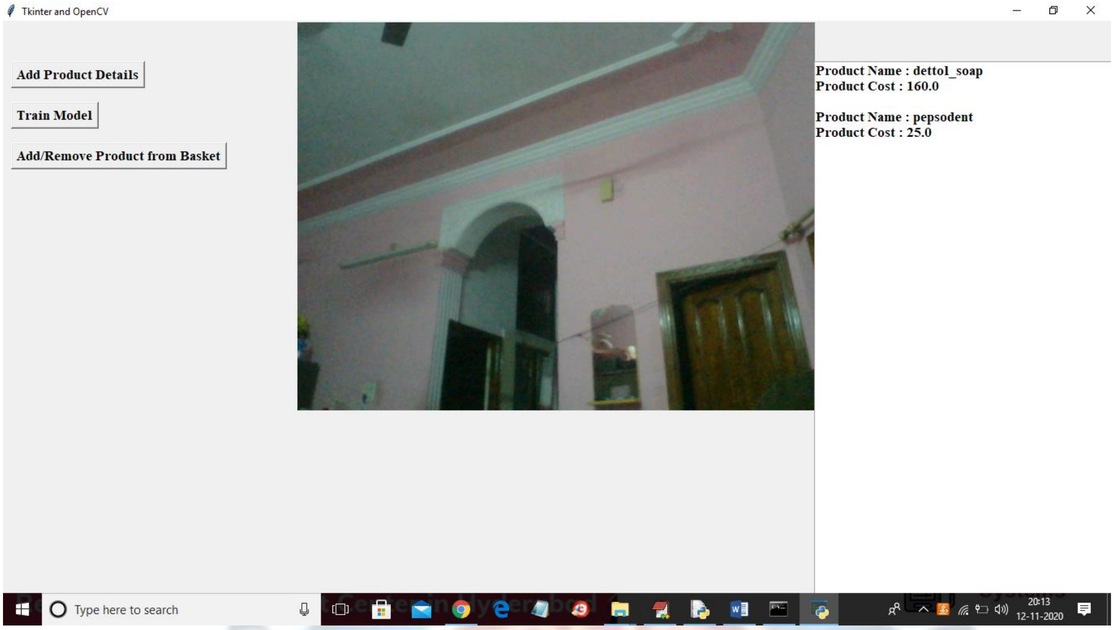
In above screen in text area we can consider as basket and the name of product and cost is displaying and now try with other product.



Product Name : dettol_soap
Product Cost : 160.0

FIG. 8.7 ADDING NEW PRODUCT 2

In above screen showing another image and after clicking on ‘Add/Remove Product from basket’ button will get below screen.

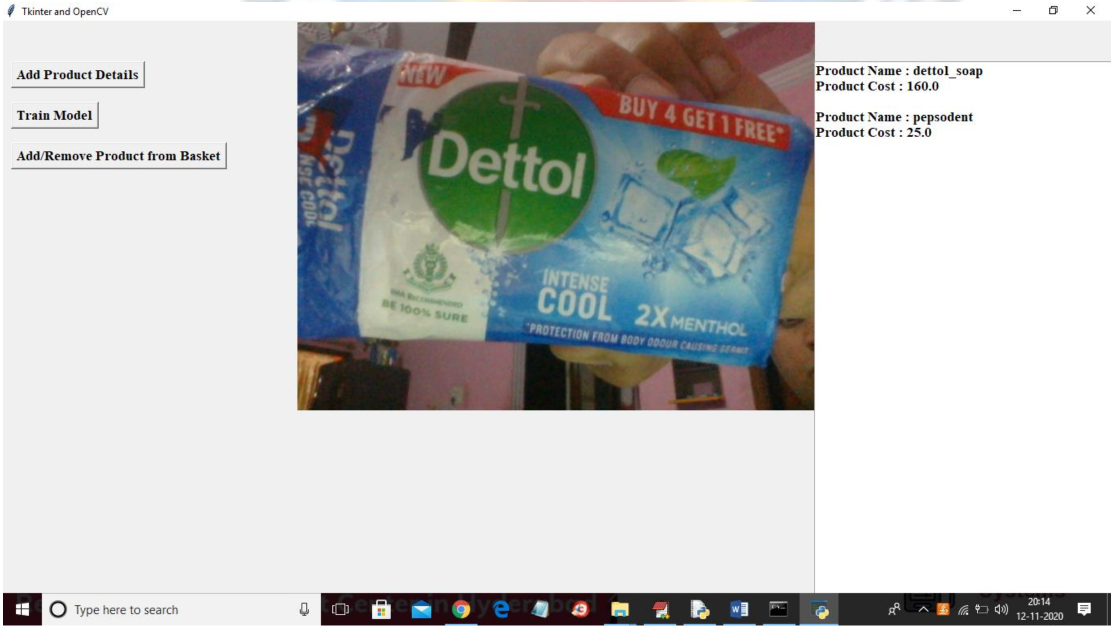


Product Name : dettol_soap
Product Cost : 160.0

Product Name : pepsodent
Product Cost : 25.0

FIG. 8.8 ADDING PRODUCT 2 DETAILS

In above screen we can see two products added to basket and now show same product again to remove from basket.

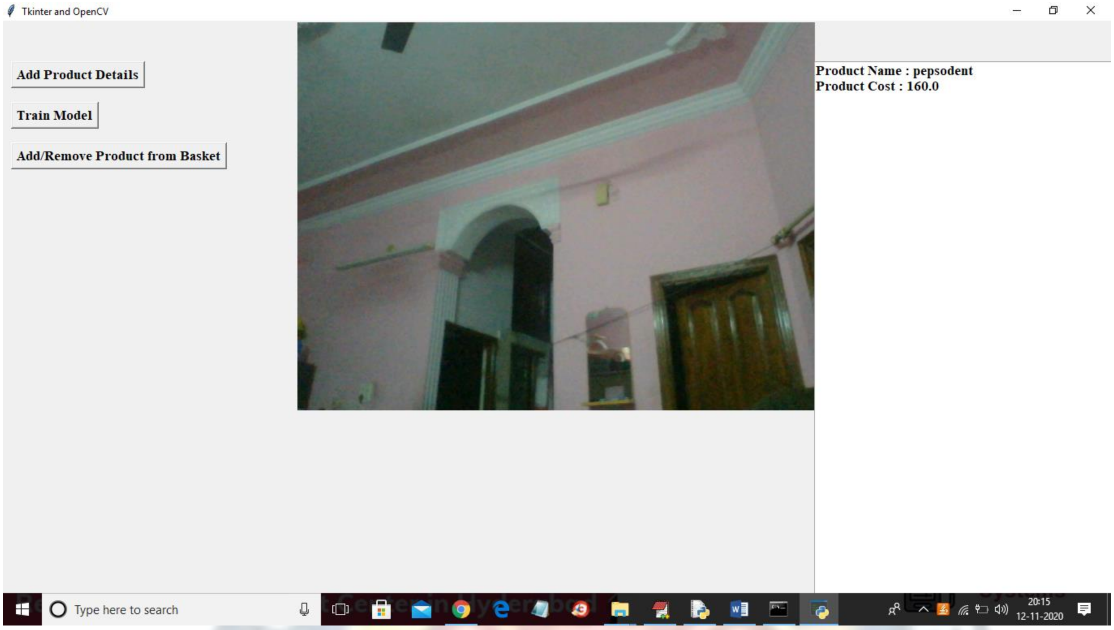


Product Name : dettol_soap
Product Cost : 160.0

Product Name : pepsodent
Product Cost : 25.0

FIG. 8.9 REMOVING PRODUCT 1

In above screen I am showing same product again and then application identified this item from basket and removed it and see the below output screen after removing item.



Product Name : pepsodent
Product Cost : 160.0

FIG. 8.10 AFTER REMOVING PRODUCT 1

Similarly u can test with other products also and if u have new images then send to us we will rebuild model as per your image and send.

9. CONCLUSION

In this Python project, the users are also provided an option to purchase items from the supermarket. The user can view items and then purchase the items which they need. To buy an item, the user needs to enter the product name and then click enter to confirm. This system then displays a message saying the user to pay the price of the item in the counter. In the modern era, people have more income to spend and lesser time to spend, so they typically opt for supermarkets for grocery. Truly the client is in a position & absolves to opt for product from large on the market varieties which attract the large customers mainly in big cities thus therefore long queues of shoppers are seen at these stores. In several cases, the barcode is either broken or there is also downside in reading barcode because of lighting effects, low resolution etc.

10. FUTURE ENHANCEMENT

In the modern era, the people have more income to spend and lesser time to spend, so they generally typically opt for supermarkets for grocery. Truly the client is in a position & absolves to opt for product from large on the market varieties which attract the large customers mainly in big cities thus therefore long queues of shoppers are seen at these stores. In several cases, the barcode is either broken or there is also downside in reading barcode because of lighting effects, low resolution etc. A bar code based billing system is also expensive as it requires bar coding of all products. The planet is moving towards an era of automation and creature is a great asset which should be utilized in additional intellectual works instead of manual, monotonous works.

11. REFERENCES

[1] Atzori, L., Iera, A., & Morabito, G, “The internet of things: A survey,” Computer Networks, vol. 54, no. 15, 2010, pp. 2787–2805

[2] Lizheng Liu1, Bo Zhou2, Zhuo Zou1, Shih-Ching Yeh1, Lirong Zheng" Image Processing System for Automatic Segmentation and Yield Prediction of fruits using Open CV." International Conference on Emerging Trends & Innovations in Engineering and Technological Research (2018).

[3] Sarvini T, Sneha T, Sukanya Gowthami G S, Sushmita S & R Kumar "Performance Comparison of weed Detection Algorithm" IEEE, International Conference on Communication and Signal Processing, April 4-6, India. 2019.

[4] Gorbunov Vladimir(&) , Ionov Evgen(&) , and Naing Lin Aung " Automatic Detection & classification of weaving fabric defects based on digital image processing." Second International Conference on green computing (2019).

[5] https://www.amazon.com/b?ie=UTF8&node=16008589011

[6] Zhang, Yanan, H. Wang, and F. Xu. "Object detection and recognition of intelligent service robot based on deep learning." IEEE International Conference on Cybernetics and Intelligent Systems IEEE, 2018.

[7] Martinez-Martin, Ester, and A. P. D. Pobil. "Object Detection and Recognition for Assistive Robots." IEEE Robotics & Automation Magazine PP.99(2017):1-1.

[8] Zhang, Shuai, et al. "New Object Detection, Tracking, and Recognition Approaches for Video Surveillance Over Camera Network." IEEE Sensors Journal 15.5(2015):2679-2691.

[9] Oliveira, Bernardo A. G. De, F. Magalhaes, and C. A. P. D. S. Martins. "Fast and Lightweight Object Detection Network: Detection and recognition on resource constrained devices." IEEE Access PP.99(2018):1-1. [10] Ren, S., He, K., Girshick, R., Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In: NIPS. 2

[10] Jerry B, Andrea C. Bitcoin: A Primer for Policymakers. Mercatus Center, George Mason University, 2013.

A

PROJECT REPORT

On

USING DEEP LEARNING TO PREDICT PLANT GROWTH AND YIELD IN GREEN HOUSE ENVIRONMENTS

Submitted by

Ms.VEMULA BHOOMIKA	**(17K81A1254)**
Ms.RAJARAMGARI THANMAYA	**(17K81A1245)**
Mr.TEEGALA LIMBAREDDY	**(17K81A1251)**
Mr.KALAL SAI CHARAN GOUD	**(17K81A1222)**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

MR.V.CHANDRAPRAKASH

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN'S ENGINEERING COLLEGE
An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **USING DEEP LEARNING TO PREDICT PLANT GROWTH AND YIELD IN GREEN HOUSE ENVIRONMENTS**, is being submitted by Ms.VEMULA BHOOMIKA (17K81A1254), Ms.RAJARAMGARI THANMAYA (17K81A1245), Mr.TEEGALA LIMBAREDDY (17K81A1251), Mr.KALAL SAI CHARAN GOUD (17K81A1222) In partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

	Head of the Department
MR.V.CHANDRAPRAKASH	DR.R.NAGARAJU
Department of Information Technology	Department of Information Technology

Internal Examiner

External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of Information Technology, session: <2017 – 2021>, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **USING DEEP LEARNING TO PREDICT PLANT GROWTH AND YIELD IN GREEN HOUSE ENVIRONMENTS** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

VEMULA BHOOMIKA	17K81A1254
RAJARAMGARI THANMAYA	17K81A1245
TEEGALA LIMBAREDDY	17K81A1251
KALAL SAI CHARAN GOUD	17K81A1222

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **DR.R.NAGARAJU**, Head of the Department, Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **DR.D.BABU RAO**, in Information Technology, for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **MR.V.CHANDRAPRAKASH,** Assistant Professor, Information Technology, St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

VEMULA BHOOMIKA	17K81A1254
RAJARAMGARI THANMAYA	17K81A1245
TEEGALA LIMBAREDDY	17K81A1251
KALAL SAI CHARAN GOUD	17K81A1222

TABLE OF CONTENTS

LASYA INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **K.SAI CHARAN GOUD** WITH ROLL NO.**17K81A1222, R.THANMAYA** WITH ROLL NO.**17K81A1245**, **T.LIMBA REDDY** WITH ROLL NO.**17K81A1251**, **V.BHOOMIKA** WITH ROLL NO.**17K81A1254**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**USING DEEP LEARNING TO PREDICT PLANT GROWTH AND YIELD IN GREEN HOUSE ENVIRONMENTS**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd. ★ Kompally-500014.★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

ABSTRACT

Effective plant growth and yield prediction is an essential task for greenhouse growers and for agriculture in general. Developing models which can effectively model growth and yield can help growers improve the environmental control for better production, match supply and market demand and lower costs. Recent developments in Machine Learning (ML) and, in particular, Deep Learning (DL) can provide powerful new analytical tools. The proposed study utilizes ML and DL techniques to predict yield and plant growth variation across two different scenarios, yield forecasting and growth, in controlled greenhouse environments. We deploy a new deep recurrent neural network (RNN), using the Long Short-Term Memory (LSTM) neuron model, in the prediction formulations. Both the former yield, growth and stem diameter values, as well as the microclimate conditions, are used by the RNN architecture to model the targeted growth parameters. A comparative study is presented, using ML methods, such as support vector regression and random forest regression, utilizing the mean square error criterion, in order to evaluate the performance achieved by the different methods.

LIST OF FIGURES

1. INTRODUCTION

1.1 PROJECT OVERVIEW

As with many bio-systems, plant growth is a highly complex and dynamic environmentally linked system. Therefore, growth and yield modeling is a significant scientific challenge . Modeling approaches vary in a number of aspects (including, scale of interest, level of description, integration of environmental stress, etc.). Two basic modeling approaches are possible, namely, "knowledge-driven" or "data-driven" modeling. The knowledge driven approach relies mainly on existing domain knowledge. In contrast, a data-driven modeling approach is capable of formulating a model solely from gathered data without necessarily using domain knowledge.

Data driven models (DDM) include classical Machine Learning techniques, artificial neural networks, support vector machines, and generalized linear models. Those methods have many desirable characteristics, such as imposing fewer restrictions, or assumptions, the ability to approximate nonlinear functions, strong predictive abilities, and the flexibility to adapt to inputs of a multivariate system. According to Singh et al., 2016 and reviewed by Liakos et al., 2018 Machine Learning (ML), linear polarization, wavelet-based filtering, vegetation indices (NDVI) and regression analysis are the most popular techniques used for analyzing agricultural data. However and besides the mentioned techniques, a new methodology which is recently gaining momentum is deep learning (DL). DL belongs to the machine learning computational field and is similar to ANN. However, DL is about “deeper” neural networks that provide a hierarchical representation of the data by means of various operations. This allows larger learning capabilities, and thus higher performance and precision. A strong advantage of DL is feature learning, i.e., automatic feature extraction from raw data, with features from higher levels of the hierarchy being formed by composition of lower level features. DL can solve more complex problems particularly well, because of the more complex related models (Pan and Yang, 2010). These complex models employed in DL can increase classification accuracy and reduce error in regression problems, provided there are adequately large data-sets available describing the problem.

1.2 PROJECT OBJECTIVES

Effective plant growth and yield prediction is an essential task for greenhouse growers and for agriculture in general. Developing models which can effectively model growth and yield can help growers improve the environmental control for better production, match supply and market demand and lower costs. Recent developments in Machine Learning (ML) and, in particular, Deep Learning (DL) can provide powerful new analytical tools.

The proposed study utilizes ML and DL techniques to predict yield and plant growth variation across two different scenarios, yield forecasting and growth, in controlled greenhouse environments. We deploy a new deep recurrent neural network (RNN), using the Long Short-Term Memory (LSTM) neuron model, in the prediction formulations. Both the former yield, growth and stem diameter values, as well as the microclimate conditions, are used by the RNN architecture to model the targeted growth parameters.

A comparative study is presented, using ML methods, such as support vector regression and random forest regression, utilizing the mean square error criterion, in order to evaluate the performance achieved by the different methods.

1.3 SCOPE OF THE PROJECT

3.4.1 EXISTING SYSTEM:

Effective plant growth and yield prediction is an essential task for greenhouse growers and for agriculture in general. Developing models which can effectively model growth and yield can help growers improve the environmental control for better production, match supply and market demand and lower costs. SVR and RF are the traditional old algorithms whose performance of prediction will be low due to unavailable of deep learning techniques.

3.4.2 PROPOSED SYSTEM:

We deploy a new deep recurrent neural network (RNN), using the Long Short-Term Memory (LSTM) neuron model, in the prediction formulations.Both the former yield, growth and stem diameter values, as well as the microclimate conditions, are used by the RNN architecture to model the targeted growth parameters.

Deep Learning extends classical ML by adding more "depth" (complexity) into the model, as well as transforming the data using various functions that create data representations in a hierarchical way, through several levels of abstraction. DL can solve complex problems particularly well and fast, due to the more complex models used, which also allow massive parallelization.

1.4 ORGANIZATION OF CHAPTERS

1.4.1 INTRODUCTION

Effective plant growth and yield prediction is an essential task for greenhouse growers and for agriculture in general. Developing models which can effectively model growth and yield can help growers improve the environmental control for better production, match supply and market demand and lower costs. Recent developments in Machine Learning (ML) and, in particular, Deep Learning (DL) can provide powerful new analytical tools.

1.4.2 LITERATURE SURVEY

S.V Baria [1] built up a strategy to demonstrate how a high-resolution satellite imagery is essential to isolate rice cultivation. LAI's multi-regression model was taken as an input and NDVI or any other vegetation index calculated from red and near-infrared spectral reflection was taken as another influence under normal environmental conditions and common agronomic practices. The top methodology of estimating rice yield using satellite imagery could be during the period of maximum vegetative growth.

1.4.3 REQUIREMENTS SPECIFICATION

SOFTWARE REQUIREMENTS

• Operating System	: Windows family
• Technology	: Python 3.6
• IDE	: Anaconda

HARDWARE REQUIREMENTS

• Processor	: Any Updated Processor
• RAM	: 4 GB
• Space on Hard Disk	: minimum 80GB

1.4.4 SOFTWARE DEVELOPMENT ANALYASIS

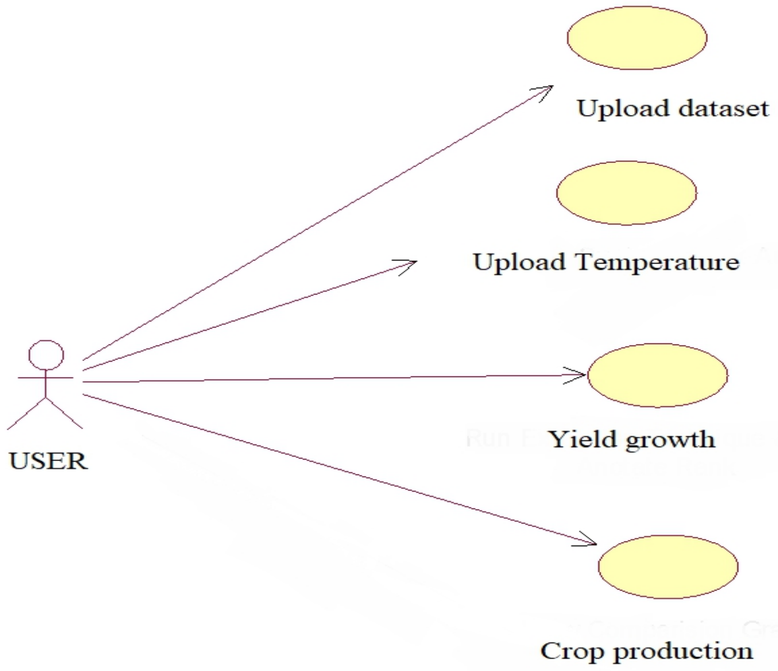
SUPPORT VECTOR REGRESSION

Support Vector Regression is a supervised learning algorithm that is used to predict discrete values. Support Vector Regression uses the same principle as the SVMs. The basic idea behind SVR is to find the best fit line. In SVR, the best fit line is the hyperplane that has the maximum number of points. Unlike other Regression models that try to minimize the error between the real and predicted value, the SVR tries to fit the best line within a threshold value. The threshold value is the distance between the hyperplane and boundary line. The fit time complexity of SVR is more than quadratic with the number of samples which makes it hard to scale to datasets with more than a couple of 10000 samples.

1.4.5 PROJECT SYSTEM DESIGN

USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



1.4.6 PROJECT CODING

```
from pathlib import Path

from scipy import stats

import geopandas as gpd

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from shapely.geometry import MultiPoint, Point, Polygon

from shapely.ops import split, snap, nearest_points

data_path = Path('../data')

sentinel_path = data_path/'sentinel'

output_path = Path('output')
```

```
shape_path = Path('output/shapes')

patch_path = Path('output/patches')

feature_path = Path('output/features')

feature_path.mkdir(exist_ok=True, parents=True)

shape_path.mkdir(exist_ok=True, parents=True)

def calculate_stats(data, aggs, aggregation_axis=0,
stacking_axis=-1):

stats_list = []

for agg in aggs:

stats_list.append(agg(data, axis=aggregation_axis))

return np.stack(stats_list, axis=stacking_axis)

In [63]: aggs = [np.mean,

np.min,

np.max,

np.median,

np.std]

field_ids = []
```

1.4.7 PROJECT TESTING

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.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

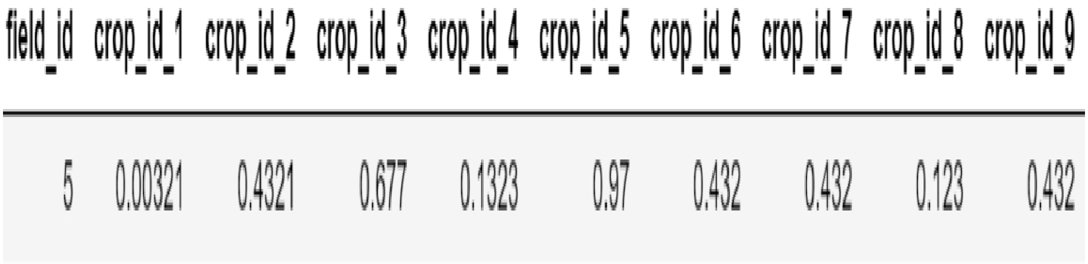
Test Results: All the test cases mentioned above passed successfully. No defects encountered.

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.

1.4.8 OUTPUT SCREEN



field_id	crop_id_1	crop_id_2	crop_id_3	crop_id_4	crop_id_5	crop_id_6	crop_id_7	crop_id_8	crop_id_9
5	0.00321	0.4321	0.677	0.1323	0.97	0.432	0.432	0.123	0.432

Figure 1.4.8 Output Screen

1.4.9 CONCLUSIONS

We have developed a DL approach using LSTM for plant growth and crop yield prediction, achieving high prediction accuracy in both problems. Experimental results were presented that show that the DL technique (using a LSTM model) outperformed other traditional ML techniques, such as SVR and RF. Hence, the main aim of our project is to develop DL methodologies to predict plants growth and yield in greenhouse environment.

2. LITERATURE SURVEY

2.1 REVIEW OF RELATED LITERATURE

Some of the most valuable documents and papers after extensive background work are:

S.V Baria [1] built up a strategy to demonstrate how a high-resolution satellite imagery is essential to isolate rice cultivation. LAI's multi-regression model was taken as an input and NDVI or any other vegetation index calculated from red and near-infrared spectral reflection was taken as another influence under normal environmental conditions and common agronomic practices. The top methodology of estimating rice yield using satellite imagery could be during the period of maximum vegetative growth.

SnehalS.Darikar et al. [2]discussed in their paper the use of Artificial Neural Network to predict crop yield. The paper senses the parameters of the regional soil and the various atmospheric conditions. Then it furthers analyses by using feed forward back propagation ANN. By using Mat lab ANN approach was made more efficient. They structure a system that accurately links climate effects to crop yield, can be used to estimate long - term or short-term crop production and can also obtain an ANN with adequate and useful data.

Agaarwal Sachin [3] discovered the air quality index by utilizing neural system-based air quality factors, which worked with a fractional number of informational indexes and are sufficiently strong to deal with information with noise and inaccuracies. Various accessible varieties of neural system models, for example, Repetitive Network Model(RNM), Consecutive System Development Demonstrate (SNCM), Change Point Discovery Display with RNM (CPDM), and Self Sorting out Element Maps (SOFM) are executed in this paper for anticipating air quality. The created models were utilized to reproduce and figure the air quality index dependent on long haul (yearly) and in addition present moment (every day) informational collections. The models can accurately anticipate air quality patterns. B.A.

Smith et al. [4] discussed about all year or long-haul atmospheric temperature expectation models that were produced for estimating forecast horizons of 1 to 12 utilizing Ward-style Artificial Neural System. These models were intended for general support in decision making. The variations of the ANN plan described here provide greater precision compared to previously developed winter models amid the winter time frame. The models that had included precipitation terms likewise as a part of the air prediction model in the input vector were progressively exact.

B. J I et al. [5] The aim of this study was to: (1) research whether Artificial neural system (ANN) models could effectively and efficiently forecast Fujian rice yield for characteristic mountainous climate and atmospheric conditions, (2) assess the performance of the ANN model in comparison to varieties of rising parameters and (3).Compare the effectiveness of multiple linear models of regression with models of ANN. The models were developed using historical harvest data from several locations in Fujian Field-explicit rainfall information and the climate conditions were utilized at every location for the rice yield prediction.

Lillian Kay Peterson [6] Here, Lillian created satellite investigation methodologies and programming devices to forecast crop yields two to four months ahead of time, before the harvest. This procedure estimated relative vegetation condition dependent on pixel-level customary irregularities of NDVI, EVI and NDWI indices.

2.2 SOFTWARE REQUIREMENT

For developing the project, the following are the Software Requirements:

- Python

2.2.1 PYTHON

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. C Python, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all its variant implementations. C Python is managed by the non-profit Python Software Foundation. Python features a dynamic type of 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.

2.3 ALGORITHMS FOR RESEARCH ACTIVITY

DEEP LEARNING

Deep learning is a branch of machine learning which is completely based on artificial neural networks, as neural network is going to mimic the human brain so deep learning is also a kind of mimic of human brain. In deep learning, we don't need to explicitly program everything. The concept of deep learning is not new. It has been around for a couple of years now. It's on hype nowadays because earlier we did not have that much processing power and a lot of data. As in the last 20 years, the processing power increases exponentially, deep learning and machine learning came in the picture. A formal definition of deep learning is- neurons.

Deep learning is a particular kind of machine learning that achieves great power and flexibility by learning to represent the world as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations computed in terms of less abstract ones.

In human brain approximately 100 billion neurons all together this is a picture of an individual neuron and each neuron is connected through thousand of their neighbors. The question here is how do we recreate these neurons in a computer. So, we create an artificial structure called an artificial neural net where we have nodes or neurons. We have some neurons for input value and some for output value and in between, there may be lots of neurons interconnected in the hidden layer.

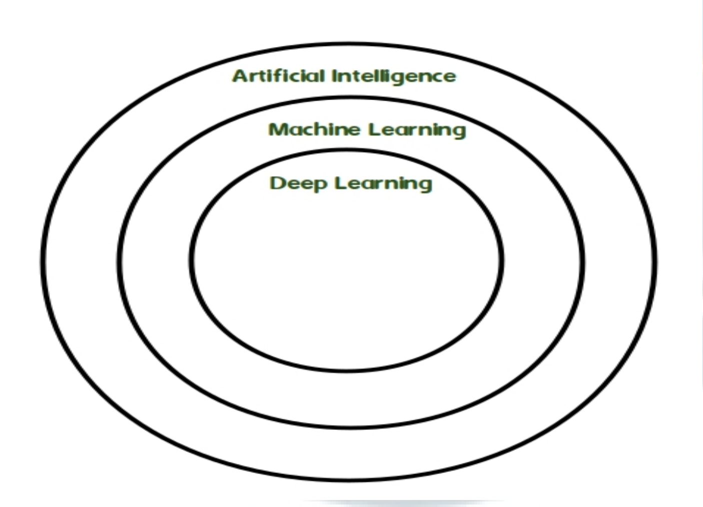


Figure 2.3.1 Deep learning

SUPPORT VECTOR REGRESSION

Support Vector Regression is a supervised learning algorithm that is used to predict discrete values. Support Vector Regression uses the same principle as the SVMs. The basic idea behind SVR is to find the best fit line. In SVR, the best fit line is the hyperplane that has the maximum number of points. Unlike other Regression models that try to minimize the error between the real and predicted value, the SVR tries to fit the best line within a threshold value. The threshold value is the distance between the hyperplane and boundary line.

The fit time complexity of SVR is more than quadratic with the number of samples which makes it hard to scale to datasets with more than a couple of 10000 samples.

RANDOM FOREST REGRESSION

Random forest is a supervised learning algorithm. The “forest” it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result. Put simply, random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. One big advantage of random forest is that it can be used for both classification and regression problems, which form the majority of current machine learning systems.

LONG SHORT TERM MEMORY

Long Short Term Memory is a kind of recurrent neural network. In RNN output from the last step is fed as input in the current step. It tackled the problem of long-term dependencies of RNN in which the RNN cannot predict the word stored in the long term memory but can give more accurate predictions from the recent information. As the gap length increases RNN does not give efficient performance. LSTM can by default retain the information for long period of time. It is used for processing, predicting and classifying on the basis of time series data.

2.4 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

2.4.1 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.

2.4.2 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 a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 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.

3. REQUIREMENTS SPECIFICATION

3.1 SOFTWARE REQUIREMENTS

▪ Operating System	: Windows family
▪ Technology	: Python 3.6
▪ IDE	: Anaconda

3.2 HARDWARE REQUIREMENTS

▪ Processor	: Any updated Processor
▪ RAM	: 4 GB
▪ Space on Hard Disk	: minimum 80GB

4. SOFTWARE REQUIREMENTS ANALASIS

4.1 DEFINE THE PROBLEM

Effective plant growth and yield prediction is an essential task for greenhouse growers and for agriculture in general. Developing models which can effectively model growth and yield can help growers improve the environmental control for better production, match supply and market demand and lower costs.

In this project, We deploy a new deep recurrent neural network (RNN), using the Long Short-Term Memory (LSTM) neuron model, in the prediction formulations. Both the former yield, growth and stem diameter values, as well as the microclimate conditions, are used by the RNN architecture to model the targeted growth parameters.

4.2 DEFINE THE MODULES

4.2.1 UPLOAD DATASET:

Using this module we will upload plant dataset.

4.2.2 DATASET CLEANING:

In using this module we will find out empty values in the dataset and replace with mean or 0 values.

4.2.3 TRAIN AND TEST SPLIT:

Using this module we will split dataset into two parts called and training and testing. All machine learning algorithms take 80% dataset to train classifier and 20% dataset issued to test classifier prediction accuracy. If classifier prediction accuracy high then Mean Square Error, Root Mean Square Error and Mean Absolute Error will be dropped**.**

4.2.4 RUN SVR CLASSIFIER:

Using this module we will train SVR classifier with splitted 80% data and used 20% data to calculate it performance.

4.2.5 RUN RANDOM FOREST CLASSIFIER:

Using this module we will train Random Forest classifier with splitted data.

4.2.4 RUN LSTM CLASSIFIER:

Using this module we will train LSTM classifier with splitted data.

4.2.7 PREDICT PLANT AND YIELD GROWTH:

Using this module we will upload test data and then apply LSTM classifier to predict it growth value.

4.3 MODULE FUNTIONALITIES

1. Functional Requirements -Graphical User interface with the User.

2. Non-Functional Requirements

A. **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.

B. **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.

C. **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.

D. **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high.

E. **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.

F. **Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life. However, the allowable wattage cannot be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

5 SOFTWARE DESIGN

5.1 SYSTEM ARCHITECTURAL DESIGN

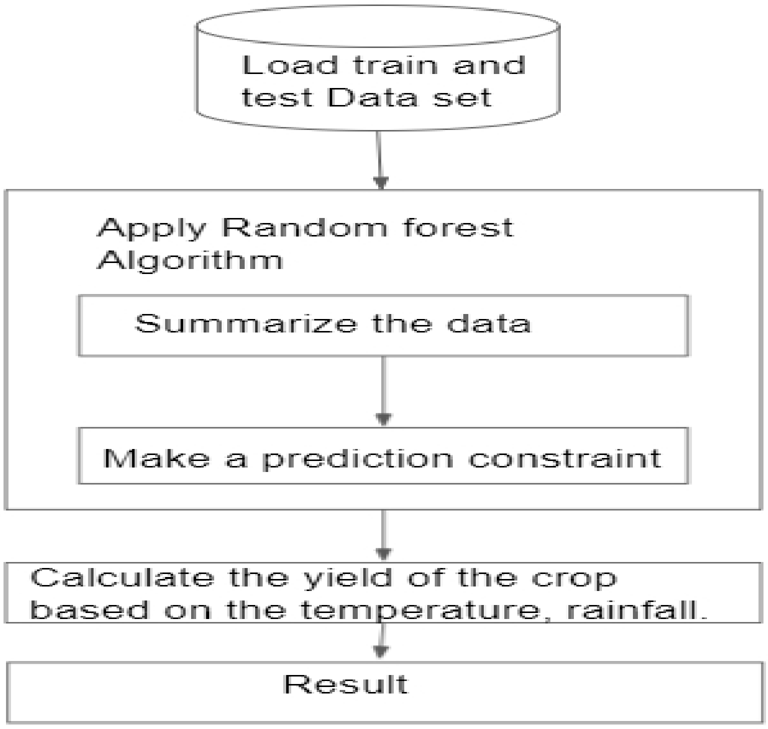

Load train and test Data set
Apply Random forest Algorithm
Summarize the data
Make a prediction constraint
Calculate the yield of the crop based on the temperature, rainfall.
Result

Figure 5.1 System Architectural Design

5.2 CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

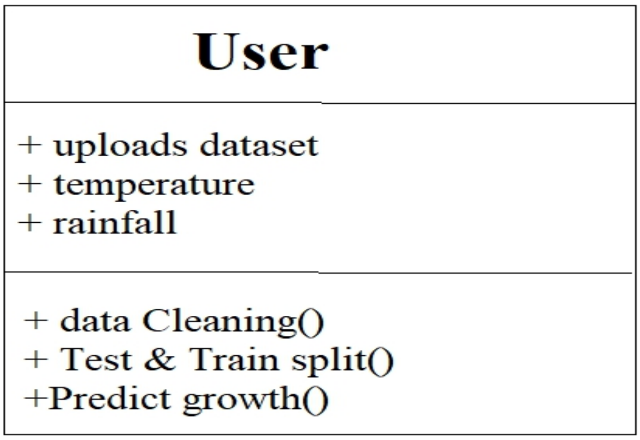


Figure 5.2 Class Diagram

5.3 USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

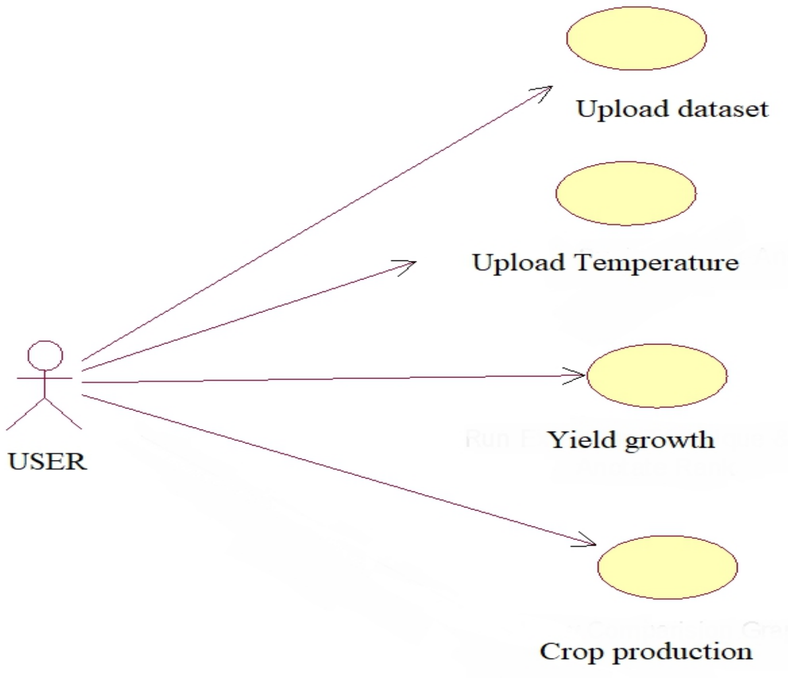


Figure 5.3 Use Case Diagram

5.4 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

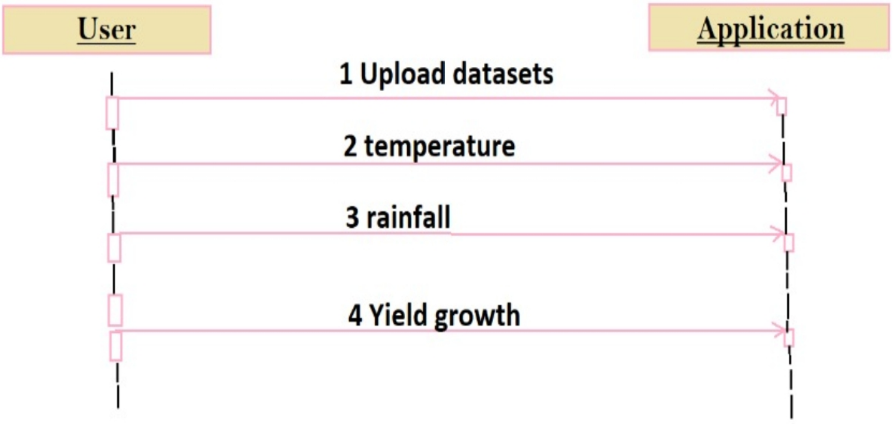


Figure 5.4 Sequence Diagram

5.5 COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

A Communication diagram models the interactions between objects or parts in terms of sequenced messages. Communication diagrams represent a combination of information taken from Class, Sequence, and Use Case Diagrams describing both the static structure and dynamic behaviour of a system.

However, communication diagrams use the free-form arrangement of objects and links as used in Object diagrams. In order to maintain the ordering of messages in such a free-form diagram, messages are labelled with a chronological number and placed near the link the message is sent over. Reading a communication diagram involves starting at message 1.0 and following the messages from object to object.

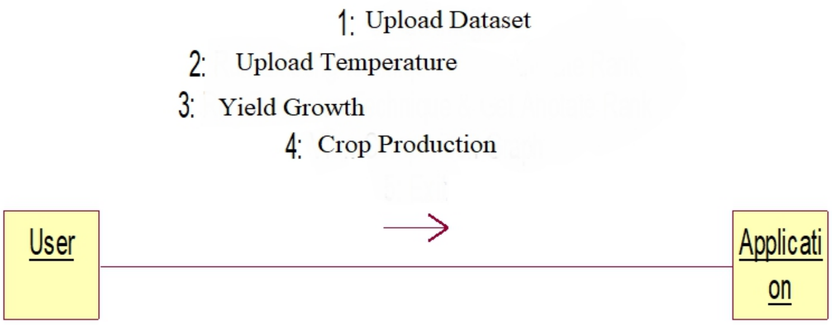


Figure 5.5 Collaboration Diagram

5.6 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions[1] with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities.[2][3] Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.[citation needed]Activity diagrams are graphical representations of workflows of stepwise activities and actions[

with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities.[2][3] Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.[citation needed]

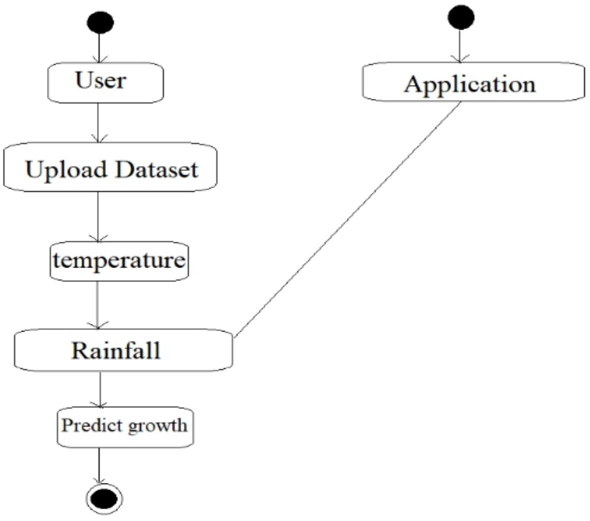


Figure 5.6 Activity Diagra

5.7 PACKAGE DIAGRAM

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.

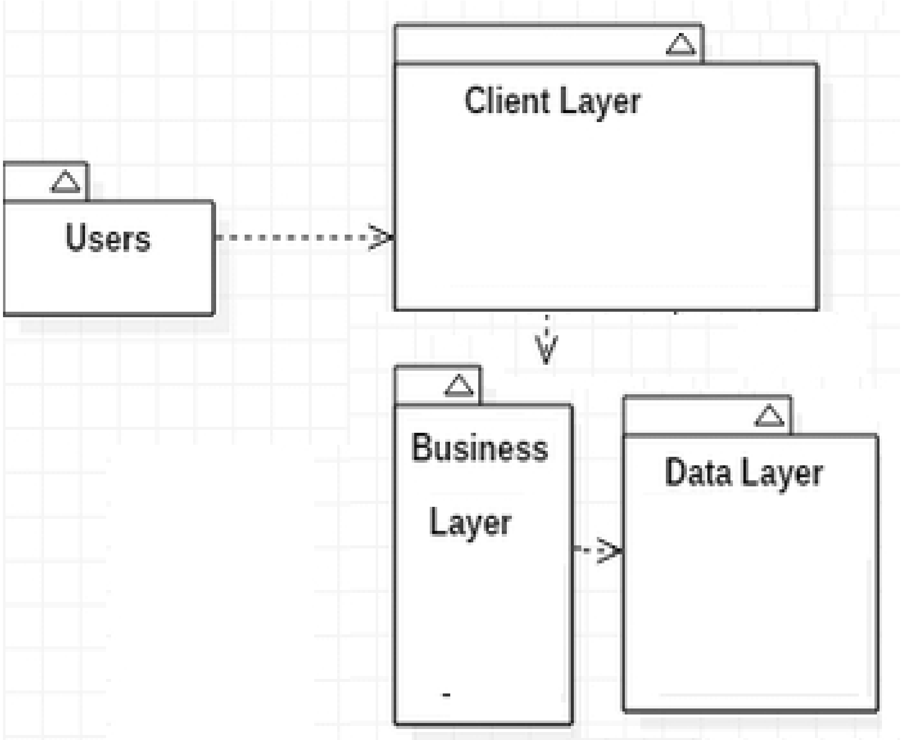


Figure 5.7 Package Diagram

6. CODING/CODE TEMPLATES

IMPLEMTATION OF CODE

```
from pathlib import Path
from scipy import stats
import geopandas as gpd
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from shapely.geometry import MultiPoint, Point, Polygon
from shapely.ops import split, snap, nearest_points
data_path = Path('../data')
sentinel_path = data_path/'sentinel'
output_path = Path('output')
shape_path = Path('output/shapes')
patch_path = Path('output/patches')
feature_path = Path('output/features')
feature_path.mkdir(exist_ok=True, parents=True)
shape_path.mkdir(exist_ok=True, parents=True)
def calculate_stats(data, aggs, aggregation_axis=0,
stacking_axis=-1):
stats_list = []
for agg in aggs:
```

```
stats_list.append(agg(data, axis=aggregation_axis))

return np.stack(stats_list, axis=stacking_axis)

In [63]: aggs = [np.mean,

np.min,

np.max,

np.median,

np.std]

field_ids = []

features_stats = []

composite_max_ndvi = []

composite_min_ndvi = []

composite_max_ndvi_lbp = []

composite_min_ndvi_lbp = []

composite_max_ndvi_hog = []

composite_min_ndvi_hog = []

argmax_b4 = []

argmin_b4 = []

argmax_ndvi = []

argmin_ndvi = []

argmax_ndvi_slope = []

argmin_ndvi_slope = []

stf = []
```

```
for i in range(12):

patch = EOPatch.load(feature_path/f'eopatch_{i}')

field_id_mask = (patch.mask_timeless['FIELD_ID'] > 0).squeeze()

field_ids .append(patch.mask_timeless['FIELD_ID'][field_id_mask]

features_stats .append(calculate_stats(patch.data['FEATURES'], aggs=ag

composite_max_ndvi .append(patch.data['COMPOSITE_MAX_NDVI'][0, field_id_ma]

composite_min_ndvi .append(patch.data['COMPOSITE_MIN_NDVI'][0, field_id_ma]

composite_max_ndvi_lbp .append(patch.data['COMPOSITE_MAX_NDVI_LBP'][0,field_i]

composite_min_ndvi_lbp .append(patch.data['COMPOSITE_MIN_NDVI_LBP'][0,field_i]

composite_max_ndvi_hog .append(patch.data['COMPOSITE_MAX_NDVI_HOG'][0,field_i]

composite_min_ndvi_hog .append(patch.data['COMPOSITE_MIN_NDVI_HOG'][0,field_i]

argmax_b4 .append(patch.data_timeless['ARGMAX_B4'][field_id_mask]

argmin_b4 .append(patch.data_timeless['ARGMIN_B4'][field_id_mask]

argmax_ndvi .append(patch.data_timeless['ARGMAX_NDVI'][field_id_ma]
argmin_ndvi .append(patch.data_timeless['ARGMIN_NDVI'][field_i
d_ma]

argmax_ndvi_slope .append(patch.data_timeless['ARGMAX_NDVI_SLO
PE'][field]

argmin_ndvi_slope .append(patch.data_timeless['ARGMIN_NDVI_SLO
PE'][field ]

stf .append(patch.data_timeless['STF'][field_id_mask,:])
```

Create dataset with statistics for each field

```
field_ids_arr = np.concatenate(field_ids , axis = 0)

features_stats_arr = np.concatenate(features_stats , axis = 0)

composite_max_ndvi_arr = np.concatenate(composite_max_ndvi ,
axis = 0)

composite_min_ndvi_arr = np.concatenate(composite_min_ndvi ,
axis = 0)

composite_max_ndvi_lbp_arr =
np.concatenate(composite_max_ndvi_lbp , axis = 0)

composite_min_ndvi_lbp_arr =
np.concatenate(composite_min_ndvi_lbp , axis = 0)

composite_max_ndvi_hog_arr =
np.concatenate(composite_max_ndvi_hog , axis = 0)

composite_min_ndvi_hog_arr =
np.concatenate(composite_min_ndvi_hog , axis = 0)
```

```
argmax_b4_arr = np.concatenate(argmax_b4 , axis = 0)
argmin_b4_arr = np.concatenate(argmin_b4 , axis = 0)
argmax_ndvi_arr = np.concatenate(argmax_ndvi , axis = 0)
argmin_ndvi_arr = np.concatenate(argmin_ndvi , axis = 0)
argmax_ndvi_slope_arr = np.concatenate(argmax_ndvi_slope ,
axis = 0)
argmin_ndvi_slope_arr = np.concatenate(argmin_ndvi_slope ,
axis = 0)
stf_arr = np.concatenate(stf , axis = 0)
red_arr = features_stats_arr[:,0,:]
green_arr = features_stats_arr[:,1,:]
blue_arr = features_stats_arr[:,2,:]
nir_arr = features_stats_arr[:,3,:]
ndvi_arr = features_stats_arr[:,4,:]
norm_arr = features_stats_arr[:,5,:]
def mode(a, axis=0):
return stats.mode(a,axis=axis)[0][0]
arr_list = [
('red_t_med' , red_arr , [np.median]),
('green_t_med' , green_arr , [np.median]),
('blue_t_med' , blue_arr , [np.median]),
('nir_t_med' , nir_arr , [np.median]),
('ndvi_t_med' , ndvi_arr , [np.median]),
('norm_t_med' , norm_arr , [np.median]),

('composite_max_ndvi' , composite_max_ndvi_arr , aggs),

('composite_min_ndvi' , composite_min_ndvi_arr , aggs),

('composite_max_ndvi_lbp' , composite_max_ndvi_lbp_arr , aggs),

('composite_min_ndvi_lbp' , composite_min_ndvi_lbp_arr , aggs),

('composite_max_ndvi_hog' , composite_max_ndvi_hog_arr , aggs),

('composite_min_ndvi_hog' , composite_min_ndvi_hog_arr , aggs),

('argmax_b4' , argmax_b4_arr , [mode]),

('argmin_b4' , argmin_b4_arr , [mode]),

('argmax_ndvi' , argmax_ndvi_arr , [mode]),

('argmin_ndvi' , argmin_ndvi_arr , [mode]),

('argmax_ndvi_slope' , argmax_ndvi_slope_arr , [mode]),

('argmin_ndvi_slope' , argmin_ndvi_slope_arr , [mode]),

('stf_arr' , stf_arr , aggs)]5/20/2021

rows = []

for fid in train_test_gpdf.Field_Id:

fid_mask = field_ids_arr==fid

fid_dic = {'Field_Id': fid}

for name, arr, funcs in arr_list:

for i in range(arr.shape[-1]):

for f in funcs:

fid_dic[f'{name}_{f.__name__}_{i}'] = f(arr[fid_mask,i],axis=0)
```

```
rows.append(fid_dic)

poly_img_stats_df = pd.DataFrame(rows)

poly_img_stats_df.to_csv(data_path/'polygon_img_statistics.csv
', index=False)

poly_img_stats_df =
pd.read_csv(data_path/'polygon_img_statistics.csv')

train_test_shp = gpd.read_file(data_path/'train_test_shp')

orange_river_segment_shp =
gpd.read_file('../data/orange_river_segment/'

).to_crs({'init': 'epsg:32734'})

train_shp =
gpd.read_file(data_path/'train').dropna().to_crs({'init':
'epsg:32734

test_shp = gpd.read_file(data_path/'test').to_crs({'init':
'epsg:32734'})

Out[29]: ([], <a list of 0 Text yticklabel objects>)

ax = test_shp.plot(figsize=(100,100), color = 'grey')

train_shp.plot(figsize=(100,100), color = 'green', ax=ax)

orange_river_segment_shp.plot(ax=ax, color = 'blue', alpha
=0.5)

plt.xticks([])

plt.yticks([])
```

```
 def nearest_neighbor_within(others, point, max_distance):

"""Find nearest point among others up to a maximum distance.

Args:

others: a list of Points or a MultiPoint

point: a Point

max_distance: maximum distance to search for the nearest
neighbor

Returns:

A shapely Point if one is within max_distance, None otherwise

"""

search_region = point.buffer(max_distance)

interesting_points =
search_region.intersection(MultiPoint(others))

if not interesting_points:

closest_point = None

elif isinstance(interesting_points, Point):

closest_point = interesting_points

else:

distances = [point.distance(ip) for ip in interesting_points

if point.distance(ip) > 0]

closest_point =
interesting_points[distances.index(min(distances))]

return closest_point
```

```
def get_distance_along_line(gdf_line, poly, tolerance=10000):
# union all geometries
line = gdf_line.geometry.unary_union
point = poly.centroid
split_line = split(line, nearest_neighbor_within(line.coords,point,tolerance)
return split_line[0].length
def get_distance_from_line(gdf_line, poly, tolerance=10000):
# union all geometries
line = gdf_line.geometry.unary_union
point = poly.centroid
return point.distance(nearest_neighbor_within(line.coords,point,tolerance))
def get_side_of_line(gdf_line, poly, tolerance=10000):
# union all geometries
line = gdf_line.geometry.unary_union
point = poly.centroid
nearest_point = nearest_neighbor_within(line.coords,point,tolerance)
if (nearest_point.coords.xy[0] < point.coords.xy[0]
and nearest_point.coords.xy[1] < point.coords.xy[1]):
return 'NE'
```

```
else:

return 'SW'
```

Count number of each crop in a radius around polygon

```
return point.distance()

train_test_shp['distance_along_river'] =
train_test_shp.geometry.apply(

lambda x: get_distance_along_line(orange_river_segment_shp, x))

train_test_shp['distance_from_river'] =
train_test_shp.geometry.apply(

lambda x: get_distance_from_line(orange_river_segment_shp, x))

train_test_shp['side_of_river'] =
train_test_shp.geometry.apply(

lambda x: get_side_of_line(orange_river_segment_shp, x))

train_test_shp['centroid_x'] =
train_test_shp.geometry.apply(lambda x: x.centroid)

train_test_shp['centroid_y'] =
train_test_shp.geometry.apply(lambda x: x.centroid )

train_test_shp['corner_count'] =
train_test_shp.geometry.apply(lambda x: len(x.bo)

train_test_shp['squareness'] =
train_test_shp.geometry.apply(lambda x: (x.area**.)

def count_nearby_crops(row):

search_region = row.geometry.buffer(300)
```

```
crop_list = []

for i, r in train_test_shp.iterrows():

if row.Field_Id != r.Field_Id and
r.geometry.within(search_region):

crop_list.append(r.Crop_Id_Ne)

crop_counts = {}

crops = ['1', '2', '3', '4', '5', '6', '7', '8', '9']

for crop in crops:

crop_counts[f'nearby_crop_{crop}_count'] =
crop_list.count(crop)

crop_counts['total_nearby_crop_count'] = sum([c for c in
crop_counts.values(

for crop in crops:

crop_counts[f'nearby_crop_{crop}_prop'] =
crop_counts[f'nearby_crop_{crop

crop_counts['total_nearby_field_count'] = len(crop_list)

crop_counts['Field_Id'] = row.Field_Id

return crop_counts
```

Join with polygon image stats

```
Out[41]: (3568, 380)

from joblib import Parallel, delayed

from tqdm import tqdm

def apply_count_nearby_crops(df):
```

```
return list(df.apply(count_nearby_crops,axis=1))

bs = 223

dfs =
[train_test_shp.iloc[i*bs:min((i+1)*bs,len(train_test_shp))]

for i in range

lists = Parallel(n_jobs=-
1)(delayed(apply_count_nearby_crops)(df)

for df in dfs)

crop_count_df = pd.concat([pd.DataFrame(l) for l in lists])

crop_count_df.to_csv(data_path/'nearby_crop_count.csv',
index=False)

shape_features = train_test_shp.merge(crop_count_df,
how='left', on='Field_Id')

shape_and_img_features =
shape_features.merge(poly_img_stats_df, how='left', on='

shape_and_img_features.to_csv(data_path/'features.csv',
index=False)

shape_and_img_features.shape
```

TRAIN MODEL.PY

```
import numpy as np

import pandas as pd

import sklearn

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import log_loss
```

```python
import lightgbm as lgbm
import xgboost as xgb
from catboost import CatBoostClassifier
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import StratifiedKFold
from category_encoders import TargetEncoder
from pathlib import Path
from skopt import BayesSearchCV
from skopt import gp_minimize # Bayesian optimization using Gaussian Processes
from skopt.space import Real, Categorical, Integer
from skopt.utils import use_named_args # decorator to convert a list of parameter
from skopt.callbacks import DeadlineStopper # Stop the optimization before runnin
from skopt.callbacks import VerboseCallback # Callback to control the verbosity
from skopt.callbacks import DeltaXStopper # Stop the optimization If the last two
from time import time
from pprint import pprint
from sklearn.metrics import average_precision_score
from sklearn.metrics import make_scorer
from imblearn.over_sampling import SMOTE, SMOTENC

data_path = Path('../data')

features_df = pd.read_csv(data_path/'features.csv')



categorical_cols =
['Subregion','tile','side_of_river','Crop_Id_Ne']

dtypes = {c:'category' for c in categorical_cols}

features_df = pd.read_csv(data_path/'features.csv',
dtype=dtypes)

cols_to_drop = ['Crop_Id_Ne',

'Field_Id',

'train_test',

'geometry']

features_dummies_df =
pd.get_dummies(features_df.drop(cols_to_drop,axis=1))

X = features_dummies_df[features_df.train_test ==
'train']#.drop(cols_to_drop, ax

y = features_df[features_df.train_test ==
'train'].Crop_Id_Ne.values

X_test = features_dummies_df[features_df.train_test ==
'test']#.drop(cols_to_drop

categorical_idx = list(range(373,386))

sm = SMOTENC(random_state=42,
categorical_features=categorical_idx)
```

```
X_res, y_res = sm.fit_resample(X, y)

skf = StratifiedKFold(n_splits=5,shuffle=True,random_state=42)

skf.get_n_splits(X,y)

Fold: 1

lgbs=[]

i=0

for train_index, valid_index in skf.split(X,y):

i=i+1

print(f'Fold: {i}')

X_train, X_valid = X_res[train_index], X_res[valid_index]

y_train, y_valid = y_res[train_index], y_res[valid_index]

lgb = lgbm.LGBMClassifier(n_estimators=100000,

categorical_feature=categorical_idx,

class_weight='balanced',

max_bin=5,

num_leaves=15,

min_data_in_leaf=140,

min_sum_hessian_in_leaf=2,

bagging_fraction=0.7,

bagging_freq=1,

feature_fraction=0.4,

lambda_l1=0.6,
lambda_l2=0,

min_gain_to_split=0.01,

max_depth=15

)

lgb.fit(X_train, y_train,
eval_set=[(X_train,y_train),(X_valid,y_valid)],earl

lgbs.append(lgb)

0 l l

0 068377

lid ti

1 l l

0 578594

xgbs=[]

i=0

for train_index, valid_index in skf.split(X,y):

i=i+1

print(f'Fold: {i}')

X_train, X_valid = X_res[train_index], X_res[valid_index]

y_train, y_valid = y_res[train_index], y_res[valid_index]

xg =
xgb.XGBClassifier(n_estimators=10000)#,colsample_bytree=0.3)

xg.fit(X_train, y_train, eval_set=[(X_train,y_train),
(X_valid,y_valid)],
eval_metric='mlogloss', early_stopping_rounds=30,verbose=10)

xg.fit(X_train, y_train)

xgbs.append(xg)

In [171]:

In [172]:

importances = []

for k, l in enumerate(lgbs):

importances.append(pd.DataFrame(

{'model': f'lgbm fold {k}',

'feature': features_dummies_df.columns,

'importance': l.feature_importances_}))

for k, x in enumerate(xgbs):

importances.append(pd.DataFrame(

{'model': f'xgboost fold {k}',

'feature': features_dummies_df.columns,

'importance': x.feature_importances_}))

importance_df = pd.concat(importances,
axis=0).groupby('feature').importance.sum(

importance_df['importance %'] =
importance_df.importance/importance_df.importance

importance_df.sort_values(by='importance',
ascending=False)5/20/2021
```

field_id crop_id_1 crop_id_2 crop_id_3 crop_id_4 crop_id_5 crop_id_6 crop_id_7 crop_id_8

0

5

0.00321

0.4321

0.677

0.1323

0.97

0.432

0.432

0.123

```
all_preds = []

for clf in [*lgbs]:

all_preds.append(clf.predict_proba(X_test.values))

for clf in [*xgbs]:

all_preds.append(clf.predict_proba(X_test.values))

preds=np.stack(all_preds).mean(axis=0)

pd.read_csv(data_path/'sample_submission_fixed.csv').head(1)

field_ids =
features_df[features_df.train_test=='test'].Field_Id.values

preds_df = pd.DataFrame(preds)

preds_df.columns = [f'crop_id_{c+1}' for c in preds_df.columns]

preds_df['field_id'] = field_ids

predictions = preds_df[['field_id',
'crop_id_1',

'crop_id_2',

'crop_id_3',

'crop_id_4',

'crop_id_5',

'crop_id_6',

'crop_id_7',

'crop_id_8',

'crop_id_9']]

predictions.to_csv(data_path/'submission.csv',index=False)
```

7 PROJECT 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 tests. Each test type addresses a specific testing requirement.

7.1 VARIOUS TESTING CASES

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.

2. INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they 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.

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.

Systems/Procedure: 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.

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.

5. 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.

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.

7. UNIT TESTING

Unit testing is usually conducted as part of a combined code and unit test phase of the software life cycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

7.2 TEST STRATEGY AND APPROACH

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

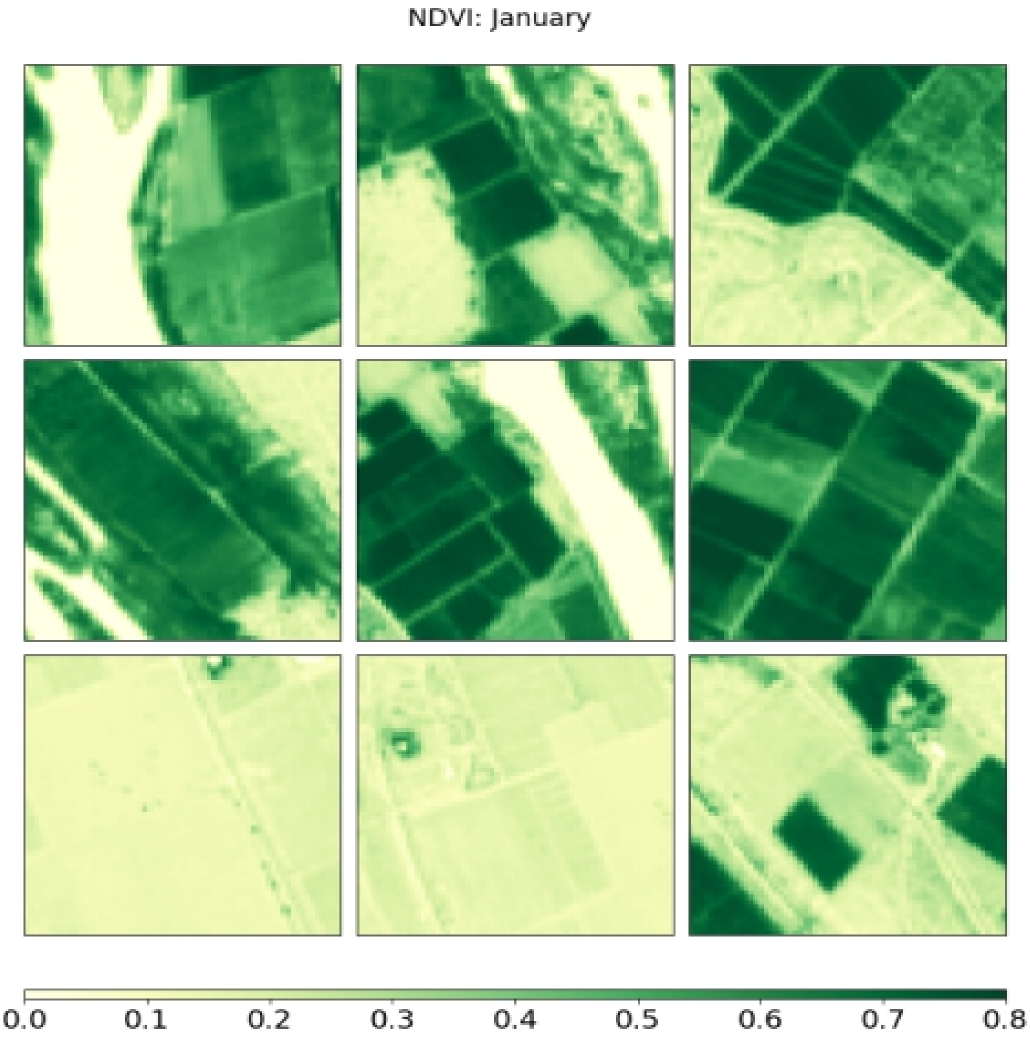
Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

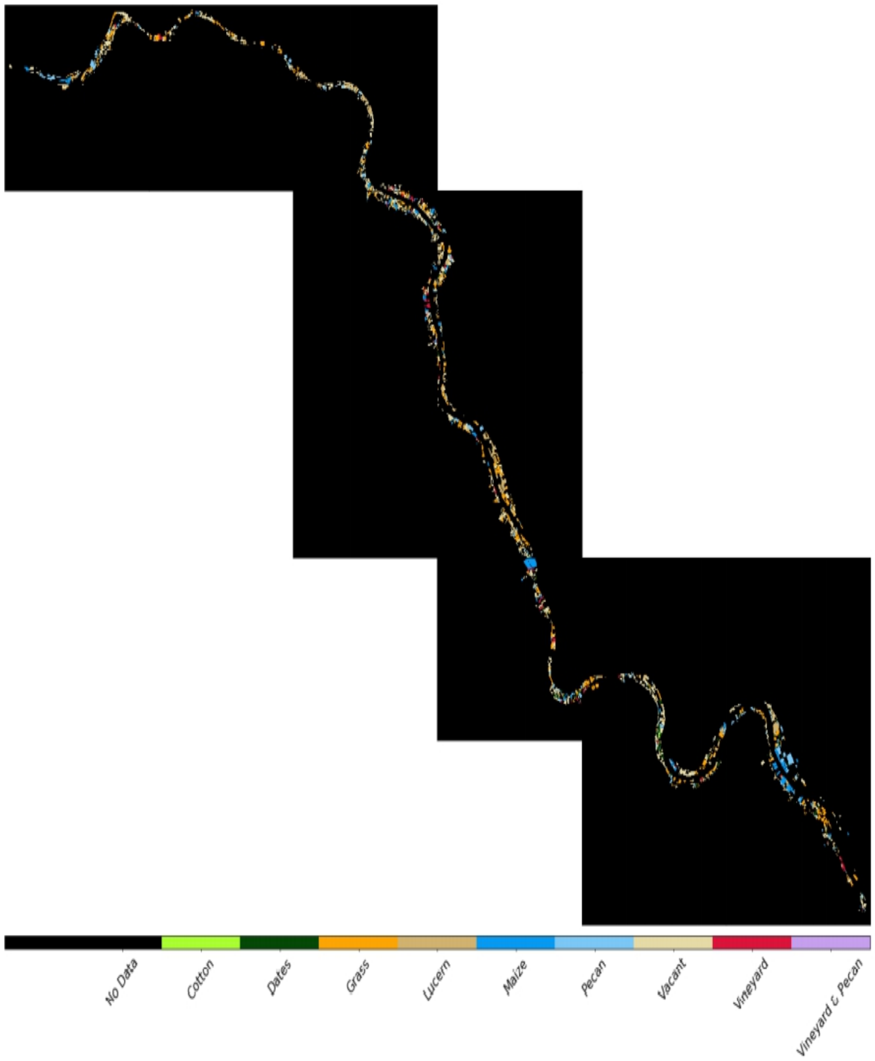
Features to be tested.

- 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.

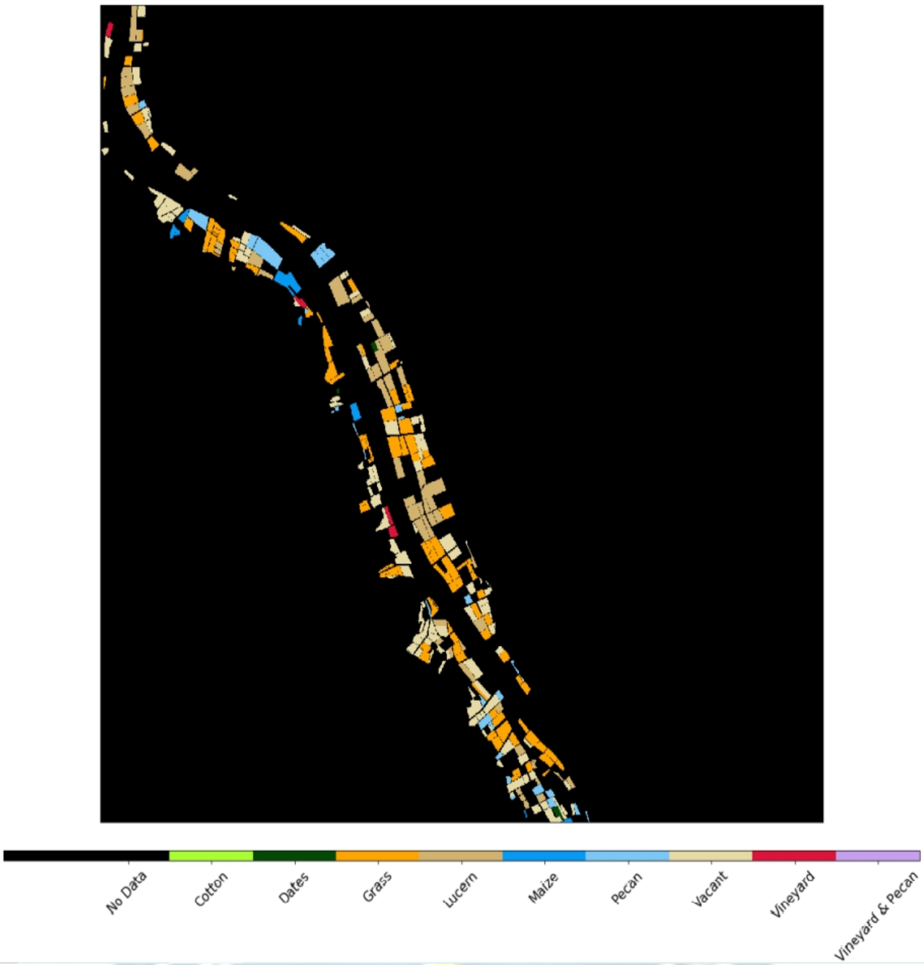
8 RESULTS/OUTPUT



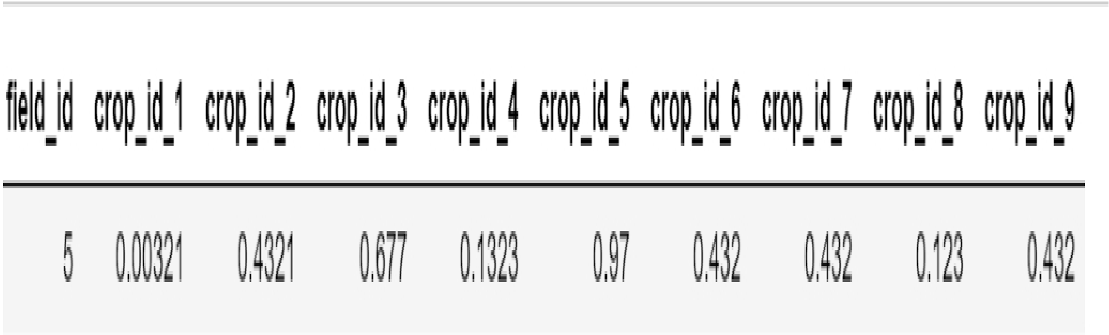
In above screen we can see how the farm looks in the month of January we have considered different parts of land, also we can observe the scale which shows the greenery.



In above screen it gives us the crop types.

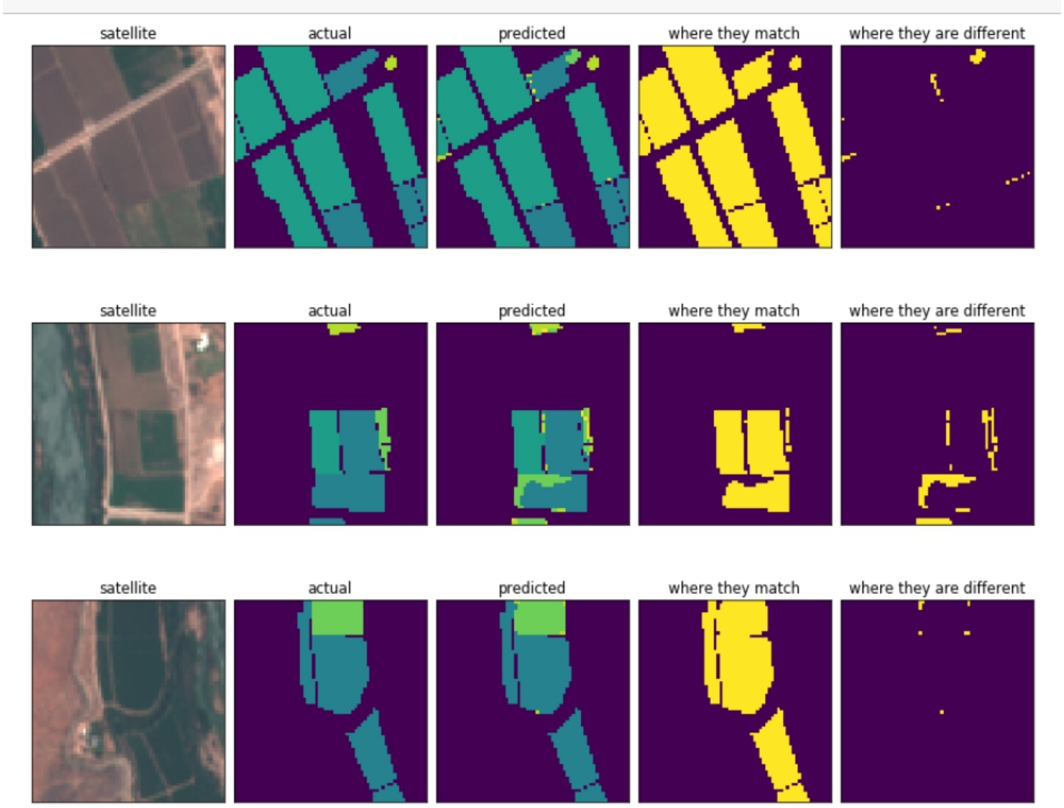


In above image we can see the different variety of plants and the place where the plants are located and the shadings are made according to the particular plant location.



field_id	crop_id_1	crop_id_2	crop_id_3	crop_id_4	crop_id_5	crop_id_6	crop_id_7	crop_id_8	crop_id_9
5	0.00321	0.4321	0.677	0.1323	0.97	0.432	0.432	0.123	0.432

In above screen we can see the prediction of the plants with respect to each plant which is represented using crop id.



In above screen we can observe the difference between the predicted and the actual one and also places where they match and where they are different.

9. CONCLUSION

We have developed a DL approach using LSTM for plant growth and crop yield prediction, achieving high prediction accuracy in both problems. Experimental results were presented that show that the DL technique (using a LSTM model) outperformed other traditional ML techniques, such as SVR and RF. Hence, the main aim of our project is to develop DL methodologies to predict plants growth and yield in greenhouse environment.

10. FUTURE ENHANCEMENT

Future studies looking at the continuity of :

a) greatly increase the number of collected data that are used for training the proposed DL methods;

b) extending the DL method so as to perform multi-step (at a weekly, or a multiple of weeks basis) prediction of growth and yield in a large variety.

11. REFERENCES

1.Abreu, P., Meneses, J. & Gary, C. 1998, "Tompousse, a model of yield prediction for tomato crops: calibration study for unheated plastic greenhouses", *XXV International Horticultural Congress, Part 9: Computers and Automation, Electronic Information in Horticulture 519*, pp. 141.

2.Adams, S. 2001, "Predicting the weekly fluctuations in glasshouse tomato yields", *IV International Symposium on Models for Plant Growth and Control in Greenhouses: Modeling for the 21st Century-Agronomic and 593*, pp. 19.

3.Atanasova, N., Todorovski, L., Džeroski, S. & Kompare, B. 2008, "Application of automated model discovery from data and expert knowledge to a real-world domain: Lake Glumsø", *Ecological Modelling,* vol. 212, no. 1-2, pp. 92-98.

4.Barandiaran, I. 1998, "The random subspace method for constructing decision forests", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 20, no. 8.

5.Breiman, L. 2001, "Random forests", *Machine Learning,* vol. 45, no. 1, pp. 5-32.

6.Buhmann, M.D. 2003, *Radial basis functions: theory and implementations,* Cambridge university press.

7.Chlingaryan, A., Sukkarieh, S. & Whelan, B. 2018, "Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review", *Computers and Electronics in Agriculture,* vol. 151, pp. 61-69.

8.Cortes, C. & Vapnik, V. 1995, "Support-vector networks", *Machine Learning,* vol. 20, no. 3, pp. 273-297.

9.Daniel, J., Andrés, P., Héctor, S., Miguel, B. & Marco, T. 2008, "A survey of artificial neural network-based modeling in agroecology" in *Soft Computing applications in industry* Springer, , pp. 247-269.

A

PROJECT REPORT

On

ROBOTIZING E-GOVERNMENT UTILIZING AI

Submitted by

Ms. B. PAVANI	**(17K81A1203)**
Mrs. B. LAYA	**(17K81A1204)**
Mr. K. SAI LIKITH REDDY	**(17K81A1221)**
Mr. G. NEERAJ VARMA	**(17K81A1214)**

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Mrs SREE VRINDA G.M

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN’S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **ROBOTIZING E-GOVERNMENT UTILIZING AI**, is being submitted by **B.PAVANI (17K81A1203) ,B.LAYA (17K81A1204), K.SAI LIKITH REDDY (17K81A1221), G.NEERAJ VARMA(17K81A1214)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN** INFORMATION TECHNOLOGY is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Project Guide	Head of the Department
SREE VRINDA G.M	**DR.R.NAGARAJU**
Department of Information Technology	Department of Information Technology

Internal Examiner	External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of Information Technology, session: 2017 – 2021, St. Martin’s Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled ROBOTIZING E-GOVERNMENT UTILIZING AI is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

B. PAVANI	**17K81A1203**
B. LAYA	**17K81A1204**
K.SAI LIKITH REDDY	**17K81A1221**
G. NEERAJ VARMA	**17K81A1214**

LASYA
INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **B.PAVANI** WITH ROLL NO.**17K81A1221**, **BOBBASANI LAYA** WITH ROLL NO.**17K81A1204**, **G.NEERAJ VARMA** WITH ROLL NO.**17K81A1214**, **K.SAI LIKITH REDDY** WITH ROLL NO.**17K81A1221**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**AUTOMATING E-GOVERNANCE USING AI**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd. ★ Kompally-500014 ★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin’s Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr.Mr.R.Nagaraju**, Head of the Department, Information Technology , St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Mrs.SREE VRINDA G.M**, Assistant Professor, Department of Information Technology, St. Martin’s Engineering College, Dhulapally, for her support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

B. PAVANI	**17K81A1203**
B. LAYA	**17K81A1204**
K.SAI LIKITH REDDY	**17K81A1221**
G. NEERAJ VARMA	**17K81A1214**

TABLE OF CONTENTS

ABSTRACT

Artificial Intelligence (AI) has recently advanced the state-of-art results in an ever-growing number of domains. However, it still faces several challenges that hinder its deployment in the e-government applications–both for improving the e-government systems and the e-government-citizens interactions. In this paper, we address the challenges of e-government systems and propose a framework that utilizes AI technologies to automate and facilitate e-government services. E-government is the application of employing advanced electronic techniques and web services to present, exchange, and advance the government's services for citizens and businesses with a goal of improving the productivity while reducing the cost. E-government plays a critical role in advancing the economy of the government, citizens, and industry, especially for developing countries. It facilitates the business-to-business transactions and tasks (B2B), brings customers closer to businesses (B2C), allow productive interactions between the government and citizens (G2C), government and enterprises (G2B), and inter-agency and relationships (G2G) in more convenient, transparent and economic way specifically, we first outline a framework for the management of e-government information resources. Second, we develop a set of deep learning models that aim to automate several e-government services. Third, we propose a smart e-government platform architecture that supports the development and implementation of AI applications of e-government. Our overarching goal is to utilize trustworthy AI techniques in advancing the current state of e-government services in order to minimize processing times, reduce costs, and improve citizens' satisfaction..

LIST OF FIGURES

1.INTRODUCTION

AI and E-governance have been a buzz since the United Nations called its members to integrate intelligent applications and to enhance their governance procedures to act more closely to citizens for offering better services. The UN aims at implementing e-governance to help developing and under-developed countries in saving the cost and time of the government as it can be a better economic growth driving factor. AI and its sub- domain technologies have the potential to ameliorate the several existing structural inefficiencies to offer satisfied government functionalities. AI and E-governance could be a revolution in integrating and embedding technologies.

Among the humongous list of government functions and responsibilities, implementation of evolving policies, passing files to various fronts, delivering services, enabling businesses to sustain, maintaining law & order, reducing the cost of living, etc., would be top priorities. The communication and interaction gap between government, public, business before enforcing a law, rule, or policy is the biggest barrier. E-governance refers to the strategic integration of intelligent systems to create a simple, moral, accountable, reasonable, responsive, and transparent environment that is comfortable and less costly for interacting between citizens, businesses, and government.

Importance of AI and E-governance:

- It simplifies the process of gathering and accumulation of government information regarding any department to the citizens and business.
- It helps citizens and businesses to participate in the processes of decision making, before developing or implementing any new law or policy.
- It is the best way to eliminate corruption by automating the services and ensuring transparency in the information communicated and it is easily accessible to the public.
- Ease of availability of government services 24*7 for every citizen through online applications.

1.1 PROJECTOVERVIEW

E-government is the application of employing advanced electronic techniques and webservices to present, exchange, and advance the government's services for citizens and businesses with a goal of improving the productivity while reducing the cost. E-government plays a critical role in advancing the economy of the government, citizens, and industry, especially for developing countries. It facilitates the business-to-business transactions and tasks(B2B), brings customers closer to businesses (B2C), allow productive interactions between the government and citizens (G2C), government and enterprises (G2B), and inter-agency and relationships (G2G) in more convenient, transparent and economic ways. ability and efficiency of the government services while reducing cost. Moreover, implementing e-government applications can foster several other advantages including, but not limited to, the following:

- Transparency
- Trust
- Citizen participation
- Environment support
- Lack of experts
- Inaccessibility.
- Security

1.2PROJECTOBJECTIVES

- The main objective of this application is to analyze users post and detect the feedback on the government.
- We are using CNN (convolutional neural network) algorithm to detect handwritten images, facial expression and feedback on the government
- With the help of CNN algorithm we are able to recognize the handwritten images, we are able to detect the facial expression of the user and in feedback we can analyze analyze users post and give result as negative orpositive

1.3 SCOPE OF THEPROJECT

- Man-made reasoning (ai) has as of late progressed the condition of-craftsmanship brings about a steadily developing number of areas.
- It actually faces a few difficulties that block its sending in the e-government applications–both for improving the e-government frameworks and the e-government- residents collaborations.
- we presented the meanings of man-made consciousness and e-government, momentarily examined the present status of e-government lists all throughout the planet, and afterward proposed our answers for advance the present status of e-government,
- A system for the board of government data assets that help deal with the e-government lifecycle start to finish.
- A bunch of profound learning procedures that can help work with and robotize a few driven organizations.
- The stage with coordinate late advances in ai strategies in the e-government frameworks and administrations to improve the general trust, straightforwardness, and proficiency of e-government.

1.4ORGANIZATION OFCHAPTERS

1.4.1 INTRODUCTION

AI and E-governance have been a buzz since the United Nations called its members to integrate intelligent applications and to enhance their governance procedures to act more closely to citizens for offering better services. The UN aims at implementing e-governance to help developing and under-developed countries in saving the cost and time of the government as it can be a better economic growth driving factor. AI and its sub- domain technologies have the potential to ameliorate the several existing structural inefficiencies to offer satisfied government functionalities. AI and E-governance could bea revolution in integrating and embedding technologies. Importance of AI and E-governance:

1. It simplifies the process of gathering and accumulation of government information regarding any department to the citizens and business.

2. It helps citizens and businesses to participate in the processes of decision making, before developing or implementing any new law or policy.

3. It is the best way to eliminate corruption by automating the services and ensuring transparency in the information communicated and it is easily accessible to the public.

4. Ease of availability of government services 24*7 for every citizen through online applications.

5. E-governance helps business access information that is important at a click away.

1.4.2 LITERATURESURVEY

The public value of E-Government – A literature review

This study organizes existing research on the public value of e-government in order to investigate the current state and what value e-government is supposed to yield. The two questions that guided the research were: (1) What is the current state of research on the public value of e-government? And (2) What value is e-government supposed to yield? Six, sometimes overlapping, values were found: Improved public services; improved administrative efficiency; Open Government (OG) capabilities; improved ethical behaviour and professionalism; improved trust and confidence in government; and improved social value and well-being. These six public value dimensions were thereafter generalized into three overarching, and also overlapping, public value dimensions of Improved Public Services, Improved Administration, and Improved Social Value.

Engendering inclusive e-government use through citizen IT training programs

This study is motivated by two objectives:

1) To evaluate whether citizen participation in government training programs is associated with greater e-government use among participants.

2) To assess whether the strength of this relationship varies according to whether a citizen is elderly, disabled, or not – those who are elderly or disabled tend to use e-

government the least.

1.4.3 SOFTWARE & HARDWAREREQUIREMENTS

SOFTWARE REQUIREMENTS

□ Operating system	: Windows 7 Ultimate.
□ Coding Language	: Python.
□ Designing	:Html,css,javascript.
□ DataBase	:Files

HARDWARE REQUIREMENTS

□ Processor	: Pentium IV 2.4GHz.
□ HardDisk	: 1 TB
□ Ram	: 4GB

1.4.4 SOFTWARE DEVELOPMENTANALYASIS

Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It is used for web development (server-side),software development ,mathematics ,system scripting.

Artificial Intelligence

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

1.4.5 PROJECT SYSTEMDESIGN

- **Generate Hand Written Digits Recognition Deep Learning Model**: It using this model we are building CNN based hand written model which take digit image as input and then predict the name of digit.
- **Generate Text & Image Based Sentiment Detection Deep Learning Model**: It using this module we will generate text and image based sentiment detection model.
- **Upload Test Image & Recognize Digit**: By using this module we will upload text image and apply train model to recognize digit.
- **Write Your Opinion About Government Policies**: By using this module we will accept user's opinion and then save that opinion inside application to detect sentiment from opinion.
- **View Peoples Sentiments From Opinions**: By using this module user can see all users opinion and their sentiments detected through CNNmodel.
- **Upload Your Face Expression Photo About Government Policies**: By using this module user will upload his image with facial expression which indicates whether user is satisfy with this scheme ornot.

1.4.6 PROJECT CODING

Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It is used for web development (server-side),software development, mathematics, system scripting.

The libraries used in our project are

Tkinter -it is the standard Python interface to the Tk GUI toolkit.

matplotlib.pyplot -it provides a MATLAB-like plotting frame work.

Numpy -it is the fundamental package for scientific computing in
Python Joblib -it is a set of tools to provide lightweight pipelining in
Python keras.models -it represents the actual neural network model.

cv2 -itis a library which is used to bindings designed to solve computer vision problems-

1.4.7 PROJECT 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.

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.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program

Functional testing

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

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.

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.

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.

1.4.8 INPUTSCREENS

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 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.

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.

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.

1.4.9 OUTPUTSCREENS

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

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.

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. Select methods for presenting information. 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.

1.4.10 CONCLUSIONS

With the recent advances in AI and deep learning technologies, more government agencies are starting to use such technologies to improve their systems and services However, a large set of challenges hinder the adoption of such technologies, including the lack of experts, computational resources, trust, and AI interpretability

In this paper, we introduced the definitions of artificial intelligence and e-government, briefly discussed the current state of e-government indices around the world, and then proposed our solutions to advance the current state of e-government, considering the Gulf Countries as a case-study. We proposed a framework for management of government information resources that help manage the e-government lifecycle end-to-end. Then, we proposed a set of deep learning techniques that can help facilitate and automate several e- government services. After that, we proposed a smart platform for AI development and implementation in e-government. The overarching goal of this paper is to introduce new frameworks and platform to integrate recent advances in AI techniques in the e- government systems and services to improve the overall trust, transparency, and efficiency ofe-government.

2. LITERATURESURVEY

2.1 SURVEY ONBACKGROUND

The public value of E-Government – A literature review

This study organizes existing research on the public value of e-government in order to investigate the current state and what value e-government is supposed to yield. The two questions that guided the research were: (1) What is the current state of research on the public value of e-government? And (2) What value is e-government supposed to yield? Six, sometimes overlapping, values were found: Improved public services; improved administrative efficiency; Open Government (OG) capabilities; improved ethical behavior and professionalism; improved trust and confidence in government; and improved social value and well-being. These six public value dimensions were thereafter generalized into three overarching, and also overlapping, public value dimensions of Improved Public Services, Improved Administration, and Improved Social Value.

Engendering inclusive e-government use through citizen IT training programs

This study is motivated by two objectives:

1) To evaluate whether citizen participation in government training programs is associated with greater e-government use among participants.

2) To assess whether the strength of this relationship varies according to whether a citizen is elderly, disabled, or not – those who are elderly or disabled tend to use e-government the least.

2.2 CONCLUSION ONSURVEY

Artificial Intelligence and Machine Learning are products of both science and myth. The idea that machines could think and perform tasks just as humans do is thousands of years old. The cognitive truths expressed in AI and Machine Learning systems are not new either. It may be better to view these technologies as the implementation of powerful and long-established cognitive principles through engineering.

We should accept that there is a tendency to approach all important innovations as a

Rorschach test upon which we impose anxieties and hopes about what constitutes a good or happy world. But the potential of AI and machine intelligence for good does not lie exclusively, or even primarily, within its technologies. It lies mainly in its users. If we trust (in the main) how our societies are currently being run then we have no reason not to trust ourselves to do good with these technologies. And if we can suspend presentism and accept that ancient stories warning us not to play God with powerful technologies are instructive then we will likely free ourselves from unnecessary anxiety about their use.

3.SOFTWARE AND HARDWAREREQUIREMENTS

3.1 SOFTWAREREQUIREMENTS

- Operating system : Windows 7Ultimate.
- Coding Language : Python.
- Designing :Html,css,javascript.
- DataBase :Files

3.2 HARDWAREREQUIREMENTS

- Processor : Pentium IV 2.4GHz.
- Hard Disk : 1 TB
- Ram : 4GB

Functional Requirements

- Graphical User interface with the User.

Debugger and Emulator

- Any Browser (ParticularlyChrome)

4.SOFTWARE DEVELOPMENTANALYASIS

4.1 OVERVIEW OF PROBLEM

Recently, many countries have adopted e-government services in various departments and many autonomous applications. While there are several studies conducted for enhancing e-government services. only a few of them address utilizing recent advances in AI and deep learning in the automation of e-government services. Therefore, there is still an urgent need to utilize state-of-the-art AI techniques and algorithms to address e-government challenges and needs.

4.2 DEFINE THEPROBLEM

In this paper, we propose a novel framework that utilizes recent advances in AI to improve the e-government systems and their interactions with the citizens. First, we propose a framework to automate and facilitate the management of e-government systems using AI techniques. Second, we develop and present several deep learning models that aim at automating e-government services including automatic recognition of hand-written digit sand letters and sentiment analysis. Third, we propose an platform for smart e-government services development and implementation.

This paper author describing concept to automate government services with Artificial Intelligence technology such as Deep Learning algorithm called Convolution Neural Networks (CNN). Government can introduce new schemes on internet and peoples can read news and notifications of such schemes and then peoples can write opinion about such schemes and this opinions can help government in taking better decisions. To detect public opinions about schemes automatically we need to have software like human brains which can easily understand the opinion which peoples are writing is in favor of positive or negative.

4.3 MODULEOVERVIEW

The modules used in our project are

1. Generate Hand Written Digits Recognition Deep Learning Model:
2. Generate Text & Image Based Sentiment Detection Deep Learning Model:

3. Upload Test Image & Recognize Digit
4. Write Your Opinion About Government Policies
5. View Peoples Sentiments From Opinions
6. Upload Your Face Expression Photo About Government Policies

4.4 DEFINE THEMODULES

- **Generate Hand Written Digits Recognition Deep Learning Model**: It using this model we are building CNN based hand written model which take digit image as input and then predict the name of digit. CNN model can be generated by taking two types of images called train (train images contain all possible shapes of digits human can write in all possible ways) and test (Using test images train model will be tested whether its giving better prediction accuracy). Using all train images CNN will build the training model. While building model we will extract features from train images and then build a model. While testing also we will extract features from test image and then apply train model on that test image to classify it.

- **Generate Text & Image Based Sentiment Detection Deep Learning Model**: It using this module we will generate text and image based sentiment detection model. All possible positive and negative words will be used to generate text based sentiment model. All different types of facial expression images will be used to generate image based sentiment model. Whenever we input text or image then train model will be applied on that input to predict its sentiments.

- **Upload Test Image & Recognize Digit**: By using this module we will upload text image and apply train model to recognize digit.

- **Write Your Opinion About Government Policies**: By using this module we will accept user’s opinion and then save that opinion inside application to detect sentiment from opinion.

- **View Peoples Sentiments From Opinions**: By using this module user can see all users opinion and their sentiments detected through CNN model.

- **Upload Your Face Expression Photo About Government Policies**: By using this module user will upload his image with facial expression which indicates whether user is satisfy with this scheme or not.

4.5 MODULESFUNCTIONALITY

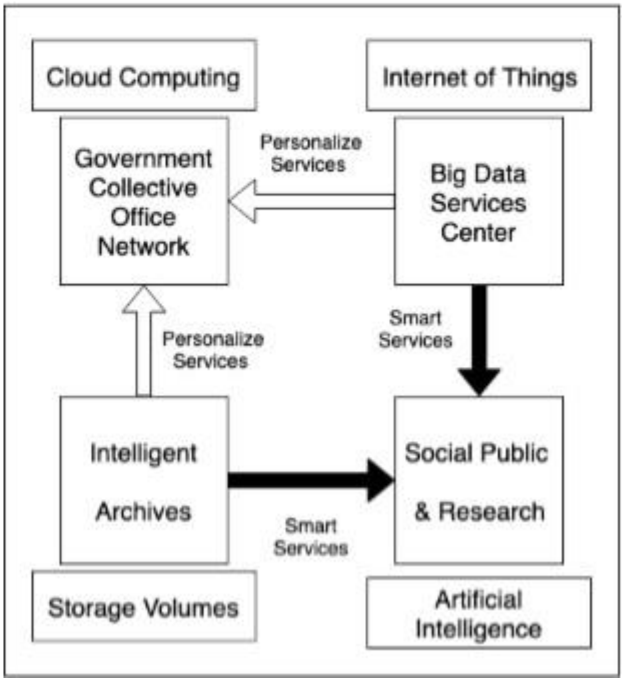


FIGURE 4.5 CNN ARCHITECTURE

It illustrates our proposed framework for a centralized management of e-government information resources. It consists of four main components: Government Collective Office Network, Big Data Services Center, Social Public and Research, and Intelligent Archives. These components utilize the advances in cutting-edge technology to enhance and facilitate the production, processing, and presentation of e-government resources, including, Cloud Computing services, Internet of Things, AI, and Storage utilities. We address in this paper AI technology being one of the active areas at the moment in addition to the challenges we mentioned in the Introduction Section. We also present several applications (i.e., deep learning models) that depict how AI applications can help automating several e-government services (we present our models in the next Section). The Government Collective Office Network is responsible to implement and ensure the correctness of e-government polices and services in alignment with all government offices and agencies. Big Data Services Center is responsible for all processes and policies regarding Big Data (collecting, storing, processing, transmitting). Moreover, this unit plays a critical role in ensuring the privacy and security of the citizens and government data. Social Public and Research is the unit responsible for providing e-services for the citizens and research organizations. It also includes a research agency concerned with advancing the current state of e-government ecosystem. Intelligent Archive unit is responsible to digitize paper documents and applications and provide smart and personalized services to other units that require accessing and consuming digital data.

5. PROJECTSYSTEM DESGIN

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modeling language.
- Encourage the growth of OO tools market.
- Support higher level development concepts such as collaborations, frameworks, patterns and components.

- Integrate best practices.

5.1.1 CLASSDIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

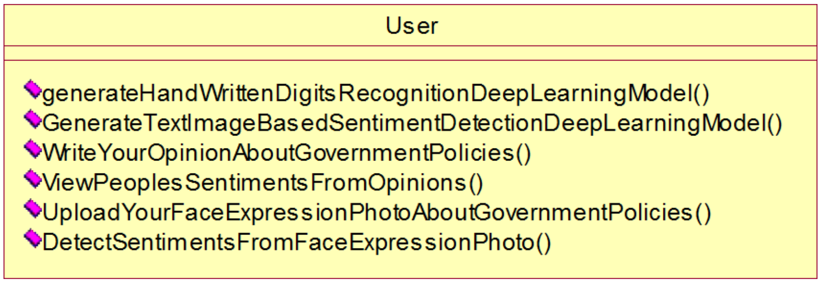

User
generateHandWrittenDigitsRecognitionDeepLearningModel()
GenerateTextImageBasedSentimentDetectionDeepLearningModel()
WriteYourOpinionAboutGovernmentPolicies()
ViewPeoplesSentimentsFromOpinions()
UploadYourFaceExpressionPhotoAboutGovernmentPolicies()
DetectSentimentsFromFaceExpressionPhoto()

Figure 5.1.1 CLASS DIAGRAM

5.1.2 USE CASEDIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

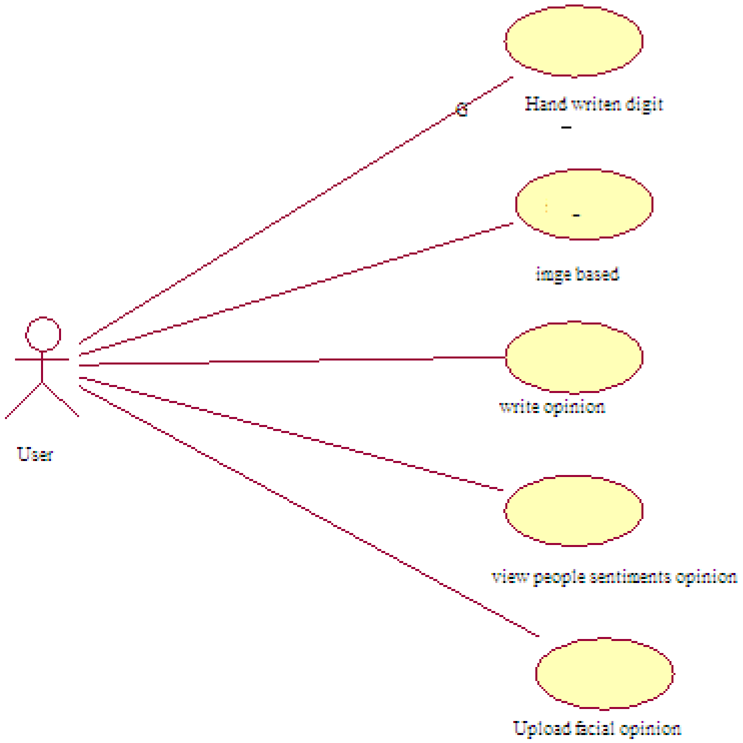


Figure 5.1.2 USE CASE DIAGRAM

5.1.3 SEQUENCEDIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

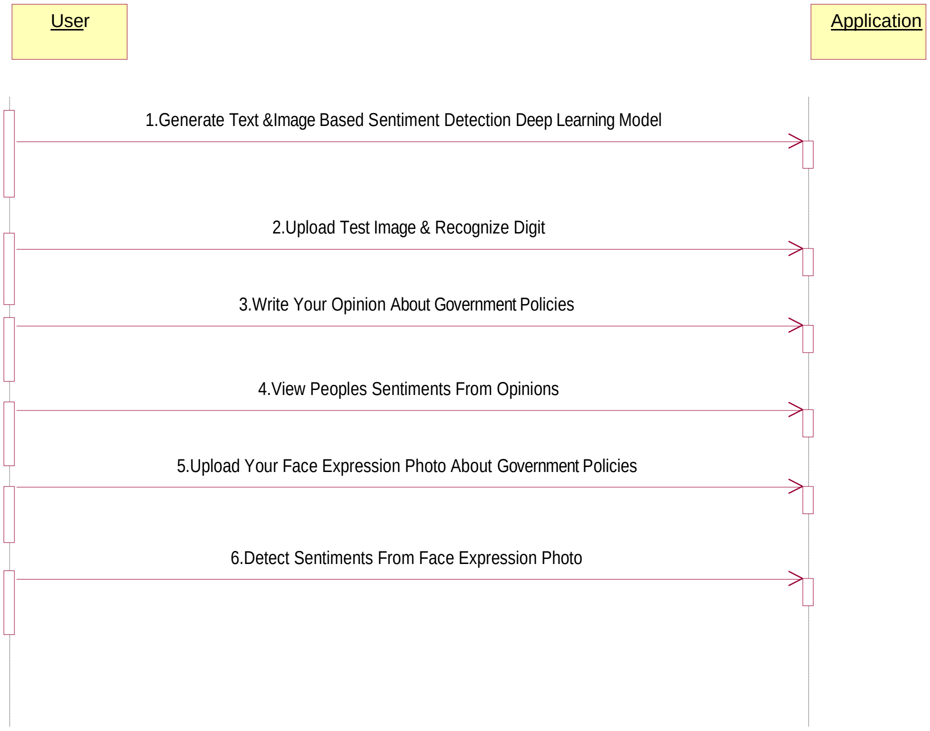

User
Application
1.Generate Text &Image Based Sentiment Detection Deep Learning Model
2.Upload Test Image & Recognize Digit
3.Write Your Opinion About Government Policies
4.View Peoples Sentiments From Opinions
5.Upload Your Face Expression Photo About Government Policies
6.Detect Sentiments From Face Expression Photo

Figure 5.1.3 SEQUENCE DIAGRAM

5.1.4 COLLABORATIONDIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

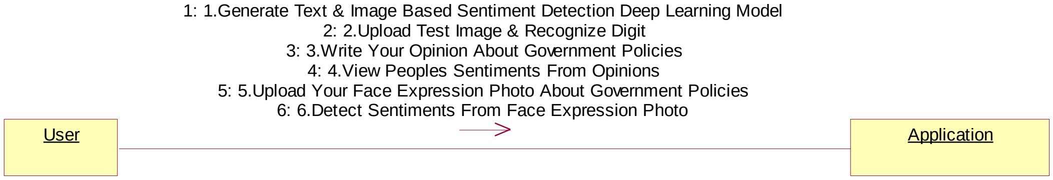

1: 1.Generate Text & Image Based Sentiment Detection Deep Learning Model
2: 2.Upload Test Image & Recognize Digit
3: 3.Write Your Opinion About Government Policies
4: 4.View Peoples Sentiments From Opinions
5: 5.Upload Your Face Expression Photo About Government Policies
6: 6.Detect Sentiments From Face Expression Photo
User
Application

Figure 5.1.4 COLLABORATION DIAGRAM

5.1.5 COMMUNICATION DIAGRAM

A Communication diagram models the interactions between objects or parts in terms of sequenced messages. Communication diagrams represent a combination of information taken from Class, Sequence, and Use Case Diagrams describing both the static structure and dynamic behavior of a system.

However, communication diagrams use the free-form arrangement of objects and links as used in Object diagrams. In order to maintain the ordering of messages in such a free-form diagram, messages are labeled with a chronological number and placed near the link the message is sent over. Reading a communication diagram involves starting at message 1.0, and following the messages from object to object.

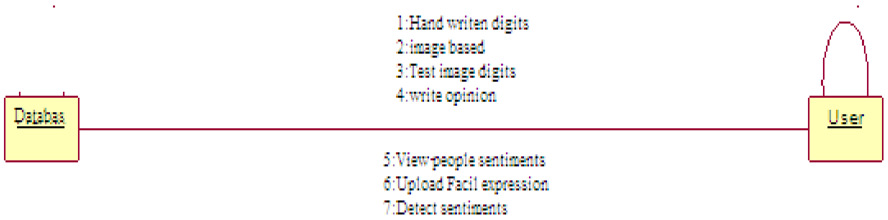


Figure 5.1.5 COMMUNICATION DIAGRAM

5.1.6 DEPLOYMENTDIAGRAM

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.[1] To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

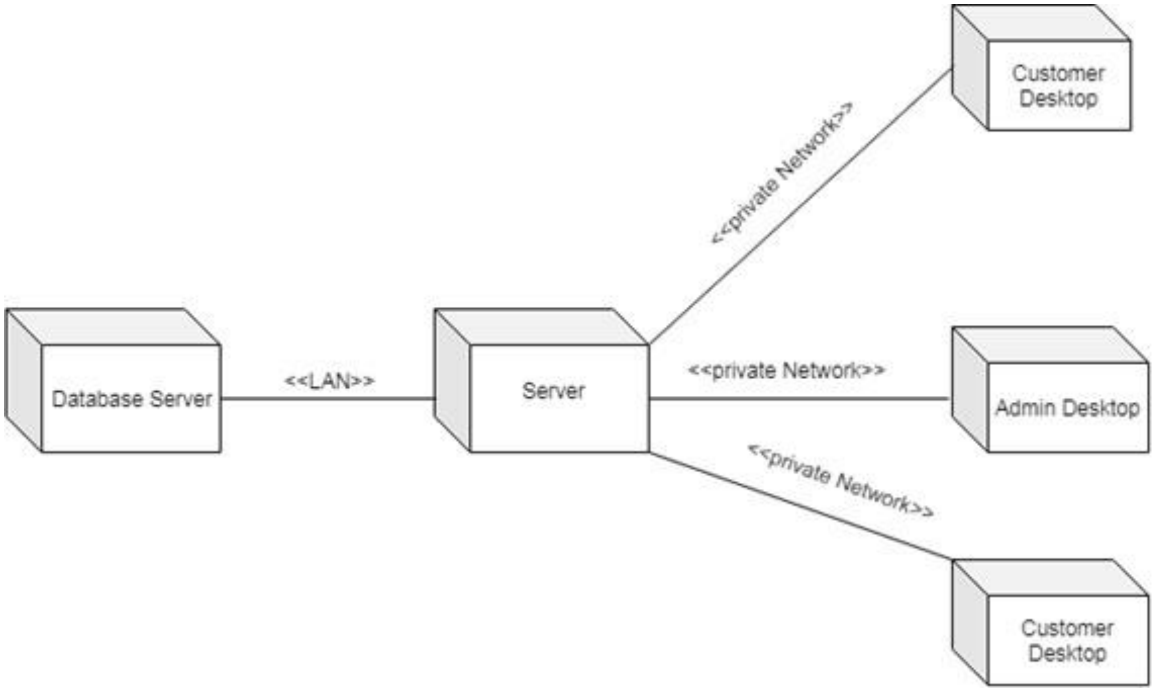


Figure 5.1.6 DEPLOYMENT DIAGRAM

5.1.7 PACKAGE DIAGRAM

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element ,dependency ,element import, package import, package merge.

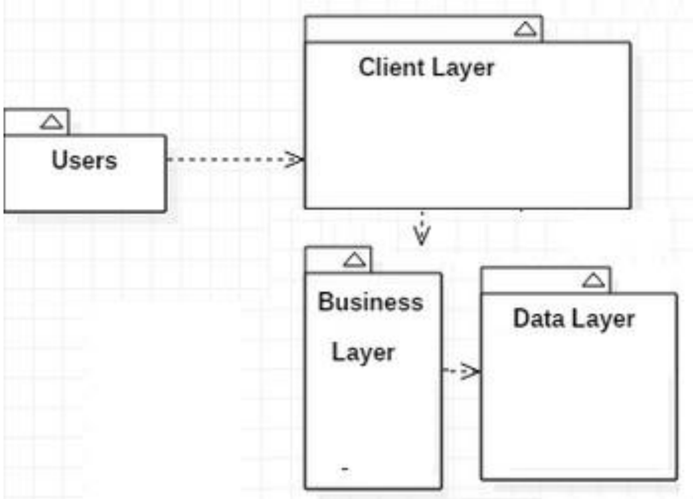


Figure 5.1.7 PACKAGE DIAGRAM

5.1.8 PROFILE DIAGRAM

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connectors and their components.

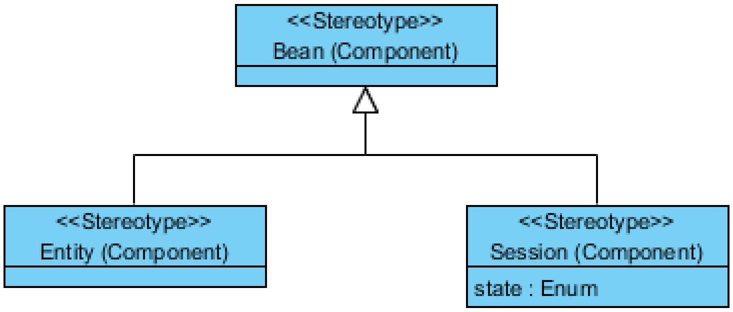


Figure 5.1.8 PROFILE DIAGRAM

6. PROJECT CODING

6.1 TECHNOLOGY

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. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython 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

What is Python

Python is a popular programming language. It was created by Guido van Rossum, 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 work flows.
- 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**.**

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for thispurpose.

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 Quickstart

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.pyThe 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"

Virtual Environments and Packages
Introduction

Python applications will often use packages and modules that don’t come as part of the standard library. Applications will sometimes need a specific version of a library, because the application may require that a particular bug has been fixed or the application may be written using an obsolete version of the library’s interface.

This means it may not be possible for one Python installation to meet the requirements of every application. If application A needs version 1.0 of a particular module but application B needs version 2.0, then the requirements are in conflict and installing either version 1.0 or 2.0 will leave one application unable to run.

The solution for this problem is to create a virtual environment, a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.

Different applications can then use different virtual environments. To resolve the earlier example of conflicting requirements, application A can have its own virtual environment with version 1.0 installed while application B has another virtual environment with version 2.0. If application B requires a library be upgraded to version 3.0, this will not affect application A’s environment.

Creating Virtual Environments

The module used to create and manage virtual environments is called venv. venv will usually install the most recent version of Python that you have available. If you have multiple versions of Python on your system, you can select a specific Python version by running python3 or whichever version you want.

To create a virtual environment, decide upon a directory where you want to place it, and run the venv module as a script with the directory path:

```
python3 -m venv tutorial-env
```

This will create the tutorial-env directory if it doesn't exist, and also create directories inside it containing a copy of the Python interpreter, the standard library, and various supporting files.

A common directory location for a virtual environment is .venv. This name keeps the directory typically hidden in your shell and thus out of the way while giving it a name that explains why the directory exists. It also prevents clashing with .env environment variable definition files that some tooling supports.

Once you've created a virtual environment, you may activate it. On Windows, run:

```
tutorial-env\Scripts\activate.bat
```

On Unix or MacOS, run:source tutorial-env/bin/activate

(This script is written for the bash shell. If you use the csh or fish shells, there are alternate activate.csh and activate.fish scripts you should use instead.)

Activating the virtual environment will change your shell's prompt to show what virtual environment you're using, and modify the environment so that running python will get you that particular version and installation of Python. For example:

```
$ source ~/envs/tutorialenv/bin/activate (tutorial-env) $ python
Python 3.5.1 (default, May 6 2016, 10:59:36)...
>>> import sys
>>>sys.path
['', '/usr/local/lib/python35.zip', ..., '~/envs/tutorial-env/lib/python3.5/site-packages']
>>>
```

You can install, upgrade, and remove packages using a program called pip. By default pip will install packages from the Python Package Index, <https://pypi.org>. You can browse the Python Package Index by going to it in your web browser, or you can use pip's limited searchfeature:

```
(tutorial-env) $ pip search astronomy
Skyfield                    - Elegant astronomy forPython
Gary                        - Galactic astronomy and gravitationaldynamics.
```

novas - The United States Naval Observatory NOVAS astronomylibrary

astroobs - Provides astronomy ephemeris to plan telescope observations

PyAstronomy - A collection of astronomy related tools forPython.

Artificial Intelligence

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 secton.

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.

Trending AI Articles:

- Cheat Sheets for AI, Neural Networks, Machine Learning, Deep Learning &Big Data

- Data Science Simplified Part 1: Principles and Process
- Getting Started with Building Realtime API Infrastructure
- AI & NLP Workshop

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.
- NaturalLanguageProcessing−Interactwiththecomputerthatunderstands natural language spoken by humans.
- Expert Systems − Machine or software provide explanation and advice to the users.
- VisionSystems−Systemsunderstand,explain,anddescribevisualinputonthe 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

CNN ALOGRITHM

CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "full connectivity" of these networks make them prone to overfitting data. Typical ways of regularization, or preventing overfitting, include: penalizing parameters during training (such as weight decay) or trimming connectivity (skipped connections, dropout, etc.) CNNs take a different approach towards regularization: they take advantage of the hierarchical pattern in data and assemble patterns of increasing complexity using smaller and simpler patterns embossed in their filters. Therefore, on a scale of connectivity and complexity, CNNs are on the lower extreme. Convolutional networks were inspired by biological processes[9][10][11][12]in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. Individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field. The receptive fields of different neurons partially overlap such that they cover the entire visual field.

CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns to optimize the filters (or kernels) through automated learning, whereas in traditional algorithms these filters are hand-engineered. This independence from prior knowledge and human intervention in feature extraction is a major advantage.

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.

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.

Decision Tree Learning

For general use, decision trees are employed to visually represent decisions and show or inform decision making. When working with machine learning and data mining, decision trees are used as a predictive model. These models map observations about data to conclusions about the data’s target value.

The goal of decision tree learning is to create a model that will predict the value of a target based on input variables.

In the predictive model, the data’s attributes that are determined through observation are represented by the branches, while the conclusions about the data’s target value are represented in the leaves.

When “learning” a tree, the source data is divided into subsets based on an attribute value test, which is repeated on each of the derived subsets recursively. Once the subset at a node has the equivalent value as its target value has, the recursion process will be complete. Let’s look at an example of various conditions that can determine whether or not someone should go fishing. This includes weather conditions as well as barometric pressure conditions.

In the simplified decision tree above, an example is classified by sorting it through the tree to the appropriate leaf node. This then returns the classification associated with the particular leaf, which in this case is either a Yes or a No. The tree classifies a day’s conditions based on whether or not it is suitable for going fishing.

A true classification tree data set would have a lot more features than what is outlined above, but relationships should be straightforward to determine. When working with decision tree learning, several determinations need to be made, including what features to choose, what conditions to use for splitting, and understanding when the decision tree has reached a clear ending.

Introduction to Deep Learning

What is deep learning

Deep learning is a branch of machine learning which is completely based on artificial neural networks, as neural network is going to mimic the human brain so deep learning is also a kind of mimic of human brain. In deep learning, we don’t need to explicitly program everything. The concept of deep learning is not new. It has been around for a couple of years now. It’s on hype nowadays because earlier we did not have that much processing power and a lot of data. As in the last 20 years, the processing power increases exponentially, deep learning and machine learning came in the picture. A formal definition of deep learning is-neurons

Deep learning is a particular kind of machine learning that achieves great power and flexibility by learning to represent the world as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations computed in terms of less abstract ones.

In human brain approximately 100 billion neurons all together this is a picture of an individual neuron and each neuron is connected through thousand of their neighbours.

The question here is how do we recreate these neurons in a computer. So, we create an artificial structure called an artificial neural net where we have nodes or neurons. We have some neurons for input value and some for output value and in between, there may be lots of neurons interconnected in the hiddenlayer**.**

6.2 CODING

```
from tkinter import messagebox

from tkinter import *

from tkinter import simpledialog

import tkinter

from tkinter import filedialog

from tkinter.filedialog import askopenfilename

import matplotlib.pyplot as plt

import numpy as np

import joblib

from keras.models import load_model

from keras.preprocessing.image import img_to_array

import cv2

from keras.models import model_from_json

from keras.preprocessing import image

from keras.optimizers import Adam
from keras.utils import np_utils

from keras.preprocessing import image

import os

from numpy import dot

from numpy.linalg import norm

from keras.models import Sequential

from keras.layers import Dense, Conv2D, Dropout, Flatten, MaxPooling2D

import imutils

import nltk



main = tkinter.Tk()

main.title("Automating E-Government")

main.geometry("1300x1200")


global filename

global text_sentiment_model

EMOTIONS = ["angry","disgust","scared", "happy", "sad", "surprised","neutral"]

global face_detection

global image_sentiment_model

global digits_cnn_model


def digitModel():

    global digits_cnn_model

    with open('models/digits_cnn_model.json', "r") as json_file:

loaded_model_json = json_file.read()
```

```
digits_cnn_model = model_from_json(loaded_model_json)


digits_cnn_model.load_weights("models/digits_cnn_weights.h5")
    #digits_cnn_model._make_predict_function()
    print(digits_cnn_model.summary())
text.insert(END,'Digits based Deep Learning CNN Model generated\n')


def sentimentModel():
    global text_sentiment_model
    global image_sentiment_model
    global face_detection
text_sentiment_model = joblib.load('models/sentimentModel.pkl')
text.insert(END,'Text based sentiment Deep Learning CNN Model generated\n')


face_detection = cv2.CascadeClassifier('models/haarcascade_frontalface_default.xml')
image_sentiment_model = load_model('models/_mini_XCEPTION.106-0.65.hdf5',
compile=False)
text.insert(END,'Image based sentiment Deep Learning CNN Model generated\n')
    print(image_sentiment_model.summary())


def digitRecognize():
    global filename
    filename = filedialog.askopenfilename(initialdir="testImages")
pathlabel.configure(text=filename)
text.delete('1.0', END)
text.insert(END,filename+" loaded\n");
```

```
imagetest = image.load_img(filename, target_size = (28,28), grayscale=True)
imagetest = image.img_to_array(imagetest)
imagetest = np.expand_dims(imagetest, axis = 0)
pred = digits_cnn_model.predict(imagetest.reshape(1, 28, 28, 1))
    predicted = str(pred.argmax())
imagedisplay = cv2.imread(filename)
orig = imagedisplay.copy()
    output = imutils.resize(orig, width=400)
    cv2.putText(output, "Digits Predicted As : "+predicted, (10, 25), cv2.FONT_HERSHEY_SIMPLEX,0.7, (0, 255, 0), 2)
    cv2.imshow("Predicted Image Result", output)
    cv2.waitKey(0)



def opinion():
    user = simpledialog.askstring("Please enter your name", "Username")
    opinion = simpledialog.askstring("Government Service Opinion", "Please write your Opinion about government services & policies")
    f = open("Peoples_Opinion/opinion.txt", "a+")
f.write(user+"#"+opinion+"\n")
f.close()
messagebox.showinfo("Thank you for your opinion", "Your opinion saved for reviews")



def stem(textmsg):
    stemmer = nltk.stem.PorterStemmer()
textmsg_stem = ''
textmsg = textmsg.strip("\n")
    words = textmsg.split(" ")
    words = [stemmer.stem(w) for w in words]
textmsg_stem = ' '.join(words)
    return textmsg_stem


def viewSentiment():
text.delete('1.0', END)
    with open("Peoples_Opinion/opinion.txt", "r") as file:
        for line in file:
            line = line.strip('\n')
            line = line.strip()
arr = line.split("#")
text_processed = stem(arr[1])
            X =  [text_processed]
            sentiment = text_sentiment_model.predict(X)
            predicts = 'None'
            if sentiment[0] == 0:
                predicts = "Negative"
            if sentiment[0] == 1:
                predicts = "Positive"
text.insert(END,"Username : "+arr[0]+"\n");
text.insert(END,"Opinion  : "+arr[1]+" : Sentiment Detected As : "+predicts+"\n\n")
```

```
def uploadPhoto():
    filename = filedialog.askopenfilename(initialdir="expression_images_to_upload")
    user = simpledialog.askstring("Please enter your name", "Username")
    policy = simpledialog.askstring("Please enter Government Policy name related to Facial Expression", "Please enter Government Policy name related to Facial Expression")
img = cv2.imread(filename)
    cv2.imwrite("sentimentImages/"+user+"-"+policy+".jpg",img);
messagebox.showinfo("Your facial expression image accepted for reviews", "Your facial expression image accepted for reviews")


def photoSentiment():
    filename = 'sentimentImages'
    for root, dirs, files in os.walk(filename):
      for fdata in files:
          frame = cv2.imread(root+"/"+fdata)
          faces = face_detection.detectMultiScale(frame,scaleFactor=1.1,minNeighbors=5,minSize=(30,30),flags=cv2.CASCADE_SCALE_IMAGE)
          msg = ''
          if len(faces) > 0:
              faces = sorted(faces, reverse=True,key=lambda x: (x[2] - x[0]) * (x[3] - x[1]))[0]
              (x, y, w, h) = faces
              cv2.rectangle(frame, (x,y), (x+w,y+h), (0,0,255), 2)
              temp = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
roi = temp[y:y + h, x:x + w]
roi = cv2.resize(roi, (48, 48))
roi = roi.astype("float") / 255.0
roi = img_to_array(roi)
roi = np.expand_dims(roi, axis=0)
preds = image_sentiment_model.predict(roi)[0]
emotion_probability = np.max(preds)
        label = EMOTIONS[preds.argmax()]
        msg = "Sentiment detected as : "+label
img_height, img_width = frame.shape[:2]
        cv2.putText(frame, msg, (50,40), cv2.FONT_HERSHEY_SIMPLEX, 0.5,(0,0,255), 2)
        cv2.imshow(fdata,frame)
messagebox.showinfo(fdata, "Sentiment predicted from Facial expression as : "+label)
        if cv2.waitKey(10) & 0xFF == ord('q'):
            break
   cv2.waitKey(0)
   cv2.destroyAllWindows()



font = ('times', 16, 'bold')
title = Label(main, text='Automating E-Government Services With Artificial Intelligence',anchor=W, justify=CENTER)
title.configure(bg='yellow4', fg='white')
title.configure(font=font)
title.configure(height=3, width=120)
title.place(x=0,y=5)
```

```
font1 = ('times', 14, 'bold')
digitButton = Button(main, text="Generate Hand Written Digits Recognition Deep Learning Model", command=digitModel)
digitButton.place(x=50,y=100)
digitButton.configure(font=font1)

pathlabel = Label(main)
pathlabel.configure(bg='yellow4', fg='white')
pathlabel.configure(font=font1)
pathlabel.place(x=50,y=150)

sentimentButton = Button(main, text="Generate Text & Image Based Sentiment Detection Deep Learning Model", command=sentimentModel)
sentimentButton.place(x=50,y=200)
sentimentButton.configure(font=font1)

recognizeButton = Button(main, text="Upload Test Image & Recognize Digit", command=digitRecognize)
recognizeButton.place(x=50,y=250)
recognizeButton.configure(font=font1)

opinionButton = Button(main, text="Write Your Opinion About Government Policies", command=opinion)
opinionButton.place(x=50,y=300)
opinionButton.configure(font=font1)

viewButton = Button(main, text="View Peoples Sentiments From Opinions",
command=viewSentiment)
viewButton.place(x=50,y=350)
viewButton.configure(font=font1)

photoButton = Button(main, text="Upload Your Face Expression Photo About Government Policies", command=uploadPhoto)
photoButton.place(x=50,y=400)
photoButton.configure(font=font1)

photosentimentButton = Button(main, text="Detect Sentiments From Face Expression Photo", command=photoSentiment)
photosentimentButton.place(x=50,y=450)
photosentimentButton.configure(font=font1)


font1 = ('times', 12, 'bold')
text=Text(main,height=15,width=80)
scroll=Scrollbar(text)
text.configureure(yscrollcommand=scroll.set)
text.place(x=600,y=100)
text.configure(font=font1)


main.configure(bg='magenta3')
main.mainloop()
```

7.PROJECT 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.

7.1 VARIOUS TEST CASES

Test Case 1:

We are entering the feedback on the government

Test Case 2:

We are uploading hand written digits and predicting the values

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.

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.

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.

System Test

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

7.2 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.

7.3 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.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

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.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

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

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.

8. OUTPUTSCREENS

To run this project double click on ‘run.bat’ file to get below screen

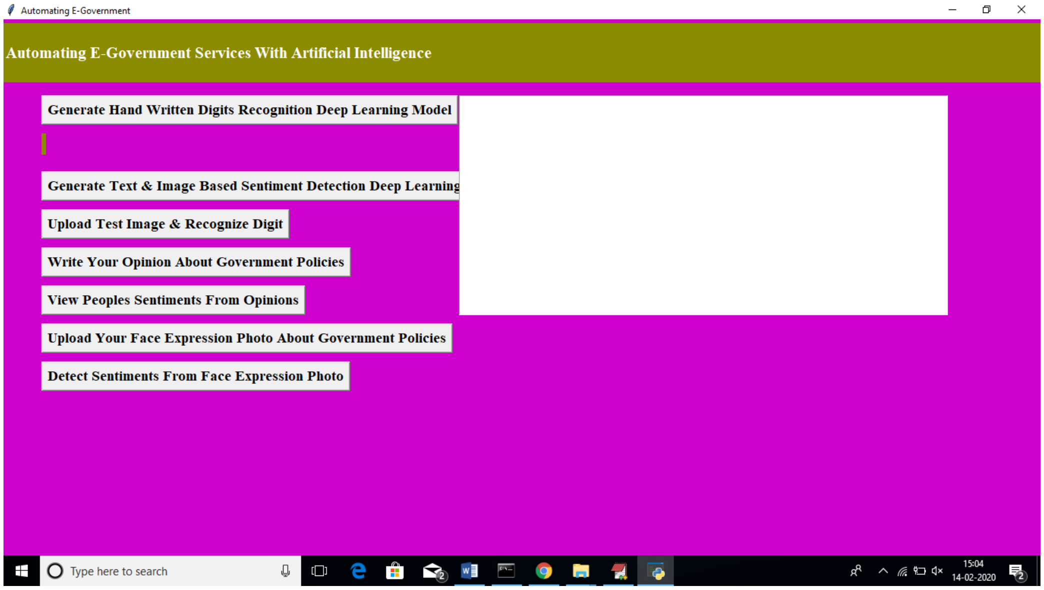
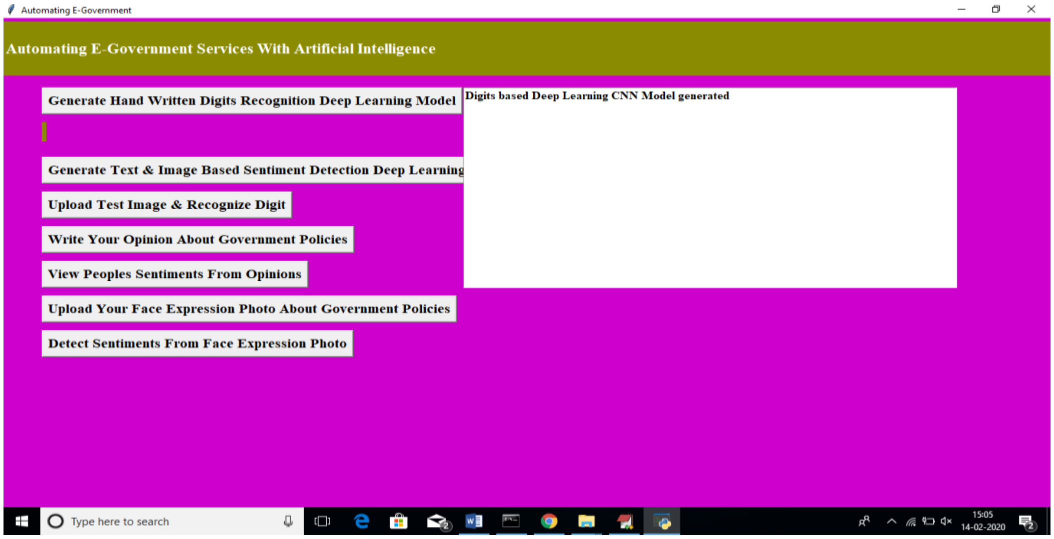

Automating E-Government
Automating E-Government Services With Artificial Intelligence
Generate Hand Written Digits Recognition Deep Learning Model
Generate Text & Image Based Sentiment Detection Deep Learning
Upload Test Image & Recognize Digit
Write Your Opinion About Government Policies
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About Government Policies
Detect Sentiments From Face Expression Photo
Type here to search
15:04
14-02-2020

Figure 8.1 ACTUALOUTPUT

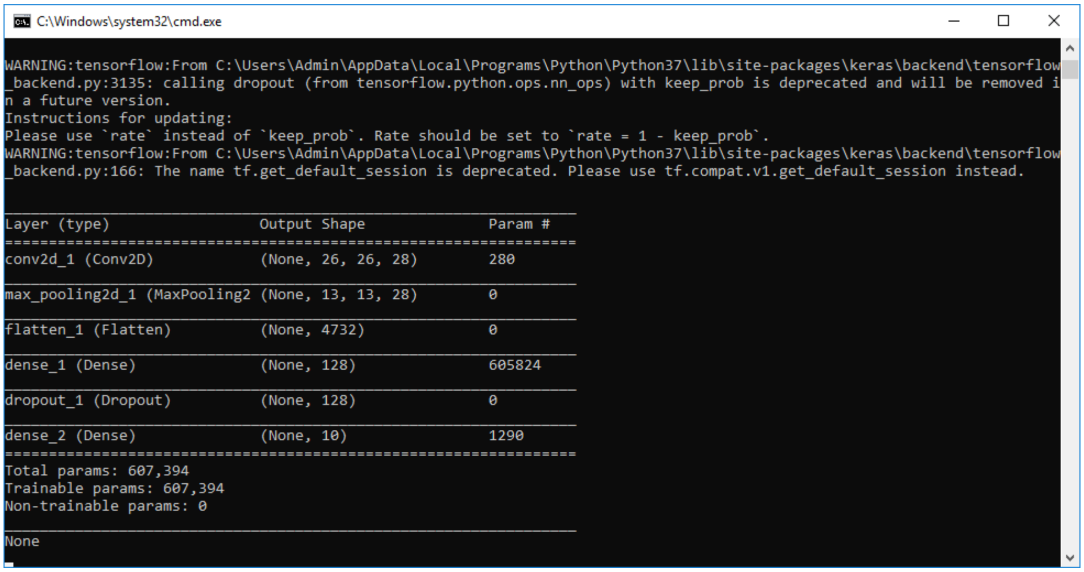
In above screen click on ‘Generate Hand Written Digits Recognition Deep Learning Model’ button to generate CNN digits recognition model


Automating E-Government
Automating E-Government Services With Artificial Intelligence
Generate Hand Written Digits Recognition Deep Learning Model
Generate Text & Image Based Sentiment Detection Deep Learning
Upload Test Image & Recognize Digit
Write Your Opinion About Government Policies
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About Government Policies
Detect Sentiments From Face Expression Photo
Type here to search
15:05
14-02-2020

Digits based Deep Learning CNN Model generated

Figure 8.2 CNN ALGORITHM GENERATED FOR HAND WRITTEN DIGITS RECOGNITION

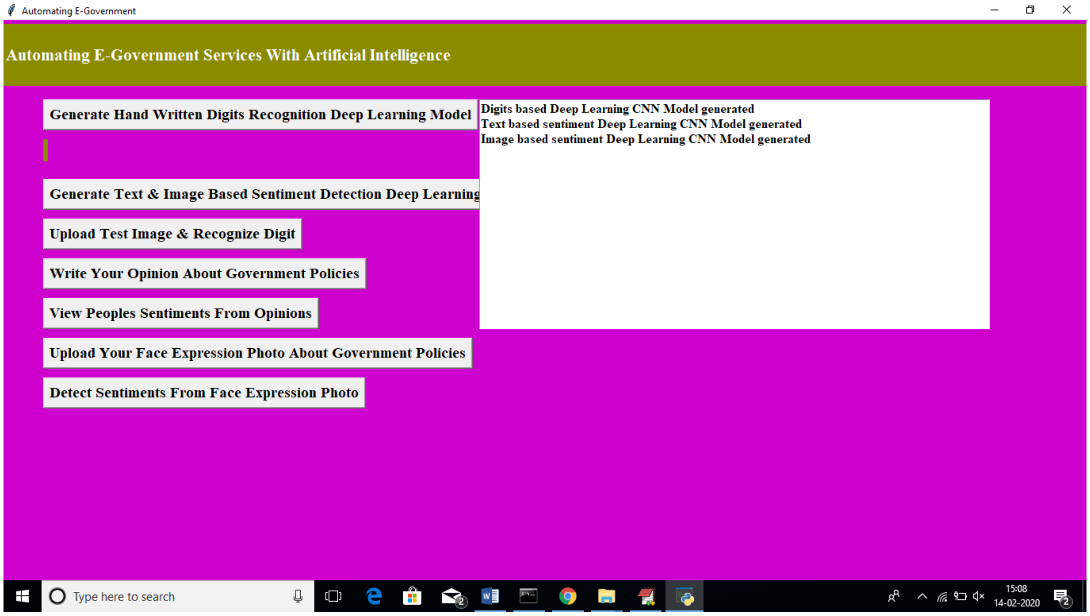
In above screen we can see digits model generated and CNN layer details you can see ablack console



```
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow
_backend.py:3135: calling dropout (from tensorflow.python.ops.nn_ops) with keep_prob is deprecated and will be removed i
n a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow
_backend.py:166: The name tf.get_default_session is deprecated. Please use tf.compat.v1.get_default_session instead.

_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 26, 26, 28)        280
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 13, 13, 28)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 4732)              0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               605824
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 10)                1290
=================================================================
Total params: 607,394
Trainable params: 607,394
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 8.3 COMMAND PROMPT FOR CNN ALGORITHM GENERATED FOR HANDWRITTEN DIGITS RECOGNITION

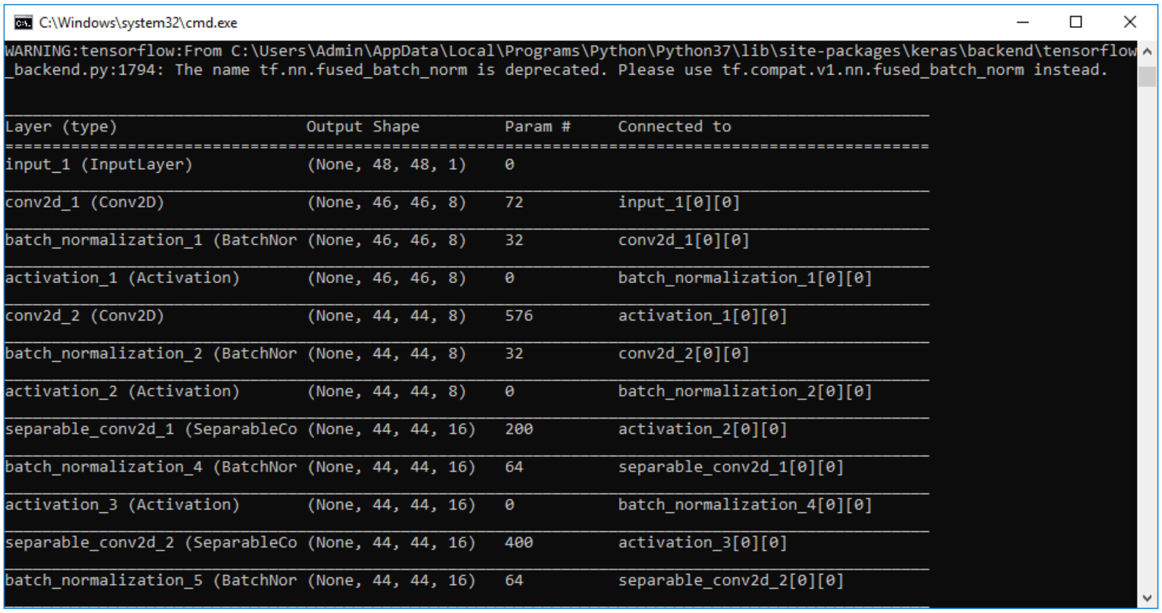
In above screen we can see Conv2d means convolution or CNN generate image features layer from different size as first layer generate with image size 26, 26 and second generated with 13 and 13 and goes on. Now click on ‘Generate Text & Image Based Sentiment Detection Deep Learning Model’ button to generate CNN for text andimage based sentiment detection model.


Automating E-Government
Automating E-Government Services With Artificial Intelligence
Generate Hand Written Digits Recognition Deep Learning Model
Generate Text & Image Based Sentiment Detection Deep Learning
Upload Test Image & Recognize Digit
Write Your Opinion About Government Policies
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About Government Policies
Detect Sentiments From Face Expression Photo
Type here to search
15:08
14-02-2020

Digits based Deep Learning CNN Model generated
Text based sentiment Deep Learning CNN Model generated
Image based sentiment Deep Learning CNN Model generated

Figure 8.4 CNN ALGORITHM GENERATED FOR IMAGERECOGNITION

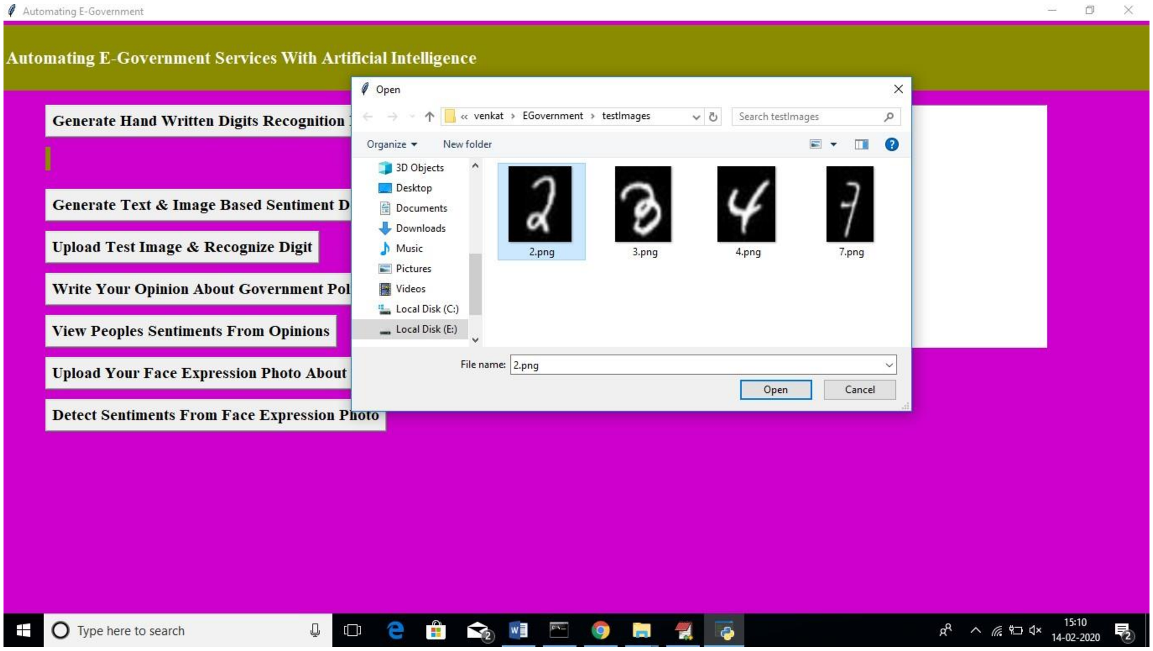
In above screen we can see text and image based CNN model generated. See black screen for more details



```
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow
_backend.py:1794: The name tf.nn.fused_batch_norm is deprecated. Please use tf.compat.v1.nn.fused_batch_norm instead.

__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            (None, 48, 48, 1)    0
__________________________________________________________________________________________________
conv2d_1 (Conv2D)               (None, 46, 46, 8)    72          input_1[0][0]
__________________________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 46, 46, 8)    32          conv2d_1[0][0]
__________________________________________________________________________________________________
activation_1 (Activation)       (None, 46, 46, 8)    0           batch_normalization_1[0][0]
__________________________________________________________________________________________________
conv2d_2 (Conv2D)               (None, 44, 44, 8)    576         activation_1[0][0]
__________________________________________________________________________________________________
batch_normalization_2 (BatchNor (None, 44, 44, 8)    32          conv2d_2[0][0]
__________________________________________________________________________________________________
activation_2 (Activation)       (None, 44, 44, 8)    0           batch_normalization_2[0][0]
__________________________________________________________________________________________________
separable_conv2d_1 (SeparableCo (None, 44, 44, 16)   200         activation_2[0][0]
__________________________________________________________________________________________________
batch_normalization_4 (BatchNor (None, 44, 44, 16)   64          separable_conv2d_1[0][0]
__________________________________________________________________________________________________
activation_3 (Activation)       (None, 44, 44, 16)   0           batch_normalization_4[0][0]
__________________________________________________________________________________________________
separable_conv2d_2 (SeparableCo (None, 44, 44, 16)   400         activation_3[0][0]
__________________________________________________________________________________________________
batch_normalization_5 (BatchNor (None, 44, 44, 16)   64          separable_conv2d_2[0][0]
__________________________________________________________________________________________________
```

Figure 8.5 COMMAND PROMPT CNN ALGORITHM GENERATED FOR IMAGE RECOGNITION

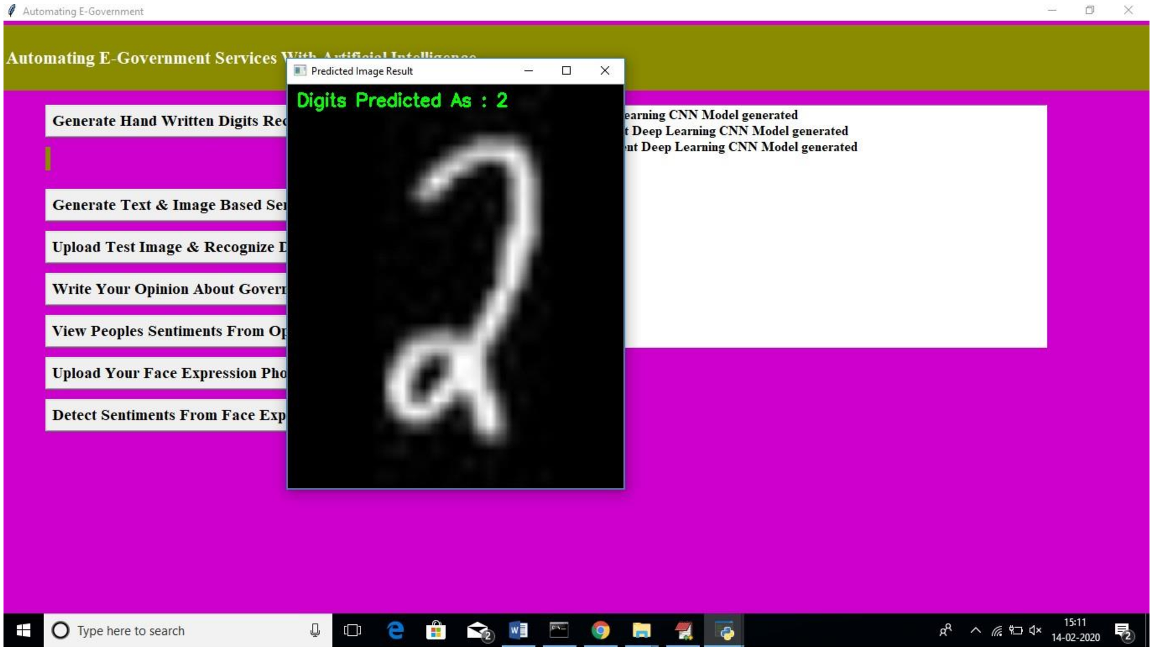
Now click on ‘Upload Test Image & Recognize Digit’ button to upload digit images and to get name of that digit. All digit images saved inside test Images folder


Automating E-Government
Automating E-Government Services With Artificial Intelligence
Generate Hand Written Digits Recognition
Generate Text & Image Based Sentiment D
Upload Test Image & Recognize Digit
Write Your Opinion About Government Pol
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About
Detect Sentiments From Face Expression Photo
Open
« venkat › EGovernment › testImages
Search testImages
Organize ▾
New folder
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
2.png
3.png
4.png
7.png
Open
Cancel
Type here to search
15:10
14-02-2020

File name: 2.png

Figure 8.6 UPLOADING HAND WITTEN DIGITS

In above screen I am uploading image which contain digit 2 and below is the output of detection

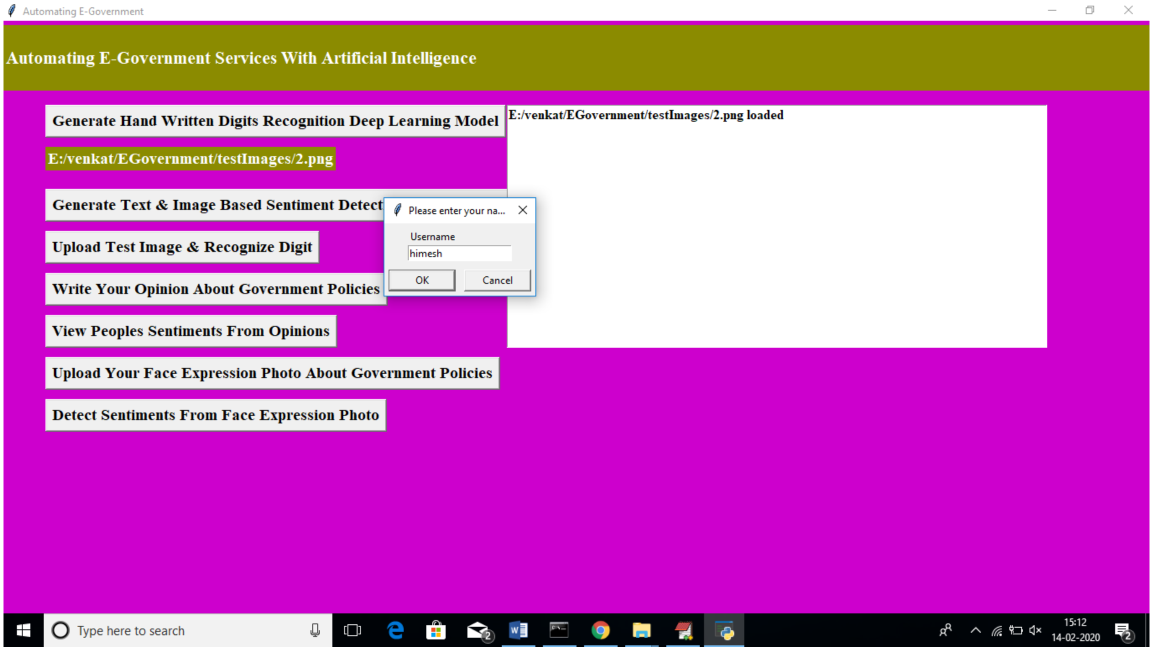

Automating E-Government
Automating E-Government Services With Artificial Intelligence
Predicted Image Result
Generate Hand Written Digits Rec
Generate Text & Image Based Se
Upload Test Image & Recognize D
Write Your Opinion About Gover
View Peoples Sentiments From Op
Upload Your Face Expression Pho
Detect Sentiments From Face Exp
Type here to search
15:11
14-02-2020

Digits Predicted As : 2

earning CNN Model generated
t Deep Learning CNN Model generated
nt Deep Learning CNN Model generated

Figure 8.7 PREDICTS THE VALUE OF HAND WRITTEN

In above screen we can see Digits Predicted as: 2. Now click on 'Write Your Opinion About Government Policies' button to write some comments on government policy


Automating E-Government

Automating E-Government Services With Artificial Intelligence

E:/venkat/EGovernment/testImages/2.png

E:/venkat/EGovernment/testImages/2.png loaded

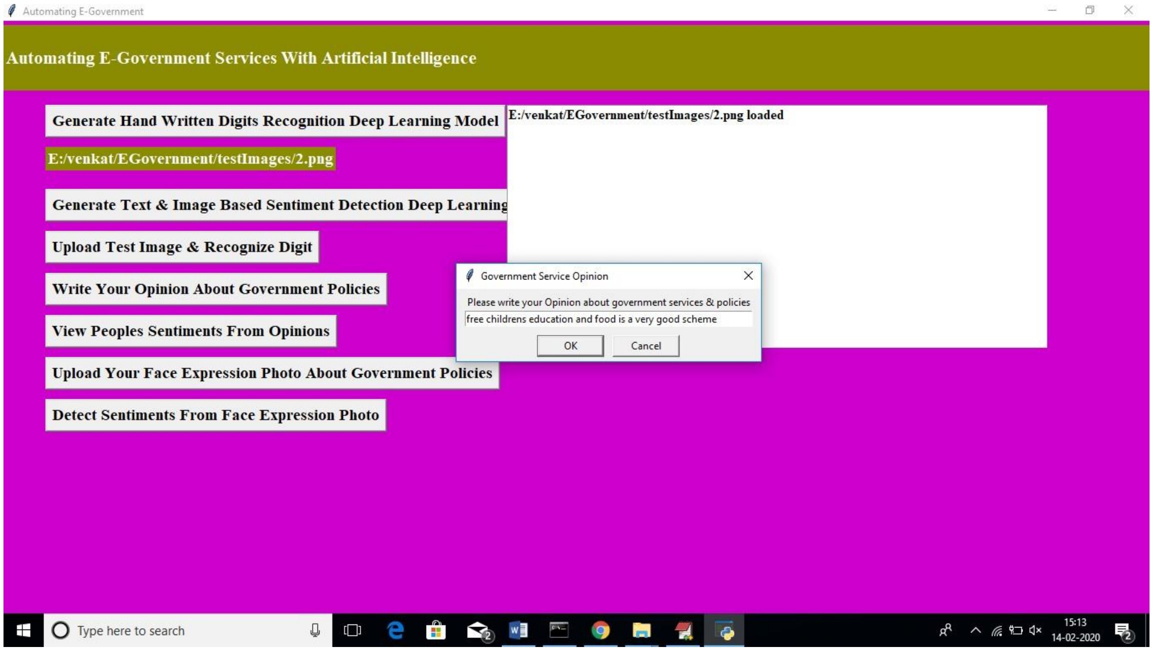
Please enter your na...

Username: himesh

Generate Hand Written Digits Recognition Deep Learning Model
Generate Text & Image Based Sentiment Detect
Upload Test Image & Recognize Digit
Write Your Opinion About Government Policies
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About Government Policies
Detect Sentiments From Face Expression Photo
OK
Cancel
Type here to search
15:12
14-02-2020

Figure 8.8 UPLOADING FEEDBACK ON THE GOVERNMENT

In above screen before writing opinions we need to write username after writing username click ok button to get below screen


Automating E-Government

Automating E-Government Services With Artificial Intelligence

E:/venkat/EGovernment/testImages/2.png

E:/venkat/EGovernment/testImages/2.png loaded

Government Service Opinion

Please write your Opinion about government services & policies

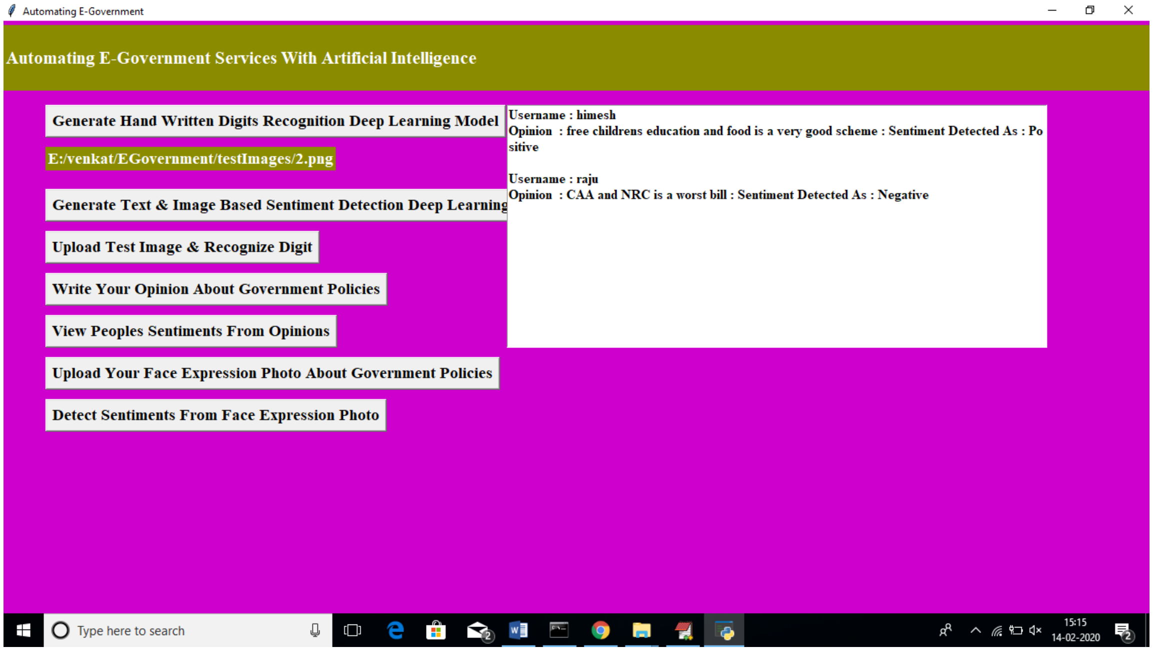
free childrens education and food is a very good scheme

Generate Hand Written Digits Recognition Deep Learning Model
Generate Text & Image Based Sentiment Detection Deep Learning
Upload Test Image & Recognize Digit
Write Your Opinion About Government Policies
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About Government Policies
Detect Sentiments From Face Expression Photo
OK
Cancel
Type here to search
15:13
14-02-2020

Figure 8.9 UPLOADING FEEDBACK ON THE GOVERNMENT-2

In above screen I wrote some comment on some scheme and application detect sentiment from it as positive or negative. Now click on ‘View Peoples Sentiments From

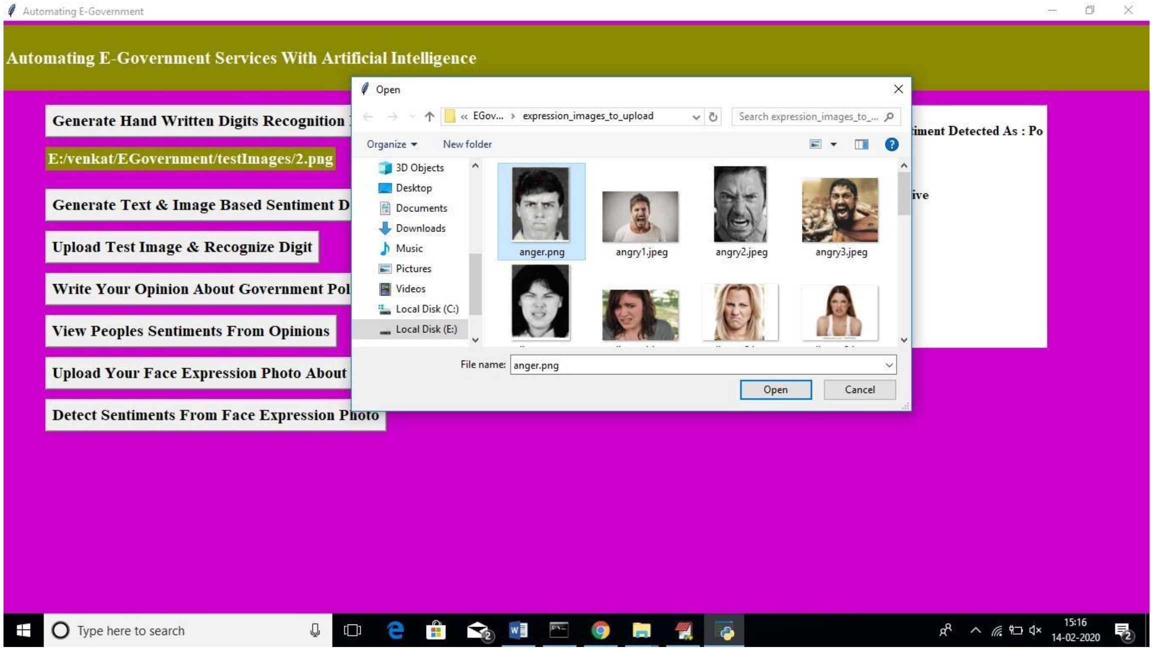
Opinions’ button to view all opinions from past users.


Automating E-Government
Automating E-Government Services With Artificial Intelligence
Generate Hand Written Digits Recognition Deep Learning Model
E:/venkat/EGovernment/testImages/2.png
Generate Text & Image Based Sentiment Detection Deep Learning
Upload Test Image & Recognize Digit
Write Your Opinion About Government Policies
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About Government Policies
Detect Sentiments From Face Expression Photo
Type here to search
15:15
14-02-2020

```
Username : himesh
Opinion  : free childrens education and food is a very good scheme : Sentiment Detected As : Po
sitive

Username : raju
Opinion  : CAA and NRC is a worst bill : Sentiment Detected As : Negative
```

Figure 8.10 VIEWING THE FEEDBACK ON THE GOVERNMENT

In above screen text area we can see opinions from all users and in first opinion we got sentiment detected as positive which means user is satisfy with that scheme and for second opinion we got sentiment as negative which means user not happy. Similarly user can upload their image with facial expression which describe whether user is happy or angry


Automating E-Government
Automating E-Government Services With Artificial Intelligence
Generate Hand Written Digits Recognition
Generate Text & Image Based Sentiment D
Upload Test Image & Recognize Digit
Write Your Opinion About Government Pol
View Peoples Sentiments From Opinions
Upload Your Face Expression Photo About
Detect Sentiments From Face Expression Photo
Open
« EGov... › expression_images_to_upload
Search expression_images_to_...
Organize
New folder
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Local Disk (E:)
anger.png
angry1.jpeg
angry2.jpeg
angry3.jpeg
Open
Cancel
Type here to search
15:16
14-02-2020

E:/venkat/EGovernment/testImages/2.png

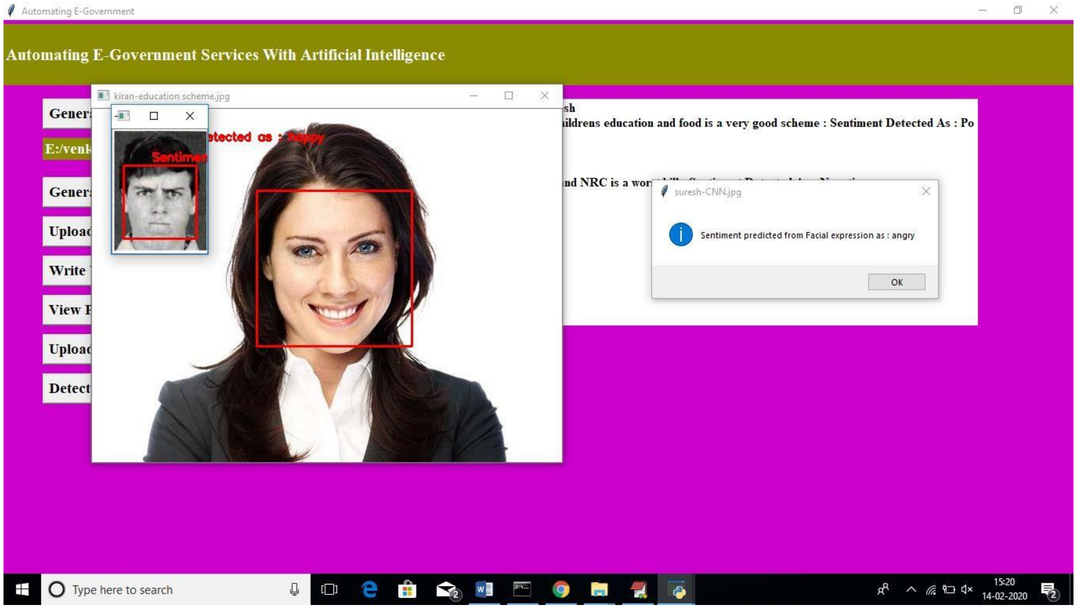
iment Detected As : Po

ive

File name: anger.png

Figure 8.11 UPLOADING IMAGE FOR FEEBACK ON THE GOVERNMENT

In above screen I am uploading one anger face image and then application ask to write username and referring scheme name. similarly any number of users can upload their images. Now click on ‘Detect Sentiments From Face Expression Photo’ button to get all images and its detected sentiments



E:/venk

Sentimen

etected as : happy

sh
ildrens education and food is a very good scheme : Sentiment Detected As : Po

nd NRC is a wor

Sentiment predicted from Facial expression as : angry

Figure 8.12 VIEW THE FACIAL EXPRESSION RESULT

In above screen we can see all images with facial expression are identified with their sentiments. In dialog box also we can see sentiment result.

Similarly you can enter any number of comments or facial images to detect their sentiments.

9.CONCLUSION AND FUTUREENHANCEMENT

With the recent advances in AI and deep learning technologies, more government agencies are starting to use such technologies to improve the ir systems and services However, a large set of challenges hinder the adoption of such technologies, including the lack of experts, computational resources, trust, and AI interpretability.

In this paper, we introduced the definitions of artificial intelligence and e- government, briefly discussed the current state of e-government indices around the world, and then proposed our solutions to advance the current state of e-government, considering as a case-study. We proposed a framework for management of government information resources that help manage the e-government lifecycle end-to-end. Then, we proposed a set of deep learning techniques that can help facilitate and automate several e-government services. After that, we proposed a smart platform for AI development and implementation in e-government. The overarching goal of this paper is to introduce new frameworks and platform to integrate recent advances in AI techniques in the e-government systems and services to improve the overall trust, transparency, and efficiency ofe-government.

10.REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, ‘‘Deep residual learning for image recognition,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp.770–778.

[2] Y.-D. Zhang, Y. Zhang, X.-X. Hou, H. Chen, and S.-H. Wang, ‘‘Sevenlayer deep neural network based on sparse autoencoder for voxelwise detection of cerebral microbleed,’’ Multimedia Tools Appl., vol. 77, no. 9, pp. 10521–10538, May2018.

[3] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, ‘‘Translating videos to natural language using deep recurrent neural networks,’’ 2014, arXiv:1412.4729. [Online]. Available: https://arxiv.org/abs/1412.4729

[4] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche,J.Schrittwieser,I.Antonoglou,V.Panneershelvam,M.Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K.Kavukcuoglu,

T. Graepel, and D. Hassabis, ‘‘Mastering the game of Go with deep neural networks and tree search,’’ Nature, vol. 529, no. 7587, pp. 484–489, 2016.

[5] C. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer,2006.

[6] Y. LeCun, Y. Bengio, and G. Hinton, ‘‘Deep learning,’’ Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[7] G.D.Abowd,A.K.Dey,P.J.Brown,N.Davies,M.Smith,andP.Steggles,

‘‘Towards a better understanding of context and context-awareness,’’ in Proc.Int.Symp.HandheldUbiquitousComput.Berlin,Germany:Springer, 1999, pp. 304– 307.

[8] C. Dwork, ‘‘Differential privacy,’’ in Encyclopedia of Cryptography and Security,H.

C. A. van Tilborg and S. Jajodia, Eds. Boston, MA, USA: Springer, 2011.

[9] L. Bottou, ‘‘Large-scale machine learning with stochastic gradient descent,’’ in Proc. COMPSTAT, 2010, pp. 177–186.

[10]A. Kankanhalli, Y. Charalabidis, and S. Mellouli, ‘‘IoT and AI for smart government: A research agenda,’’ Government Inf. Quart., vol. 36, no. 2, pp. 304–309, 2019

[11] J. B. Lee and G. A. Porumbescu, ‘‘Engendering inclusive e-government usethroughcitizenITtrainingprograms,’’GovernmentInf.Quart.,vol.36, no. 1, pp. 69–76, 2019.

[12] R. Santa, J. B. MacDonald, and M. Ferrer, ‘‘The role of trust in e-Government effectiveness, operational effectiveness and user satisfaction: Lessons from Saudi Arabia in e-G2B,’’ Government Inf. Quart., vol. 36, no. 1, pp. 39–50,2019.

[13] J. D. Twizeyimana and A. Andersson, ‘‘The public value of E-Government–A literature review,’’ Government Inf. Quart., vol. 36, no. 2, pp. 167–178,2019.

A

Project report

On

PRIVACY PRESERVING MEDICAL TREATMENT THROUGH NON-DETERMINISTIC FINITE AUTOMATA

Submitted by

Ms. A. KEERTHANA	(17K81A1202)
Ms. K. AKHILA	(17K81A1220)
Ms.D.V.S. MOUNI SAI	(17K81A1258)
Ms. K. THEJESWI NAIDU	(17K81A1259)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

Mr. R. Prashanth Kumar

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST. MARTIN’S ENGINEERING COLLEGE

(An Autonomous Institute)

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **PRIVACY PRESERVING MEDICAL TREATMENT THROUGH NON-DETERMINISTIC FINITE AUTOMATA**, is being submitted by **A. KEERTHANA (17K81A1202), K. AKHILA (17K81A1220), D.V.S. MOUNI SAI (17K81A1258), K. THEJESWI NAIDU (17K81A1259)** in partial fulfilment of the requirement for the award of the degree of **BACHELOR OFTECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Project Guide	Head of the Department
R. PRASHANTH KUMAR	Dr. R. NAGARAJU
Department of Information Technology	Department of Information Technology

Internal Examiner	External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department **of Information Technology**, session: 2017 – 2021, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **Privacy Preserving Medical Treatment Through Non-Deterministic Finite Automata** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

Ms. A. KEERTHANA	(17K81A1202)
Ms. K. AKHILA	(17K81A1220)
Ms.D.V.S. MOUNI SAI	(17K81A1258)
Ms. K. THEJESWI NAIDU	(17K81A1259)

LASYA
INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
SMEC
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **A.KEERTHANA** WITH ROLL NO.**17K81A1202, D.V.S MOUNI SAI** WITH ROLL NO.**17K81A1258**, **K.AKHILA** WITH ROLL NO.**17K81A1220**, **K.THEJESWI NAIDU** WITH ROLL NO.**17K81A1259**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT OF ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**PRIVACY-PRESERVING MEDICAL TREATMENT SYSTEM THROUGH NONDETERMINISTIC FINITE AUTOMATA**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd. ★ Kompally-500014. ★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

ACKNOWLEDGEMEN

The satisfaction and euphoria that accompanies the successful completion of any task would b incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr.P. SANTOSH KUMARPATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr**. **R. NAGARAJU**, Head of the Department, **Information Technology** St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **Mr. R. Prashanth Kumar**, Assistant Professor, Department of **Information Technology,** St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

Ms. A. KEERTHANA	**(17K81A1202)**
Ms. K. AKHILA	**(17K81A1220)**
Ms.D.V.S. MOUNI SAI	**(17K81A1258)**
Ms. K. THEJESWI NAIDU	**(17K81A1259)**

TABLEOFCONTENTS

ABSTRACT

With the popularity of wearable devices, along with the development of clouds and cloudlet technology, there has been increasing need to provide better medical care. The processing chain of medical data mainly includes data collection, data storage and data sharing, etc. Traditional healthcare system often requires the delivery of medical data to the cloud, which involves users' sensitive information and causes communication energy consumption. Practically, medical data sharing is a critical and challenging issue. Thus in this paper, we build up a novel healthcare system by utilizing the flexibility of cloudlet. The functions of cloudlet include privacy protection, data sharing and intrusion detection. In the stage of data collection, we first utilize Number Theory Research Unit (NTRU) method to encrypt user's body data collected by wearable devices. Those data will be transmitted to nearby cloudlet in an energy efficient fashion. Secondly, we present a new trust model to help users to select trustable partners who want to share stored data in the cloudlet. The trust model also helps similar patients to communicate with each other about their diseases. Thirdly, we divide users' medical data stored in remote cloud of hospital into three parts, and give them proper protection. Finally, in order to protect the healthcare system from malicious attacks, we develop a novel collaborative intrusion detection system (IDS) method based on cloudlet mesh, which can effectively prevent the remote healthcare big data cloud from attacks. Our experiments demonstrate the effectiveness of the proposed scheme.

LIST OF TABLES

LIST OF FIGURES

CHAPTER-1: INTRODUCTION

The maturing of populace and pervasiveness of ongoing sick nesses have exacerbated numerous social issues. Far off conclusion and treatment frameworks, which utilize information innovation to give available, savvy, and top notch clinical medical care benefits distantly, can be conveyed to lighten a portion of the issues. Such a framework makes it feasible for proceeded with therapy in a home climate and builds patient adherence to clinical proposal. The clinical Internet of Things (mIoT) assumes a basic part in far off clinical conclusion and treatment by sending remote wearable (or implantable) sensors on a patient to gather the fundamental signs and physiological information. The observed physiological boundaries are shipped off emergency clinic for clinical conclusion, which supplies rich longitudinal wellbeing records than the short disease portrayal. Utilizing the definite observing information, doctors can improve a much guess for the patient and suggest treatment, early mediation what's more, drug changes that are compelling for sickness recuperation. The vital factor for the precision of far off clinical finding also, treatment is the doctor's skill and expert experience. A clinical model is planned as per level headed and quantifiable perception to give clinically helpful data about the course of the ailment over the long haul also, direct explicit medicines for the condition, which plays a critical part in directing the treatment cycle and giving expense rate medical care administrations. Limited automata (FA) is one of the standard technologies that can be utilized to address clinical models. Thought about with the stream graph or square outline based model, a FA-based clinical model has the upside of regularized portrayal, adaptability in ailment state assessment and great expansibility. FA can be sorted into two kinds: deterministic limited automata (DFA) and nondeterministic limited automata (NFA). The expression "deterministic" in DFA implies that it can just travel to each state in turn (for example for some given input); "nondeterministic" in NFA implies it can travel to numerous states without a moment's delay.

1.1 PROJECT OVERVIEW

We are proposing a cloud based health care guidance platform named as “Privacy Preserving Medical Treatment Through Non-Deterministic Finite Automata(NFA).

Our project main endeavour is to reduce the time complexity, giving a secure and better treatment for users by this platform.

The main technological innovation designed in this project is by using Secure protocols such as:

1. Secure Illness State Match Protocol(SSM)

2. Secure Treatment Procedure Traverse Algorithm(TPT)

3. Secure Treatment Procedure Weight Calculation Protocol(TPW)

4. Secure Top-K Best Treatment Procedure Selection(BPS-K)

5. Privacy Preserving Error Resistant Gene Match Protocol(P-GENE)

By this users can access the platform from anywhere so that he/she(user’s)will be benefited with secure and best treatment procedure.In the day to day life everything is been chained with technology. Every sector is been bounded with the latest Technologies and among those Cloud computing technology is currently taking up every where

The main feature will be securing the data and detecting intruders it is done by means of secure protocolsSecure Illness State Match Protocol(SSM)Secure Treatment Procedure Traverse Algorithm(TPT)Secure Treatment Procedure Weight Calculation Protocol(TPW)Secure Top-K Best Treatment Procedure Selection(BPS-K)Privacy Preserving Error Resistant Gene Match Protocol(P-GENE) which leads to secure and faster result

1.2 PROJECT 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

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.

1.3 SCOPE OF THE PROJECT

- The aging of population and prevalence of chronic illnesses have exacerbated many social problems.
- In the day to day life everything is pivot upon technology. Considering that scenario our project is developed.
- We bring forward a system which is designed to provide a secure and best procedural

treatment of illness state to the user which will act as a good user interface.

- The main aim with the system is it acts as a viaduct between patient and doctor and to safeguard the data from intruders.

1.4 ORGANIZATION OF CHAPTERS

1.4.1 INTRODUCTION

The aging of population and prevalence of chronic illnesses have exacerbated many social problems. Remote diagnosis and treatment systems, which make use of information technology to provide accessible, cost-effective, and highquality clinical healthcare services remotely, can be deployed to alleviate some of the problems. Such a system makes it possible for continued treatment in a home environment and increases patient adherence to medical recommendations. The medical Internet of Things (mIoT) plays a critical role in distant medical diagnosis and treatment by deploying wireless wearable (or implantable) sensors on a patient to collect the vital signs

1. We propose a Privacy Preserving Medical Treatment System using Non-Deterministic Finite Automata(NFA),here after referred as P-med, designed for remote medical environment.

2. P-med makes use of NFA to flexibly represent medical model which represents illness states, treatment methods and state transitions caused by exerting different methods.

Moreover, a new privacy preserving NFA evaluation method is given in P-Med to get a confidential match result.

3To build the Privacy Preserving Medical Treatment Through NonDeterministic Finite Automata we need to connect to cloud which represents as a bridge between patient and doctor

.4. We use cloud not only to represent as a bridge but also as a defender to track the intruder and also safeguard the data

5. We are describing concept of implementing Cloud based health care platform by including modules of patient , doctor , cloudlets and intruder

1.4.2 LITERATURE SURVEY

”Wearable medical device for telephone healthcare”(2004)

The world's ageing population and prevalence of diseases have lead to high demand for tele-home healthcare, in which vital-signs monitoring is essential. An overview of state-of-art wearable technologies for remote patient-monitoring is presented, followed by case studies on a cuffless blood pressure meter, ring-type heart rate monitor, and Bluetooth/spl trade/-based ECG monitor. Aim of our project is to develop a tele-home healthcare system which utilizes wearable devices, wireless communication technologies, and multisensor data fusion methods.

“Cloud-supported monitoring”(2015)

The potential of cloud-supported cyber-physical systems (CCPSs) has drawn a great deal of interest from academia and industry. CCPSs facilitate the seamless integration of devices in the physical world with cyberspace. This enables a range of emerging applications or systems such as patient or health monitoring, which require patient locations to be tracked.

“Security models and requirements for healthcare application clouds”

In this paper we discuss important concepts related to EHR sharing and integration in healthcare clouds and analyzes the arising security and privacy issues in access and management of EHRs. We describe an EHR security reference model for managing security issues in healthcare clouds, which highlights three important core components in securing an EHR cloud.

1.4.3 SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements

For developing the application the following are the Software Requirements:

- Operating system : - Windows XP/7.
- Coding Language : JAVA/J2EE
- Data Base : MYSQL

Hardware Requirements

For developing the application the following are the Hardware Requirements:

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40 GB.
- Floppy Drive : 1.44 Mb.
- Monitor : 15 VGA Colour.
- Mouse : Logitech.
- Ram : 512 Mb.

1.4.4 SOFTWARE DEVELOPMENT ANALYASIS

Java Technology

Java technology is both a programming language and a platform.

The Java Programming Language

The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

Cloud computing:

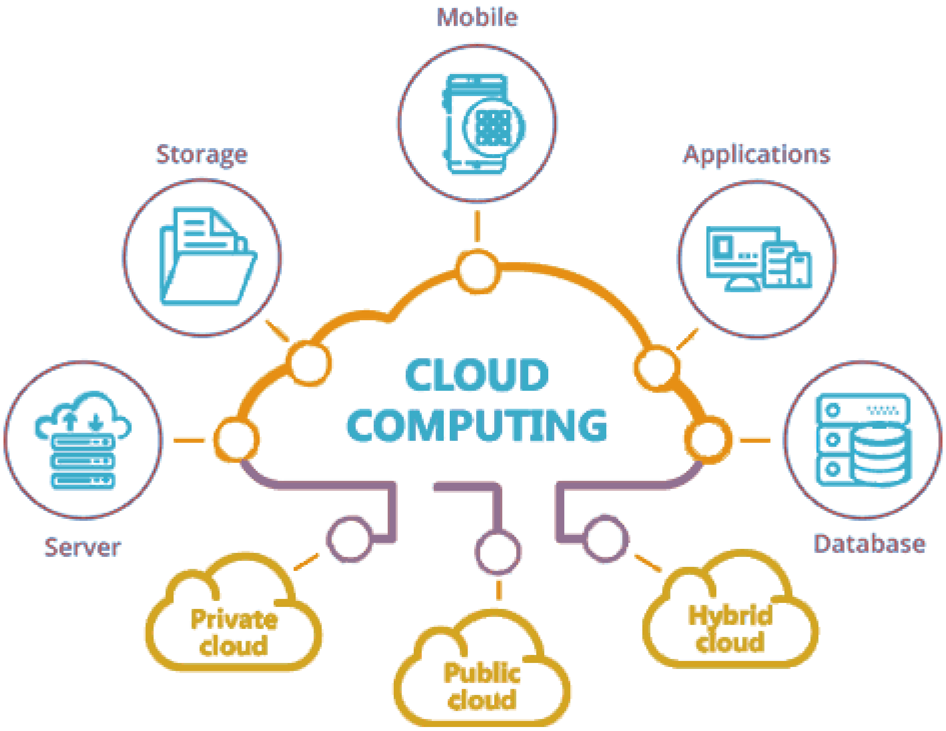
Cloud computing is the on-demand availability of computer system resources, especially data storage (cloud storage) and computing power, without direct active management by the user. The term is generally used to describe data centers available to many users over the Internet. Large clouds, predominant today, often have functions distributed over multiple locations from central servers. If the connection to the user is relatively close, it may be designated an edge server.

Clouds may be limited to a single organization (enterprise clouds), or be available to multiple organizations (public cloud).

Cloud computing relies on sharing of resources to achieve coherence and economies of scale.

Cloud computing, often referred to as simply *the cloud* is the delivery of on-demand computing resources – everything from applications to data centers – over the internet on a pay-for-use basis.

- Elastic resources: Scale up or down quickly and easily to meet changing demand.
- Metered services: Pay only for what you use.
- Self-service: Find all the IT resources you need, with self-service access.



Cloud Computing

1.4.5 PROJECT SYSTEM DESIGN

Patient:

In this module, there are n numbers of patient are present. Patient should register before doing some operations. And register user details are stored in user module. After registration successful he has to login by using authorized user name and password. Login successful he will do some operations like Send AppoinmentRequest , AccessRequest, Receive Prescription

Doctor:

In this doctor module, we develop the following functionalities:

Login

View Patient Request

Send Access Request to Cloudlet 1 or 2 or 3

View patient records

Update patient health records like ecg,Send prescription details to user

CloudLet:

In this module, the **CloudLet** has to login by using valid name and password. After login successful he can do some operations such as Add Docter, View all Docter Information, view Patient, and view the Intruder Detection Details.

Intruder:

In this Intruder module, we develop the following functionalities:

Login, view patient record means it show in encrypted format and try modify data

1.4.6 PROJECT CODING

HTML

Html is a language which is used to create web pages with html marking up a page to indicate its format, telling the web browser where you want a new line to begin or how you want text or images aligned and more are possible.

We used the following tags in our project.

Table:

Tables are so popular with web page authors is that they let you arrange the elements of a web page in such a way that the browser won't rearrange them web page authors frequently use tables to structure web pages.

TR:

TRis used to create a row in a table encloses <TH> and <TD> elements. <TR> contain many attributes. Some of them are,

- ALIGN: specifies the horizontal alignment of the text in the table row.
- BGCOLOR: Specifies the background color for the row.
- BORDERCOLOR: Sets the external border color for the row.
- VALIGN: Sets the vertical alignment of the data in this row.

TH:

TH is used to create table heading.

- ALIGN: Sets the horizontal alignment of the content in the table cell. Sets LEFT, RIGHT, CENTER.
- BACKGROUND: Species the back ground image for the table cell.

- BGCOLOR: Specifies the background color of the table cell
- VALIGN: Sets the vertical alignment of the data. Sets to TOP, MIDDLE, BOTTOM or BASELINE.
- WIDTH: Specifies the width of the cell. Set to a pixel width or a percentage of the display area.

TD:

TD is used to create table data that appears in the cells of a table.

- ALIGN: Species the horizontal alignment of content in the table cell. Sets to LEFT, CENTER, RIGHT.
- BGCOLOR: Specifies the background image for the table cell.
- BGCOLOR: sets the background color of the table cells.
- WIDTH: Species the width of the cell

Frames:

Frames are used for either run off the page or display only small slices of what are supposed to be shown and to configure the frame we can use <FRAMESET>There are two important points to consider when working with <FRAMESET>.

- <FRAMESET> element actually takes the place of the <BODY> element in a document.

Form:

The purpose of FORM is to create an HTML form; used to enclose HTML controls, like buttons and text fields.

Attribute:

- ACTION: Gives the URL that will handle the form data.
- NAME: Gives the name to the form so you can reference it in code set to an alphanumeric string.
- METHOD: method or protocol is used to sending data to the target action URL. The GET method is the default, it is used to send all form name/value pair information in an URL. Using the POST method, the content of the form are encoded as with the GET method, but are sent in environment variables.

Java Script:

Java script originally supported by Netscape navigator is the most popular web scripting language today. Java script lets you embedded programs right in your web pages and run these programs using the web browser. You place these programs in a <SCRIPT> element, usually within the <HEAD> element. If you want the script to write directly to the web page, place it in the <BODY> element.

MySQL:

MySQL is an open source relational database management system (RDBMS).This is the most popular database system used with PHP. MySQL is distributed and supported by Oracle Corporation.

MySQL runs on almost all platforms including Linux, Unix and Windows. Although it can be used in a wide range of applications, MySQL is often associated with web applications and online publishing.

The Java Platform

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

JAVA SERVER PAGES (JSP):

Java Server Pages (JSP) technology enables you to mix regular, static HTML with dynamically generated content. You simply write the regular HTML in the normal manner, using familiar Web-page-building tools. You then enclose the code for the dynamic parts in special tags, most of which start with <% and end with %>.

Jakarta Tomcat:

Tomcat is the Servlet/JSP container. Tomcat implements the Servlet 2.4 and JavaServer Pages 2.0 specification. It also includes many additional features that make it a useful platform for developing and deploying web applications and web services.

1.4.7 PROJECT 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

TYPES OF TESTS

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.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program.

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.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results.

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.

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..

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.

1.4.8 INPUT SCREENS

The **Input Screen** allows users to search for transactions using three search options, Quick Search, Passenger Search, and Transaction Search. Common Search Options are applied to these search options to refine the search results.

Objective

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.

1.4.9OUTPUT SCREENS

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.

1.4.10CONCLUSIONS

In this paper, we investigated the problem of privacy protection and sharing large medical data in cloudlets and the remote cloud. We developed a system which does not allow users to transmit data to the remote cloud in consideration of secure collection of data, as well as low communication cost. However, it does allow users to transmit data to a cloudlet, which triggers the data sharing problem in the cloudlet.

Firstly, we can utilize wearable devices to collect users' data, and in order to protect users privacy, we use NTRU mechanism to make sure the transmission of users' data to cloudlet in security.

Secondly, for the purpose of sharing data in the cloudlet, we usetrust model to measure users' trust level to judge whether to sharedata or not.

Thirdly, for privacy-preserving of remote cloud data, we partition the data stored in the remote cloud and encrypt the data in different ways, so as to not just ensure data protection but also accelerate the efficacy of transmission.

Finally, we propose collaborative IDS based on cloudlet mesh to protect the whole system. The proposed schemes are validated with simulations and experiments.

CHAPTER - 2: LITERATURE SURVEY

2.1 SURVEY ON BACKGROUND

”Wearable medical device for tele home healthcare”(2004)

The world's ageing population and prevalence of diseases have lead to high demand for tele-home healthcare, in which vital-signs monitoring is essential. An overview of state-of-art wearable technologies for remote patient-monitoring is presented, followed by case studies on a cuffless blood pressure meter, ring-type heart rate monitor, and Bluetooth/spl trade/-based ECG monitor. Aim of our project is to develop a tele-home healthcare system which utilizes wearable devices, wireless communication technologies, and multisensor data fusion methods.

“Cloud-supported monitoring”(2015)

The potential of cloud-supported cyber-physical systems (CCPSs) has drawn a great deal of interest from academia and industry. CCPSs facilitate the seamless integration of devices in the physical world with cyberspace. This enables a range of emerging applications or systems such as patient or health monitoring, which require patient locations to be tracked.

“Security models and requirements for healthcare application clouds”

In this paper we discuss important concepts related to EHR sharing and integration in healthcare clouds and analyzes the arising security and privacy issues in access and management of EHRs. We describe an EHR security reference model for managing

Security issues in healthcare clouds, which highlights three important core components in securing an EHR cloud.

“Security models and requirements for healthcare application clouds”

In this paper we discuss important concepts related to EHR sharing and integration in healthcare clouds and analyzes the arising security and privacy issues in access and management of EHRs. We describe an EHR security reference model for managing security issues in healthcare clouds, which highlights three important core components in securing an EHR cloud. We illustrate the development of the EHR security reference model through a use-case scenario and describe the corresponding security countermeasures and state of art security techniques that can be applied as basic security guards.

“Big video data for light-field-based 3d telemedicine”

Big data and 3D technologies have been successfully leveraged in a variety of industries to improve their efficiency and quality. The healthcare sector has lagged in the uptake of these new technologies. In this article, we propose a novel light field (LF)-based 3D telemedicine system. To solve the challenges in storage and analysis of LFV, we extend the standard multi-view video coding (MVC) approach to LF-MVC, which is able to achieve up to a 23 percent higher compression rate when compared to standard MVC. Furthermore, a big data analysis framework is proposed to integrate LFV into conventional telemedicine analysis, which can achieve improved classification, statistics gathering, prediction, and cognitive analysis for healthcare applications.

2.1 CONCLUSION ON SURVEY

In this paper, we investigated the problem of privacy protection and sharing large medical data in cloudlets and the remote cloud. We developed a system which does not allow users to transmit data to the remote cloud in consideration of secure collection of data, as well as low communication cost. However, it does allow users to transmit data to a cloudlet, which triggers the data sharing problem in the cloudlet. Firstly, we can utilize wearable devices to collect users’ data, and in order to protect users privacy, we use NTRU mechanism to make sure the transmission of users’ data to cloudlet in security. Secondly, for the purpose of sharing data in the cloudlet, we use trust model to measure users’ trust level to judge whether to share data or not. Thirdly, for privacy-preserving of remote cloud data, we partition the data stored in the remote cloud and encrypt the data in different ways, so as to not just ensure data protection but also accelerate the efficacy of transmission. Finally, we propose collaborative IDS based on cloudlet mesh to protect the whole system. The proposed schemes are validated with simulations and experiments.

CHAPTER-3: Software and Hardware requirements

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

3.1 Software Requirements

For developing the application the following are the Software Requirements:

- Operating system : - Windows XP/7.
- Coding Language : JAVA/J2EE
- Data Base : MYSQL
- Tool :Net beans 7.2.1

3.2 Hardware Requirements

For developing the application the following are the Hardware Requirements:

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40 GB.
- Floppy Drive : 1.44 Mb.
- Monitor : 15 VGA Colour.
- Mouse : Logitech.
- Ram : 512 Mb.

CHAPTER - 4: SOFTWARE DEVELOPMENT ANALYASIS

4.1 EXISTING SYSTEM

In the existing system, In Cao et al. [11], an MRSE (multi keyword ranked search over encrypted data in cloud computing) privacy protection system was presented, which aims to provide users with a multi-keyword method for the cloud's encrypted data. Although this method can provide result ranking, in which people are interested, the amount of calculation could be cumbersome. In Zhang et al. [24], a priority based health data aggregation (PHDA) scheme was presented to protect and aggregate different types of healthcare date in cloud assisted wireless boby area network (WBANs). The article in the existing system investigates security and privacy issues in mobile healthcare networks, including the privacy-protection for healthcare data aggregation, the security for data processing and misbehavior. The system describes a flexible security model especially for data centric applications in cloud computing based scenario to make sure data confidentiality, data integrity and fine grained access control to the application data. The system gives a systematic literature review of privacy-protection in cloud-assisted healthcare system.

4.2 PROPOSED SYSTEM

- **I**n this paper,The proposed system, a cloudlet based healthcare system is presented, where the privacy of users' physiological data and the efficiency of data transmissions are our main concern. The system uses NTRU for data protection during data transmissions to the cloudlet.
- In order to share data in the cloudlet, we use users' similarity and reputation to build up trust model. Based on the measured users' trust level, the system determines whether data sharing is performed.
- The proposed system divides data in remote cloud into different kinds and utilizes encryption mechanism to protect them respectively.
- The Proposed system proposes collaborative IDS based on cloudlet mesh to protect the whole healthcare system against malicious attacks.

4.2.1 SYSTEM ARCHITECTURE

We build up a novel healthcare system by utilizing the flexibility of cloudlet. The functions of cloudlet include privacy protection, data sharing and intrusion detection. In the stage of data collection, we first utilize Number Theory Research Unit (NTRU) method to encrypt user's body data collected by wearable devices. Those data will be transmitted to nearby cloudlet in an energy efficient fashion. Secondly, we present a new trust model to help users to select trustable partners who want to share stored data in the cloudlet. The trust model also helps similar patients to communicate with each other about their diseases. Thirdly, we divide users' medical data stored in remote cloud of hospital into three parts, and give them proper protection. Finally, in order to protect the healthcare system from malicious attacks, we develop a novel collaborative intrusion detection system (IDS) method based on cloudlet mesh, which can effectively prevent the remote healthcare big data cloud from attacks. Our experiments demonstrate the effectiveness of the proposed scheme.

.

- to build the privacy preserving medical treatment through non deterministic finite automata we need to connect to cloud which represents as a bridge between patient and doctor

- we use cloud not only to represent as a bridge but also as a defender to track the intruder and also safeguard the data

- we are describing concept of implementing cloud based health care platform by including modules of patient , doctor , cloudlets and intruder

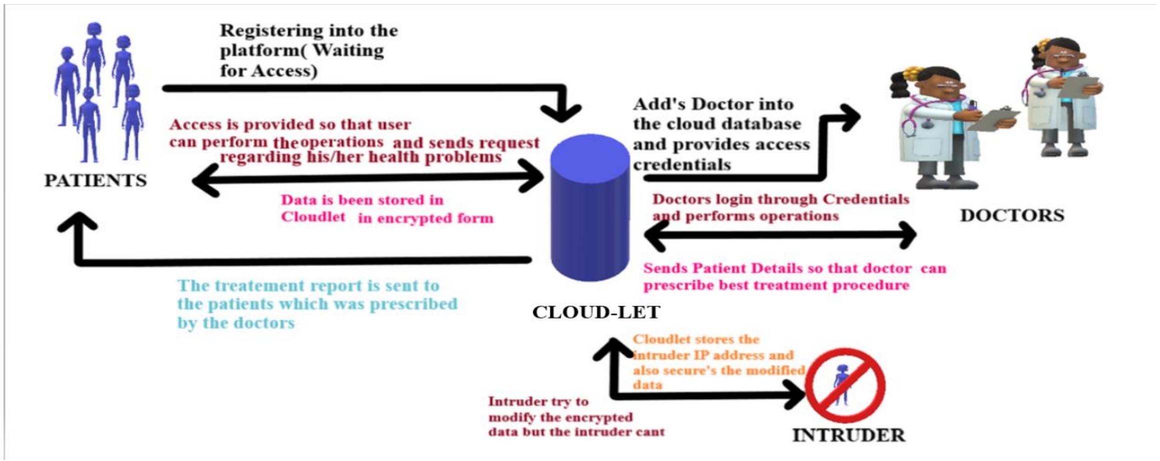

Registering into the platform(Waiting for Access)
PATIENTS
Access is provided so that user can perform theoperations and sends request regarding his/her health problems
Data is been stored in Cloudlet in encrypted form
The treatement report is sent to the patients which was prescribed by the doctors
CLOUD-LET
Add's Doctor into the cloud database and provides access credentials
DOCTORS
Doctors login through Credentials and performs operations
Sends Patient Details so that doctor can prescribe best treatment procedure
Cloudlet stores the intruder IP address and also secure's the modified data
Intruder try to modify the encrypted data but the intruder cant
INTRUDER

Fig4.2.1 SYSTEM ARCHITECTURE

4.3 MODULES FUNCTIONALITY

We are describing concept of implementing Cloud based health care platform by including Modules of

- Patient
- Doctor
- CloudLet
- Intruder

Patient

In this module, there are n numbers of patient are present. Patient should register before doing some operations. And register user details are stored in user module. After registration successful he has to login by using authorized user name and password. Login successful he will do some operations like Send AppoinmentRequest ,AccessRequest,Receive Prescription.

Doctor:

In this doctor module, we develop the following functionalities:

Login, View Patient Request

Send Access Request to Cloudlet 1 or 2 or 3

View patient records

Update patient health records like ECG,Send prescription details to user

Cloudlet:

In this module, the **Cloudlet** has to login by using valid name and password. After login successful he can do some operations such as Add Doctor, View all Doctor Information, view Patient, and view the Intruder Detection Details.

Intruder:

In this Intruder module, we develop the following functionalities:

Login, View patient records means it is showing only encrypted format

Try to modify data means alert mail send to patient or cloud let.

CHAPTER – 5: PROJECT SYSTEM DESIGN

5.1 DFDs in case of database projects

DATA FLOW DIAGRAM:

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

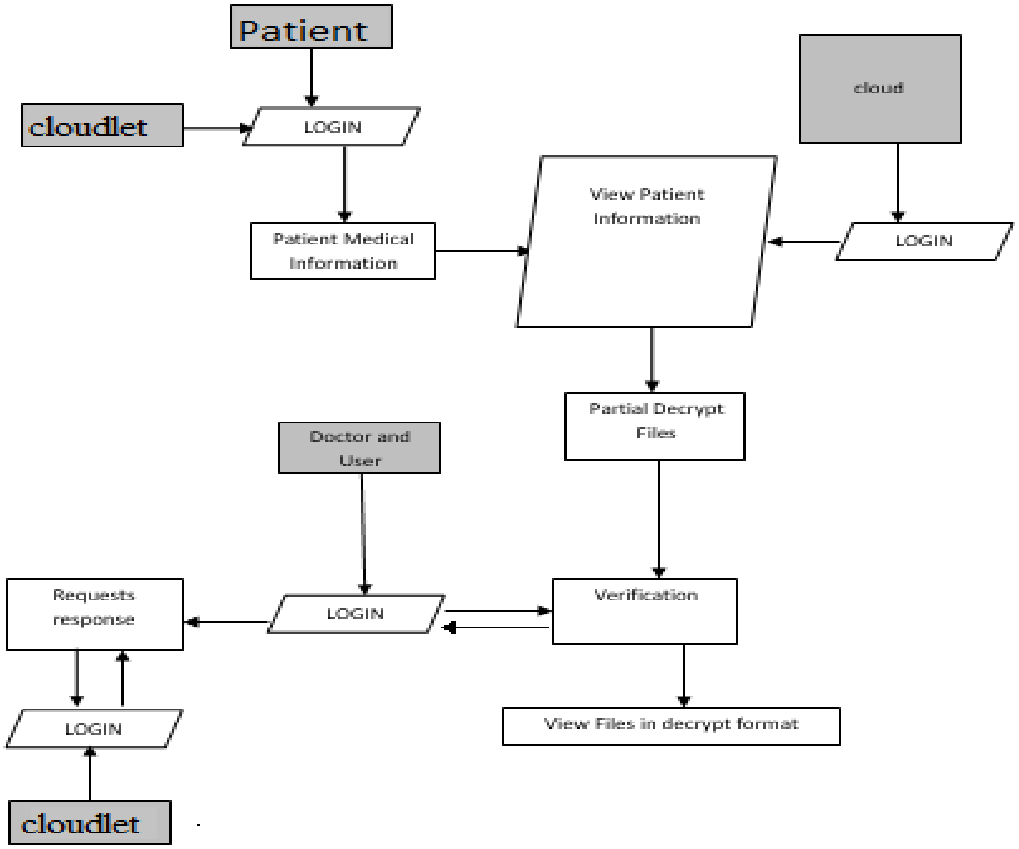


Fig: 5.1 DATA FLOW DIAGRAM

5.2 E-R Diagrams

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Lets have a look at a simple ER diagram to understand this concept.

Components of e-r diagrams are as follows:

1.Entity

2.Attribute

3. Relationship

1. Entity

An entity is an object or component of data. An entity is represented as rectangle in an ER diagram.
For example: In the following ER diagram we have two entities Student and College and these two entities have many to one relationship as many students study in a single college. We will read more about relationships later, for now focus on entities.

2. Attribute

An attribute describes the property of an entity. An attribute is represented as Oval in an ER diagram. There are four types of attributes:

1. Key attribute
2. Composite attribute
3. Multivalued attribute
4. Derived attribute

3. Relationship

A relationship is represented by diamond shape in ER diagram, it shows the relationship among entities. There are four types of relationships:

1. One to One
2. One to Many
3. Many to One
4. Many to Many

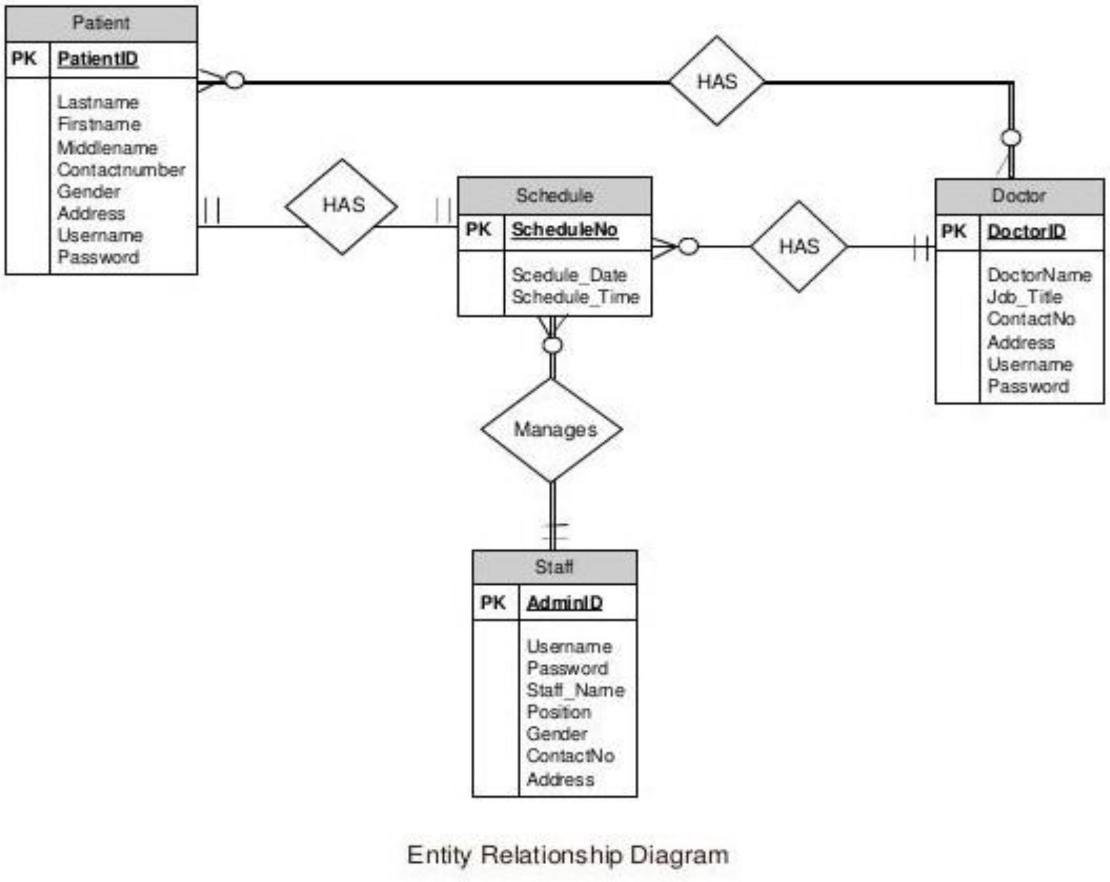


Fig 5.2.1 Entity relationship diagram for doctor and patient

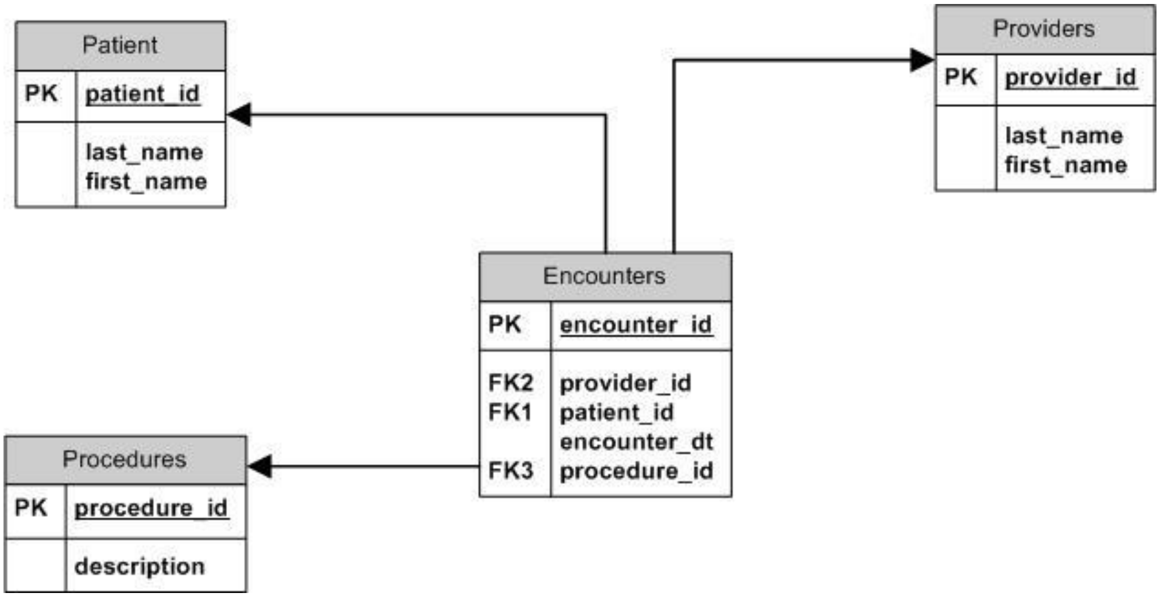


Fig 5.2.2 sample ER Diagram for patient/doctor relationship

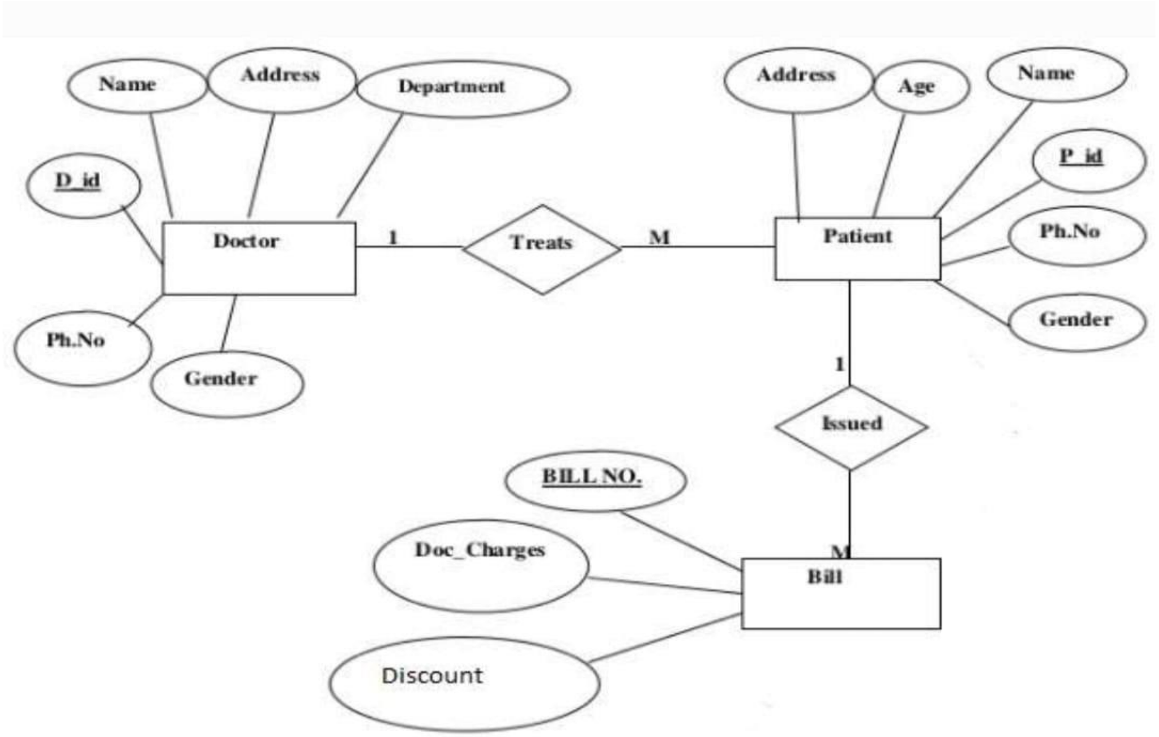


Fig 5.2.3 Online Medical Appointment Booking Application

5.3 UML diagrams

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modeling language.
- Encourage the growth of OO tools market.
- Support higher level development concepts such as collaborations, frameworks, patterns and components.

5.3.1 CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

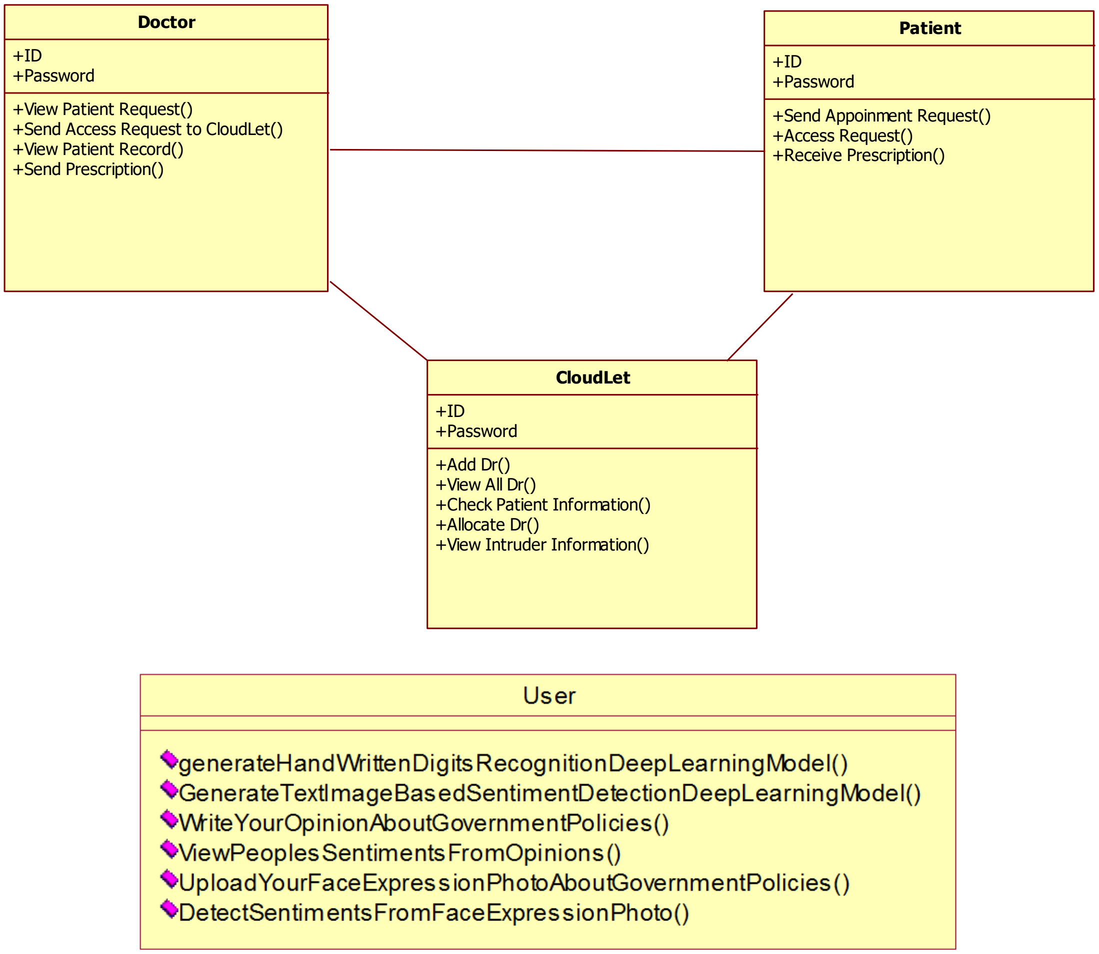

Doctor
+ID
+Password
+View Patient Request()
+Send Access Request to CloudLet()
+View Patient Record()
+Send Prescription()
Patient
+ID
+Password
+Send Appoinment Request()
+Access Request()
+Receive Prescription()
CloudLet
+ID
+Password
+Add Dr()
+View All Dr()
+Check Patient Information()
+Allocate Dr()
+View Intruder Information()
User
generateHandWrittenDigitsRecognitionDeepLearningModel()
GenerateTextImageBasedSentimentDetectionDeepLearningModel()
WriteYourOpinionAboutGovernmentPolicies()
ViewPeoplesSentimentsFromOpinions()
UploadYourFaceExpressionPhotoAboutGovernmentPolicies()
DetectSentimentsFromFaceExpressionPhoto()

Fig 5.3.1 class diagram

5.3.2 USE CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

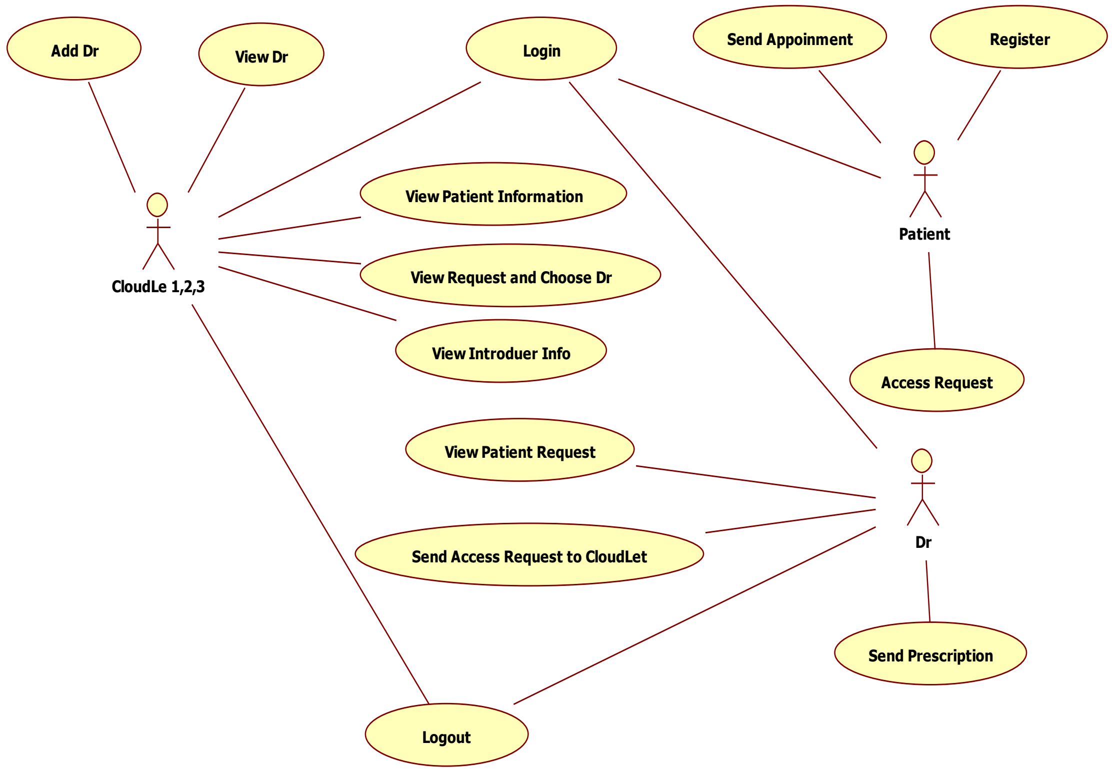


Fig 5.3.2 usecase diagram

5.3.3SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

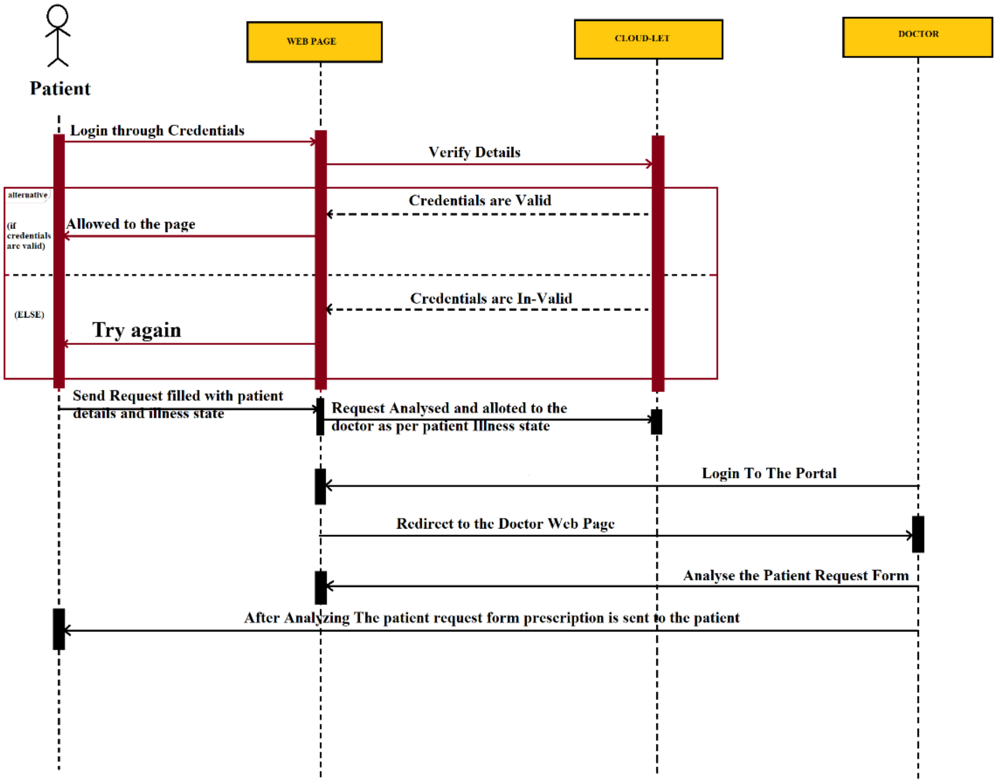

Patient
WEB PAGE
CLOUD-LET
DOCTOR
Login through Credentials
Verify Details
alternative
Credentials are Valid
(if credentials are valid)
Allowed to the page
(ELSE)
Credentials are In-Valid
Try again
Send Request filled with patient details and illness state
Request Analysed and alloted to the doctor as per patient Illness state
Login To The Portal
Redirect to the Doctor Web Page
Analyse the Patient Request Form
After Analyzing The patient request form prescription is sent to the patient

Fig 5.3.3 sequence diagram

5.3.4 COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

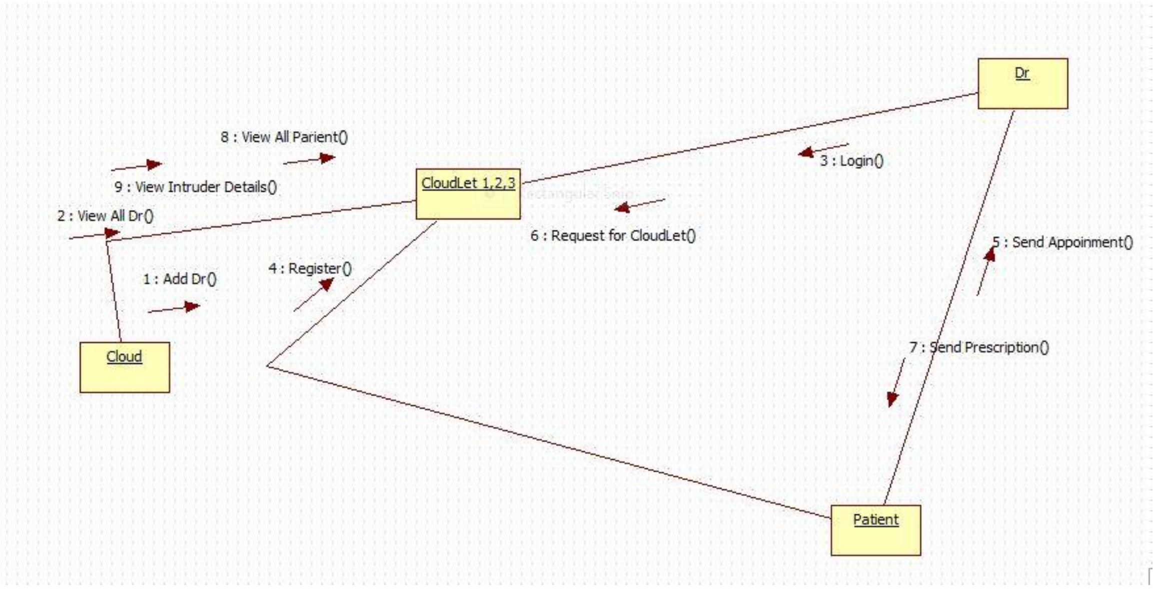


Fig 5.3.4 collaboration diagram

5.3.5 ACTIVITY DIAGRAM

AActivity diagrams are graphical representation of workflows of stepwise activities ans actions. it is a behavioral diagram i.e. it depicts the behavior of the system. an activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

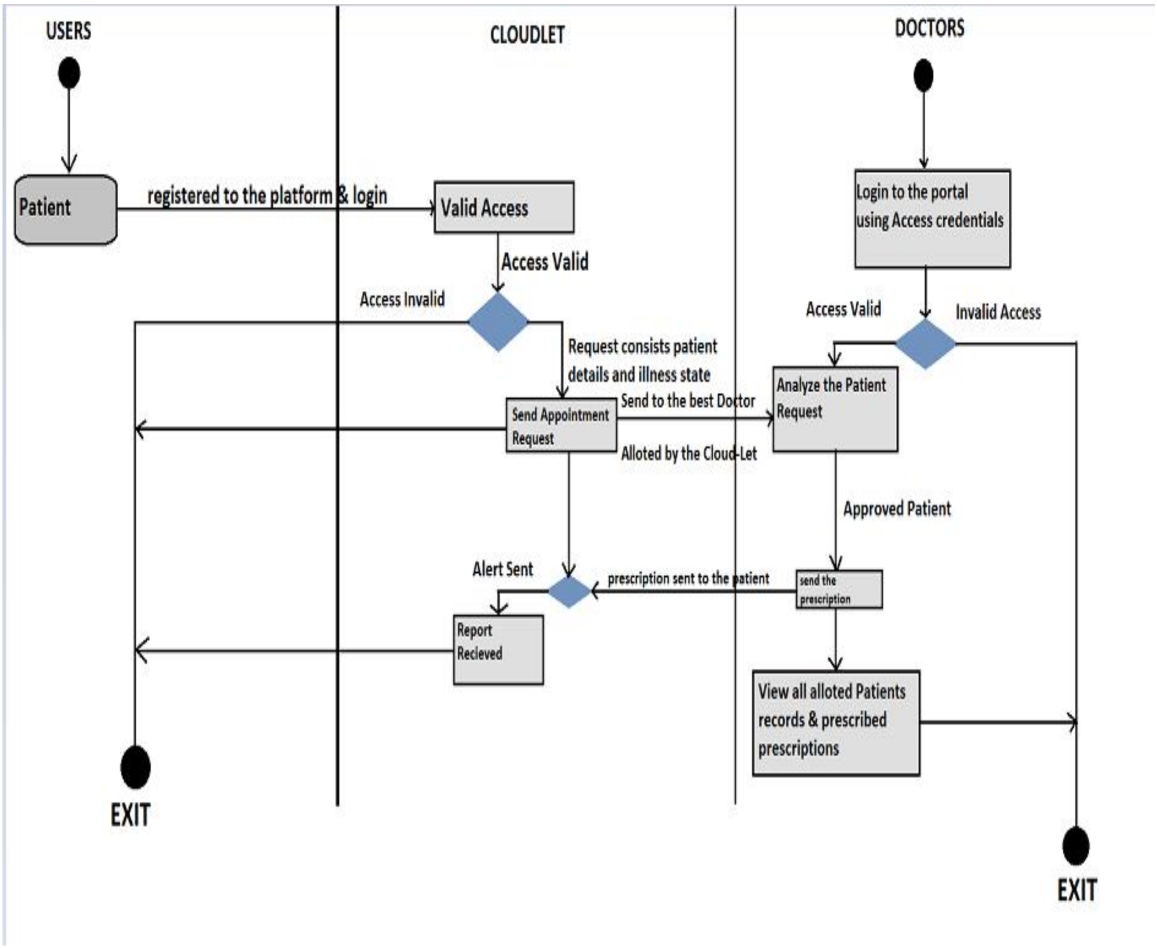

USERS
CLOUDLET
DOCTORS
Patient
registered to the platform & login
Valid Access
Access Valid
Access Invalid
Request consists patient details and illness state
Send Appointment Request
Send to the best Doctor
Alloted by the Cloud-Let
Login to the portal using Access credentials
Access Valid
Invalid Access
Analyze the Patient Request
Approved Patient
send the prescription
prescription sent to the patient
Alert Sent
Report Recieved
View all alloted Patients records & prescribed prescriptions
EXIT
EXIT

Fig 5.3.5 activity diagram

5.3.6 DEPLOYMENT DIAGRAM

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.[1] To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have subnodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

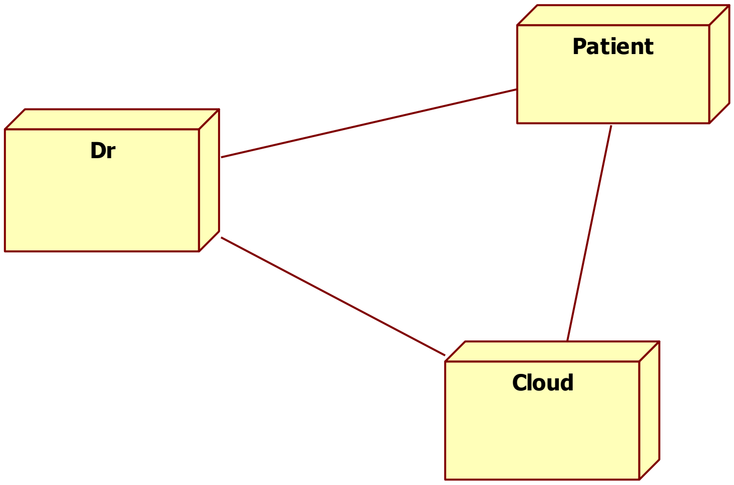


Fig 5.6 deployment diagram

CHAPTER – 6: PROJECT CODING

6.1 TECHNOLOGY

Java Technology

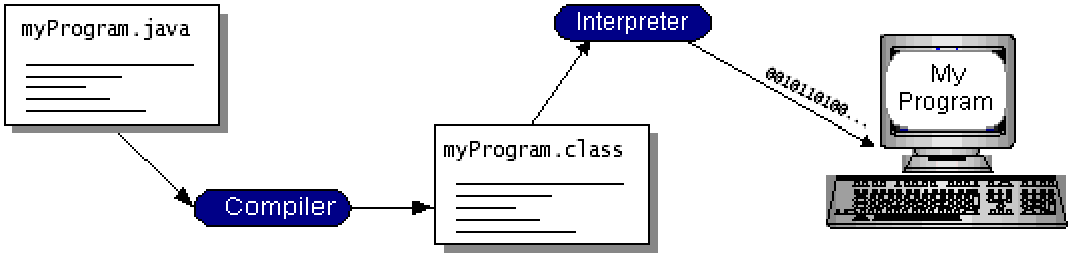
Java technology is both a programming language and a platform.

The Java Programming Language

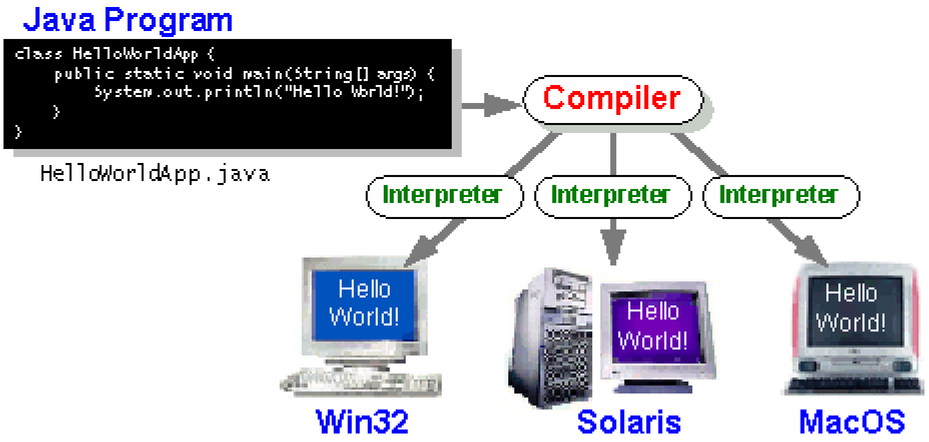
The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called *Java byte codes* —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.



You can think of Java byte codes as the machine code instructions for the *Java Virtual Machine* (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make "write once, run anywhere" possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.



```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

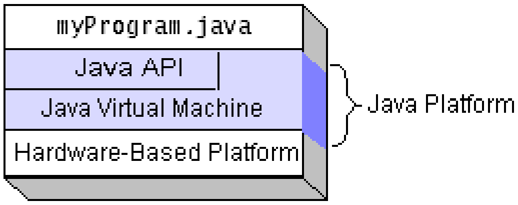
The Java Platform

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

- The *Java Virtual Machine* (Java VM)
- The *Java Application Programming Interface* (Java API)

The following figure depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.



Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

What Can Java Technology Do?

The most common types of programs written in the Java programming language are *applets* and *applications*. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

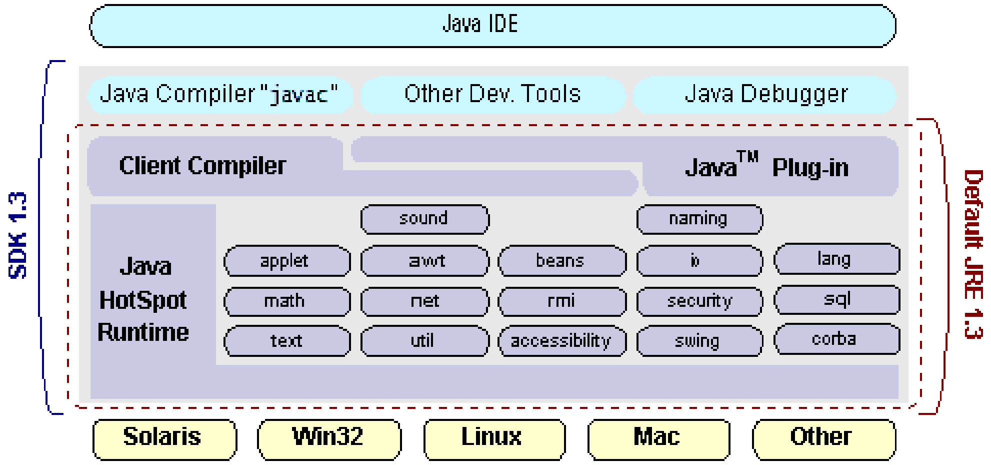
However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a *server* serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a *servlet*. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

- **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
- **Applets**: The set of conventions used by applets.
- **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
- **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
- **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
- **Software components**: Known as JavaBeans, can plug into existing component architectures.
- **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
- **Java Database Connectivity (JDBC^{TM})**: Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.



How Will Java Technology Change My Life?

We can't promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

- **Get started quickly**: Although the Java programming language is a powerful object-oriented language, it's easy to learn, especially for programmers already familiar with C or C++.
- **Write less code**: Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.
- **Write better code**: The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extendible API let you reuse other people's tested code and introduce fewer bugs.
- **Develop programs more quickly**: Your development time may be as much as twice as fast versus writing the same program in C++. Why? You write fewer lines of code and it is a simpler programming language than C++.

- **Avoid platform dependencies with 100% Pure Java**: You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure Java^{TM}Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.
- **Write once, run anywhere**: Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.
- **Distribute software more easily**: You can upgrade applets easily from a central server. Applets take advantage of the feature of allowing new classes to be loaded "on the fly," without recompiling the entire program.

ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto* standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made the choice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needs suddenly change.

JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of "plug-in" database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. ***SQL Level API***

   The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to "generate" JDBC code and to hide many of JDBC's complexities from the end user.

2. ***SQL Conformance***

   SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard functionality in a manner that is suitable for its users.

3. ***JDBC must be implemental on top of common database interfaces***

   The JDBC SQL API must "sit" on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

4. ***Provide a Java interface that is consistent with the rest of the Java system***

   Because of Java's acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

5. ***Keep it simple***

   This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

6. ***Use strong, static typing wherever possible***

   Strong typing allows for more error checking to be done at compile time; also, less error appear at runtime.

7. ***Keep the common cases simple***

   Because more often than not, the usual SQL calls used by the programmer are simple SELECT's, INSERT's, DELETE's and UPDATE's, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible.

CLOUD COMPUTING

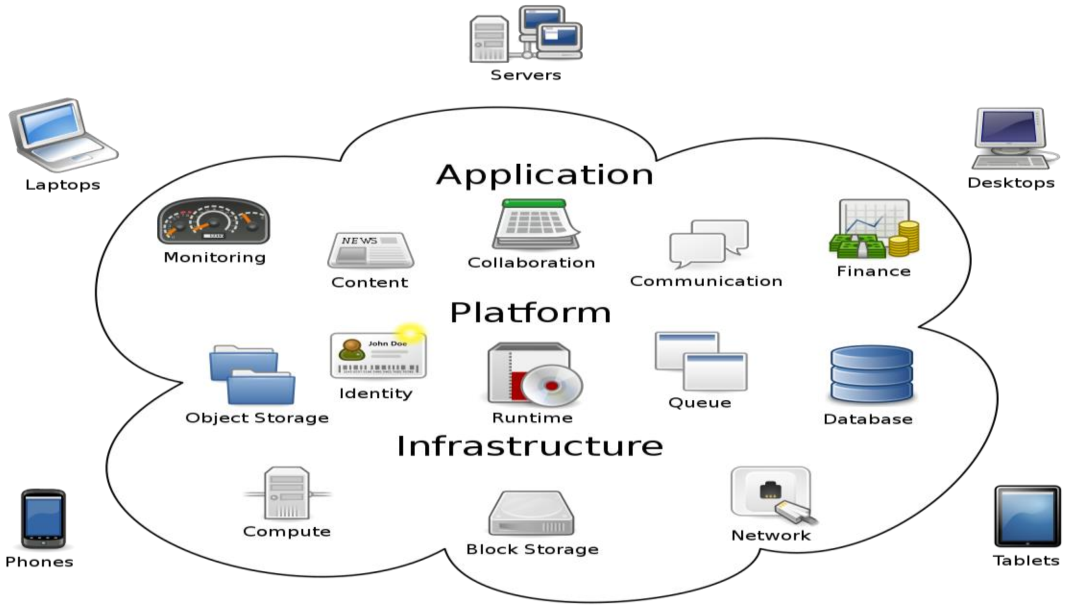
Cloud computing is the on-demand availability of computer system resources, especially data storage (cloud storage) and computing power, without direct active management by the user. The term is generally used to describe data centers available to many users over the Internet. Large clouds, predominant today, often have functions distributed over multiple locations from central servers. If the connection to the user is relatively close, it may be designated an edge server.

Clouds may be limited to a single organization (enterprise clouds), or be available to multiple organizations (public cloud).

Cloud computing relies on sharing of resources to achieve coherence and economies of scale.

Cloud computing, often referred to as simply *the cloud* is the delivery of on-demand computing resources – everything from applications to data centers – over the internet on a pay-for-use basis.

- Elastic resources: Scale up or down quickly and easily to meet changing demand.
- Metered services: Pay only for what you use.
- Self-service: Find all the IT resources you need, with self-service access.



Structure of Cloud Computing

Benefits of the Cloud

Cloud computing fundamentally changes the way that IT services are delivered to organizations. Instead of both owning and managing IT services for themselves, or using an outsourcing approach built around dedicated hardware, software, and support services, organizations can use cloud computing to meet their IT requirements using a flexible, on-demand, and rapidly scalable model that requires neither ownership on their part, nor provision of dedicated resources.

Some of the benefits that Cloud Computing brings are as follows:

1. Reduced Cost: Cost is a clear benefit of cloud computing, both in terms of CapEx and OpEx. The reduction in CapEx is obvious because an organization can spend in increments of required capacity and does not need to build infrastructure for maximum (or burst) capacity. For most enterprises, OpEx constitutes the majority of spending; therefore, by utilizing a cloud provider or adopting cloud paradigms internally, organizations can save operational and maintenance budgets.

2. Flexibility: Flexibility benefits derive from rapid provisioning of new capacity and rapid relocation or migration of workloads. In public sector settings, cloud computing provides agility in terms of procurement and acquisition process and timelines.

3. Improved Automation: Cloud computing is based on the premise that services can not only be provisioned but also de-provisioned in a highly automated fashion. This specific attribute offers significant efficiencies to enterprises.

4.Focus on Core Competency: Government agencies can reap the benefits of cloud computing in order to focus on its core mission and core objectives and leverage IT resources as a means to provide services to citizens.

5.Sustainability: The poor energy efficiency of most existing data centers, due to poor design or poor asset utilization, is now understood to be environmentally and economically unsustainable. Through leveraging economies of scale and the capacity to manage assets more efficiently, cloud computing consumes far less energy and other resources than a traditional IT data center.

6.2 ALOGRITHMS

A. Key Distribution and User Authorization

KGC runs the KeyGen algorithm in PCTD to generate thepublic parameter PP = (g;N) and master secret key SK = _for the system, and S Key algorithm to generate the partialstrong keys SK1 = _1 and SK2 = _2 for CP and CSP, respectively. KGC generates the secret/public key pair skA =a; pkA = ga for hospital A, and skB = b; pkB = gb for patientB, where a; b are randomly selected from ZN.If a patient B wishes to request service from hospitalA, A defines a valid service time period in the formof SP = “20190101-20191231”. Then, KGC generates acertificate number CN, and a certificate CERA;B: hcer =(CN; A;B; SP; pk_); Sig(cer; SK)i, where pk_ = gsk_ ,sk_ 2R ZN, and sk_ is confidentially sent to B.

B. Encryption of Medical Model and Query

Hospital A encrypts the weighted NFAbasedmedical model to [[M]]pkA =([[Q]]pkA; [[_]]pkA; [[q0]]pkA; [[F]]pkA; [[_]]pkA; [[_]]pkA), wherethe encrypted states [[Q]] = ([[q0]];

_ _ _ ; [[qn1]]), the encryptedtreatment methods [[_]] = ([[y_1]]; _ _ _ ; [[y_n2]]), the encryptedaccept states [[F]] = ([[q%1]]; _ _ _ ; [[q%n3]]), the encryptedtransition weights [[W]] = ([[w1]]; _ _ _ ; [[wn5]]), and the emptysymbol _ is encrypted to [[_]].When patient B queries the telemedical service, the illnessstates _ = (_1; _ _ _ ; _m) in the last few days are encryptedinto [[_]]pkB = ([[_1]]pkB; _ _ _ ; [[_m]]pkB) and sent to CP, whichis used for diagnosis and treatment recommendation.

C. Illness State Representation and Match Test

Illness State Representation. In the healthcare domain, theillness state can be expressed by symptoms and a set ofphysiological index, where the former can be described bythe patient and the latter can be monitored by the mIoT. PMedadopts this method,the representation of q, _ and the encryption of them. Fivevital signs of human body are body temperature (BT), bloodpressure (BP), blood glucose level (BG), respiratory rate (RR)and heart rate (HR), which have frequently-used units _C,mmHg, mmol/L, breaths per minute and beats per minute, respectively. In the illness state q in medical modelutilizes intervals to describe the five vital signs and severalmedical terminologies (in lexicographical order) to describethe symptoms. The illness state _ of patient B is representedby the concrete physiological index rather than interval.

Algorithm 1: SECURE TREATMENT PROCEDURE TRAVERSE
ALGORITHM (TPT)

```
Input: [[M]]pkA, MV isit, MState.
Output: [[TP]]pkA = ([[T P1]]pkA; _ _ _ ; [[T Pn]]pkA).
set the arrays value(_; _), weight(_; _) according to the state
Transition table of the encrypted NFA [[M]]pkA;
set the stacks Q; Y;W to be empty and set n = 0;
fori = 0 to n1 do
set counti = 0;
for k = 0 to MV isit do
fori = 0 to n1 do
```

```
for j = 0 to n1 do
set visitk;i;j = 0;
Q.push ([[q0]] pkA), count0 = count0 + 1;
while Q 6=; do
set _ = Q:peak:element, _ = □1;
fori = 1 to n1 do
if (value;i 6= ?) & (visitcount[_];_;i = 0) then
set _ = i, visitcount[_];_;_ = 1;
if _ = □1 then
for j = 0 to n1 do
set visitcount[_];_;j = 0;
Q.pop;
 count_ = count_ □ 1;
if Y 6= ; then
 Y .pop, W.pop;
else if (_ 6= □1) & (count_ < MV isit) then
 Y .push (value_;_), W.push (weight_;_), Q.push
([[q_]]pkA), count_ = count_ + 1;
if Q 6= ; then
set _0 = Q:peak:element;
if ([[q_0 ]]pkA 2 [[F]]pkA) then
 n = n + 1, [[Qn]]pkA = Q, [[Yn]]pkA = Y ,
[[Wn]]pkA = W;
Q.pop, Y .pop, W.pop, count_0 = count_0 □ 1;
else if ([[q_0]]pkA =2 [[F]]pkA) & (Q:size = MState)en
Q.pop, Y .pop, W.pop, count_0 = count_0 □ 1;
set [[T Pi]]pkA = ([[Qi]]pkA; [[Yi]]pkA; [[Wi]]pkA) (1 _ i _ n);
 Return [[TP]]pkA = ([[T P1]]pkA; _ _ _ ; [[T Pn]]pkA).
```

the above algorithm (TPT) Secure treatment procedure traverse algorithm is used to share data among doctor and patient.

Algorithm 2: SECURE TOP-k BEST TREATMENT PROCEDURES
SELECTION (BPS-k)

```
Input: [[ETP]]pkA.
Output: [[ETPMin]]pk_ .
 Set S = [[ETP]]pkA;
fori = 1 to k do
 CP and CSP jointly calculate
[[ET PMini ]]pk_   SMinn(S);
for j = 1 to n do
 CP randomly selects rj 2 ZN and computes
lj = ([[WMini ]]pk_ )rj _ ([[Wj ]]pk_ )N□rj ,
l0j
= PD1SK1 (lj), where L(rj) < L(N)=4 □ 1;
 Permute (lj ; l0j
) using permutation function _i and get
(l_i(j); l0
_i(j)) for 1 _ j _ n, which are sent to CSP;
 CSP computes l00
_i(j) = PD2SK2 (l_i(j); l0
_i(j));
 If l00
_i(j) = 0, set A_i(j) = [[MWeight]]pk_ ; otherwise,
A_i(j) = [[1]]pk_ . CSP sends A_i(j) to CP, 1 _ j _ n;
 CP obtains (A1; _ _ _ ;An) by using permutation _□1
i ;
 Refresh ([[W1]]pk_ ; _ _ _ ; [[Wn]]pk_ ) in S by computing
[[Wj ]]pk_   SMD([[Wj ]]pk_ ;Aj), 1 _ j _ n;
 Return [[ETPMin]]pk_ .
```

the above secure top-k best treatment proceduresselection (bps-k) algorithm is used to give top-k best treatments

6.3 CODE IMPLEMENTATION

```
<!DOCTYPE HTML>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Medical Data Sharing</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="description" content="">
<meta name="title" content="">
<!-- Favicon -->
<link rel="shortcut icon" href="images/favicon.ico">
<!--Google Font-->
<link href='http://fonts.googleapis.com/css?family=Roboto+Condensed:400,300,700|Roboto:500,100,900&subset=latin,cyrillic-ext,greek-ext,greek,vietnamese,cyrillic,latin-ext' rel='stylesheet' type='text/css'>
<!--End Google Font-->
<!-- Main Styles -->
<link href="css/style.css" rel="stylesheet">
<link rel="stylesheet" type="text/css" href="css/box_style.css" />
<!--Common Jquery --><script src="js/jquery.min.js"></script><!--Common Jquery -->
<!-- Main Menu Styles -->
<link rel="stylesheet" href="css/menu/core.css" type="text/css" media="screen">
<link rel="stylesheet" href="css/menu/styles/lgray.css" type="text/css" media="screen">
<!--[if (gt IE 9)|!(IE)]><!-->
<link rel="stylesheet" href="css/menu/effects/slide.css" type="text/css" media="screen">
<!--<![endif]-->
<!--[if lte IE 9]>
<style type="text/css">
.menu .cols4,.menu .col4{width:989px; left:-287px;}
.menu .cols3,.menu .col3{width:989px; left:-78px;}
```

```
.menu .col3{
        width:989px;
        float:left;
        background:#191919;
        }
.menu .menuarrow{
        width:989px;
        height:10px;
        float:left;
        background:url(../../images/menu_arrow.png) 176px top no-repeat;
        }
.menu .menuarrow2{
        width:989px;
        height:10px;
        float:left;
        background:url(../../images/menu_arrow.png) 375px top no-repeat;
        }
.lgray>li>a{
        color:#FFF;
        font-weight:bold;
        font-size:12px;
        line-height:18px;
        padding:11px 28px 11px 28px;
        text-transform:uppercase;
        background:#000;
        margin-right:1px;
        }
.lgray>li:hover>a{
        background-color:#e31e24;
        border-left:none;
        padding-left:28px;
        }
</style>
<!--<![endif]-->

<!-- This piece of code, makes the CSS3 effects available for IE -->
<!--[if lte IE 9]>

<script src="js/menu.min.js" type="text/javascript" charset="utf-8"></script>
<script type="text/javascript" charset="utf-8">
        $(function() {
                $("#menu").menu({'effect' : 'slide'});
        });
</script>
<![endif]-->
<!--End Main Menu Styles -->

<!--Sanket change start -->
<link media="screen" rel="stylesheet" href="css/colorbox.css" />
<scriptsrc="js/jquery.colorbox.js"></script>
<script>
var y = jQuery.noConflict();
</script>
<script>
        y(document).ready(function() {
                y(".inline").colorbox({inline:true, width:"948px", height:"316px"});
                y(".inline").colorbox().trigger('click');

                setTimeout(function(){
                        y.colorbox.close();
                }, 15000);
        });
</script>
<!--Sanket change end -->
<script type="text/javascript" src="js/functions.js"></script>
<script type="text/javascript" src="js/slider.js"></script>
<script>
  (function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){
  (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o),
  m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)
})(window,document,'script','//www.google-analytics.com/analytics.js','ga');
ga('create', 'UA-47071863-1', 'apolloahd.com');
ga('send', 'pageview');
</script>
</head>
<body>
<!--******************** HEADER ********************-->
<div class="wrapper">
        <header>
<!--Navbar-->
<nav   style="border-top-left-radius:   120px;border-bottom-right-radius:   131px;background-
color: darkcyan;">
<ul class="menu lgray slide" id="menu">
        <li><a href="#" style="margin-left: 241px;"> Home </a>
        </li>
<li><a href="Cloud1.jsp">CloudLet 1</a>
</li>
<li><a href="Cloud1.jsp">CloudLet 2</a>
</li>
<li><a href="Cloud1.jsp"  title="CENTRES OF EXCELLENCE">CloudLet 3 </a>
```

```
<li  data-transition="slidedoun" data-masterspeed="300" data-slotamount="1" >
<imgsrc="images/homebanner5.jpg" alt="" />
<div  class="caption  sfrshopper_caption_underline_light"   data-x="100"  data-y="45"  data-speed="500" data-start="500" data-easing="easeOutBack"
<div   class="caption   fade   shopper_small_text_light"   data-x="105"   data-y="110"   data-speed="500" data-start="1100" data-easing="easeOutExpo">
<div   class="caption   fade   shopper_small_text_light"   data-x="103"   data-y="210"   data-speed="500" data-start="1300" data-easing="easeOutExpo">
<!--<a href="#"><imgsrc="images/readmorebtn.png" alt="read more"></a>-->
</div>
</li>
</ul>
<div class="tp-bannertimertp-top"></div>
</div>
</div>
</div>
<script type="text/javascript">
var CONFIG_REVOLUTION = {
delay:9000,
startwidth:1170,
startheight:460,
hideThumbs:200,
navigationType:"bullet",
navigationArrows:"verticalcentered",
navigationStyle:"round",
touchenabled:"on",
shuffle:"off",
navOffsetHorizontal:0,
navOffsetVertical:-14,
onHoverStop:"on",
thumbWidth:100,
thumbHeight:50,
thumbAmount:5,
hideCaptionAtLimit:0,
hideAllCaptionAtLilmit:0,
hideSliderAtLimit:0,
stopAtSlide:-1,
stopAfterLoops:-1,
fullWidth:"on"
    };
</script>
<!-- SLIDESHOW EOF -->
</div>
<!-- Content Area -->
<div id="homepageContent" style="
background-color: honeydew;
">
            <div class="wrapper">
<div class="homepageLeft">
<div class="content">
                        <p>PRIVACY-PRESERVING MEDICAL TREATMENT SYSTEM THROUGH NONDETERMINISTIC FINITE AUTOMATA<b>
                        <div class="clearfloat"></div><br><br>
                                    <div class="readmorebtn"><a href="about.php" title="read more"></a></div>
</div>
</div>
<div class="homepageRight">
                                    <div class="stent">
                                    </div>
<div class="whatsnewBlock">
            <div class="innerholder">
```

```
	<div class="head">Admin Home Page</div>
<ul>
	<li title=" Total Knee Replacement implant pricing as per NPPA guidelines">
	<a href="AddDr.jsp">Add Doctor</a>
</li>
	<li title="Coronary Stent pricing as per DPCO & NPPA guidelines">
	<a href="ViewDr.jsp">View Doctor Details</a>
</li>
		<div class="mask"><h2> Apollo City Center </h2><a
href="apollo_city_center.php" class="info">Read More</a></div>
	</div>
<div class="view view-first">
		<imgsrc="images/apollo_pharmacy.png" />
		<div class="mask"><h2> Apollo Pharmacy </h2><a
href="apollo_pharmacy.php" class="info">Read More</a></div>
</div>
<div class="view view-first" style="margin-right:0px;">
		<imgsrc="images/health_check.png" />
		<div class="mask"><h2> Health Check </h2><a href="health_check.php"
class="info">Read More</a></div>
</div>
</div>
</div>
<div class="homeboxbttom">
<div class="view2 view-first2">
	<imgsrc="images/find_doctor.png" />
		<div class="mask2"><h2> FIND A DOCTOR </h2><a
href="find_a_doctor.php" class="info2">Read More</a></div>
	</div>
<div class="view2 view-first2">
		<imgsrc="images/edoc.png" />
```

```
        <script type="text/javascript" src="js/jquery.bxslider.min.js"></script>
<script>
var k = jQuery.noConflict();
</script>
        <script>
        k(document).ready(function($){
k('.testimonials-slider').bxSlider({
                        slideWidth:978,
                        minSlides: 1,
                        maxSlides: 1,
                        //slideMargin:32,
                        auto: true,
                        autoControls: true,
                                        });
});
</script>
<!-- END Testimonials JS -->
<script type="text/javascript">
//<![CDATA[
optionalZipCountries = ["HK","IE","MO","PA"];
//]]>
</script>
<script type="text/javascript">//<![CDATA[
var Translator = new Translate([]);
      //]]></script><script type="text/javascript">
   //<![CDATA[
var Shopper = {};
Shopper.price_circle = 1;
Shopper.fixed_header = 1;
Shopper.totop = 1;
Shopper.responsive = 1;
Shopper.quick_view = 0;
Shopper.shopby_num = '5';
Shopper.text = {};
Shopper.text.more = 'more...';
Shopper.text.less = 'less...';
Shopper.anystretch_bg = '';
        //]]>
</script>
<!-- End Slider -->

<!-- Footer Area -->
<footer>
            <div class="innerholder">
            <div class="footerblock" style="width:19%;">
</div>
<div class="footerblock">
</div>
<div class="footerblock" style="width:16%;">
</div>
<div class="footerblock" style="width:20%;">
</div>
<div class="footerblock" style="width:17%;">
</div>
<div class="clearfloat"></div>

<div class="yellowblock">
            <div class="innerholder">
<div id="controls_left">
</div
</div>
</div>
```

```
<input type="hidden" value="" name="hidevalueurl" id="hidevalueurl" />
</footer>
<!-- Google Code for Remarketing Tag -->
<!--------------------------------------------------
Remarketing tags may not be associated with personally identifiable information or placed on pages related to sensitive categories. See more information and instructions on how to setup the tag on: http://google.com/ads/remarketingsetup
--------------------------------------------------->
<script type="text/javascript">
/* <![CDATA[ */
vargoogle_conversion_id = 878261851;
vargoogle_custom_params = window.google_tag_params;
vargoogle_remarketing_only = true;
/* ]]> */
</script>
<script type="text/javascript" src="//www.googleadservices.com/pagead/conversion.js">
</script>
<noscript>
<div style="display:inline;">
<img height="1" width="1" style="border-style:none;" alt="" src="//googleads.g.doubleclick.net/pagead/viewthroughconversion/878261851/?guid=ON&amp;script=0"/>
</div>
</noscript>
</body>
</html>
```

CHAPTER -7:PROJECT 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.

7.1 TYPES OF TESTS

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.

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.

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.

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.

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.

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.

7.2 Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

7.3 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.

7.4 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

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

Acceptance Testing

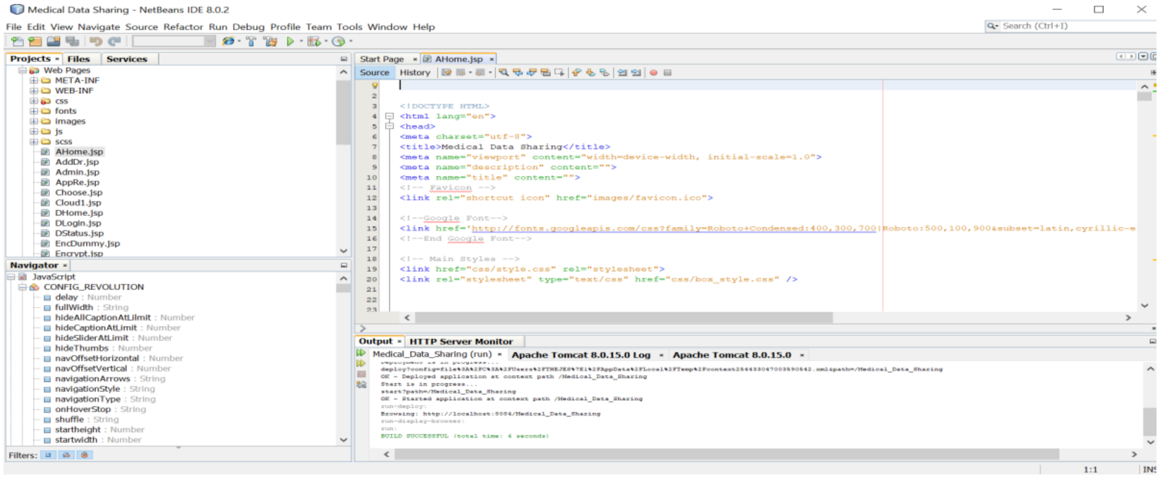
User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

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

CHAPTER – 8: OUTPUT SCREENS

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.

8.1 HOME:



```

<!DOCTYPE HTML>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Medical Data Sharing</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="description" content="">
<meta name="title" content="">
<!-- Favicon -->
<link rel="shortcut icon" href="images/favicon.ico">

<!--Google Font-->
<link href='http://fonts.googleapis.com/css?family=Roboto+Condensed:400,300,700|Roboto:500,100,900&subset=latin,cyrillic-e
<!--End Google Font-->

<!-- Main Styles -->
<link href="css/style.css" rel="stylesheet">
<link rel="stylesheet" type="text/css" href="css/box_style.css" />
```

```
deploy?config=file%3A%2FC%3A%2FUsers%2FTHEJES%7E1%2FAppData%2FLocal%2FTemp%2Fcontext2544330470035590842.xml&path=/Medical_Data_Sharing
OK - Deployed application at context path /Medical_Data_Sharing
Start is in progress...
start?path=/Medical_Data_Sharing
OK - Started application at context path /Medical_Data_Sharing
run-deploy:
Browsing: http://localhost:8084/Medical_Data_Sharing
run-display-browser:
run:
BUILD SUCCESSFUL (total time: 4 seconds)
```

Fig 8.1 Home Evaluation

In above screen we can see home evaluation which is in AHOME.JSP

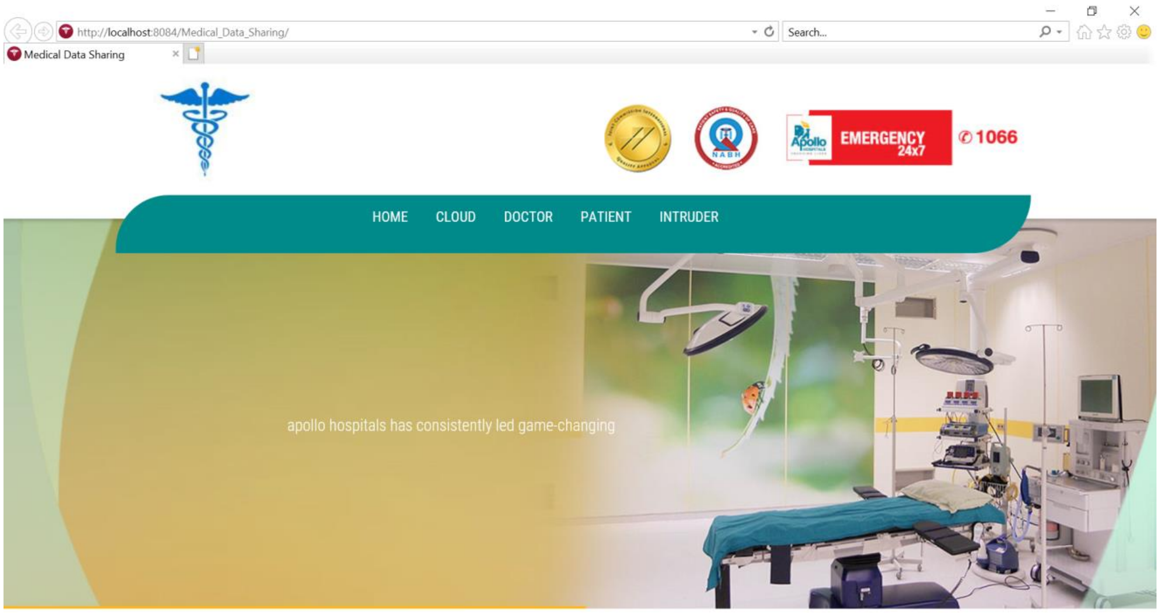
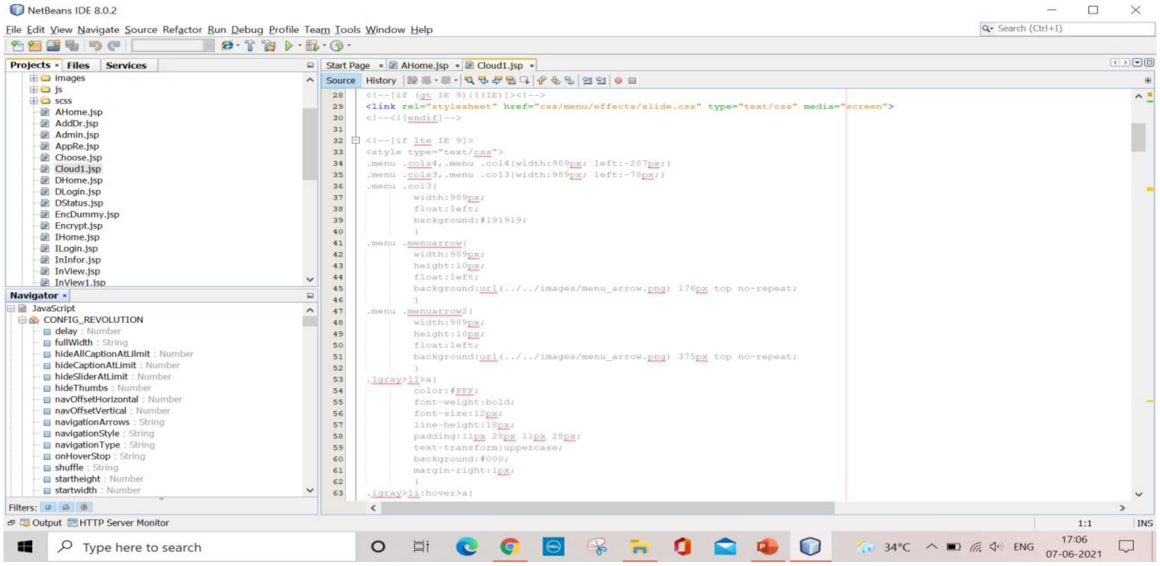

http://localhost:8084/Medical_Data_Sharing/
Search...
Medical Data Sharing
EMERGENCY 24x7
1066
HOME
CLOUD
DOCTOR
PATIENT
INTRUDER
apollo hospitals has consistently led game-changing

Fig 8.1.2 Home page

In above screen we can see the output of home evaluation i.e, Home page in this we have some operations such as HOME,CLOUD,DOCTOR,PATIENT,INTRUDER these are the modules in this project

8.2 CLOUD:



```
<!--[if (gt IE 9)|!(IE)]><!-->
<link rel="stylesheet" href="css/menu/effects/slide.css" type="text/css" media="screen">
<!--<![endif]-->

<!--[if lte IE 9]>
<style type="text/css">
.menu .cols4,.menu .col4{width:989px; left:-287px;}
.menu .cols3,.menu .col3{width:989px; left:-78px;}
.menu .col3{
        width:989px;
        float:left;
        background:#191919;
        }
.menu .menuarrow{
        width:989px;
        height:10px;
        float:left;
        background:url(../../images/menu_arrow.png) 176px top no-repeat;
        }
.menu .menuarrow2{
        width:989px;
        height:10px;
        float:left;
        background:url(../../images/menu_arrow.png) 375px top no-repeat;
        }
.lgray>li>a{
        color:#FFF;
        font-weight:bold;
        font-size:12px;
        line-height:18px;
        padding:11px 28px 11px 28px;
        text-transform:uppercase;
        background:#000;
        margin-right:1px;
        }
.lgray>li:hover>a{
```

Fig 8.2.1 cloud-let evaluation

In above screen we can see Cloud-let evaluation process this is in the process of cloud1.jsp through this we get a cloud-let page as shown below

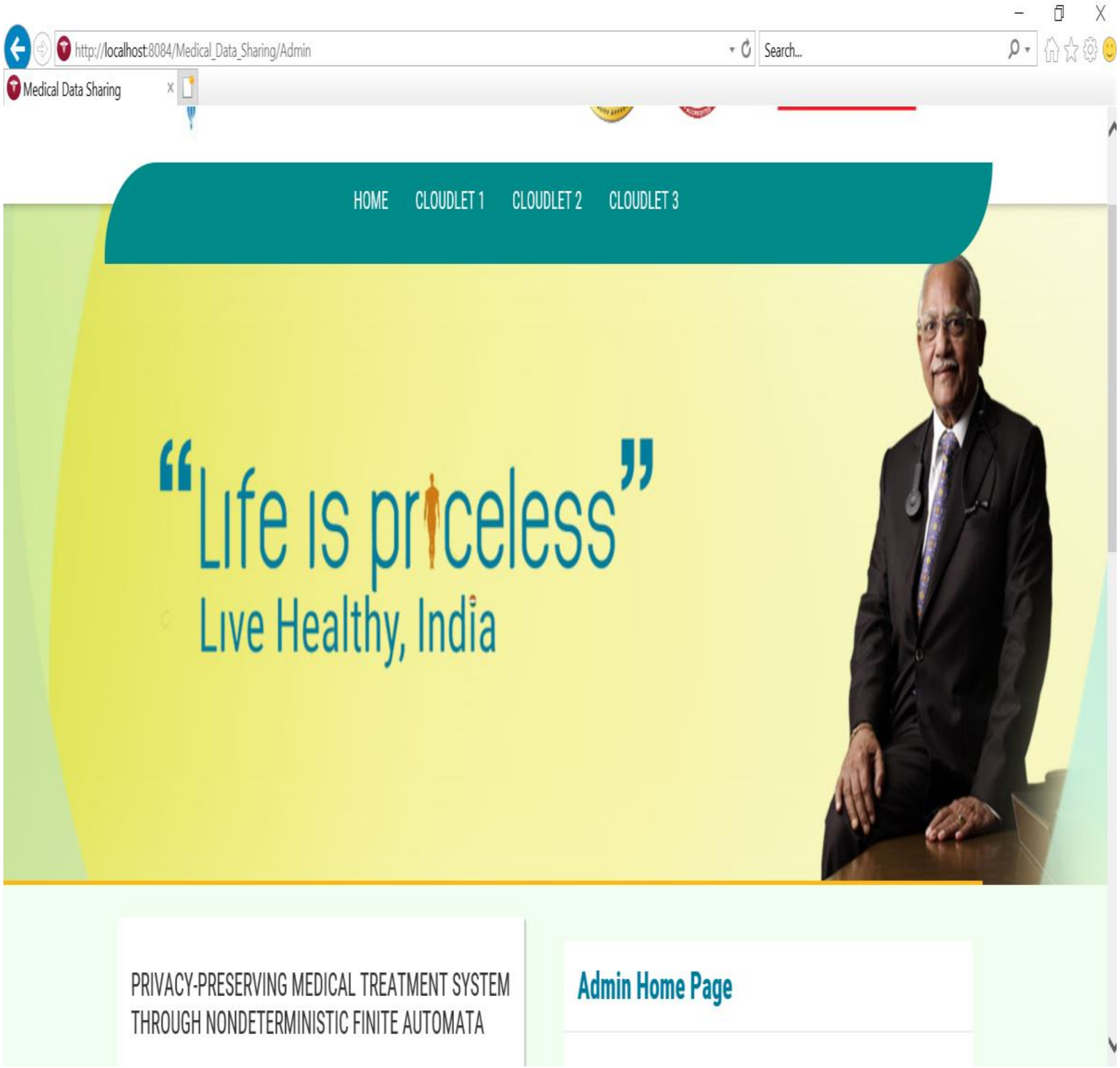
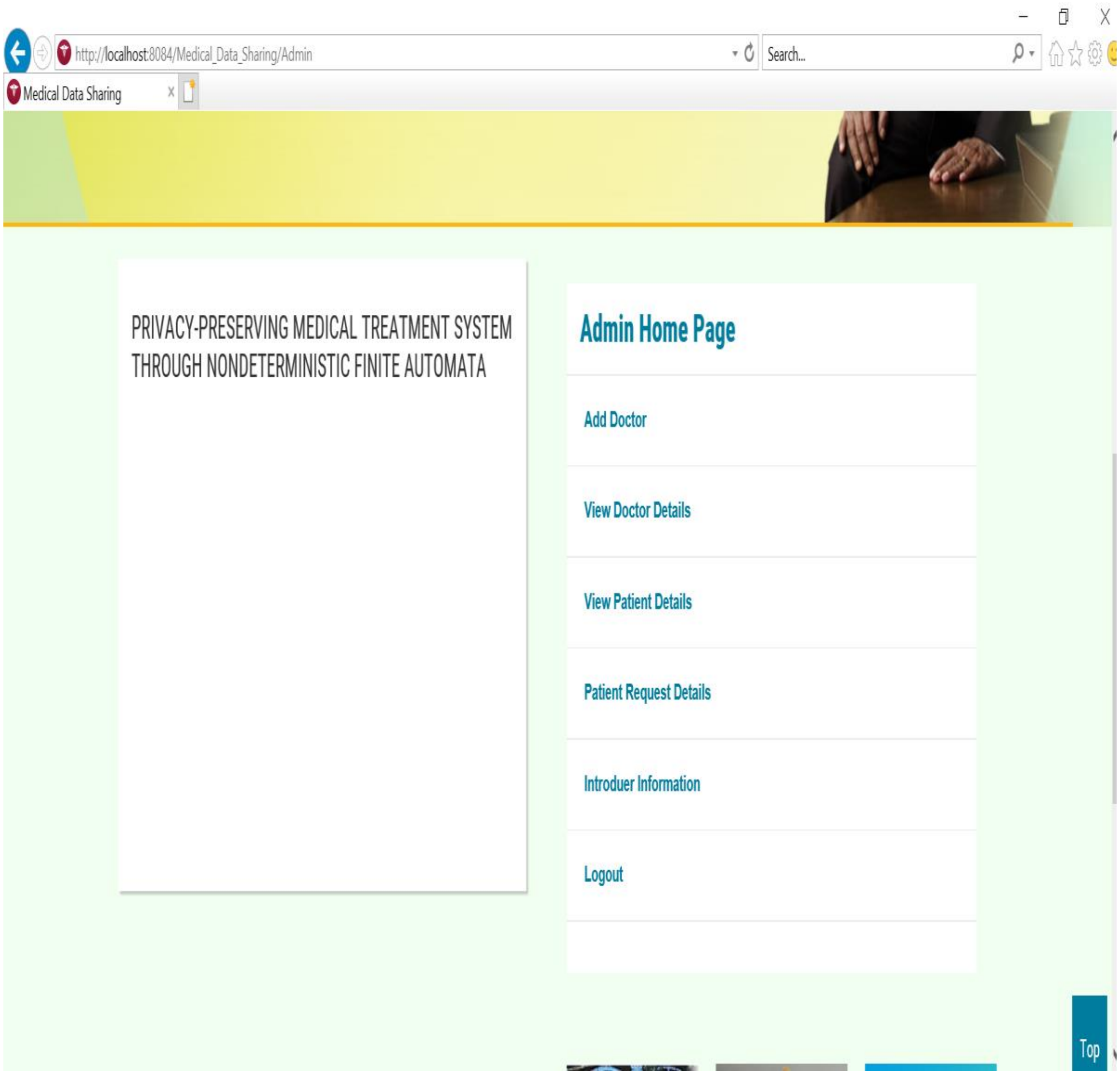

http://localhost:8084/Medical_Data_Sharing/Admin
Search...
Medical Data Sharing
HOME
CLOUDLET 1
CLOUDLET 2
CLOUDLET 3
"Life is priceless"
Live Healthy, India
PRIVACY-PRESERVING MEDICAL TREATMENT SYSTEM THROUGH NONDETERMINISTIC FINITE AUTOMATA
Admin Home Page

Fig 8.2.2 cloud-let page

In above screen we can see the cloud page in this we have cloudlet1,cloudlet2, and cloudlet3



PRIVACY-PRESERVING MEDICAL TREATMENT SYSTEM THROUGH NONDETERMINISTIC FINITE AUTOMATA

Admin Home Page

Add Doctor

View Doctor Details

View Patient Details

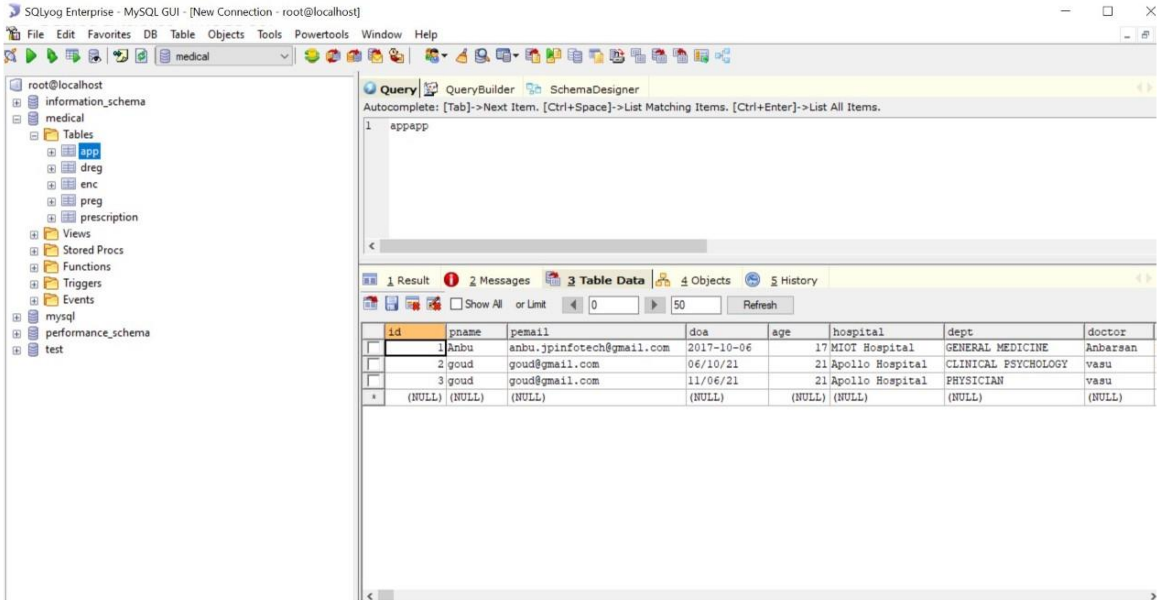
Patient Request Details

Introduer Information

Logout

Fig 8.2.3 Admin home page

The above screen is a admin home page in this admin can do perform some operations like add doctor, view doctor details, view patient details, patient request details, intruder information and logout



Autocomplete: [Tab]->Next Item. [Ctrl+Space]->List Matching Items. [Ctrl+Enter]->List All Items.

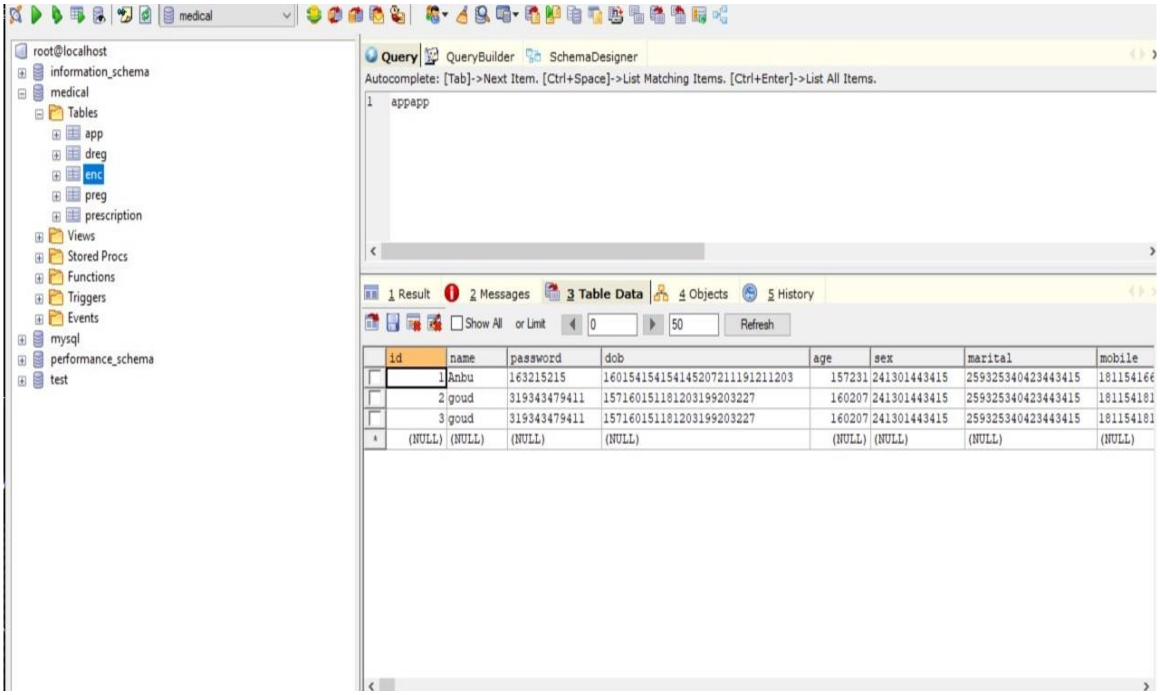
```
1  appapp
```

☐ Show All or Limit 0 50

id	pname	pemail	doa	age	hospital	dept	doctor
1	Anbu	anbu.jpinfotech@gmail.com	2017-10-06	17	MIOT Hospital	GENERAL MEDICINE	Anbarsan
2	goud	goud@gmail.com	06/10/21	21	Apollo Hospital	CLINICAL PSYCHOLOGY	vasu
3	goud	goud@gmail.com	11/06/21	21	Apollo Hospital	PHYSICIAN	vasu
(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

8.2.4TABLEDOCTOR DETAILS

the above table represent the patient details .when a patient fill the appointment request the details will be stored in this format.



Autocomplete: [Tab]->Next Item. [Ctrl+Space]->List Matching Items. [Ctrl+Enter]->List All Items.

```
1  appapp
```

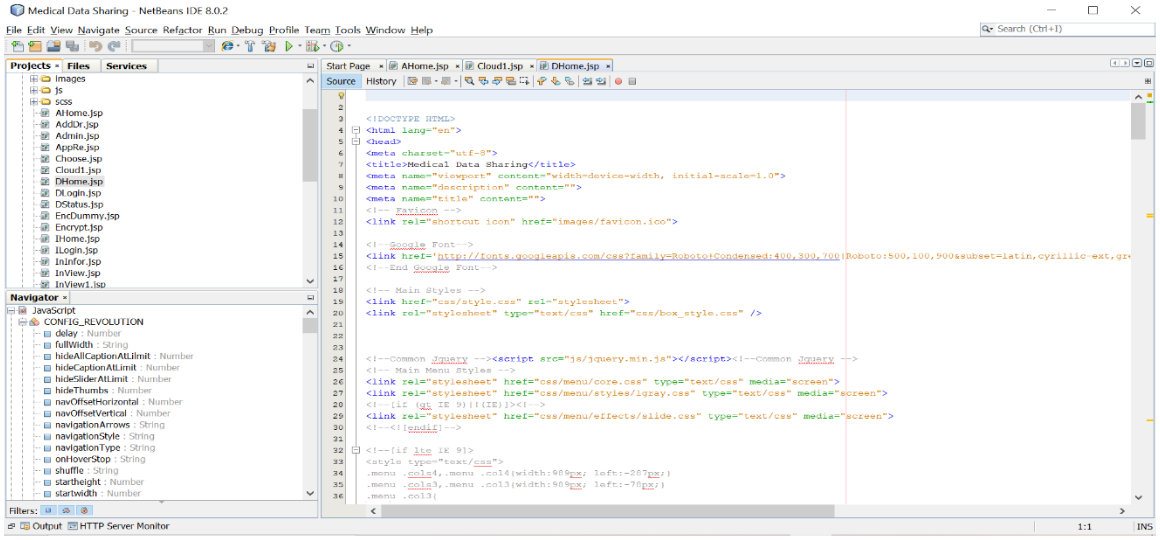
☐ Show All or Limit 0 50

id	name	password	dob	age	sex	marital	mobile
1	Anbu	163215215	16015415415414520721119121l203	157231	241301443415	259325340423443415	181154166
2	goud	319343479411	157160151181203199203227	160207	241301443415	259325340423443415	181154181
3	goud	319343479411	157160151181203199203227	160207	241301443415	259325340423443415	181154181
(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

8.2.5 ENCRYPTED TABLE

The above table is a patient details which are in the form of encrypted to make the patient details safe and secure.

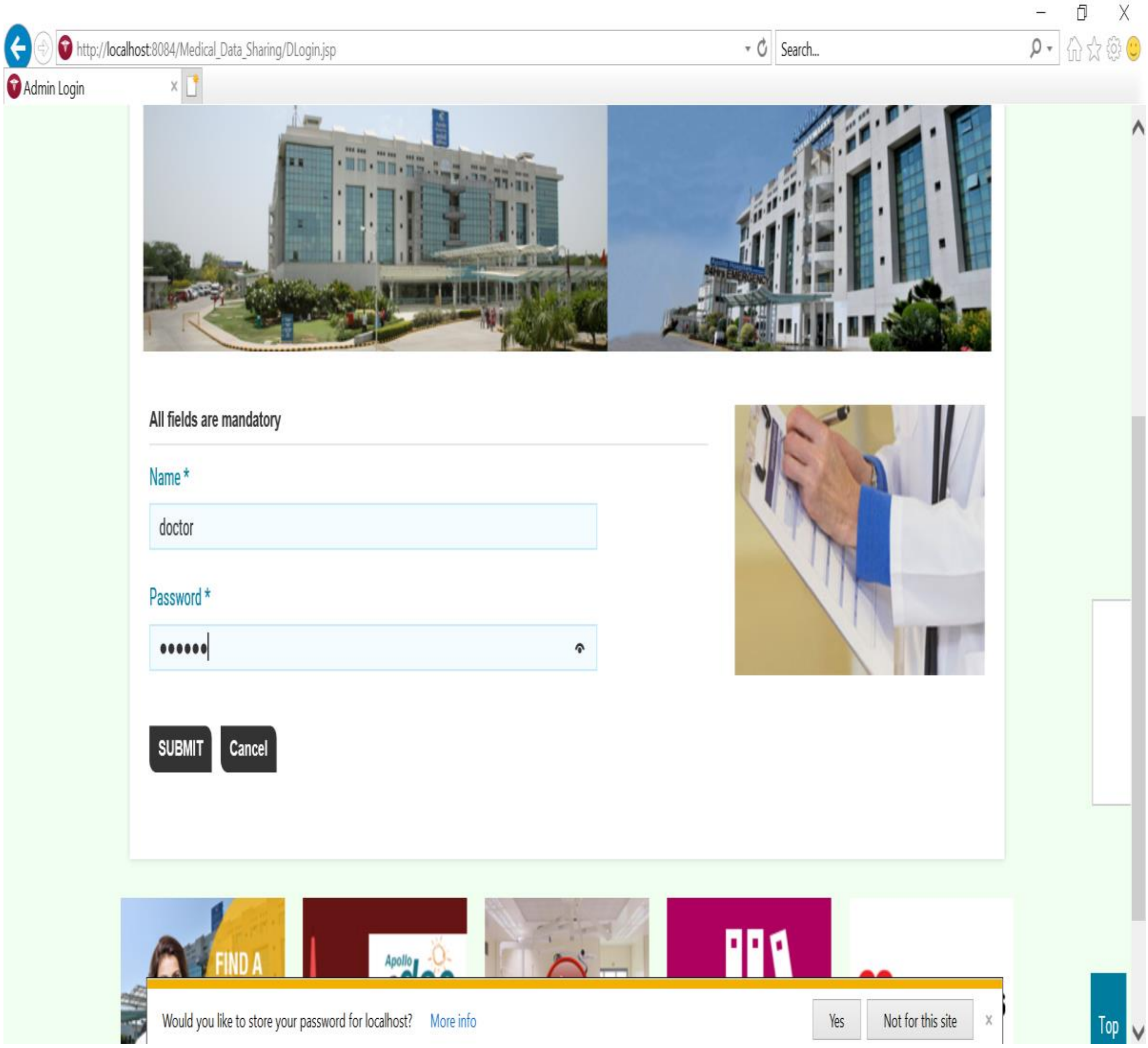
8.3 DOCTOR:



```


<!DOCTYPE HTML>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Medical Data Sharing</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="description" content="">
<meta name="title" content="">
<!-- Favicon -->
<link rel="shortcut icon" href="images/favicon.ico">

<!--Google Font-->
<link href='http://fonts.googleapis.com/css?family=Roboto+Condensed:400,300,700|Roboto:500,100,900&subset=latin,cyrillic-ext,gre
<!--End Google Font-->

<!-- Main Styles -->
<link href="css/style.css" rel="stylesheet">
<link rel="stylesheet" type="text/css" href="css/box_style.css" />



<!--Common Jquery --><script src="js/jquery.min.js"></script><!--Common Jquery -->
<!-- Main Menu Styles -->
<link rel="stylesheet" href="css/menu/core.css" type="text/css" media="screen">
<link rel="stylesheet" href="css/menu/styles/lgray.css" type="text/css" media="screen">
<!--[if (gt IE 9)|!(IE)]><!-->
<link rel="stylesheet" href="css/menu/effects/slide.css" type="text/css" media="screen">
<!--<![endif]-->

<!--[if lte IE 9]>
<style type="text/css">
.menu .cols4,.menu .col4{width:989px; left:-287px;}
.menu .cols3,.menu .col3{width:989px; left:-78px;}
.menu .col3{
```

Fig 8.3.1Doctor Evaluation

In above screen we can see the doctor evaluation process which is in the form of Dhome.jsp



All fields are mandatory

Name *

doctor

Password *

••••••

Would you like to store your password for localhost?

Fig 8.3.2Doctor login page

In above screen we can see Doctor login page in this we have all mandatory fields such as name and password .we can login through entering an authorized name and password in this page

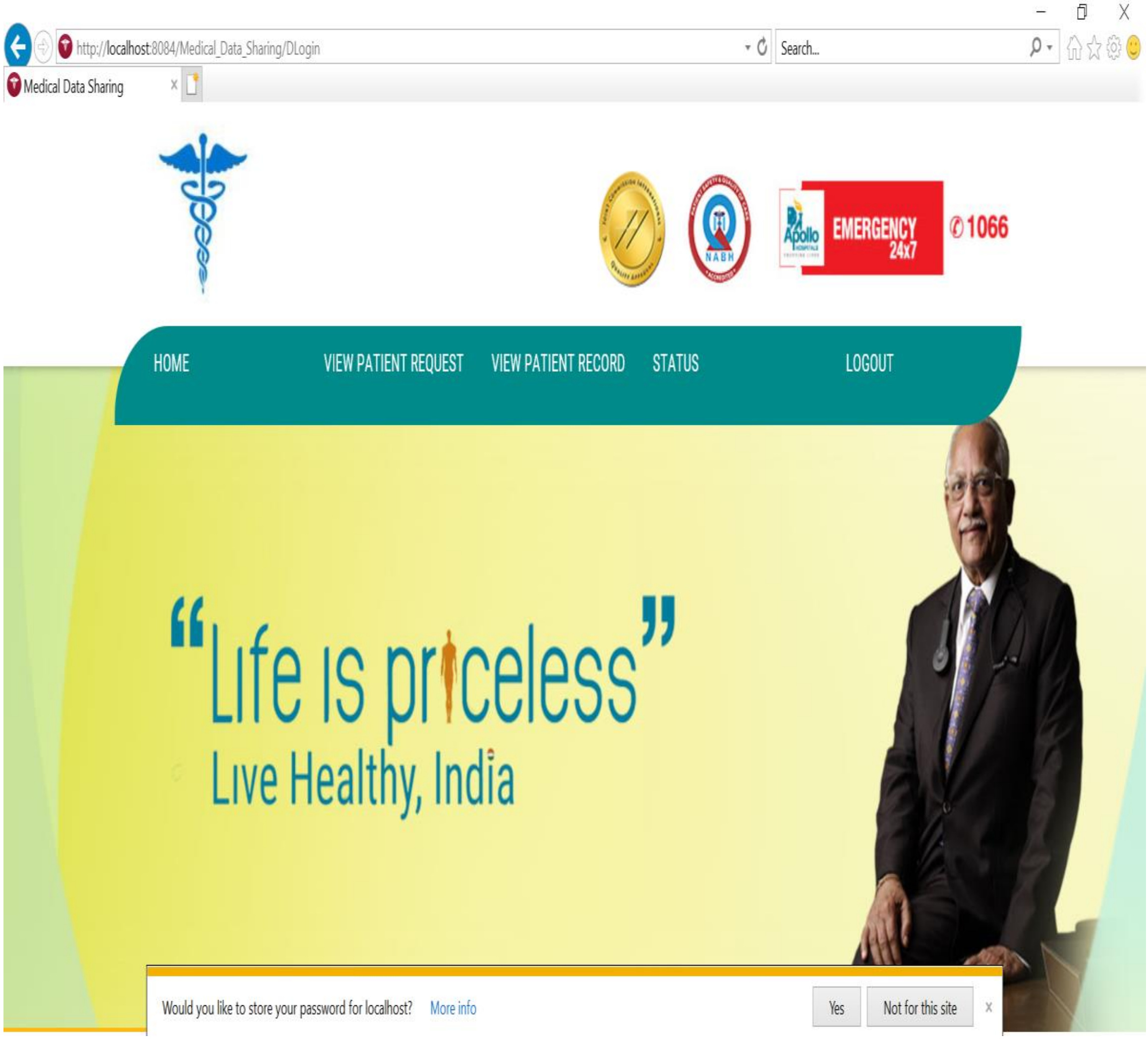
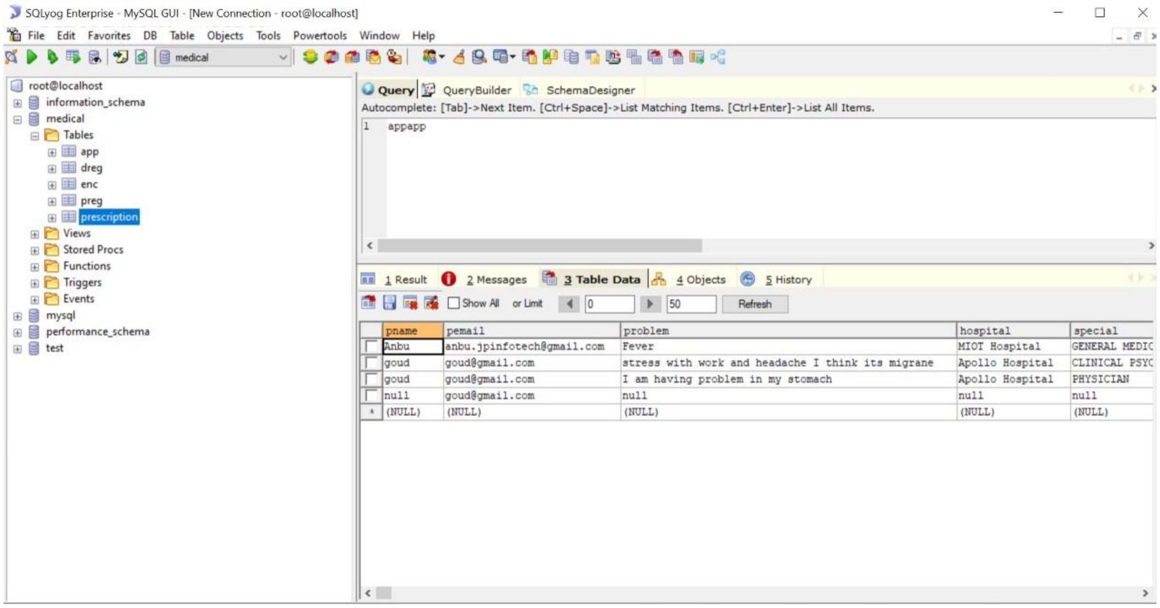


Fig 8.3.3Doctor home page

the above screen is a doctor home page after login success we come to this doctor page in this doctor can perform some operations like view patient, view patient record, view status and logout.



Autocomplete: [Tab]->Next Item. [Ctrl+Space]->List Matching Items. [Ctrl+Enter]->List All Items.

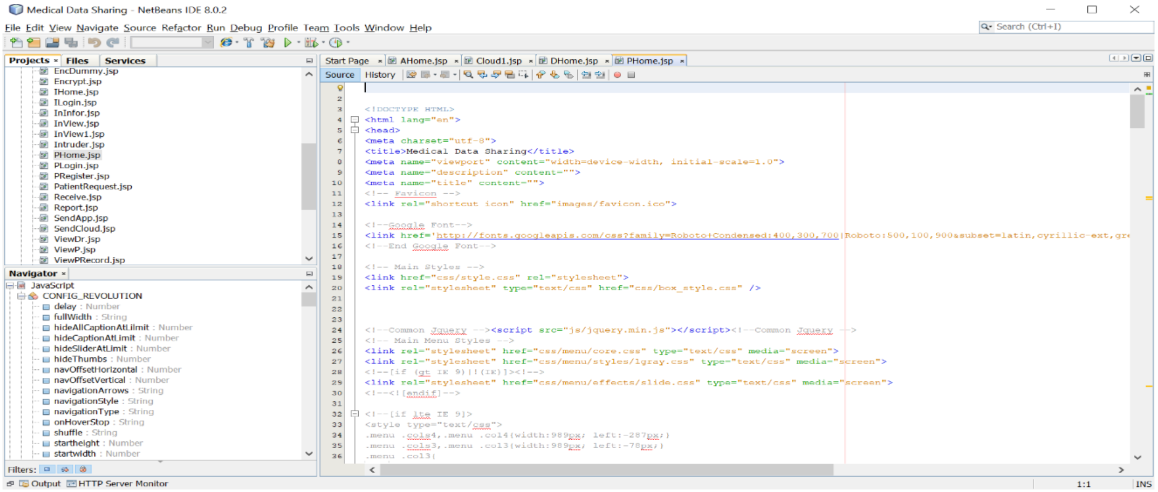
```
appapp
```

pname	pemail	problem	hospital	special
Anbu	anbu.jpinfotech@gmail.com	Fever	MIOT Hospital	GENERAL MEDIC
goud	goud@gmail.com	stress with work and headache I think its migrane	Apollo Hospital	CLINICAL PSYC
goud	goud@gmail.com	I am having problem in my stomach	Apollo Hospital	PHYSICIAN
null	goud@gmail.com	null	null	null
(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

8.3.4 PRESCRIPTION TABLE

the above table show the doctor prescription given to the patient.this will be sent to patient from doctor according to the patient record

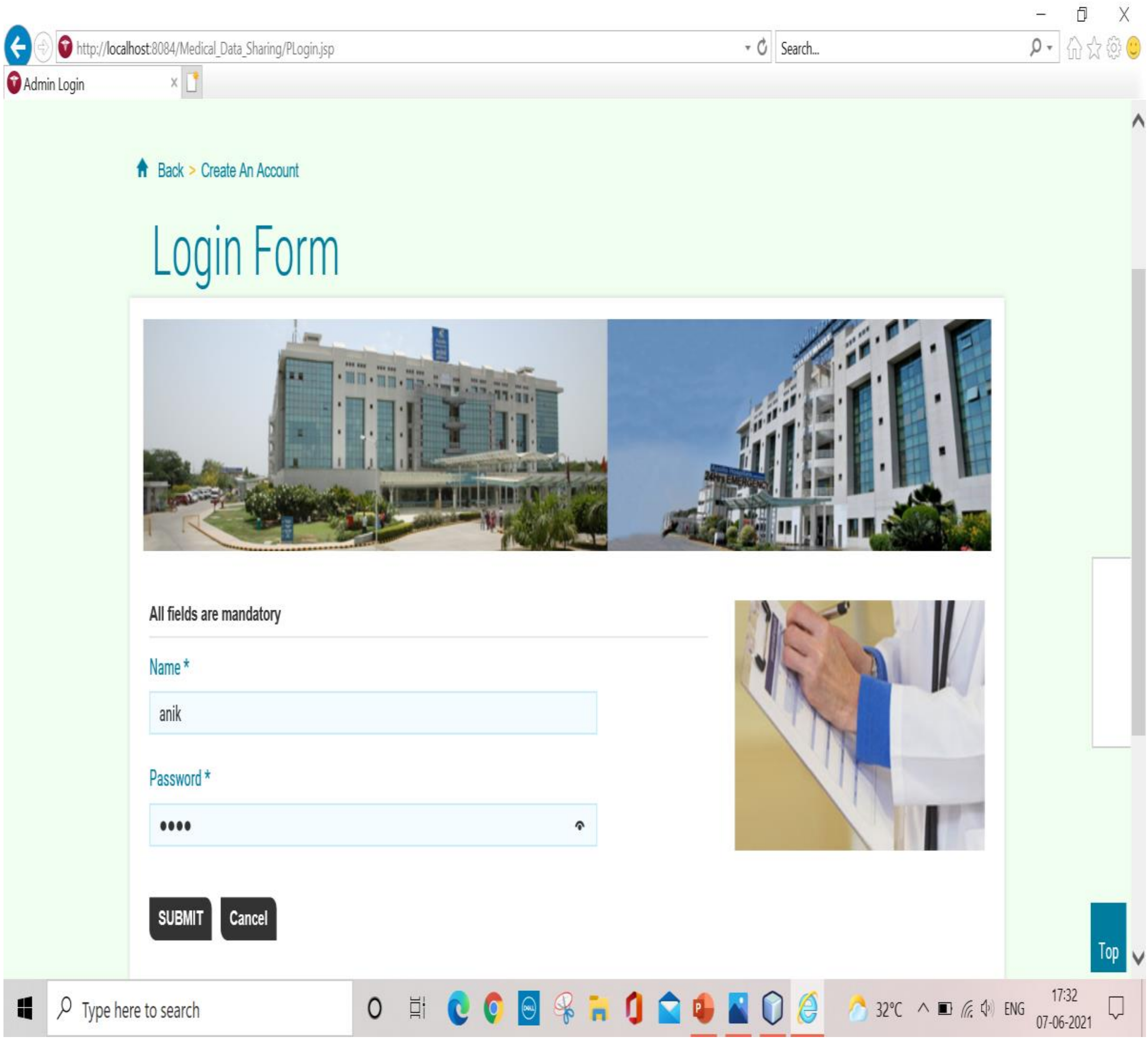
8.4 PATIENT:



```


<!DOCTYPE HTML>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Medical Data Sharing</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="description" content="">
<meta name="title" content="">
<!-- Favicon -->
<link rel="shortcut icon" href="images/favicon.ico">

<!--Google Font-->
<link href='http://fonts.googleapis.com/css?family=Roboto+Condensed:400,300,700|Roboto:500,100,900&subset=latin,cyrillic-ext,gre
<!--End Google Font-->

<!-- Main Styles -->
<link href="css/style.css" rel="stylesheet">
<link rel="stylesheet" type="text/css" href="css/box_style.css" />



<!--Common Jquery --><script src="js/jquery.min.js"></script><!--Common Jquery -->
<!-- Main Menu Styles -->
<link rel="stylesheet" href="css/menu/core.css" type="text/css" media="screen">
<link rel="stylesheet" href="css/menu/styles/lgray.css" type="text/css" media="screen">
<!--[if (gt IE 9)|!(IE)]><!-->
<link rel="stylesheet" href="css/menu/effects/slide.css" type="text/css" media="screen">
<!--<![endif]-->

<!--[if lte IE 9]>
<style type="text/css">
.menu .cols4,.menu .col4{width:989px; left:-287px;}
.menu .cols3,.menu .col3{width:989px; left:-78px;}
.menu .col3{
```

Fig 8.4.1 patient evaluation

In above screen we can see the patient evaluation process this is in the form of PHome .jsp this is the process of getting the patient login form



Login Form

All fields are mandatory

Name *

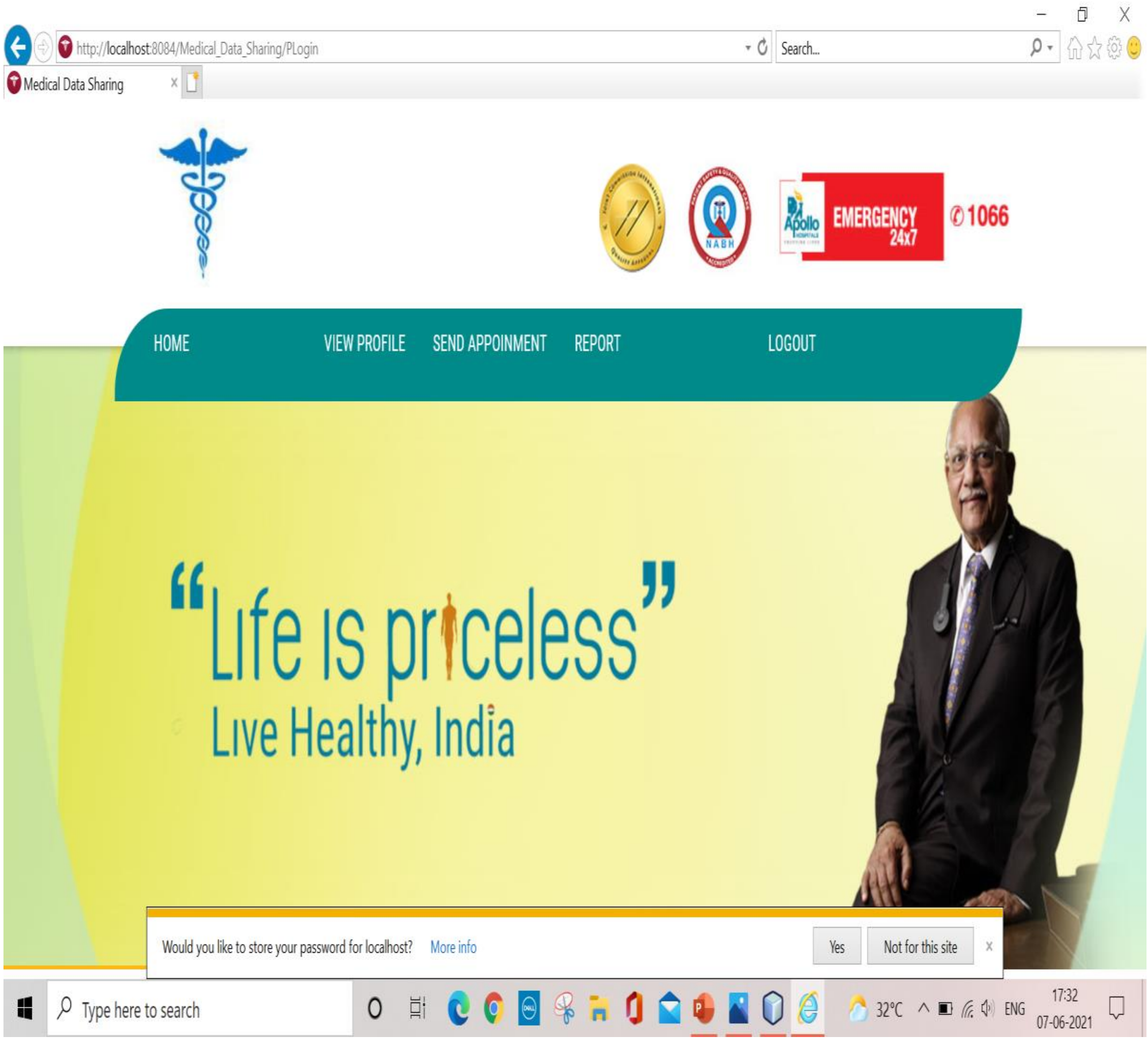
anik

Password *

••••

Fig 8.4.2 patient login form

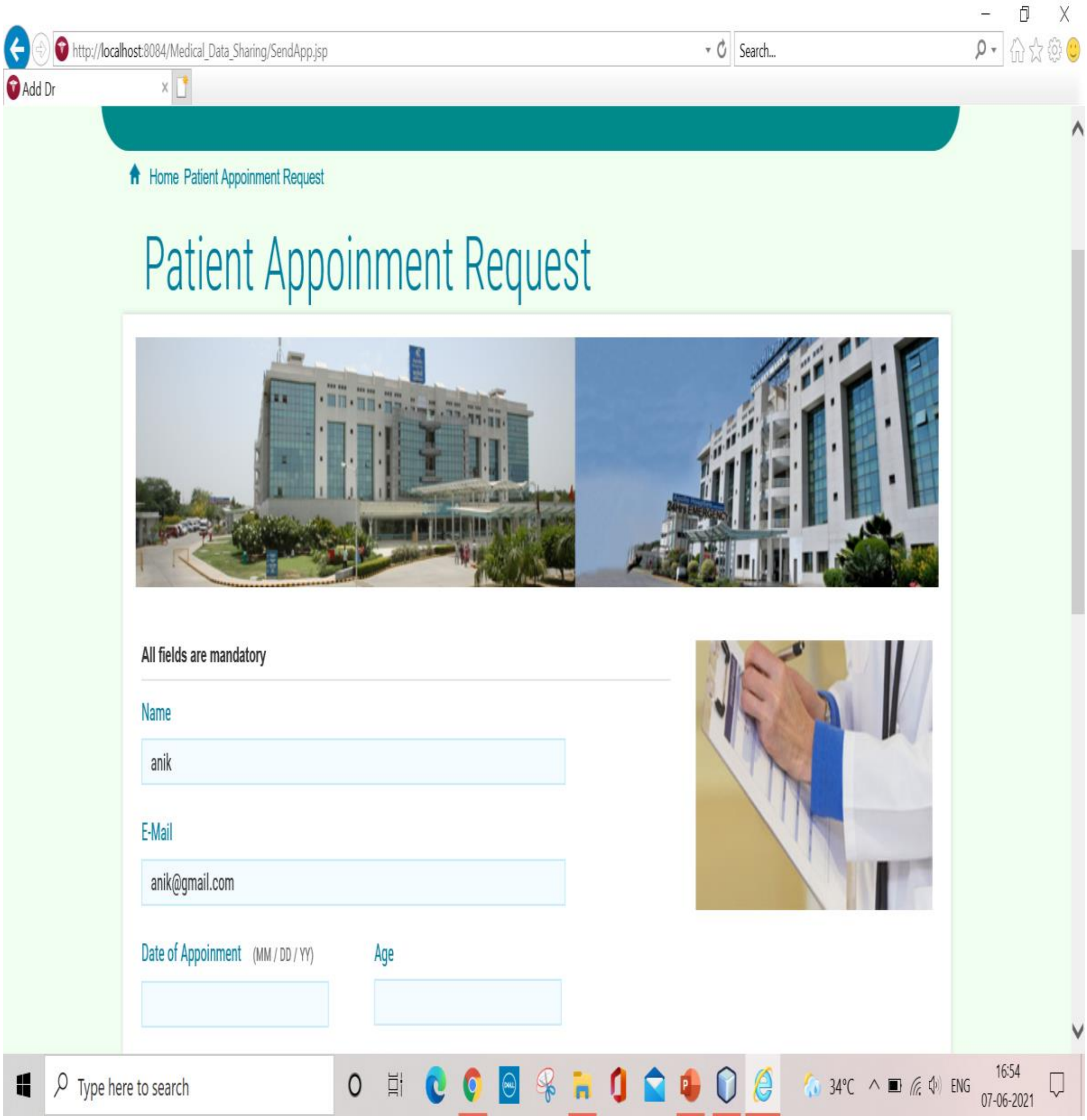
In above screen we can see patient login form in this we have all mandatory fields such as name and password, we can login through entering an authorized name and password in this page



Would you like to store your password for localhost?

Fig 8.4.3 patient home page

In above screen we have patient home page after login success we come to this page. in this patient can perform some operations like view his/her profile, send appointment request to doctor, check report and login


http://localhost:8084/Medical_Data_Sharing/SendApp.jsp
Search...
Add Dr
Home Patient Appoinment Request
Type here to search
34°C
ENG
16:54
07-06-2021

Patient Appoinment Request

All fields are mandatory

Name

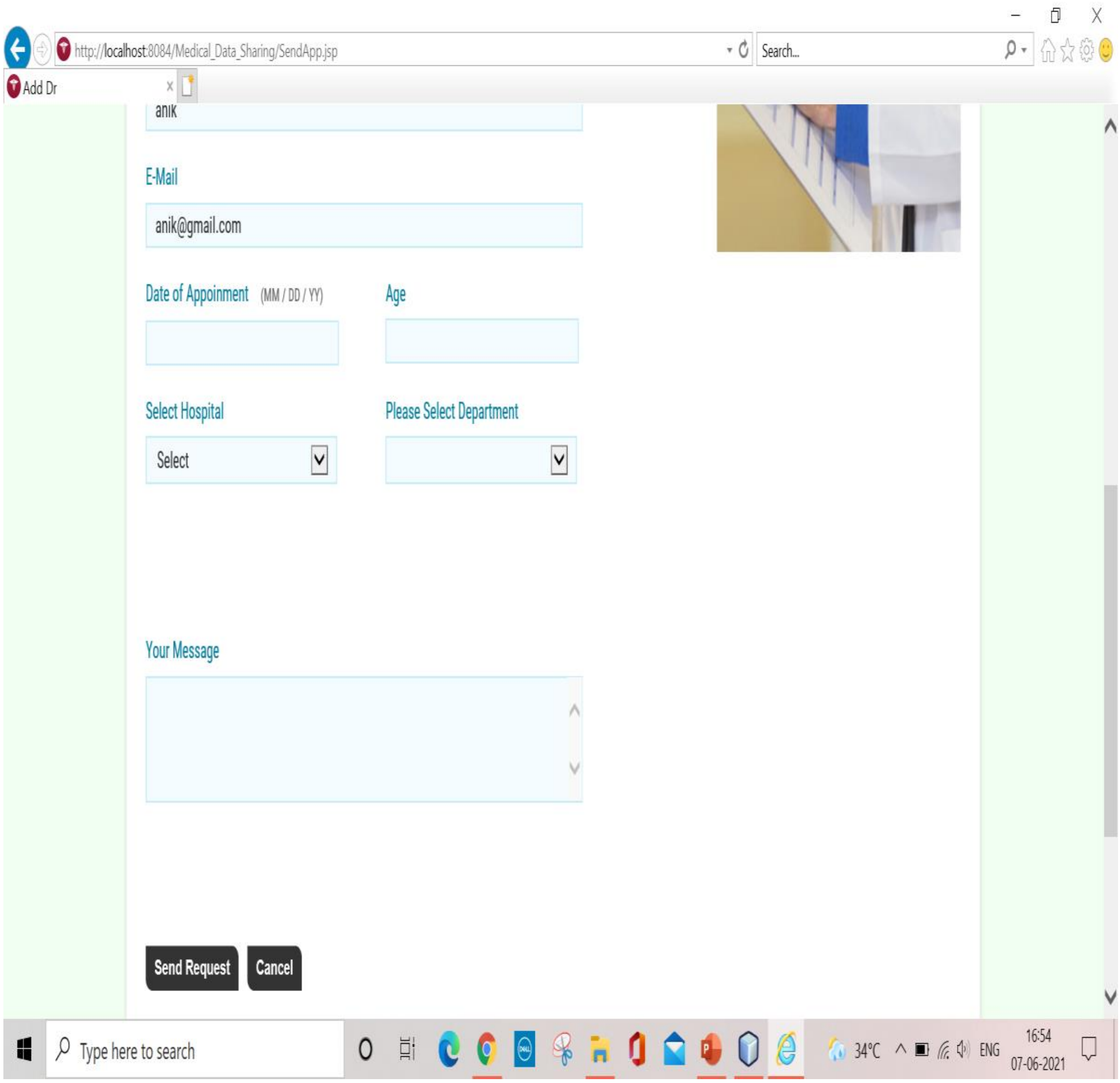
anik

E-Mail

anik@gmail.com

Date of Appoinment (MM / DD / YY)

Age



http://localhost:8084/Medical_Data_Sharing/SendApp.jsp

anik

E-Mail

anik@gmail.com

Date of Appoinment (MM / DD / YY)

Age

Select Hospital

Select

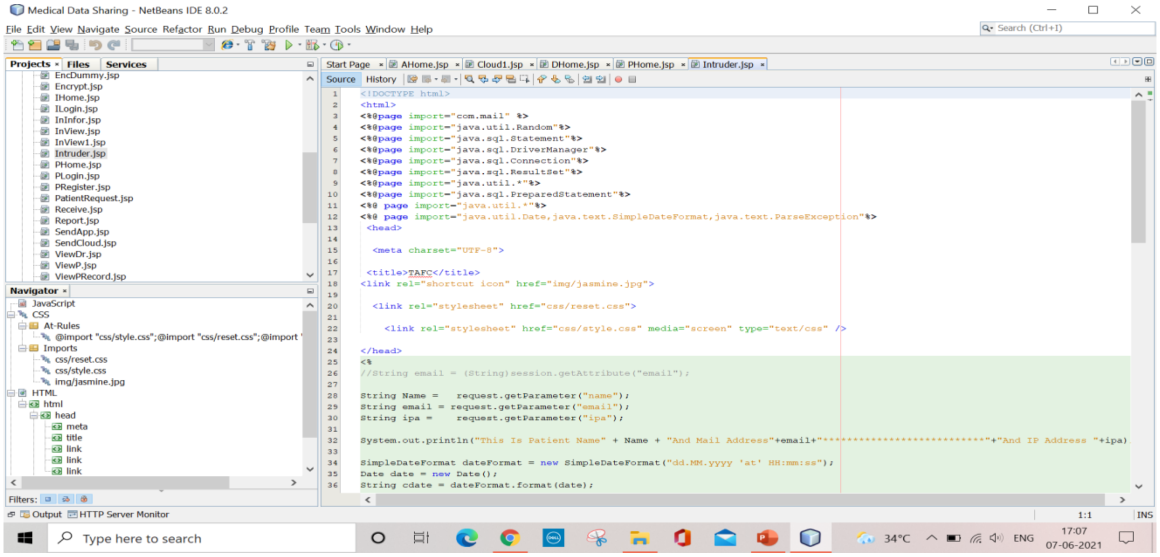
Please Select Department

Your Message

Fig 8.4.4 patient appointment request

In above screen we can see patient appointment form by filling this patient can send request to doctor. in this patient should fill all mandatory fields such as name ,email, patient age, date of appointment and patient can also select hospital, and department of doctor etc

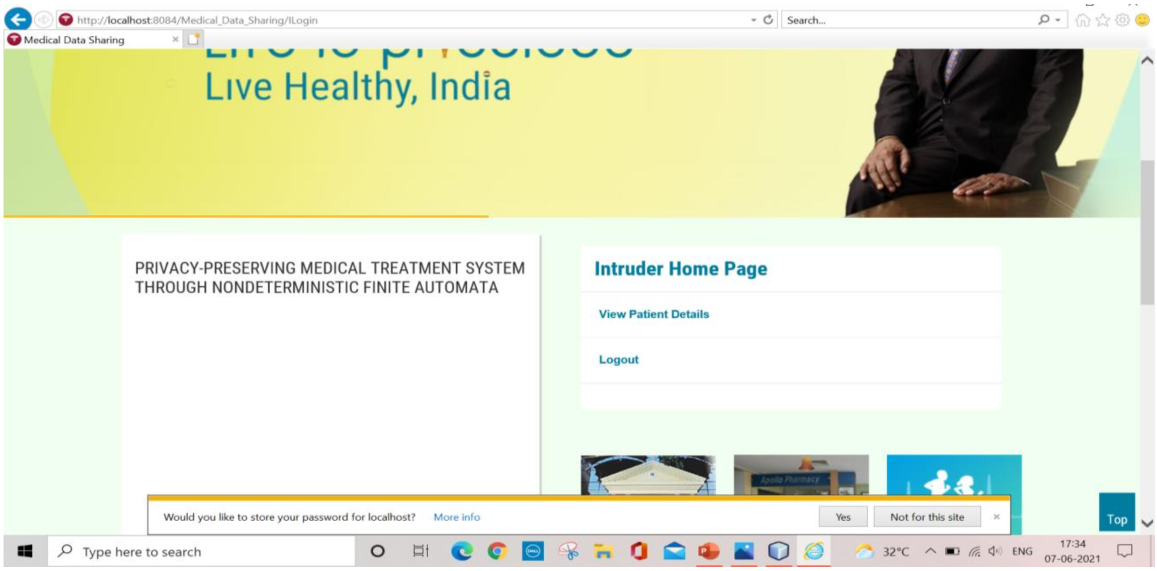
8.5 INTRUDER:



```
<!DOCTYPE html>
<html>
<%@page import="com.mail" %>
<%@page import="java.util.Random"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.DriverManager"%>
<%@page import="java.sql.Connection"%>
<%@page import="java.sql.ResultSet"%>
<%@page import="java.util.*"%>
<%@page import="java.sql.PreparedStatement"%>
<%@ page import="java.util.*"%>
<%@ page import="java.util.Date,java.text.SimpleDateFormat,java.text.ParseException"%>
 <head>

  <meta charset="UTF-8">

 <title>TAFC</title>
<link rel="shortcut icon" href="img/jasmine.jpg">

  <link rel="stylesheet" href="css/reset.css">

    <link rel="stylesheet" href="css/style.css" media="screen" type="text/css" />

</head>
<%
//String email = (String)session.getAttribute("email");

String Name =   request.getParameter("name");
String email = request.getParameter("email");
String ipa =    request.getParameter("ipa");

System.out.println("This Is Patient Name" + Name + "And Mail Address"+email+"**************************"+"And IP Address "+ipa)

SimpleDateFormat dateFormat = new SimpleDateFormat("dd.MM.yyyy 'at' HH:mm:ss");
Date date = new Date();
String cdate = dateFormat.format(date);
```

Fig 8.5.1 intruder evaluation

in the above we can see the intruder evaluation which is in the form of intruder.jsp by doing the evaluation we get the intruder home page



PRIVACY-PRESERVING MEDICAL TREATMENT SYSTEM THROUGH NONDETERMINISTIC FINITE AUTOMATA

Intruder Home Page

View Patient Details

Logout

Would you like to store your password for localhost? More info Yes Not for this site ×

Fig 8.5.2 Intruder home page

in the above screen we can see the intruder home page in this intruder can perform some operations like view patient details and logout.

CHAPTER – 9: CONCLUSION AND FUTURE ENHANCEMENT

9.1CONCLUSION

In this paper, we investigated the problem of privacy protection and sharing large medical data in cloudlets and the remote cloud. We developed a system which does not allow users to transmit data to the remote cloud in consideration of secure collection of data, as well as low communication cost. However, it does allow users to transmit data to a cloudlet, which triggers the data sharing problem in the cloudlet. Firstly, we can utilize wearable devices to collect users' data, and in order to protect users privacy, we use NTRU mechanism to make sure the transmission of users' data to cloudlet in security. Secondly, for the purpose of sharing data in the cloudlet, we use trust model to measure users' trust level to judge whether to share data or not. Thirdly, for privacy-preserving of remote cloud data, we partition the data stored in the remote cloud and encrypt the data in different ways, so as to not just ensure data protection but also accelerate the efficacy of transmission. Finally, we propose collaborative IDS based on cloudlet mesh to protect the whole system. The proposed schemes are validated with simulations and experiments.

9.2 FUTURE ENHANCEMENT

- Privacy-preserving of remote cloud data, we partition the data stored in the Remote cloud and encrypt the data in different ways, so as to not just ensure data Protection but also accelerate the efficacy of transmission.
- we propose collaborative IDS based on cloudlet mesh to protect the whole System. The proposed schemes are validated with simulations and experiments. And by adding more secure protocols there by to reduce malicious attacks.

CHAPTER -10: REFERENCES

[1] K. Hung, Y. Zhang, and B. Tai, "Wearable medical devices for tele home healthcare," in *Engineering in Medicine and Biology Society, 2004.IEMBS'04. 26th Annual International Conference of the IEEE*, vol. 2.IEEE, 2004, pp. 5384–5387.

[2] M. S. Hossain, "Cloud-supported cyber–physical localization frame work for patients monitoring," 2015.

[3] J. Zhao, L. Wang, J. Tao, J. Chen, W. Sun, R. Ranjan, J. Kołodziej,A. Streit, and D. Georgakopoulos, "A security framework in g-hadoop for big data computing across distributed cloud data centre's," *Journal of Computer and System Sciences*, vol. 80, no. 5, pp. 994–1007, 2014.

[4] M. S. Hossain and G. Muhammad, "Cloud-assisted industrial internet of things (iiot)–enabled framework for health monitoring," *Computer Networks*, vol. 101, pp. 192–202, 2016.

[5] R. Zhang and L. Liu, "Security models and requirements for healthcare application clouds," in *Cloud Computing (CLOUD), 2010 IEEE 3rdInternational Conference on*. IEEE, 2010, pp. 268–275.

[6] K. He, J. Chen, R. Du, Q. Wu, G. Xue, and X. Zhang, "Deypos: Deduplicatable dynamic proof of storage for multi-user environments,"2016.

[7] L. Griffin and E. De Leastar, "Social networking healthcare," in *Wearable Micro and Nano Technologies for Personalized Health (pHealth), 20096th International Workshop on*. IEEE, 2009, pp. 75–78.

[8] W. Xiang, G. Wang, M. Pickering, and Y. Zhang, "Big video data forlight-field-based 3d telemedicine," *IEEE Network*, vol. 30, no. 3, pp. 30–38, 2016.
[9] https://www.patientslikeme.com/.

[10] C. Zhang, J. Sun, X. Zhu, and Y. Fang, “Privacy and security for online social networks: challenges and opportunities,” *Network, IEEE*, vol. 24,no. 4, pp. 13–18, 2010.

[11] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-preserving multi-keyword ranked search over encrypted cloud data,” *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 1, pp. 222–233,2014.

[12] K. T. Pickard and M. Swan, “Big desire to share big health data: A shift in consumer attitudes toward personal health information,” in *2014 AAAI Spring Symposium Series*, 2014.

[13] T. Xu, W. Xiang, Q. Guo, and L. Mo, “Mining cloud 3d video data forinter active video services,” *Mobile Networks and Applications*, vol. 20,no. 3, pp. 320–327, 2015.

A

PROJECT REPORT

On

ADAPTIVE DIFFUSION OF SENSITIVE INFORMATION IN ONLINE SOCIAL NETWORKS

Submitted by

Ms. S Nishitha	(17K81A1238)
Ms. Pooja Singh	(17K81A1244)
Mr. M Sannith	(17K81A1236)
Mr.Abhishek Jena	(17K81A1201)

in the partial fulfillment for the award of

degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

MS. G PRIYA

ASSISTANT PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST. MARTIN’S ENGINEERING COLLEGE

(An Autonomous Institute)

Dhulapally, Secunderabad – 500 100

June 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **ENHANCEMENT OF VEHICLE SPEED DETECTION**”, is being submitted by **S NISHITHA(17K81A1238), POOJA SINGH(17K81A1244),M SANNITH(17K81A1236), ABHISHEK JENA(17K81A1201)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY** is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Ms. G PRIYA
Department of IT

Head of the Department
Dr. R. NAGARAJU
Department of IT

Internal Examiner

External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of **Information Technology**, session: 2017-2021, St. Martin’s Engineering College, Dhulapally, Kompally, Secundrabad, hereby declare that work presented in this Project Work entitled **ADAPTIVE DIFFUSION OF SESNSITIVE INFORMATION IN ONLINE SOCIAL METWORKS** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

Ms. S Nishitha	**17K81A1238**
Ms. Pooja Singh	**17K81A1244**
Mr. M Sannith	**17K81A1236**
Mr. Abhishek Jena	**17K81A1201**

LASYA
INFOTECH

ST. MARTIN'S ENGINEERING COLLEGE
TECHNOLOGY FOR PROSPERITY
UGC AUTONOMOUS

TUESDAY, 15 JUNE 2021

INTERNSHIP CERTIFICATE

THIS IS TO CERTIFY THAT **ABHISHEK JENA** WITH ROLL NO.**17K81A1201, M.SANNITH** WITH ROLL NO.**17K81A1236**, **POOJA SINGH** WITH ROLL NO.**17K81A1244**, **S.NISHITHA** WITH ROLL NO.**17K81A1238**, OF B.TECH – IV YEAR, **INFORMATION TECHNOLOGY DEPARTMENT** OF **ST. MARTIN'S ENGINEERING COLLEGE**, KOMPALLY, SECUNDERABAD HAVE COMPLETED ONE MONTH INTERNSHIP PROGRAM AT **LASYA IT SOLUTION PVT. LTD, KOMPALLY.**

DURING THE PERIOD, THEY HAVE SUCCESSFULLY COMPLETED MAJOR PROJECT TITLED "**ADAPTIVE DIFFUSION OF SENSITIVE INFORMATION IN ONLINE SOCIAL NETWORKS**" AT OUR DEVELOPMENT CENTER, KOMPALLY.

WE WISH THEM SUCCESS IN THEIR FUTURE ENDEVOUR.

Lasya IT Solutions Pvt. Ltd.
★ Kompally-500014. ★

ORUGANTI VENKAT
DIRECTOR
TRAININGS & PLACEMENTS
LASYA IT SOLUTIONS PVT LTD.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin’s Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. R. NAGARAJU**, Head of the Department, **DEPARTMENT OF INFORMATION TECHNOLOGY**, St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **MR. D. BABU RAO**, Associate Professor, in Department of Information Technology, for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **MS. G PRIYA**, Department of Information Technology, St. Martin’s Engineering College, Dhulapally, for his support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

S NISHITHA	**17K81A1238**
POOJA SINGH	**17K81A1244**
M SANNITH	**17K81A1236**
ABHISHEK JENA	**17K81A1201**

TABLE OF CONTENTS

ABSTRACT

The cascading of sensitive information such as private contents and rumors is a severe issue in online social networks. One approach for limiting the cascading of sensitive information is constraining the diffusion among social network users. However, the diffusion constraining measures limit the diffusion of non-sensitive information diffusion as well, resulting in the bad user experiences. To tackle this issue, in this paper, we study the problem of how to minimize the sensitive information diffusion while preserve the diffusion of non-sensitive information, and formulate it as a constrained minimization problem where we characterize the intention of preserving non-sensitive information diffusion as the constraint. We study the problem of interest over the fully known network with known diffusion abilities of all users and the semi-known network where diffusion abilities of partial users remain unknown in advance. By modeling the sensitive information diffusion size as the reward of a bandit, we utilize the bandit framework to jointly design the solutions with polynomial complexity in both scenarios. Moreover, the unknown diffusion abilities over the semi-known network induce it difficult to quantify the information diffusion size in algorithm design. For this issue, we propose to learn the unknown diffusion abilities from the diffusion process in real time and then adaptively conduct the diffusion constraining measures based on the learned diffusion abilities, relying on the bandit framework. Extensive experiments on real and synthetic datasets demonstrate that our solutions can effectively constrain the sensitive information diffusion and enjoy a 40% less diffusion loss of non-sensitive information comparing with four baseline algorithms.

LIST OF FIGURES

LIST OF OUTPUT SCREENS

LIST OF ABBREVATIONS

- **SYSTEM ARCHITECHTURE**
- **APACHE-TOMCAT SERVER**
- **COMPILING AND INTERPREATION OF JAVA SOURCE CODE**
- **PICTURE SHOWING THE DEVELOPMENT OF JAVA SOURCE CODE**
- **TWO-TIER MODEL**
- **TWO-TIER MODEL**

CHAPTER 1: INTRODUCTION

1.1 MOTIVATION

The sensitive information refers to any kind of information that needs to be prohibited from cascading such as rumors, personal contents, and trade secrets. The cascading of such sensitive information may cause the risk of leaking users' privacy or arising panics among public. With this concern, several social network medias (e.g., Facebook, Twitter) have claimed authorities to block accounts of users and delete some posts or tweets when they violate relevant rules about privacy or security. Thus network managers are able to take measures to prohibit the cascading of sensitive information. The existing attempts that share the closest correlation with prohibiting sensitive information diffusion belong to the rumor influence minimization, whose current strategies can mainly be classified into two aspects. The first is diffusing the truths over network to counteract rumors. However, diffusing truths is only suitable for constraining the rumors, while is not suitable for constraining the diffusion of the other kinds of sensitive information, including personal information, trade secrets, and etc.

1.2 PROBLEM DEFINITION

We assume network managers know the diffusion abilities of all users. The examples for the fully-known network lie on the social networks for enterprises (e.g., Skype) or special interest groups (SIGs) (e.g., Douban1). As the full topology of a local social network, which consists of the staff of a same enterprise or the members in a same SIG, is available to network managers, it is feasible to quantify the diffusion abilities of all users. On the contrast, the semi-known network here refers to the case that diffusion abilities of partial users remain unknown in advance. For example, the data of Facebook was reported to be utilized to influence the 2016 election in the US, which then led to a severe trust crisis for Facebook. Thus, due to the privacy concern and potential side effect, even for network managers, it is difficult to obtain the full topology of some global large scale social networks like Facebook, We chat. Unless the full network topology is known, we cannot evaluate the diffusion abilities of all users. Over the fully-known network, although we can determine the diffusion probability variations via social links through solving a constrained minimization problem, the huge size of social links in current large scale networks leads to the high complexity of the problem. Moreover, the unknown diffusion abilities of partial users over the semi-known network induce it infeasible to directly solve the constrained minimization problem for minimizing the diffusion size of sensitive information. To tackle the above challenges, we utilize the constrained combination multi-arm bandit framework to jointly design our solutions over the fully-known and semi-known networks, where we take the diffusion size of sensitive information as the reward of a bandit and model the probability variations as the arms in bandit. With this mapping, we determine the probability variations through a constrained arms picking process with the aim of minimizing the obtained rewards. Through incorporating the constraint of diffusion probability variations into the construction of the arms of bandit, we relax the problem of interest into an unconstrained minimization problem when determining the diffusion probability variations based on the arms.

1.3 OBJECTIVE OF PROJECT

We take the first look into limiting the cascading of sensitive information while preserving the diffusion of non-sensitive ones to lower the information loss. Considering the randomness of the users accepting information diffused from their social neighbors, we adopt the widely used random diffusion model that each user diffuses information to his social neighbor successfully with a diffusion probability via the social link between them. Then our technical objective is adjusting the diffusion probabilities via social links to minimize the diffusion size of sensitive information, under the constraint of keeping the value of the sum of diffusion probabilities via all social links. Corresponding to the reality, we consider a case when some advertisements in viral marketing and some rumors simultaneously diffuse over an online social network. In this case, decreasing diffusion probabilities models the measures such as deleting partial posts or fan pages re-posted by users , while the measures for increasing diffusion probabilities include sticking and adding pushes or deliveries of the posts re-posted by given users . Then, if network managers decrease the diffusion probability from a user holding rumors, the advertisements diffused from the user will inevitably be constrained as well.

CHAPTER 2: LITERATURE SURVEY

2.1 Influence maximization on social graphs: A survey

AUTHORS: Y. Li, J. Fan, Y. Wang, and K. L. Tan

Influence Maximization (IM), which selects a set of k users (called seed set) from a social network to maximize the expected number of influenced users (called influence spread), is a key algorithmic problem in social influence analysis. Due to its immense application potential and enormous technical challenges, IM has been extensively studied in the past decade. In this paper, we survey and synthesize a wide spectrum of existing studies on IM from an algorithmic perspective, with a special focus on the following key aspects: (1) a review of well-accepted diffusion models that capture the information diffusion process and build the foundation of the IM problem, (2) a fine-grained taxonomy to classify existing IM algorithms based on their design objectives,

(3) a rigorous theoretical comparison of existing IM algorithms, and (4) a comprehensive study on the applications of IM techniques in combining with novel context features of social networks such as topic, location, and time. Based on this analysis, we then outline the key challenges and research directions to expand the boundary of IM research.

2.2 Post and repost: A holistic view of budgeted influence maximization

AUTHORS: Q. Shi, C. Wang, J. Chen, Y. Feng, and C. Chen

Abstract Existing studies on influence maximization (IM) mainly focus on activating a set of influential users (seed nodes). Originated from the seed nodes' promotion actions (e.g., posting an advertising tweet) on social networks, a large influence spread might be triggered. However, in practice it is usually very expensive to have influential users posting original tweets in a promotional event. In contrast, it will incur much lower costs to have influential users reposting tweets and have ordinary users posting original tweets. Inspired by these observations, in this paper, we consider the Holistic Budgeted Influence Maximization (HBIM) problem, which maximizes the influence spread by deploying the budget to select seed nodes (for posting) and boost nodes (for reposting). To tackle the NP-hardness and non-submodularity of the problem, we devise two efficient algorithms with the data-dependent approximation ratios. Extensive experiments on real social networks demonstrate the efficiency and effectiveness of our proposed algorithms.

2.3 GLP: a novel framework for group-level location promotion in Geo-social networks

AUTHORS: X. Wu, L. Fu, Y. Yao, X. Fu, X. Wang, and G. Chen

Location-aware viral marketing is crucial in modern commercial applications for attracting customers to certain points of interests. Prior works are mainly based on formulating it into a location-aware influence maximization problem in Geo-social Networks (GSNs), where K initial seed individuals are selected in hope of maximizing the number of final influenced users. In this paper, we present the first look into the group-level location promotion, which can potentially enhance its performance, with the phenomenon that users belonging to the same geo-community share similar moving preferences. We propose GLP, a new and novel framework of group-level location promotion by virtue of geo-communities, each of which is treated as a group in GSNs. Aiming to attract more users to designated locations, GLP firstly carries out user grouping through an iterative learning approach based on information extraction from massive check-ins records. The advantage of GLP is three-folded: i) by aggregating movements of group members, GLP significantly avoids the sparsity and sporadicity of individual check-ins, and thus obtains more reliable mobility models; ii) by generalizing a new group-level social graph, GLP can exponentially reduce the computational complexity of seed nodes selection that is algorithmically executed by a greedy algorithm; iii) in comparison with prior individual-level cases, GLP is theoretically demonstrated to drastically increase influence spread under the same given budget. Extensive experiments on real datasets demonstrate that the GLP outperforms four baselines, with notably up to 10 times larger influence spread and 100 times faster seed selection over two individual-level cases, meanwhile verifying the impact of group numbers in final influence spread.

2.4 Boosting information spread: An algorithmic approach

AUTHORS: Y. Lin, W. Chen, and J. C. Lui

The majority of influence maximization (IM) studies focus on targeting influential seeders to trigger substantial information spread in social networks. In this paper, we consider a new and complementary problem of how to further increase the influence spread of given seeders. Our study is motivated by the observation that direct incentives could "boost" users so that they are more likely to be influenced by friends. We study the k-boosting problem which aims to find k users to boost so that the final "boosted" influence spread is maximized. The k-boosting problem is different from the IM problem because boosted users behave differently from seeders: boosted users are initially uninfluenced and we only increase their probability to be influenced. Our work also complements the IM studies because we focus on triggering larger influence spread on the basis of given seeders. Both the NP-hardness of the problem and the non-submodularity of the objective function pose challenges to the k-boosting problem. To tackle the problem, we devise two efficient algorithms with the data-dependent approximation ratio. We conduct extensive experiments using real social networks demonstrating the efficiency and effectiveness of our proposed algorithms. We show that boosting solutions returned by our algorithms achieves boosts of influence that are up to several times higher than those achieved by boosting solutions returned by intuitive baselines, which have no guarantee of solution quality. We also explore the "budget allocation" problem in our experiments. Compared with targeting seeders with all budget, larger influence spread is achieved when we allocation the budget to both seeders and boosted users. This also shows that our study complements the IM studies.

2.5 Data-Aware Vaccine Allocation Over Large Networks

AUTHORS: Y. Zhang, and B. A Prakash,

Given a graph, like a social/computer network or the blogosphere, in which an infection (or meme or virus) has been spreading for some time, how to select the k best nodes for immunization/quarantining immediately? Most previous works for controlling propagation (say via immunization) have concentrated on developing strategies for vaccination preemptively before the start of the epidemic. While very useful to provide insights in to which baseline policies can best control an infection, they may not be ideal to make real-time decisions as the infection is progressing. In this paper, we study how to immunize healthy nodes, in the presence of already infected nodes. Efficient algorithms for such a problem can help public-health experts make more informed choices, tailoring their decisions to the actual distribution of the epidemic on the ground.

CHAPTER 3: SOFTWARE AND HARDWARE REQUIREMENTS

This section elaborates on the functional requirements of the application. The SRS itself can be divided into module, each module having specifications. In order to carry out the project, the following hardware and software is required.

HARDWARE REQUIREMENTS:

● System	:	i3
● Hard Disk	:	40 GB.
● Floppy Drive	:	1.44 Mb.
● Monitor	:	15 VGA Colour.
● Mouse	:	Logitech.
● Ram	:	512 Mb.

SOFTWARE REQUIREMENTS:

● Technology	:	Java 2 Standard Edition, JDBC
● Web Server	:	Tomcat 7.0
● Client Side Technologies	:	HTML, CSS, JavaScript
● Server Side Technologies	:	Servlets, JSP
● Data Base Server	:	MySQL
● Editor	:	Netbeans8.1

CHAPTER 4: SYSTEM DEVELOPMENT ANALYSIS

4.1 Overview of problem

4.1.1 Existing System

The existing attempts that share the closest correlation with prohibiting sensitive information diffusion belong to the rumor influence minimization, whose current strategies can mainly be classified into two aspects. The first is diffusing the truths over network to counteract rumors. However, diffusing truths is only suitable for constraining the rumors, while is not suitable for constraining the diffusion of the other kinds of sensitive informations, including personal informations, trade secrets, and etc. The second is temporarily blocking a number of users with high diffusion abilities or blocking a number of social links among users in hope of minimizing the diffusion of a rumor. Although such strategy is effective for preventing rumors about some significant events like earthquakes, terrorist attacks and political elections, it is unrealistic for network managers to adopt this strategy on constraining the diffusion of sensitive informations with various contents that widely exist in our daily lives. If network managers take such measure, it is required to block a much larger size of users or links. Then two critical problems arise. Firstly, blocking too many users or social links will degrade user experiences and may arouse complaints for the right violation. Secondly, blocking users or social links for restraining rumors also brings the loss of the diffusion of positive informations, say information loss, which is not beneficial to the viral marketers that utilize information cascading to promote products.

4.1.2 Limitations of Existing System

1. Although previous work in Psychology has identified several crucial mental factors related to SNMDs, they are mostly examined as standard diagnostic criteria in survey questionnaires.

2. To automatically detect potential SNMD cases of OSN users, extracting these factors to assess users’ online mental states is very challenging.

3. For example, the extent of loneliness and the effect of dis-inhibition of OSN users are not easily observable.

4. The developed schemes are not designed to handle the sparse data from multiple

OSNs.

5. The SNMD data from different OSNs may be incomplete due to the heterogeneity.

4.2 Define the Problem

4.2.1 Proposed System

To tackle the above challenges, we utilize the constrained combinatorial multi-arm bandit framework to jointly design our solutions over the fully-known and semi-known networks, where we take the diffusion size of sensitive informations as the reward of a bandit and model the probability variations as the arms in bandit. With this mapping, we determine the probability variations through a constrained arms picking process with the aim of minimizing the obtained rewards. Through incorporating the constraint of diffusion probability variations into the construction of the arms of bandit, we relax the problem of interest into an unconstrained minimization problem when determining the diffusion probability variations based on the arms. This enables us to determine the probability variations via social 1. https://www.douban.com/ links with high efficiency. Furthermore, for coping with the unknown diffusion abilities over the semi-known network, we propose to iteratively learn the unknown diffusion abilities through learning the reward distributions of the arms based on the rewards obtained from previously picked arms, and then determine the diffusion probability variations based on the learned reward distributions of arms.

Our main contributions are summarized as follows:

(1) We take the first look into minimizing the diffusion size of sensitive informations while preserving the diffusion of non-sensitive ones. We formulate the problem of interest into a constrained minimization problem where we characterize the intention of preserving non-sensitive information diffusions as the constraint.

(2) We propose an efficient bandit based framework to jointly explore the solutions over the fully-known and semiknown networks within polynomial running time. Moreover, we design the distributed implementation scheme of our solutions for the further improvement of time efficiency.

(3) We further extend our bandit based solution into a “learning- determining” manner for addressing the challenge of unknown diffusion abilities in semi-known networks. We theoretically prove that the regret bound of our solution is sub-linear to

the diffusion time, indicating that the probability variations returned by our solution approximates to the optimal one with the increase of diffusion time.

(4) We perform extensive experiments on both real and synthetic social network datasets. The results demonstrate that the proposed algorithms can effectively constrain the diffusion of sensitive informations, and more importantly, enjoy a superiority over four baselines in terms of 40% less information diffusion loss.

4.2.2 Advantages of Proposed System

1. The novel STM incorporates the SNMD characteristics into the tensor model according to Tucker decomposition.
2. The tensor factorization captures the structure, latent factors, and correlation of features to derive a full portrait of user behavior.
3. We further exploit CANDECOMP/PARAFAC (CP) decomposition based STM and design a stochastic gradient descent algorithm, i.e., STM-CP-SGD, to address the efficiency and solution uniqueness issues in traditional Tucker decomposition.

4. The convergence rate is significantly improved by the proposed second-order stochastic gradient descent algorithm, namely, STM-CP-2SGD.
5. To further reduce the computation time, we design an approximation scheme of the second-order derivative, i.e., Hessian matrix, and provide a theoretical analysis.

4.2.3 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 feasibility study are:

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

4.2.3.1 Economic 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.

4.2.3.2 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 a modest requirement, as only minimal or null changes are required for implementing this system.

4.2.3.3 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.

4.3 Modules Overview

4.3.1 Admin

In this module, the Admin has to login by using valid user name and password. After login successful he can perform some operations such as Authorizing users, Login ,View all users and authorize, give click option to view all users locations in GMap using Multiple Markers ,View all Friend Request and Response ,View all users time line tweet details with Soci rank, rating and give tweet ,View all tweets by clustering based on tweet name and show tweeted details, Soci_Rank,rating and View all Relevant Term Identification on all tweets and group together(similar tweeted details for each and every created tweet) ,View all users outlier detection tweet with its tweeted details, Soci_Rank,rating and View all term frequency on all tweets count(Display the tweets which is getting tweet regularly) based on tweet name, View all tweet news Socirank in chart and View all tweet term frequency count in chart based on date and time, View all tweets tweeted socirank in chart

Friend Request & Response

In this module, the admin can view all the friend requests and responses. Here all the requests and responses will be displayed with their tags such as Id, requested user photo, requested user name, user name request to, status and time & date. If the user accepts the request then the status will be changed to accepted or else the status will remains as waiting.

4.3.2 User

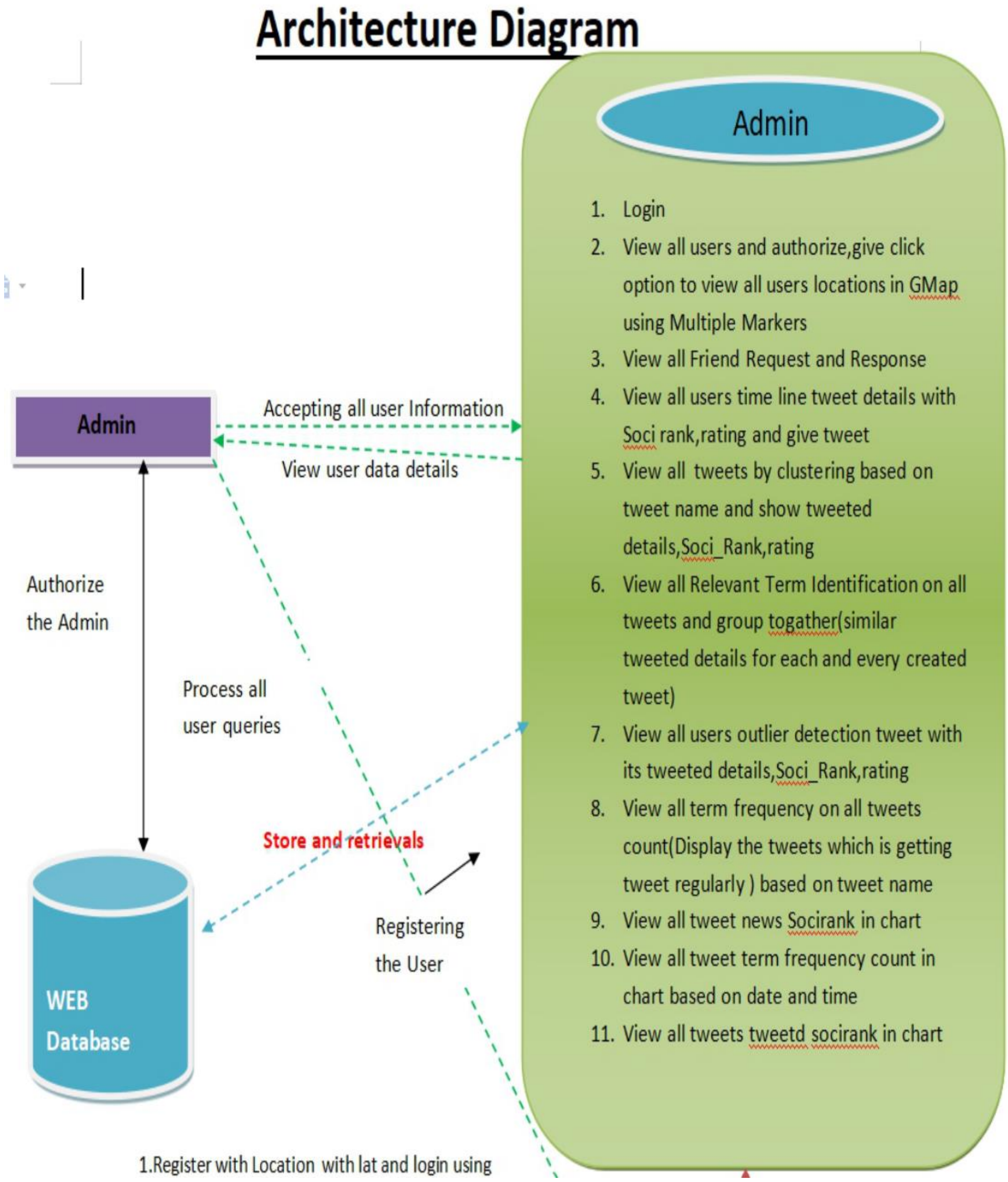
In this module, there are n numbers of users are present. User should register before performing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user can perform some operations like Register with Location with lat and login using GMap and Login, View Your Profile with location ,Search Friend and Find Friend Request, View all Your Friends Details and Location Route path from Your Location, View all your time line tweets with Soci rank, rating and give tweet, Create tweet for News like Tweet name, tweet uses, Tweet desc(enc),tweet image and View all your tweet with re tweet details,Socirank,rating,Search tweet and list all Tweets and view its details and give re tweet, give rank by hyper link and View all your friends Tweets and give Tweet

Searching Users to make friends

In this module, the user searches for users in Same Site and in the Sites and sends friend requests to them. The user can search for users in other sites to make friends only if they have permission.

CHAPTER 5: PROJECT SYSTEM DESIGN

5.1 System Architecture



1. Login
2. View all users and authorize,give click option to view all users locations in GMap using Multiple Markers
3. View all Friend Request and Response
4. View all users time line tweet details with Soci rank,rating and give tweet
5. View all tweets by clustering based on tweet name and show tweeted details,Soci_Rank,rating
6. View all Relevant Term Identification on all tweets and group togather(similar tweeted details for each and every created tweet)
7. View all users outlier detection tweet with its tweeted details,Soci_Rank,rating
8. View all term frequency on all tweets count(Display the tweets which is getting tweet regularly) based on tweet name
9. View all tweet news Socirank in chart
10. View all tweet term frequency count in chart based on date and time
11. View all tweets tweetd socirank in chart

Figure 1. System Architecture

5.2 Flow Charts

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

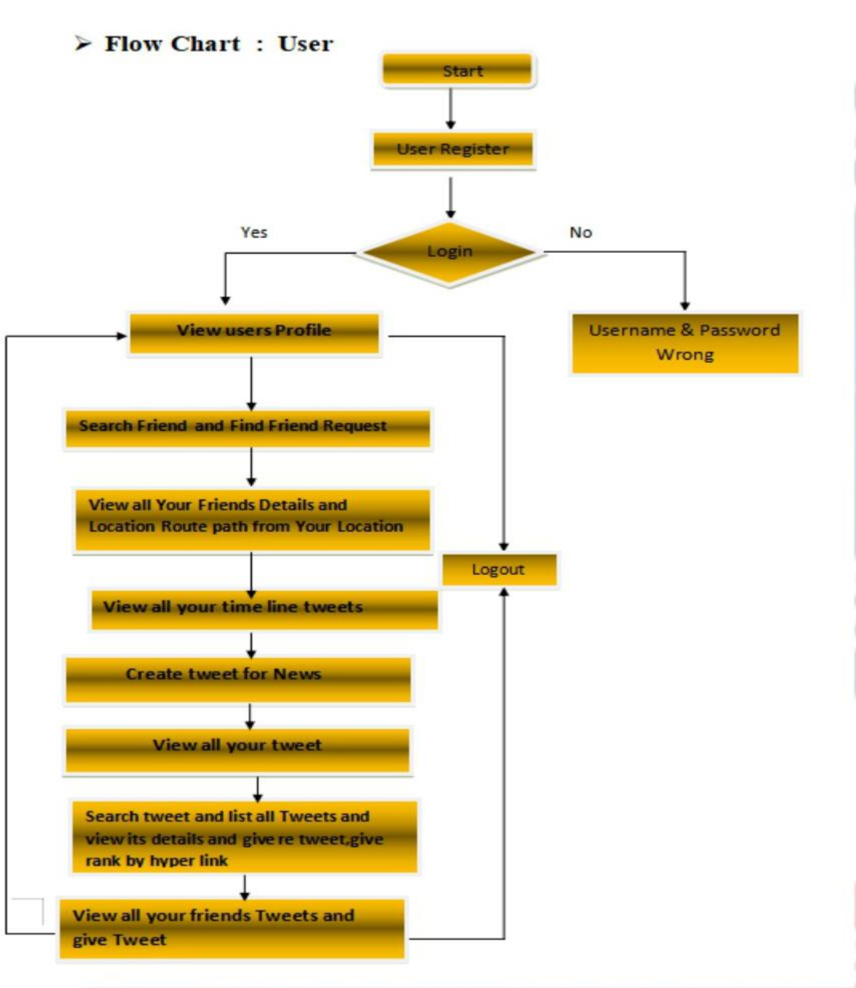

➤ Flow Chart : User
Start
User Register
Yes
Login
No
View users Profile
Username & Password Wrong
Search Friend and Find Friend Request
View all Your Friends Details and Location Route path from Your Location
Logout
View all your time line tweets
Create tweet for News
View all your tweet
Search tweet and list all Tweets and view its details and give re tweet,give rank by hyper link
View all your friends Tweets and give Tweet

Figure 2. Flow Chart : User

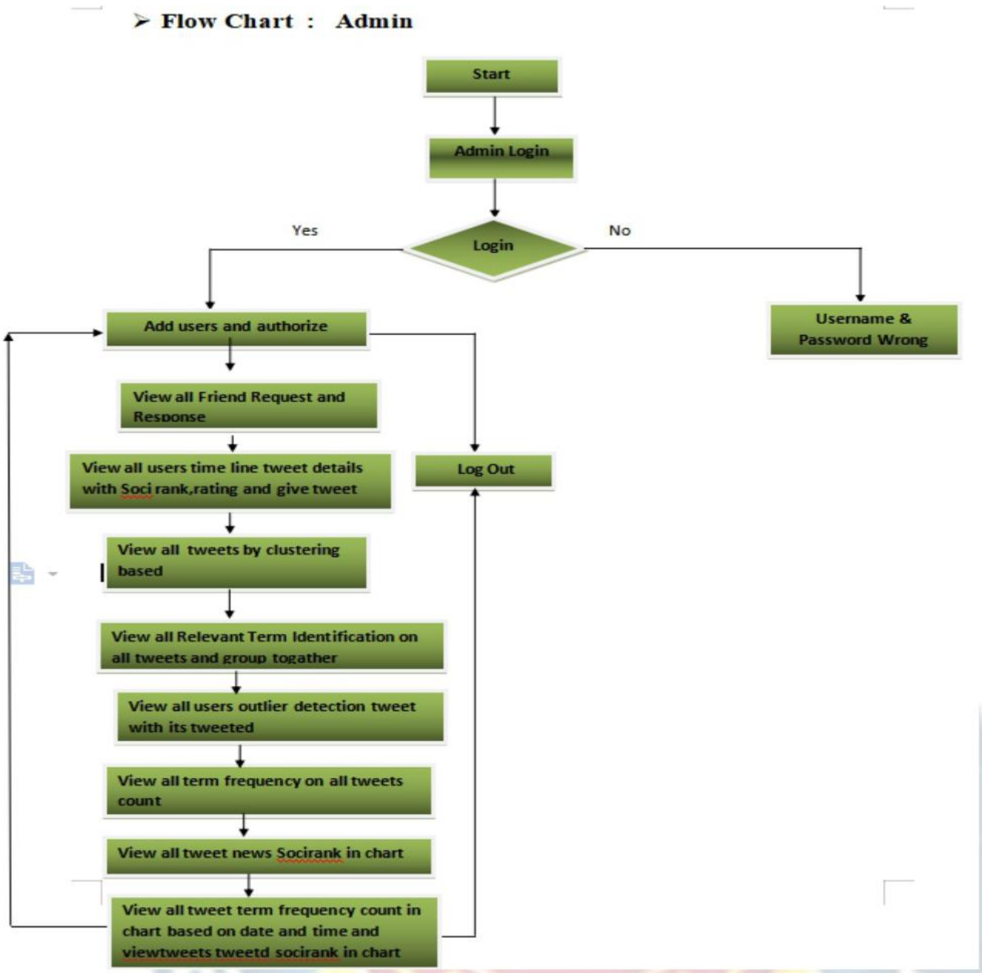

➢ Flow Chart : Admin
Start
Admin Login
Yes
Login
No
Username & Password Wrong
Add users and authorize
View all Friend Request and Response
View all users time line tweet details with Soci rank,rating and give tweet
Log Out
View all tweets by clustering based
View all Relevant Term Identification on all tweets and group togather
View all users outlier detection tweet with its tweeted
View all term frequency on all tweets count
View all tweet news Socirank in chart
View all tweet term frequency count in chart based on date and time and viewtweets tweetd socirank in chart

Figure 3. Flow Chart : Admin

5.3 UML Diagrams

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

Goals:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

5.3.1 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

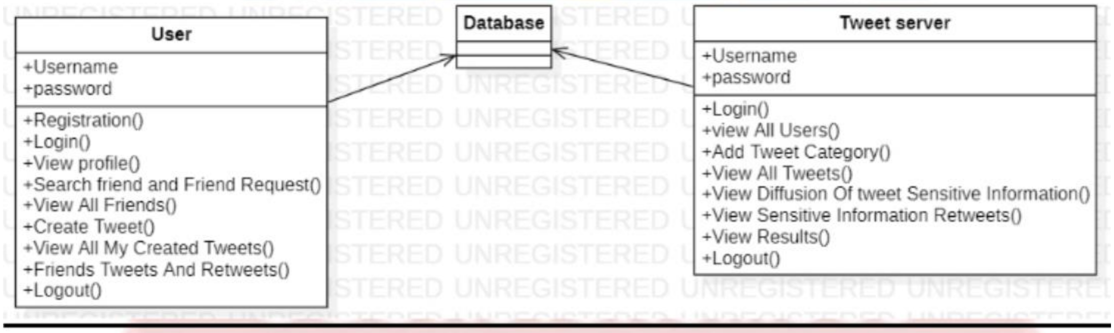


Figure 4. Class Diagram

5.3.2 Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

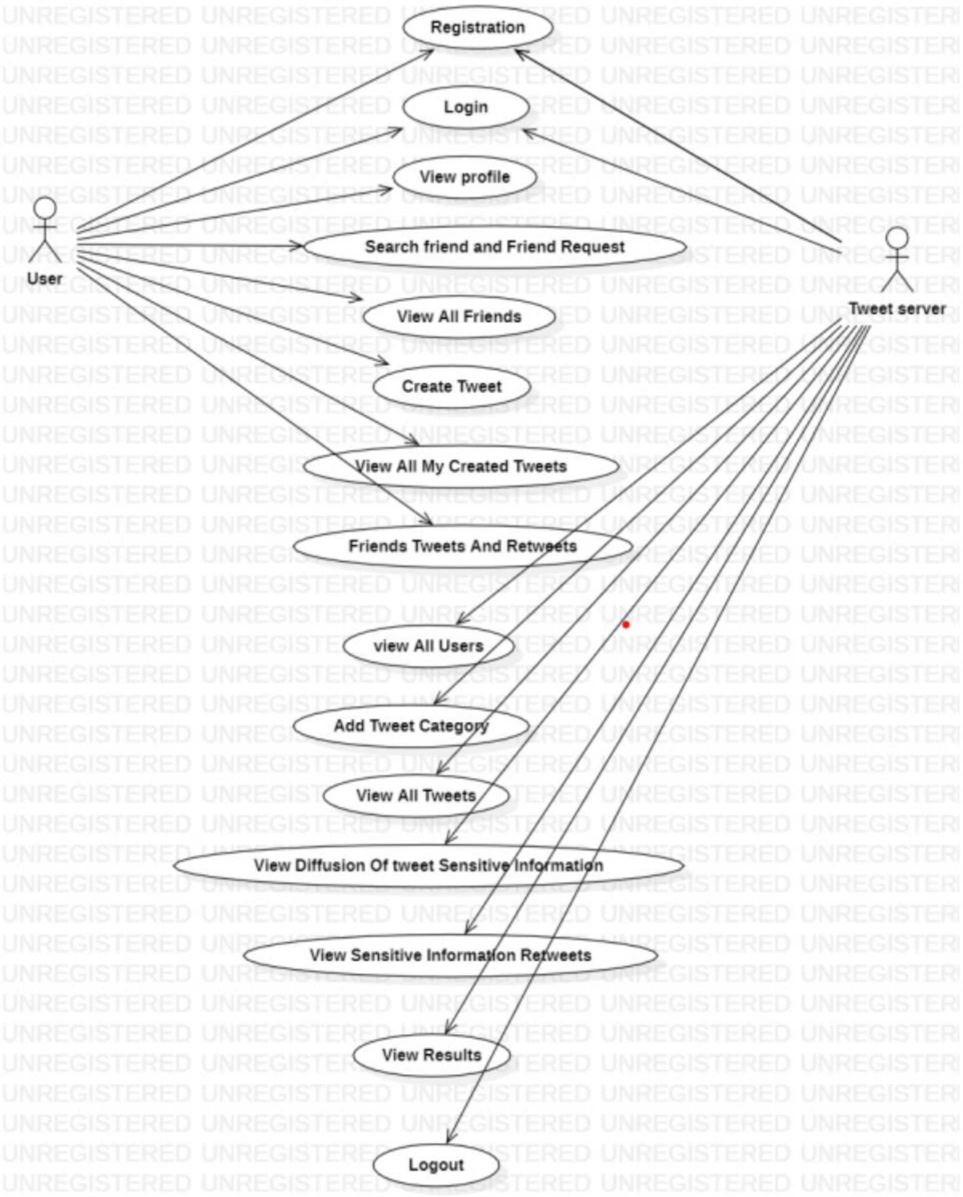

Registration
Login
View profile
Search friend and Friend Request
View All Friends
Create Tweet
View All My Created Tweets
Friends Tweets And Retweets
view All Users
Add Tweet Category
View All Tweets
View Diffusion Of tweet Sensitive Information
View Sensitive Information Retweets
View Results
Logout
User
Tweet server

Figure 5. Use Case Diagram

5.3.3 Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

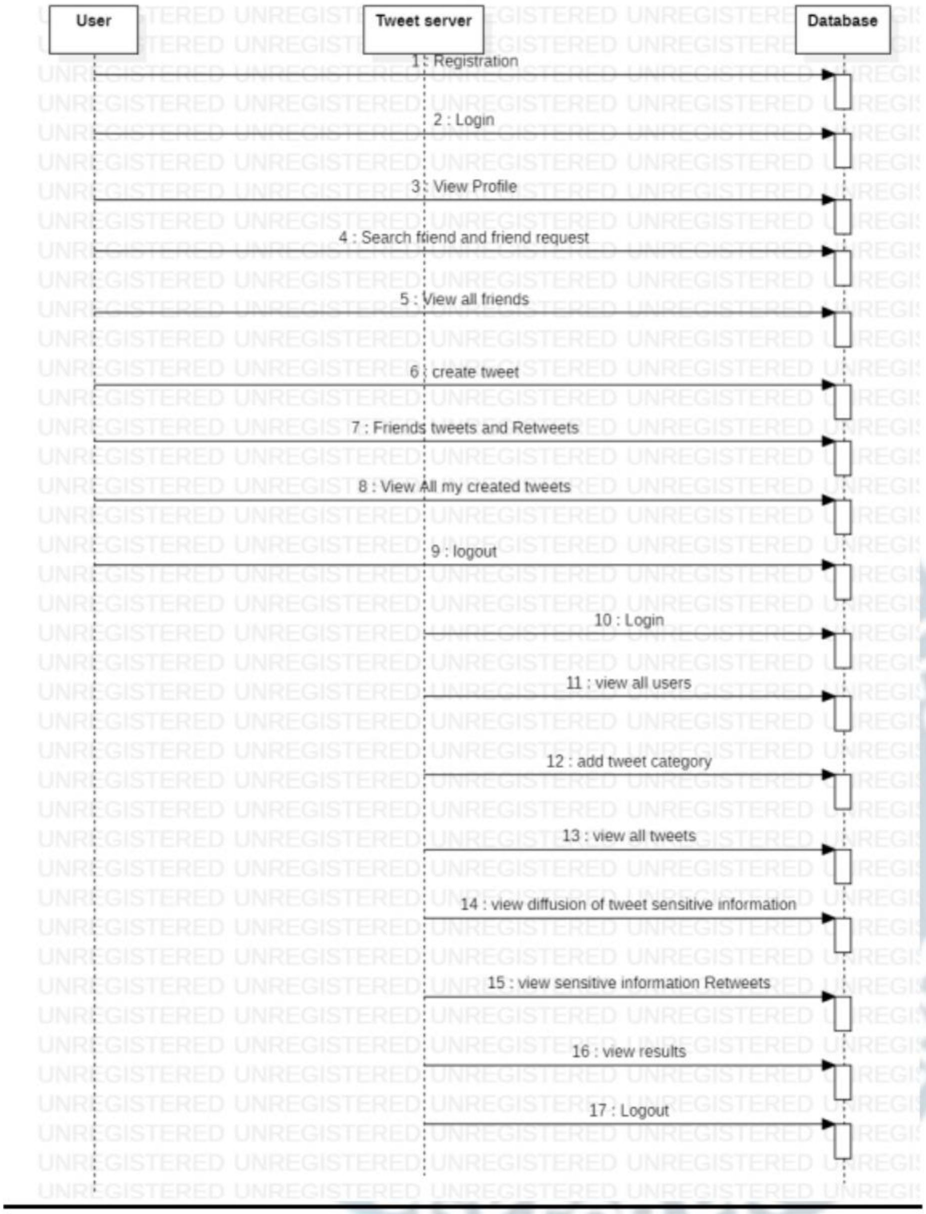


Figure 6. Sequence Diagram

5.3.4 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

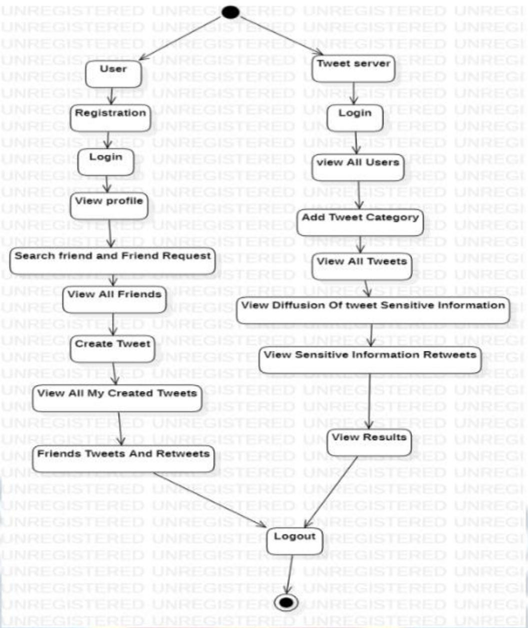

User
Registration
Login
View profile
Search friend and Friend Request
View All Friends
Create Tweet
View All My Created Tweets
Friends Tweets And Retweets
Tweet server
Login
view All Users
Add Tweet Category
View All Tweets
View Diffusion Of tweet Sensitive Information
View Sensitive Information Retweets
View Results
Logout

Figure 7. Activity Diagram

5.3.5 Deployment diagram

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.[1] To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have subnodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

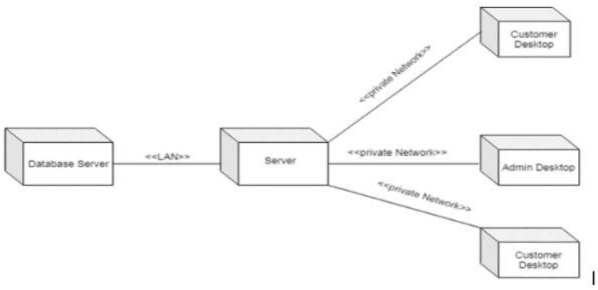


Figure 8 : Deployment Diagram

5.3.6 Package Diagram

Package diagram is UML structure diagram which shows structure of the designed system at the level of packages. The following elements are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.

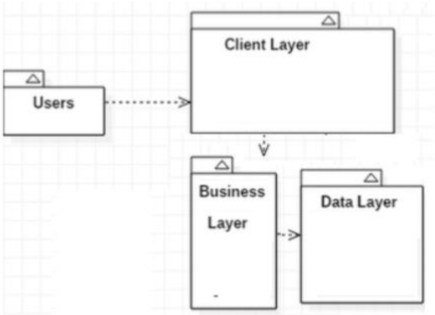


Figure 9 . Package Diagram

5.3.7 Profile Diagram

A Profile diagram is any diagram created in a «profile» Package. Profiles provide a means of extending the UML. They are based on additional stereotypes and Tagged Values that are applied to UML elements, connectors, and their components.

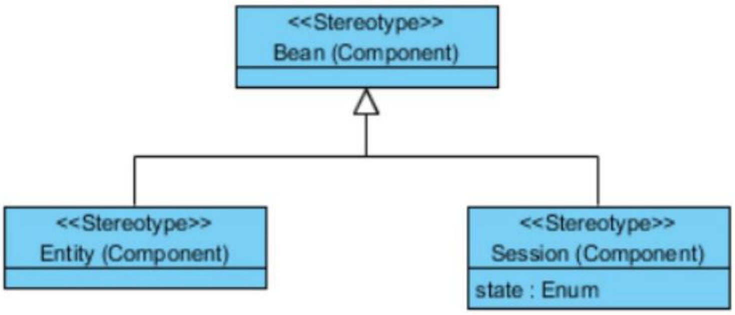


Figure 10. Profile Diagram

CHAPTER 6: PROJECT CODING

6.1 Technology

In my project, I have chosen Java language for developing the code.

About Java

Initially the language was called as “oak” but it was renamed as “Java” in 1995. The primary motivation of this language was the need for a platform-independent (i.e., architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

- Java is a programmer’s language.
- Java is cohesive and consistent.
- Except for those constraints imposed by the Internet environment, Java gives the programmer, full control.

Finally, Java is to Internet programming where C was to system programming. Importance of Java to the Internet

Java has had a profound effect on the Internet. This is because; Java expands the Universe of objects that can move about freely in Cyberspace. In a network, two categories of objects are transmitted between the Server and the Personal computer. They are: Passive information and Dynamic active programs. The Dynamic, Self-executing programs cause serious problems in the areas of Security and probability. But, Java addresses those concerns and by doing so, has opened the door to an exciting new form of program called the Applet.
Java can be used to create two types of programs

Applications and Applets: An application is a program that runs on our Computer under the operating system of that computer. It is more or less like one creating using C or C++. Java’s ability to create Applets makes it important. An Applet is an application designed to be transmitted over the Internet and executed by a Java – compatible web browser. An applet is actually a tiny Java program, dynamically downloaded across the network, just like an image. But the difference is, it is an intelligent program, not just a media file. It can react to the user input and dynamically change.

Features Of Java

Security

Every time you that you download a “normal” program, you are risking a viral infection. Prior to Java, most users did not download executable programs frequently, and those who did scanned them for viruses prior to execution. Most users still worried about the possibility of infecting their systems with a virus. In addition, another type of malicious program exists that must be guarded against. This type of program can gather private information, such as credit card numbers, bank account balances, and passwords. Java answers both these concerns by providing a “firewall” between a network application and your computer.

When you use a Java-compatible Web browser, you can safely download Java applets without fear of virus infection or malicious intent.

Portability

For programs to be dynamically downloaded to all the various types of platforms connected to the Internet, some means of generating portable executable code is needed .As you will see, the same mechanism that helps ensure security also helps create portability. Indeed, Java’s solution to these two problems is both elegant and efficient.

The Byte code

The key that allows the Java to solve the security and portability problems is that the output of Java compiler is Byte code. Byte code is a highly optimized set of instructions designed to be executed by the Java run-time system, which is called the Java Virtual Machine (JVM). That is, in its standard form, the JVM is an interpreter for byte code.

Translating a Java program into byte code helps makes it much easier to run a program in a wide variety of environments. The reason is, once the run-time package exists for a given system, any Java program can run on it.

Although Java was designed for interpretation, there is technically nothing about Java that prevents on-the-fly compilation of byte code into native code. Sun has just completed its Just In Time (JIT) compiler for byte code. When the JIT compiler is a part of JVM, it compiles byte code into executable code in real time, on a piece-by-piece, demand basis. It is not possible to compile an entire Java program into executable code all at once, because Java performs various run-time checks that can be done only at run time. The JIT compiles code, as it is needed, during execution.

Java, Virtual Machine (JVM)

Beyond the language, there is the Java virtual machine. The Java virtual machine is an important element of the Java technology. The virtual machine can be embedded within a web browser or an operating system. Once a piece of Java code is loaded onto a machine, it is verified. As part of the loading process, a class loader is invoked and does byte code verification makes sure that the code that's has been generated by the compiler will not corrupt the machine that it's loaded on. Byte code verification takes place at the end of the compilation process to make sure that is all accurate and correct. So byte code verification is integral to the compiling and executing of Java code.

Overall Description

Java byte
Java
.Class

Figure 11 . Picture showing the development process of JAVA Program

Java programming uses to produce byte codes and executes them. The first box indicates that the Java source code is located in a. Java file that is processed with a Java compiler called javac. The Java compiler produces a file called a. class file, which contains the byte code. The. Class file is then loaded across the network or loaded locally on your machine into the execution environment is the Java virtual machine, which interprets and executes the byte code.

Java Architecture

Java architecture provides a portable, robust, high performing environment for development. Java provides portability by compiling the byte codes for the Java Virtual Machine, which is then interpreted on each platform by the run-time environment. Java is a dynamic system, able to load code when needed from a machine in the same room or across the planet.

Compilation of code

When you compile the code, the Java compiler creates machine code (called byte code) for a hypothetical machine called Java Virtual Machine (JVM). The JVM is supposed to execute the byte code. The JVM is created for overcoming the issue of portability. The code is written and compiled for one machine and interpreted on all machines. This machine is called Java Virtual Machine.

Compiling and interpreting Java Source Code

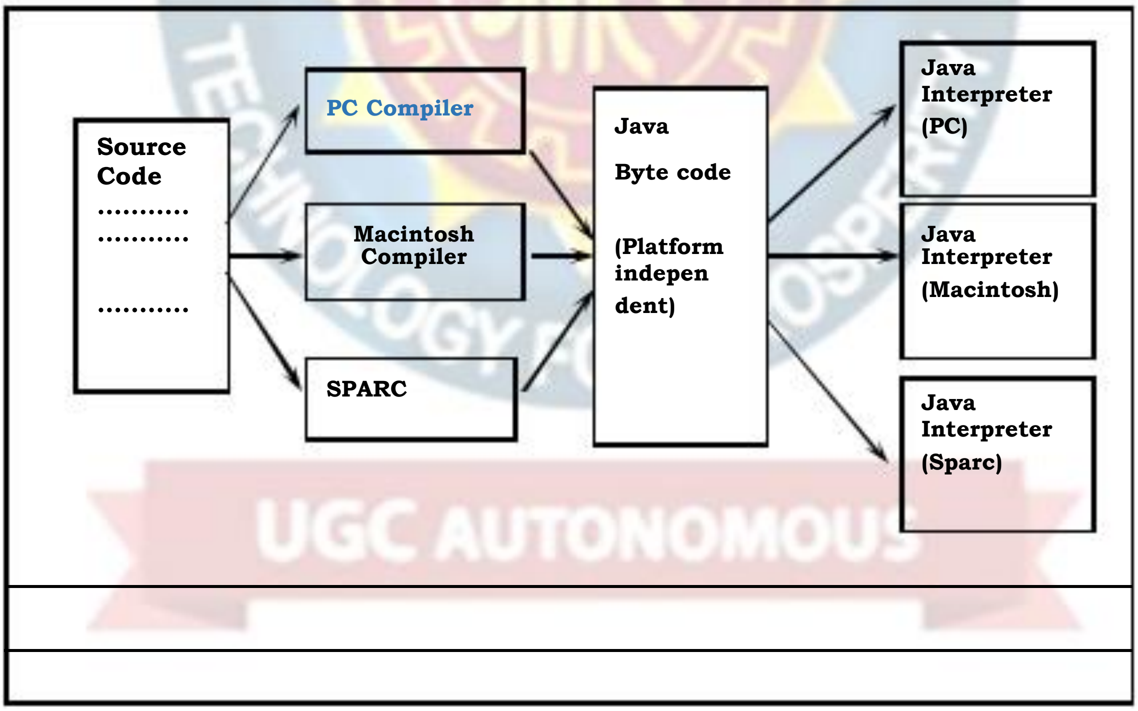


Figure 12 . Compiling and Interpretation of Java source code

During run-time the Java interpreter tricks the byte code file into thinking that it is running on a Java Virtual Machine. In reality this could be a Intel Pentium Windows 95 or Sun SARC station running Solaris or Apple Macintosh running system and all could receive code from any computer through Internet and run the Applets.

Simple

Java was designed to be easy for the Professional programmer to learn and to use effectively. If you are an experienced C++ programmer, learning Java will be even easier. Because Java inherits the C/C++ syntax and many of the object oriented features of C++. Most of the confusing concepts from C++ are either left out of Java or implemented in a cleaner, more approachable manner. In Java there are a small number of clearly defined ways to accomplish a given task.

Object-Oriented

Java was not designed to be source-code compatible with any other language. This allowed the Java team the freedom to design with a blank slate. One outcome of this was a clean usable, pragmatic approach to objects. The object model in Java is simple and easy to extend, while simple types, such as integers, are kept as high-performance non-objects.

Robust

The multi-platform environment of the Web places extraordinary demands on a program, because the program must execute reliably in a variety of systems. The ability to create robust programs was given a high priority in the design of Java. Java is strictly typed language; it checks your code at compile time and run time.

Java virtually eliminates the problems of memory management and deallocation, which is completely automatic. In a well-written Java program, all run time errors can –and should –be managed by your program.

JAVASCRIPT

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications. On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then updates the browser's display accordingly

Even though JavaScript supports both client and server Web programming, we prefer JavaScript at Client side programming since most of the browsers supports it. JavaScript is almost as easy to learn as HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags

```
<SCRIPTS>..</SCRIPT>.

<SCRIPT LANGUAGE = "JavaScript">

JavaScript statements

</SCRIPT>
```

Here are a few things we can do with JavaScript :

- Validate the contents of a form and make calculations.
- Add scrolling or changing messages to the Browser's status line.
- Animate images or rotate images that change when we move the mouse over them.
- Detect the browser in use and display different content for different browsers.
- Detect installed plug-ins and notify the user if a plug-in is required.

We can do much more with JavaScript, including creating entire application.

JavaScript vs Java

JavaScript and Java are entirely different languages. A few of the most glaring differences are:

- Java applets are generally displayed in a box within the web document; JavaScript can affect any part of the Web document itself.
- While JavaScript is best suited to simple applications and adding

interactive features to Web pages; Java can be used for incredibly complex applications.

There are many other differences but the important thing to remember is that JavaScript and Java are separate languages. They are both useful for different things; in fact they can be used together to combine their advantages

Advantages

- JavaScript can be used for Sever-side and Client-side scripting.
- It is more flexible than VBScript.
- JavaScript is the default scripting languages at Client-side since all the browsers supports it.

Hyper Text Markup Language

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produces Web pages that include text, graphics and pointer to other Web pages (Hyperlinks).

HTML is not a programming language but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but specialized to hypertext and adapted to the Web. The idea behind Hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point. We can navigate through the information based on our interest and preference. A markup language is simply a series of elements, each delimited with special characters that define how text or other items enclosed within the elements should be displayed. Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document.

HTML can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop.

HTML provides tags (special codes) to make the document look attractive. HTML tags are not case-sensitive. Using graphics, fonts, different sizes, color, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

Basic HTML Tags :

<!-- -->	Specifies comments

<A>……….</A>	Creates hypertext links
<B>……….</B>	Formats text as bold
<BIG>……….</BIG>	Formats text in large font.
<BODY>…</BODY>	Contains all tags and text in the HTML document
<CENTER>...</CENTER>	Creates text
<DD>…</DD>	Definition of a term
<DL>...</DL>	Creates definition list
<FONT>…</FONT>	Formats text with a particular font
<FORM>...</FORM>	Encloses a fill-out form
<FRAME>...</FRAME>	Defines a particular frame in a set of frames
<H#>…</H#>	Creates headings of different levels
<HEAD>...</HEAD>	Contains tags that specify information about a document
<HR>...</HR>	Creates a horizontal rule
<HTML>…</HTML>	Contains all other HTML tags
<META>...</META>	Provides meta-information about a document
<SCRIPT>…</SCRIPT>	Contains client-side or server-side script
<TABLE>…</TABLE>	Creates a table
<TD>…</TD>	Indicates table data in a table
<TR>…</TR>	Designates a table row
<TH>…</TH>	Creates a heading in a table

ADVANTAGES

- A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
- HTML is platform independent.
- HTML tags are not case-sensitive.

Java Database Connectivity

What Is JDBC?

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. The combinations of Java and JDBC lets a programmer write it once and run it anywhere. What Does JDBC Do?

Simply put, JDBC makes it possible to do three things:

- ➢ Establish a connection with a database
- ➢ Send SQL statements
- ➢ Process the results.

JDBC versus ODBC and other APIs

At this point, Microsoft's ODBC (Open Database Connectivity) API is that probably the most widely used programming interface for accessing relational databases. It offers the ability to connect to almost all databases on almost all platforms.

So why not just use ODBC from Java? The answer is that you can use ODBC from Java, but this is best done with the help of JDBC in the form of the JDBC-ODBC Bridge, which we will cover shortly. The question now becomes "Why do you need JDBC?" There are several answers to this question:

1. ODBC is not appropriate for direct use from Java because it uses a C interface. Calls from Java to native C code have a number of drawbacks in the security, implementation, robustness, and automatic portability of applications.
2. A literal translation of the ODBC C API into a Java API would not be desirable. For example, Java has no pointers, and ODBC makes copious use of them, including the notoriously error-prone generic pointer "void *". You can think of JDBC as ODBC translated into an object-oriented interface that is natural for Java programmers.
3. ODBC is hard to learn. It mixes simple and advanced features together, and it has complex options even for simple queries. JDBC, on the other hand, was designed to keep simple things simple while allowing more advanced

capabilities where required.

4. A Java API like JDBC is needed in order to enable a "pure Java" solution. When ODBC is used, the ODBC driver manager and drivers must be manually installed on every client machine. When the JDBC driver is written completely in Java, however, JDBC code is automatically installable, portable, and secure on all Java platforms from network computers to mainframes.

Two-tier and Three-tier Models

The JDBC API supports both two-tier and three-tier models for database access.

In the two-tier model, a Java applet or application talks directly to the database. This requires a JDBC driver that can communicate with the particular database management system being accessed. A user's SQL statements are delivered to the database, and the results of those statements are sent back to the user. The database may be located on another machine to which the user is connected via a network. This is referred to as a client/server configuration, with the user's machine as the client, and the machine housing the database as the server. The network can be an Intranet, which, for example, connects employees within a corporation, or it can be the Internet.

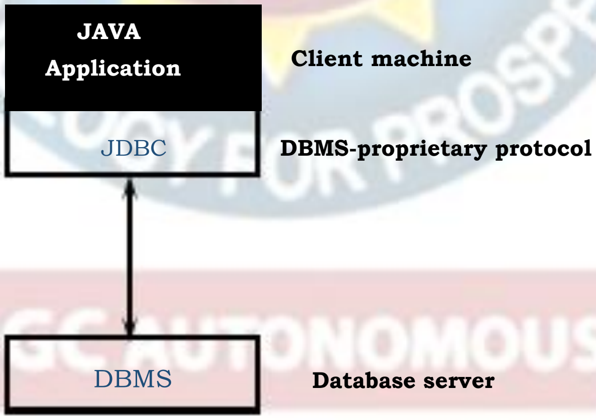


Figure 13 . Two tier model

In the three-tier model, commands are sent to a "middle tier" of services, which then send SQL statements to the database. The database processes the SQL statements and sends the results back to the middle tier, which then sends them to the user. MIS directors find the three-tier model very attractive because the middle tier makes it possible to maintain control over access and the kinds of updates that can be made to corporate data. Another advantage is that when there is a middle tier, the user can employ an easy-to-use higher-level API which is translated by the middle tier into the appropriate low-level calls. Finally, in many cases the three-tier architecture can provide performance advantages.

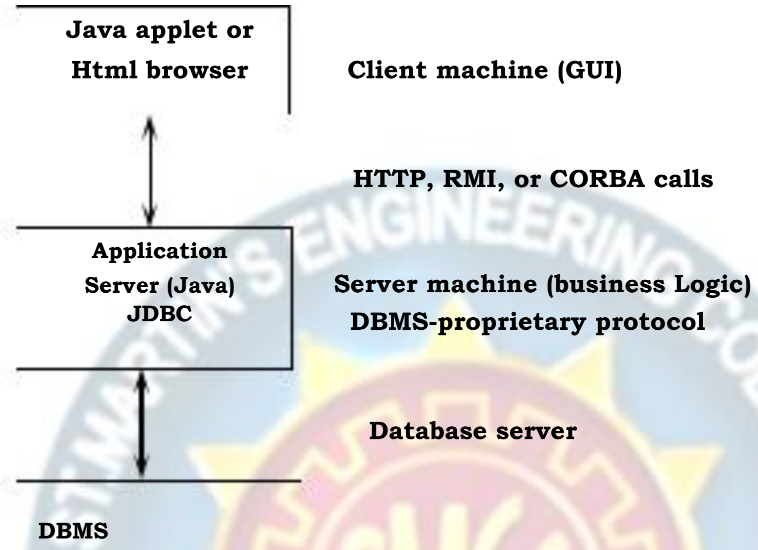


Figure 14. Three tier model

Until now the middle tier has typically been written in languages such as C or C++, which offer fast performance. However, with the introduction of optimizing compilers that translate Java byte code into efficient machine-specific code, it is becoming practical to implement the middle tier in Java. This is a big plus, making it possible to take advantage of Java's robustness, multithreading, and security features. JDBC is important to allow database access from a Java middle tier.

JDBC Driver Types

The JDBC drivers that we are aware of at this time fit into one of four categories:

- JDBC-ODBC bridge plus ODBC driver

- Native-API partly-Java driver
- JDBC-Net pure Java driver
- Native-protocol pure Java driver

JDBC-ODBC Bridge

If possible, use a Pure Java JDBC driver instead of the Bridge and an ODBC driver. This completely eliminates the client configuration required by ODBC. It also eliminates the potential that the Java VM could be corrupted by an error in the native code brought in by the Bridge (that is, the Bridge native library, the ODBC driver manager library, the ODBC driver library, and the database client library).

What Is the JDBC- ODBC Bridge?

The JDBC-ODBC Bridge is a JDBC driver, which implements JDBC operations by translating them into ODBC operations. To ODBC it appears as a normal application program. The Bridge implements JDBC for any database for which an ODBC driver is available. The Bridge is implemented as the

sun.jdbc.odbc Java package and contains a native library used to access ODBC. The Bridge is a joint development of Intersolv and JavaSoft.

Java Server Pages (JSP)

Java server Pages is a simple, yet powerful technology for creating and maintaining dynamic-content web pages. Based on the Java programming language, Java Server Pages offers proven portability, open standards, and a mature re-usable component model .The Java Server Pages architecture enables the separation of content generation from content presentation. This separation not eases maintenance headaches, it also allows web team members to focus on their areas of expertise. Now, web page designer can concentrate on layout, and web application designers on programming, with minimal concern about impacting each other’s work.

Features of JSP

Portability:

Java Server Pages files can be run on any web server or web-enabled application server that provides support for them. Dubbed the JSP engine, this support involves recognition, translation, and management of the Java Server Page lifecycle and its interaction components.

Components

It was mentioned earlier that the Java Server Pages architecture can include reusable Java components. The architecture also allows for the embedding of a scripting language directly into the Java Server Pages file. The components current supported include Java Beans, and Servlets.

Processing

A Java Server Pages file is essentially an HTML document with JSP scripting or tags. The Java Server Pages file has a JSP extension to the server as a Java Server Pages file. Before the page is served, the Java Server Pages syntax is parsed and processed into a Servlet on the server side. The Servlet that is generated outputs real content in straight HTML for responding to the client.

Access Models:

A Java Server Pages file may be accessed in at least two different ways. A client's request comes directly into a Java Server Page. In this scenario, suppose the page accesses reusable Java Bean components that perform particular well-defined computations like accessing a database. The result of the Beans computations, called result sets is stored within the Bean as properties. The page uses such Beans to generate dynamic content and present it back to the client.

In both of the above cases, the page could also contain any valid Java code. Java Server Pages architecture encourages separation of content from presentation.

Steps in the execution of a JSP Application:

1. The client sends a request to the web server for a JSP file by giving the name of the JSP file within the form tag of a HTML page.

2. This request is transferred to the JavaWebServer. At the server side JavaWebServer receives the request and if it is a request for a jsp file server gives this request to the JSP engine.
3. JSP engine is program which can understands the tags of the jsp and then it converts those tags into a Servlet program and it is stored at the server side. This Servlet is loaded in the memory and then it is executed and the result is given back to the JavaWebServer and then it is transferred back to the result is given back to the JavaWebServer and then it is transferred back to the client.

JDBC connectivity

The JDBC provides database-independent connectivity between the J2EE platform and a wide range of tabular data sources. JDBC technology allows an Application Component Provider to:

Perform connection and authentication to a database server

Manager transactions

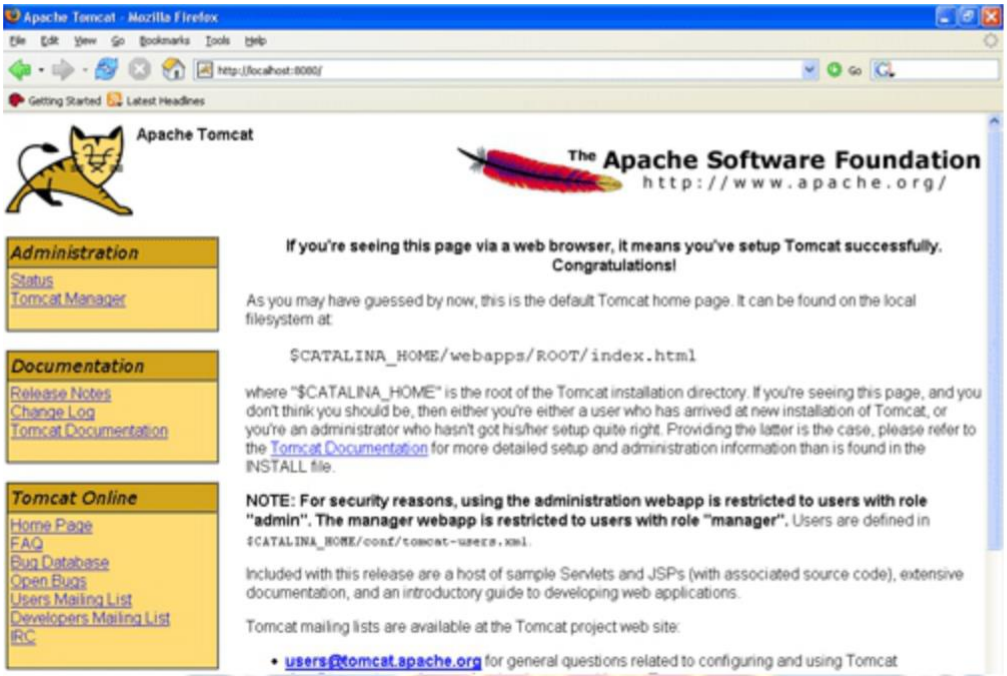
Move SQL statements to a database engine for preprocessing and execution

Execute stored procedures

Inspect and modify the results from Select statements.

Tomcat 6.0 web server

Tomcat is an open source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as well as business components (BEAs Weblogic, is one of the popular application server).To develop a web application with jsp/servlet install any web server like JRun, Tomcat etc to run your application.



If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

As you may have guessed by now, this is the default Tomcat home page. It can be found on the local filesystem at:

```
$CATALINA_HOME/webapps/ROOT/index.html
```

where "$CATALINA_HOME" is the root of the Tomcat installation directory. If you're seeing this page, and you don't think you should be, then either you're either a user who has arrived at new installation of Tomcat, or you're an administrator who hasn't got his/her setup quite right. Providing the latter is the case, please refer to the Tomcat Documentation for more detailed setup and administration information than is found in the INSTALL file.

NOTE: For security reasons, using the administration webapp is restricted to users with role "admin". The manager webapp is restricted to users with role "manager". Users are defined in `$CATALINA_HOME/conf/tomcat-users.xml`.

Included with this release are a host of sample Servlets and JSPs (with associated source code), extensive documentation, and an introductory guide to developing web applications.

Tomcat mailing lists are available at the Tomcat project web site:

- users@tomcat.apache.org for general questions related to configuring and using Tomcat

Figure 15 . Apache Tomcat Server

6.2 Code Templates

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>HOME Page</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"
/> <link href="style.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/arial.js"></script>
<script type="text/javascript" src="js/cuf_run.js"></script>
<style type="text/css">
<!--
.style3 {
        font-size: 36px;
        color: #FF0000;
}
.style4 {
        font-weight: bold;
        color: #000000;
}
.style5 {
	color: #FF0000;
	font-weight: bold;
}
.style6 {
	font-size: 18px;
	color: #0000FF;
}
.style7 {
	font-size: 24px;
	color: #FF00FF;
}
.style8 {color: #FF0000}
.style9 {font-size: 24}
.style10 {color: #FF0000; font-size: 24; }
.style11 {padding:0; margin:0; width:100%; line-height:0; clear: both;}
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo ">
     <h2 class="style3">Adaptive Diffusion of Sensitive Information In Online Social
Networks</h2>
      </div>
      <div class="clr"></div>
      <div class="menu_nav">
        <ul>
          <li class="active"><a href="index.html">Home</a></li>
          <li><a href="A_Login.jsp">Admin</a></li>
          <li><a href="U_Login.jsp">End
        User</a></li> </ul>
        <div class="search">
          <form id="form" name="form" method="post"
            action="#"> <span>
            <input name="q" type="text" class="keywords" id="textfield"
maxlength="50" value="Search..." />
            <input name="b" type="image" src="images/search.gif" class="button"
            /> </span>
          </form>
        </div>
      </div>
      <div class="clr"></div>
      <div class="header_img"> <img src="images/heder_img.jpg" alt="" width="960"
height="288" /></div>
    </div>
  </div>
  <div class="clr"></div>
  <div class="content">
    <div class="content_resize">
      <div class="mainbar">
        <div class="article">
          <h2 class="style6">Adaptive Diffusion of Sensitive Information In Online
Social Networks</h2>
          <div class="clr"></div>
          <p class="style5">Information filtering, social computing, social network
analysis, topic identification, topic ranking.</p>
          <img src="images/img_1.jpg" width="613" height="179" alt=""
          /> <div class="clr"></div>
          <p align="justify" class="style5"></p>
        </div>
        <div class="article">
          <h2 class="style7">Straightforward Approach for Identifying Topics</h2>
          <div class="clr"></div>
          <p><span class="style5">Information filtering, social computing, social
network analysis, topic identification, topic ranking.</span></p>
          <img src="images/img_1.jpg" width="613" height="179" alt=""
          /> <div class="clr"></div>
          <p align="justify" class="style8"></p>
        </div>
        </div>
      <div class="sidebar">
        <div class="gadget">
          <h2>Sidebar Menu</h2>
          <div class="clr"></div>
          <ul class="sb_menu style4">
            <li><a href="index.html">Home</a></li> <li><a
            href="A_Login.jsp">Admin</a></li> <li><a
            href="U_Login.jsp">End User </a></li>
          </ul>
        </div>
        <div class="gadget">
          <h2><span>Concepts</span></h2>
          <div class="clr"></div>
          <ul class="ex_menu">
            <li>Information filtering, </li>
            <li>Social computing, </li>
            <li>Social network analysis, </li>
            <li>Topic identification,</li>
            <li>Topic ranking.</li>
          </ul>
        </div>


                  <div class="gadget">
          <h2><span>Concepts</span></h2>
          <div class="clr"></div>
```

```
          <ul class="ex_menu">
            <li>Information filtering, </li>
            <li>Social computing, </li>
            <li>Social network analysis, </li>
            <li>Topic identification,</li>
            <li>Topic ranking.</li>
          </ul>
        </div>

                    <div class="gadget">
          <h2><span>Concepts</span></h2>
          <div class="clr"></div>
          <ul class="ex_menu">
            <li>Information filtering, </li>
            <li>Social computing, </li>
            <li>Social network analysis, </li>
            <li>Topic identification,</li>
            <li>Topic ranking.</li>
          </ul>
        </div>

      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="fbg">
    <div class="fbg_resize">
      <div class="col c1">
        <h2><span>Image Gallery</span></h2>
        <a href="#"><img src="images/gallery_1.jpg" width="58" height="58" alt=""
/></a> <a href="#"><img src="images/gallery_2.jpg" width="58" height="58" alt=""
/></a> <a href="#"><img src="images/gallery_3.jpg" width="58" height="58" alt=""
/></a> <a href="#"><img src="images/gallery_4.jpg" width="58" height="58" alt=""
```

```
/></a> <a href="#"><img src="images/gallery_5.jpg" width="58" height="58" alt=""
/></a> <a href="#"><img src="images/gallery_6.jpg" width="58" height="58" alt=""
/></a> </div>
      <div class="col c2">
        <h2 class="style9">System</h2>
        <p class="style10">We propose an unsupervised system—SociRank—which
effectively identifies news topics that are prevalent in both social media and the news
media, and then ranks them by relevance using their degrees of MF, UA, and UI. Even
though this paper focuses on news topics, it can be easily adapted<br />
          to a wide variety of fields, from science and technology to culture and
sports.</p>
  </div>
</div>
<div align=center class="style9"></div>
</body>
</html>
```

CHAPTER 7: PROJECT 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.

7.1 Types of Tests

7.1.1 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.

7.1.2 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.

The following are the types of Integration Testing:

1)Top Down Integration

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

2. Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

7.1.3 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.

7.1.4 System Testing

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.

7.1.5 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.

7.1.6 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.

7.2 Test cases:

7.2.1 Test case for Login form:

FUNCTION:	**LOGIN**
EXPECTED RESULTS:	Should Validate the user and check his existence in database
ACTUAL RESULTS:	Validate the user and checking the user against the database
LOW PRIORITY	**No**
HIGH PRIORITY	**Yes**

Table 1 . test case for login form

7.2.2 Test case for User Registration form:

FUNCTION:	**USER REGISTRATION**
EXPECTED RESULTS:	Should check if all the fields are filled by the user and saving the user to database.
ACTUAL RESULTS:	Checking whether all the fields are field by user or not through validations and saving user.
LOW PRIORITY	**No**
HIGH PRIORITY	**Yes**

Table 2 . Test case for user registration

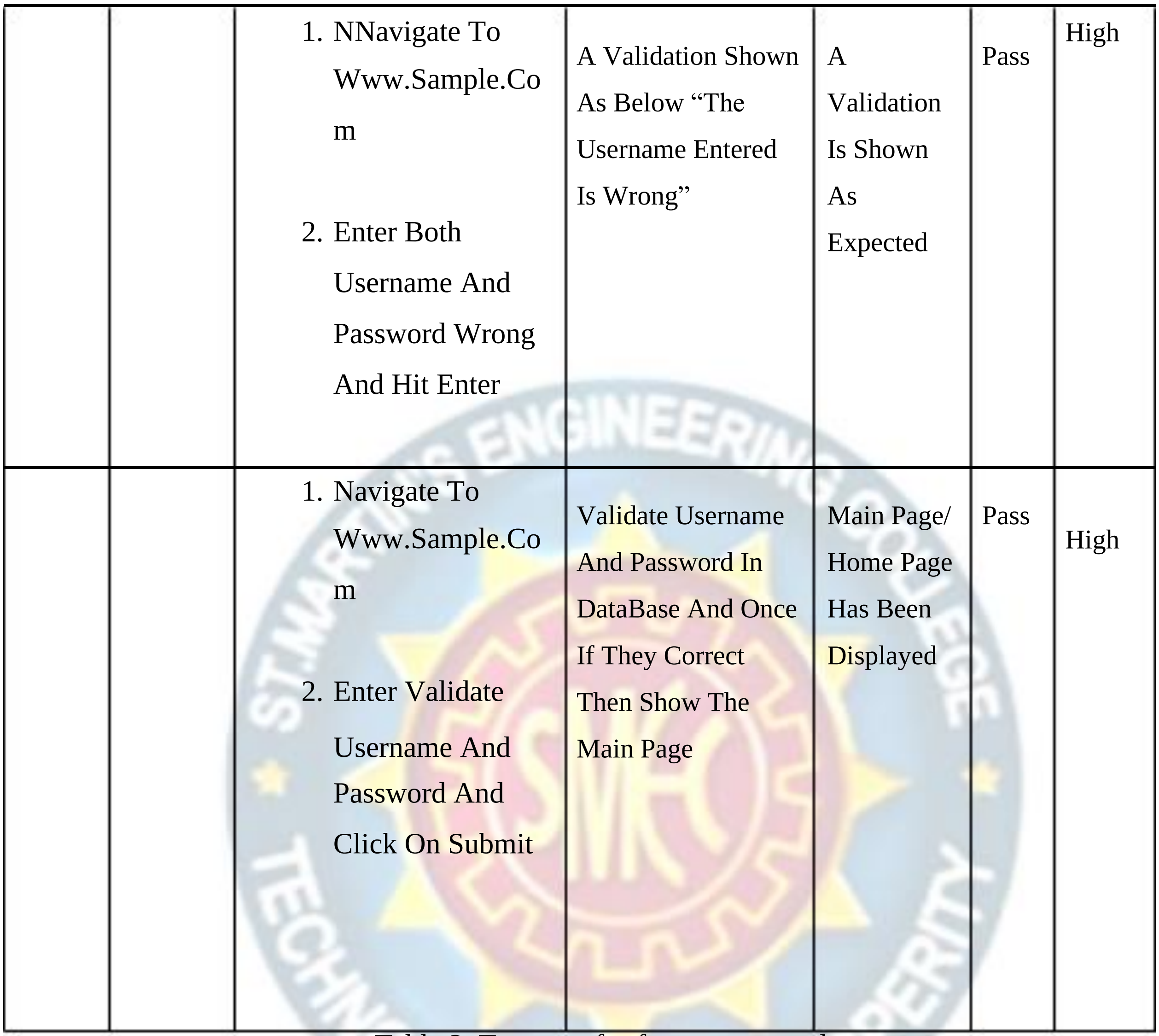
form **7.2.3 Test case for Change Password:**

When the old password does not match with the new password ,then this results in displaying an error message as “ OLD PASSWORD DOES NOT MATCH WITH THE NEW PASSWORD”.

7.2.4 Test case for Forget Password:

When a user forgets his password he is asked to enter Login name, ZIP code, Mobile number. If these are matched with the already stored ones then user will get his Original password.

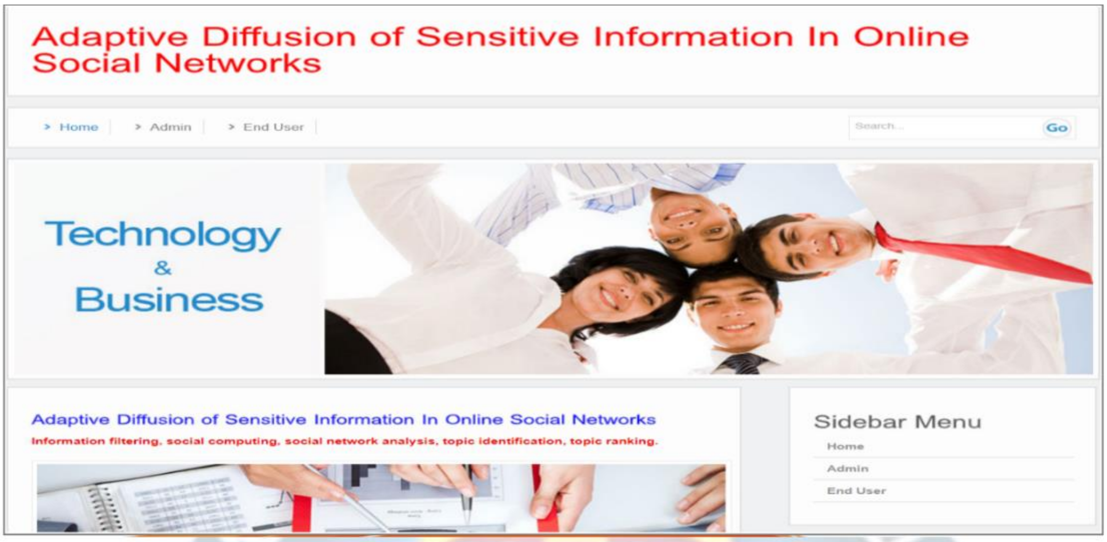
Module	Functiona	Test Case	Expected Results	Actual Resu	Result	Priority
User	Login Use	1. Navigate To Www 2. 2.Click On Submit	A Validation Should	A Validation	Pass	High
		1. aNavigate To Www 1. 2. Click On Submit	A Validation Should	A Validation	Pass	High



		1. NNavigate To Www.Sample.Com 2. Enter Both Username And Password Wrong And Hit Enter	A Validation Shown As Below “The Username Entered Is Wrong”	A Validation Is Shown As Expected	Pass	High
		1. Navigate To Www.Sample.Com 2. Enter Validate Username And Password And Click On Submit	Validate Username And Password In DataBase And Once If They Correct Then Show The Main Page	Main Page/ Home Page Has Been Displayed	Pass	High

Table 3. Test case for forgot password

CHAPTER 8: OUTPUT SCREENS

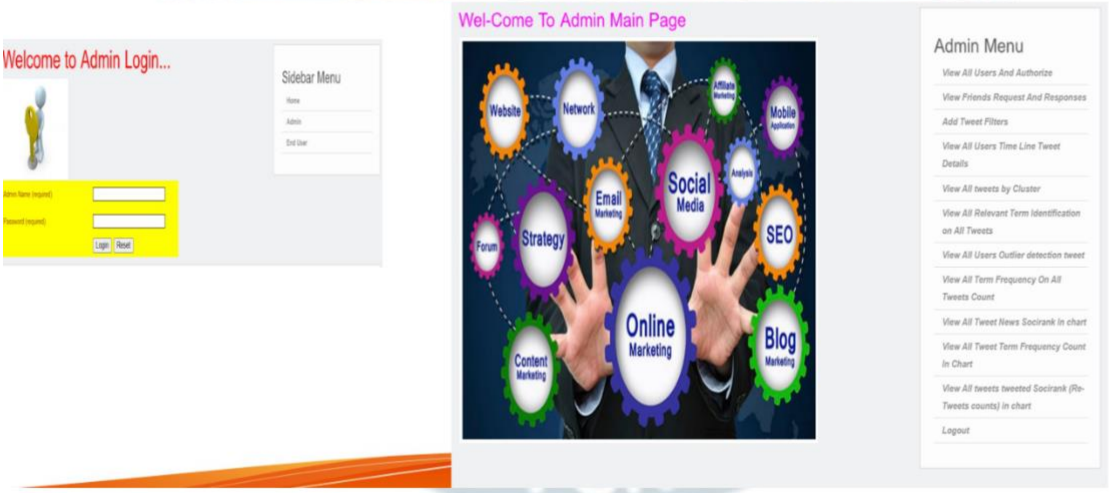


Adaptive Diffusion of Sensitive Information In Online Social Networks

Adaptive Diffusion of Sensitive Information In Online Social Networks

Information filtering, social computing, social network analysis, topic identification, topic ranking.

Figure 16. Home page



Welcome to Admin Login...

Admin Name (required)

Password (required)

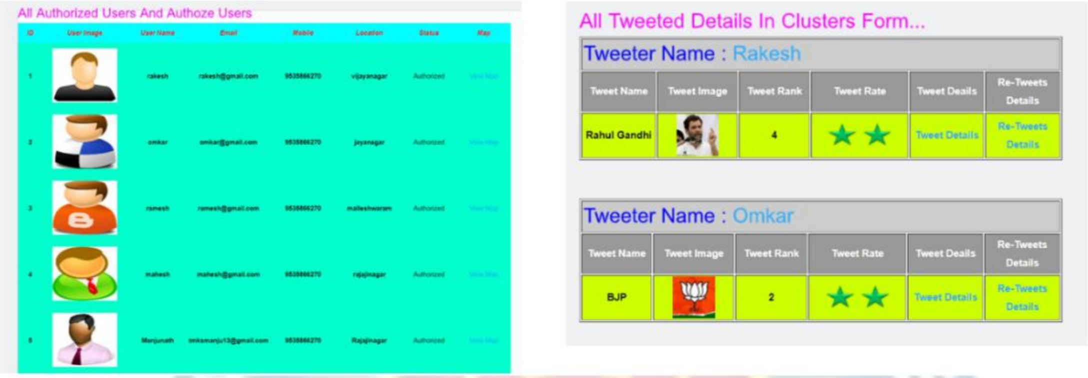
Login
Reset
Sidebar Menu
Home
Admin
End User

Wel-Come To Admin Main Page

Website
Network
Affiliate Marketing
Mobile Application
Email Marketing
Social Media
Analysis
SEO
Forum
Strategy
Online Marketing
Blog Marketing
Content Marketing
Admin Menu
View All Users And Authorize
View Friends Request And Responses
Add Tweet Filters
View All Users Time Line Tweet Details
View All tweets by Cluster
View All Relevant Term Identification on All Tweets
View All Users Outlier detection tweet
View All Term Frequency On All Tweets Count
View All Tweet News Socirank In chart
View All Tweet Term Frequency Count In Chart
View All tweets tweeted Socirank (Re-Tweets counts) in chart
Logout

Figure 17. Admin Page

CHAPTER 9: EXPERIMENTAL RESULTS



All Authorized Users And Authoze Users

ID	User Image	User Name	Email	Mobile	Location	Status	Map
1		rakesh	rakesh@gmail.com	9535866270	vijayanagar	Authorized	View Map
2		omkar	omkar@gmail.com	9535866270	jayanagar	Authorized	View Map
3		ramesh	ramesh@gmail.com	9535866270	malleshwaram	Authorized	View Map
4		mahesh	mahesh@gmail.com	9535866270	rajajinagar	Authorized	View Map
5		Manjunath	tmksmanju13@gmail.com	9535866270	Rajajinagar	Authorized	View Map

All Tweeted Details In Clusters Form...

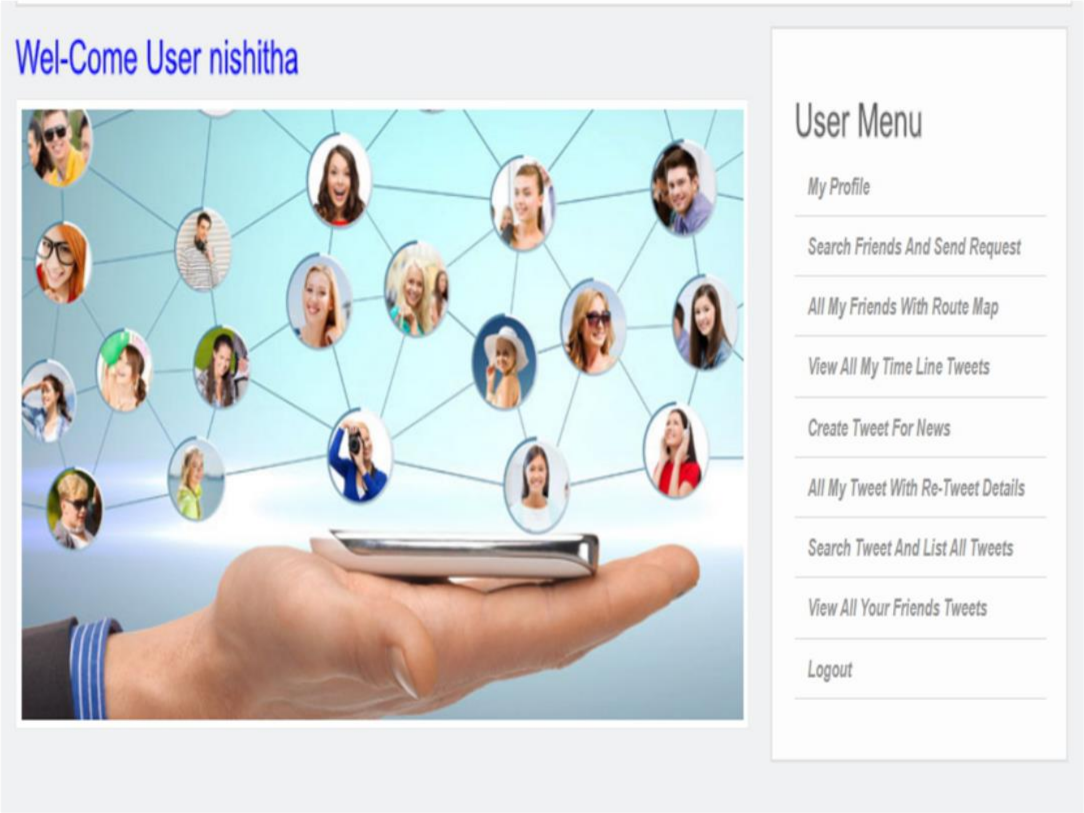
Tweeter Name : Rakesh

Tweet Name	Tweet Image	Tweet Rank	Tweet Rate	Tweet Deails	Re-Tweets Details
Rahul Gandhi		4	★ ★	Tweet Details	Re-Tweets Details

Tweeter Name : Omkar

Tweet Name	Tweet Image	Tweet Rank	Tweet Rate	Tweet Deails	Re-Tweets Details
BJP		2	★ ★	Tweet Details	Re-Tweets Details

Figure 18. Authorizing Users from Admin page



Wel-Come User nishitha

User Menu
My Profile
Search Friends And Send Request
All My Friends With Route Map
View All My Time Line Tweets
Create Tweet For News
All My Tweet With Re-Tweet Details
Search Tweet And List All Tweets
View All Your Friends Tweets
Logout

Figure 19. User Page

CHAPTER 10: CONCLUSION AND FUTURE ENHANCEMENT

Conclusion

In this paper, we study the problem of constraining the diffusion of sensitive information in social networks while preserving the diffusion of non-sensitive information. We model the diffusion constraining measures as the variations of diffusion probabilities via social links and model the problem of interest as adaptively determining the probability variations through a constrained minimization problem in multiple rounds. We utilize the CCMAB framework to jointly design our solutions in the fully known and semi known networks. Over the fully known network, we propose the CCMAB based algorithm ADFN to efficiently determine the probability variations via social links. Over the semi-known network, for tackling the challenge of unknown diffusion abilities of partial users, we propose the algorithm ADSN to iteratively learn the unknown diffusion abilities and determine the probability variations based on the learned diffusion abilities in each round. The analysis of regret bound and extensive experiments have been conducted to justify the superiority of our solutions. In addition, in the current work, we define the constraint of maintaining the sum of diffusion probabilities via edges in the objective problem, for the aim of preserving the global diffusion ability of the whole network on diffusing non sensitive information. In the future work, we will explore other relevant solutions such as simultaneously minimizing the sensitive information diffusion and maximizing the no sensitive information diffusion.

Future Enhancement

In addition, in the current work, we define the constraint of maintaining the sum of diffusion probabilities via edges in the objective problem, for the aim of preserving the global diffusion ability of the whole network on diffusing non sensitive information. In the future work, we will explore other relevant solutions such as simultaneously minimizing the sensitive information diffusion and maximizing the no sensitive information diffusion.

CHAPTER:11 REFERENCES

1. Y. Li, J. Fan, Y. Wang, and K. L. Tan, "Influence maximization on social graphs: A survey", in IEEE Transactions on Knowledge and Data Engineering (TKDE), vol. 30, no. 10, pp. 1852-1872, 2018.

2. L. Sun, W. Huang, P. S. Yu, and W. Chen, " Multi-round influence maximization", in Proc. ACM SIGKDD, 2018.

3. Q. Shi, C. Wang, J. Chen, Y. Feng, and C. Chen, "Post and repost: A holistic view of budgeted influence maximization", in Neurocomputing, vol. 338, pp. 92-100, 2019.

4. X. Wu, L. Fu, Y. Yao, X. Fu, X. Wang, and G. Chen, "GLP: a novel framework for group-level location promotion in Geo-social networks", in IEEE/ACM Transactions on Networking (TON), vol. 26, no. 6, pp. 1-14, 2018.

5. Y. Lin, W. Chen, and J. C. Lui, "Boosting information spread: An algorithmic approach", in Proc. IEEE ICDE, 2017.

6. Y. Zhang, and B. A Prakash, "Data-aware vaccine allocation over large networks", in ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 10, no. 2, article 20, 2015.

7. Q. Shi, C. Wang, J. Chen, Y. Feng, and C. Chen, "Location driven influence maximization: Online spread via offline deployment", in Knowledge-Based Systems, vol. 166, pp. 30-41, 2019.

8. H. T. Nguyen, T. P. Nguyen, T. N. Vu, and T. N. Dinh, "Outward influence and cascade size estimation in billion-scale networks, in Proc. ACM SIGMETRICS, 2017.

9. B. Wang, G. Chen, L. Fu, L. Song, and X. Wang, "Drimux: Dynamic rumor influence minimization with user experience in social networks", in IEEE Transactions on Knowledge and Data Engineering (TKDE), vol. 29, no. 10, pp. 2168-2181, 2017.

10. Q. Shi, C. Wang, D. Ye, J. Chen, Y. Feng, and C. Chen, “Adaptive Influence Blocking: Minimizing the Negative Spread by Observation-based Policies”, in Proc. IEEE ICDE, 2019.

11. S. Wen, J. Jiang, Y. Xiang, S. Yu, W. Zhou, and W. Jia, “To shut them up or to clarify: Restraining the spread of rumors in online social networks”, in IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 12, pp. 3306-3316, 2014.

12. X. He, G. Song, W. Chen, and Q. Jiang, “Influence blocking maximization in social networks under the competitive linear threshold model”, in Proc. SIAM SDM, 2012.

A

PROJECT REPORT

On

CRIME DATA ANALYSIS MAPPING USING DATA MINING

Submitted by

MS. DUSA SHEELA	(17K81A1211)
MS. SWATHI EDIGINTI	(17K81A1212)
MS. DONGARI HARSHINI	(17K81A1210)
MR. AKASH GUPTE	(17K81A1215)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Under The Guidance of

MR.G.SIVA KRISHNA

ASSIT. PROFESSOR

DEPARTMENT OF INFORMATION TECHNOLOGY

ST.MARTIN’S ENGINEERING COLLEGE

An Autonomous Institute

Dhulapally, Secunderabad – 500 100

JUNE 2021

BONAFIDE CERTIFICATE

This is to certify that the project entitled **CRIME ANALYSIS MAPPING INTRUSION DETECTION USING DATA MINING**, is being submitted by **DUSA SHEELA (17K81A1211), SWATHI EDIGINTI (17K81A1212),DONGARI HARSHINI (17K81A1210), AKASH GUPTE (17K81A1215)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY IN** INFORMATION TECHNOLOGY is recorded of bonafide work carried out by them. The result embodied in this report have been verified and found satisfactory.

Project Guide	Head of the Department
Mr. G. SIVA KRISHNA	**DR.R.NAGARAJU**
Department of Information Technology	Department of Information Technology

Internal Examiner External Examiner

Place:

Date:

DECLARATION

We, the student of **Bachelor of Technology** in Department of Information Technology, session: 2017 – 2021, St. Martin's Engineering College, Dhulapally, Kompally, Secunderabad, hereby declare that work presented in this Project Work entitled **Crime Analysis Mapping, intrusion detection Using Data Mining** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. This result embodied in this project report has not been submitted in any university for award of any degree.

DUSA SHEELA	**(17K81A1211)**
SWATHI EDIGINTI	**(17K81A1212)**
DONGARI HARSHINI	**(17K81A1210)**
AKASH GUPTE	**(17K81A1215)**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragements and guidance have crowded effects with success.

We extended our deep sense of gratitude to Principal**, Dr. P. SANTOSH KUMAR PATRA**, St. Martin's Engineering College, Dhulapally, for permitting us to undertake this project.

We are also thankful to **Dr. Mr. R. NAGARAJU**, Head of the Department, Information Technology , St. Martin's Engineering College, Dhulapally, for his support and guidance throughout our project and as well as our project coordinator **Mr. D. BABU RAO**, Associate Professor, Department of Information Technology for his valuable support.

We would like to express our sincere gratitude and indebtedness to our project supervisor **MR.G.SIVA KRISHNA** Assit. Professor, Department of Information Technology, St. Martin's Engineering College, Dhulapally, for her support and guidance throughout our project.

Finally, we express thanks to all those who have helped us successfully to completing this project. Furthermore, we would like to thank our family and friends for their moral support and encouragement.

We express thanks to all those who have helped us in successfully completing the project.

DUSA SHEELA	**(17K81A1211)**
SWATHI EDIGINTI	**(17K81A1212)**
DONGARI HARSHINI	**(17K81A1210)**
AKASH GUPTE	**(17K81A1215)**

TABLE OF CONTENTS

ABSTRACT

Data Mining plays a key role in Crime Analysis. There are many different algorithms mentioned in previous research papers, among them are the virtual identifier, pruning strategy, support vector machines, and apriori algorithms. VID is to find relation between record and vid. The apriori algorithm helps the fuzzy association rules algorithm and it takes around six hundred seconds to detect a mail bomb attack. In this research paper, we identified Crime mapping analysis based on KNN (K – Nearest Neighbor) and ANN (Artificial Neural Network) algorithms to simplify this process. Crime Mapping is conducted and Funded by the Office of Community Oriented Policing Services (COPS). Evidence based research helps in analyzing the crimes. We calculate the crime rate based on the previous data using data mining techniques. Crime Analysis uses quantitative and qualitative data in combination with analytic techniques in resolving the cases. For public safety purposes, the crime mapping is an essential research area to concentrate on. We can identity the most frequently crime occurring zones with the help of data mining techniques. In Crime Analysis Mapping, we follow the following steps in order to reduce the crime rate: 1) Collect crime data 2) Group data 3) Clustering 4) Forecasting the data. Crime Analysis with crime mapping helps in understanding the concepts and practice of Crime Analysis in assisting police and helps in reduction and prevention of crimes and crime disorders.

LIST OF FIGURES

1.INTRODUCTION

Crimes are one of the most predominant problems that is happening in most of the urban areas in the world. There are a lot of different types of crimes that happen, including robbery, theft of vehicles, etc. As crime increases, the investigation process gets longer and more complicated. Crime has been increasing day by day and everyone in the world is trying to figure out how to manage the crime rate and to work on certain cases, most of the people are trying to store the data for future reference. Human errors can occur at any point of time. There are different types of crimes law enforcement levels, such as traffic violations, sex crime, theft, violent crime, arson, gang/drug offenses, cybercrime. Different crime data mining techniques are proposed among each of them including entity extraction, clustering techniques, Association rule mining. Crime zones can be identified by occurrence of crime, by using hotspots. Patrol is needed at these hotspot areas. The data mining tool helps in reducing the crime rate drastically.

1.1 PROJECT OVERVIEW

The use of information mining methods helps in resolving most complicated criminal cases. One of the best methods is crime analysis with crime mapping. Crime analysis with crime mapping helps in understanding the concepts and practices of crime analysis in assisting police and helps in the reduction and prevention of crimes and crime disorders.

Crime mapping is conducted and funded by the Office of Community Oriented Policing Services (COPS). Evidence based research helps in analyzing the crimes. We calculate the crime rate based on the previous data using data mining techniques. Crime analysis uses quantitative and qualitative data and analytic techniques in resolving the cases.

1.2 PROJECT OBJECTIVE

For public safety purposes, the crime mapping is an essential research area to concentrate on. We can identity the highest risk crime zones with the help of data mining techniques. Security is considered to be a major issue in networks as the usage of networks has increased drastically. Data mining is used in network intrusions because of the following reasons:

To process huge amounts of data.

- It is suitable to detect the ignored and hidden information at any point of time.

1.3 SCOPE OF THE PROJECT

The Intrusion Detection System is applied to detect intrusion network related issues. Machine Learning is to deal with design and development of algorithms and in a way to allow computers to learn about the data that is fed to the machine. Machine learning is applied in areas like bioinformatics, to find the pattern match in DNA and to check for gene related data. Detection of attacks and false alarms is the main task of the Intrusion Detection System. It helps to identify authentic and falsely authentic users for maintaining privacy. False alarms and intrusion is more in recent days, and the techniques they are following is more different than the usual ones. Using data mining, intrusion detection can be resolved. For example, it is used by the US Army for tactical environments to handle constraints in military systems. Distributed denial of service attacks is one of the most common attacks on internet sites. Intrusion detection helps in identifying the network related activity and using this we can provide security against DOS attacks. There are two types of intrusion detection methods available here: misuse detection which is based on exact pattern match, and anomaly detection which requires more training related to artificial intelligence. Fuzzy intrusion recognition engine is an anomaly IDS which identifies malicious sites which are not trustworthy using fuzzy systems. Here, 3-D packet count with a 15-minute interval is used to find the regular network connections and try to indicate the intrusions, if any, at that point of time.

1.4 ORGANIZATION OF CHAPTERS

1.4.1 INTRODUCTION

Crimes are one of the most predominant problems that is happening in most of the urban areas in the world. There are a lot of different types of crimes that happen, including robbery, theft of vehicles, etc. As crime increases, the investigation process gets longer and more complicated. The use of information mining methods helps in resolving most complicated criminal cases. One of the best methods is crime analysis with crime mapping. Crime analysis with crime mapping helps in understanding the concepts and practices of crime analysis in assisting police and helps in the reduction and prevention of crimes and crime disorders. Crime mapping is conducted and funded by the Office of Community Oriented Policing Services (COPS). Evidence based research helps in analyzing the crimes. We calculate the crime rate based on the previous data using data mining techniques. Crime analysis uses quantitative and qualitative data and analytic techniques in resolving the cases. The health of a computer is analogous to that of human health: it needs protection. Fuzzy cognitive maps and fuzzy rules are used for and support causal knowledge acquisition. Computer crimes are increasing day by day, and with it the need to protect our data has also increased. The intelligence intrusion detection system is also developed as part of intrusion detection. Fuzzy cognitive map values are changed from time to time and there are causality links between nodes that represent the directed edges. In this, they used clustering techniques to identify crime patterns. In a geographical area, the need to identify the crime at a point in time is known as clustering. We can use a map to identify the plot. The largest challenge is with free text fields. It is difficult to convert the free text fields into data, but the K means technique is used for this purpose in this paper. Operational data can be extracted and transformed to another form using this technique. By doing this, it is much easier to find out the crime patterns for the detectives to identify the frauds. To identify the difference between the abnormal and normal activities, the intrusion detection system is very important. We use fuzzy logic in intrusion detection. This system along with fuzzy logic uses association rule mining, the fastest technique is to use prefix trees. It also helps in transforming and extracting the data by using timestamp and source host details. We can identify the attacks by using traffic in audit logs. TCP header is used for more improved efficiency. The pruning step is used to reduce the running time and to increase the accuracy of the system.

1.4.2 LITERATURE SURVEY

A criminal act can encompass a wide range of activities, from civil infractions such as illegal parking to internationally organized mass murder such as the 9/11 attacks. Law enforcement agencies across the US compile crime statistics using well-established standards such as the FBI's Uniform Crime Reporting System and its successor, the National Incident-Based Reporting System (www.fbi.gov/hq/cjisd/ucr.htm). As well as other criteria defined by jurisdictional needs and requirements.

Intrusion detection using data mining techniques.

As the network dramatically extended, security considered as major issue in networks. Internet attacks are increasing, and there have been various attack methods, consequently. Intrusion detection systems have been used along with the data mining techniques to detect intrusions. In this work we aim to use data mining techniques including classification tree and support vector machines for intrusion detection. As results indicate, C4.5 algorithm is better than SVM in detecting network intrusions and false alarm rate in KDD CUP 99 dataset. In recent years, internet and computers have been utilized by many people all over the world in several fields. In order to come up with efficiency and up to date issues, most organizations rest their applications and service items on internet. On the other hand, network intrusion and information safety problems are ramifications of using internet. For instance, on February 7th, 2000 the first DoS attacks of great volume where launched, targeting the computer systems of large companies like Yahoo!, eBay, Amazon, CNN, ZDnet and Dadet**.** For instance, on February 7th, 2000 the first DOS attacks of great volume where launched, targeting the computer systems of large companies like Yahoo!, eBay, Amazon, CNN, ZDnet and Dadet . In other words, network intrusion is considered as new weapon of world war. Therefore, it has become the general concern of the computer society to detect and to prevent intrusions efficiently. There are many methods to strengthen the network security at the moment, such as encryption, VPN, firewall, etc., but all of these are too static to give an effective protection. However, intrusion detection is a dynamic one, which can give dynamic protection to the network security in monitoring, attack and counter-attack. Thus, Intrusion Detection System (IDS) has been applied to detect intrusion network. Intrusion Detection technology can be defined as a system that identifies and deals with the malicious use of computer and network resources. In the case of detecting data target, intrusion detecting system can be classified as host-based and network-based . HOST-BASED IDS: Its data come from the records of various host activities, including audit record of operation system, system logs, application programs information, and so on. NETWORK-BASED IDS: Its data is mainly collected network generic stream going through network segments, such as: Internet packets.

1.4.3 SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements

For developing the application the following are the Software Requirements:

Operating Systems supported

1. Windows 7 and above.

Technologies and Languages used to Develop

1. Java
2. SQL

Debugger and Emulator

- Any Browser (Particularly Chrome)

Hardware Requirements

For developing the application the following are the Hardware Requirements:

- Processor: Pentium IV or higher
- RAM:4GB
- Hard Disk:1TB

1.4.4 SOFTWARE DEVELOPMENT ANALAYASIS

JAVA

Java programming language was originally developed by Sun Microsystems which was initiated by James Gosling and released in 1995 as core component of Sun Microsystems' Java platform (Java 1.0 [J2SE]).

The latest release of the Java Standard Edition is Java SE 8. With the advancement of Java and its widespread popularity, multiple configurations were built to suit various types of platforms. For example: J2EE for Enterprise Applications, J2ME for Mobile Applications.

The new J2 versions were renamed as Java SE, Java EE, and Java ME respectively. Java is guaranteed to be **Write Once, Run Anywhere.**

Java is −

- **Object Oriented** − In Java, everything is an Object. Java can be easily extended since it is based on the Object model.
- **Platform Independent** − Unlike many other programming languages including C and C++, when Java is compiled, it is not compiled into platform specific machine, rather into platform independent byte code. This byte code is distributed over the web and interpreted by the Virtual Machine (JVM) on whichever platform it is being run on.
- **Simple** − Java is designed to be easy to learn. If you understand the basic concept of OOP Java, it would be easy to master.
- **Secure** − With Java's secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.
- **Architecture-neutral** − Java compiler generates an architecture-neutral object file format, which makes the compiled code executable on many processors, with the presence of Java runtime system.
- **Portable** − Being architecture-neutral and having no implementation dependent aspects of the specification makes Java portable. Compiler in Java is written in ANSI C with a clean portability boundary, which is a POSIX subset.
- **Robust** − Java makes an effort to eliminate error prone situations by emphasizing mainly on compile time error checking and runtime checking.
- **Multithreaded** − With Java's multithreaded feature it is possible to write programs that can perform many tasks simultaneously. This design feature allows the developers to construct interactive applications that can run smoothly.

- **Interpreted** − Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light-weight process.
- **High Performance** − With the use of Just-In-Time compilers, Java enables high performance.
- **Distributed** − Java is designed for the distributed environment of the internet.
- **Dynamic** − Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry extensive amount of run-time information that can be used to verify and resolve accesses to objects on run-time.

SQL (Structured Query Language)

SQL (Structured Query Language) is a standardized programming language that's used to manage relational databases and perform various operations on the data in them. Initially created in the 1970s, SQL is regularly used not only by database administrators, but also by developers writing data integration scripts and data analysts looking to set up and run analytical queries.The uses of SQL include modifying database table and index structures; adding, updating and deleting rows of data; and retrieving subsets of information from within a database for transaction processing and analytics applications. Queries and other SQL operations take the form of commands written as statements -- commonly used SQL statements include select, add, insert, update, delete, create, alter and truncate.SQL became the de facto standard programming language for relational databases after they emerged in the late 1970s and early 1980s. Also known as SQL databases, relational systems comprise a set of tables containing data in rows and columns. Each column in a table corresponds to a category of data -- for example, customer name or address -- while each row contains a data value for the intersecting column.

1.4.5 PROJECT SYSTEM DESIGN

1.ADMIN :

Uploads the crime dataset on which the crime analysis is done.

2. DETECTOR :

Detector performs the crime analysis and obtains the resultant crime rate in a particular place and year.

1.4.6 PROJECT CODING

JAVA

Java is a general-purpose, class-based, object-oriented programming language designed for having lesser implementation dependencies. It is a computing platform for application development. Java is fast, secure, and reliable, therefore. It is widely used for developing Java applications in laptops, data centers, game consoles, scientific supercomputers, cell phones, etc. Java Platform is a collection of programs that help programmers to develop and run Java programming applications efficiently. It includes an execution engine, a compiler, and a set of libraries in it. It is a set of computer software and specifications. James Gosling developed the Java platform at Sun Microsystems, and the Oracle Corporation later acquired it.

Java Database Connectivity

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API. Using JDBC, it is easy to send SQL statements to virtually any relational database. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. The combinations of Java and JDBC lets a programmer write it once and run it anywhere.

What Does JDBC Do?

Simply put, JDBC makes it possible to do three things:

- Establish a connection with a database
- Send SQL statements
- Process the results.

Decision Tree

Decision Trees are a type of Supervised **Machine Learning** where the data is continuously split according to a certain parameter. The **tree** can be explained by two entities, namely **decision** nodes and leaves

Support Vector Machine

"**Support Vector Machine**" (**SVM**) is a supervised **machine learning** algorithm which can be used for both classification or regression challenges.
However, it is mostly used in classification problems.

1.4.7 PROJECT 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.

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.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program

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.

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.

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.

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.

1.4.8 OUTPUT SCREENS

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

output that is needed to meet the requirements.Select methods for presenting information.Create document, report, or other formats that contain information produced by the system 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

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.

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.

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.

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

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.

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.

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.

1.4.9 CONCLUSIONS

With the assistance of these devices, the wrongdoing information will be nourished to the information digging device for investigation and afterward comes about for two unique models will be recorded. With the assistance of the SAM instrument/tools, we will maintain a strategic distance from the distinction in the outcome and after that the subsequent information will be utilized for the finding the relations amongst those et cetera. Along these lines we will lessen false positives and false negatives in the field of the interruption identification framework utilizing the information mining in the field of wrongdoing information examination.

2. LITERATURE SURVEY

2.1 SURVEY ON BACKGROUND

Crime pattern detection using data mining

Data mining can be used to model crime detection problems. Crimes are a social nuisance and cost our society dearly in several ways. Any research that can help in solving crimes faster will pay for itself. About 10% of the criminals commit about 50% of the crimes. Here we look at use of clustering algorithm for a data mining approach to help detect the crimes patterns and speed up the process of solving crime. We will look at k-means clustering with some enhancements to aid in the process of identification of crime patterns. We applied these techniques to real crime data from a sheriff's office and validated our results. We also use semi-supervised learning technique here for knowledge discovery from the crime records and to help increase the predictive accuracy. We also developed a weighting scheme for attributes here to deal with limitations of various out of the box clustering tools and techniques. This easy to implement data mining framework works with the geospatial plot of crime and helps to improve the productivity of the detectives and other law enforcement officers. It can also be applied for counter terrorism for homeland security.

Historically solving crimes has been the prerogative of the criminal justice and law enforcement specialists. With the increasing use of the computerized systems to track crimes, computer data analysts have started helping the law enforcement officers and detectives to speed up the process of solving crimes. Here we will take an interdisciplinary approach between computer science and criminal justice to develop a data mining paradigm that can help solve crimes faster. More specifically, we will use clustering based models to help in identification of crime patterns

An improved algorithm for fuzzy data mining for intrusion detection.

We have been using fuzzy data mining techniques to extract patterns that represent normal behavior for intrusion detection. We describe a variety of modifications that we have made to the data mining algorithms in order to improve accuracy and efficiency. We use sets of fuzzy association rules that are mined from network audit data as models of "normal behavior." To detect anomalous behavior, we generate fuzzy association rules from new audit data and compute the similarity with sets mined from "normal" data. If the similarity values are below a threshold value, an alarm is issued. We describe an algorithm for computing fuzzy association rules based on Borgelt's (2001) prefix trees, modifications to the computation of support and confidence of fuzzy rules, a new method for computing the similarity of two fuzzy rule sets, and feature selection and optimization with genetic algorithms. Experimental results demonstrate that we can achieve better running time and accuracy with these modifications.An intrusion detection system (IDS) is a component of the computer and information security framework. Its main goal is to differentiate between normal activities of the system and behavior that can be classified as suspicious or intrusive [1]. IDS's are needed because of the large number of incidents reported increases every year and the attack techniques are always improving.

A comparative study of data mining algorithms for network intrusion detection

Data mining techniques are being applied in building intrusion detection systems to protect computing resources against unauthorised access. In this paper, the performance of three well known data mining classifier algorithms namely, ID3, J48 and Naive Bayes are evaluated based on the 10-fold cross validation test. Experimental results using the KDDCuppsila99 IDS data set demonstrate that while Naive Bayes is one of the most effective inductive learning algorithms, decision trees are more interesting as far as the detection of new attacks is concerned.Modern computer networks must be equipped with appropriate security mechanisms in order to protect the information resources maintained by them. Intrusion detection systems (IDSs) are integral parts of any well configured and managed computer network systems. An IDS is a combination of software and hardware components, capable of monitoring different activities in a network and analyze them for signs of security threats. There are two major approaches to intrusion detection: anomaly detection and misuse detection. Misuse detection uses patterns of well known intrusions to match and identify unlabeled data sets. In fact, many commercial and open source intrusion detection systems are misuse based. Anomaly detection, on the other hand, consists of building models from normal data which can be used to detect variations in the observed data from the normal model. The advantage with anomaly detection algorithms is that they can detect new forms of attacks which might deviate from the normal behaviour.

2.2 CONCLUSIONS ON SURVEY

This study aims to understand the crime rate in particular place and if the predicted crime rate is high certain reduction and preventive measures are taken by the law enforcement agencies. K means clustering, artificial neural network(ANN),k nearest neighbour(KNN) algorithms are used in efficient prediction of crime patterns according to its type , time and geological location of the occurred crime incident. The wrongdoing information will be nourished to the information digging device for investigation and afterward comes about for two unique models will be recorded. With the assistance of the SAM instrument/tools, we will maintain a strategic distance from the distinction in the outcome and after that the subsequent information will be utilized for the finding the relations amongst the crime inccidents. Along these lines we will lessen false positives and false negatives in the field of the interruption identification framework utilizing the information mining in the field of wrongdoing information examination.

3.SOFTWARE AND HARDWARE REQUIREMENTS

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

REQUIREMENT SPECIFICATION

Functional Requirements

- Graphical User interface with the User.

3.1 SOFTWARE REQUIREMENTS

For developing the application the following are the Software Requirements:

Operating Systems supported

- Windows 7 and above

Technologies and Languages used to Develop

- Python

Debugger and Emulator

- Any Browser (Particularly Chrome)

3.2 HARDWARE REQUIREMENTS

For developing the application the following are the Hardware Requirements:

- Processor: Pentium IV or higher
- RAM: 8GB
- Hard Disk: 1 TB

4.SOFTWARE DEVELOPMENT ANALYSIS

4.1 EXISTING SYSTEM

Crime has been increasing day by day and everyone in the world is trying to figure out how to manage the crime rate and to work on certain cases, most of the people are trying to store the data for future reference. Human errors can occur at any point of time. There are different types of crimes law enforcement levels, such as traffic violations, sex crime, theft, violent crime, arson, gang/drug offenses, cybercrime. Different crime data mining techniques are proposed among each of them including entity extraction, clustering techniques, Association rule mining. Crime zones can be identified by occurrence of crime, by using hotspots. Patrol is needed at these hotspot areas. The data mining tool helps in reducing the crime rate drastically.

4.2. PROPOSED SYSTEM

Crime Mapping helps in understanding the concepts and practice of Crime Analysis in assisting police and helps in reduction and prevention of crimes and crime disorders using data mining tools. We can use data mining tools involved using ANN (Artificial Neural Networks) and KDD (Knowledge Discovery in Databases). We collect the data from police department and try to get each and every detail, like the person’s name, height, age, sex, fingerprint details, and pattern identification number for similar types of cases. Once we get the information, we start to process the data.We get a lot of unnecessary data along with the required data. But before we start processing the data using data mining techniques and tools, we need to identify unnecessary data and remove those kinds of data to reduce or to avoid the confusion. We use the SAM tool to identify the pattern in the crime data. Here we have two classifications of data: supervised and unsupervised data. We take the data that has all the details about the case and we try to solve the other cases by training using this supervised data. We mainly collect the attributes information, like eye color, fingerprint details, characteristics, dimensions, or other features.

4.3 SYSTEM ARCHITECTURE

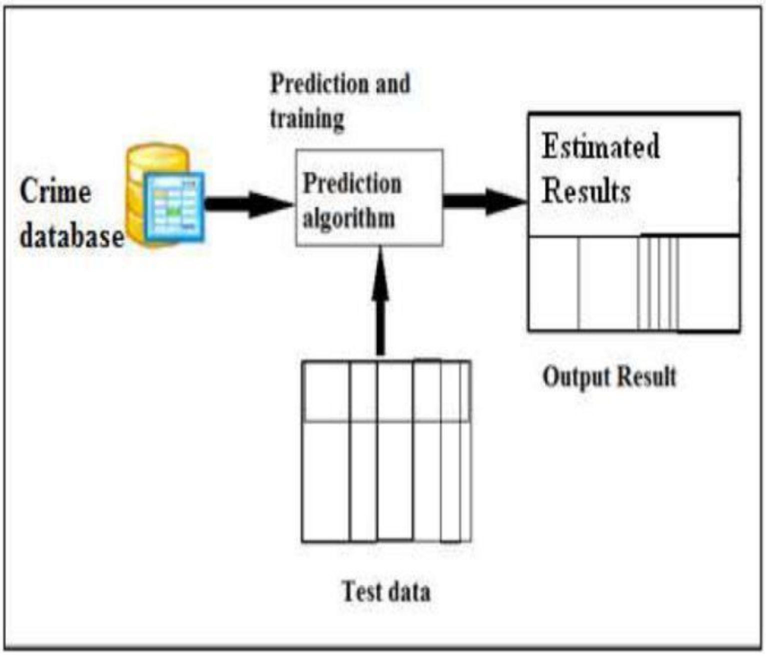


FIG 4.3.1 SYSTEM ARCHITECTURE

Intelligent criminal identification system called ICIS which can potentially distinguish a criminal in accordance with the observations collected from the crime location for a certain class of crimes. The system uses existing evidences in situations for identifying a criminal by clustering mechanism to segment crime data in to subsets, and the Nave Bayesian classification has used for identifying possible suspect of crime incidents. ICIS has been used the communication power of multi agent system for increasing the efficiency in identifying possible suspects. In order to describe the system ICIS is divided to user interface, managed bean, multi agent system and database. SQL Database is used for implementing of database, and identification of crime patterns has been implemented using Java platform.The collected crime dataset is tested using the prediction algorithms such as k nearest neighbour(KNN) and artificial neural network(ANN) algorithms which has test data sets and trained data sets from which the estimated results are obtained.

The method included are data collection, classification, pattern identification, prediction and visualization.

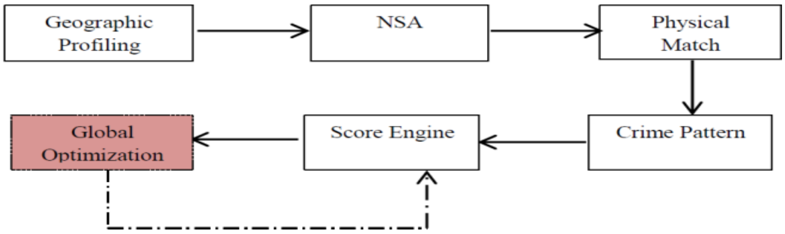


FIG 4.3.2 PROCESS OF CRIME PATTERN EXTRACTION

4.4 MODULE FUNCTIONALITY

Home: It is the main page where user Admin can login with user name and password.

Upload: We the upload the dataset here, where we browse and selects the data and uploads it.

View Dataset: The data set will be view with information.

Logout: And after the information collected logout.

same action repeats for detector login but it will do
analysis by

Clustering: technology for auditing. Automating fraud filtering can be of great value to continuous audits.

Crime Analysis: It will analyze data with different age groups at different areas and different years.

5. PROJECT SYSTEM DESIGN

5.1 UML DIAGRAM

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

USE-CASE DIAGRAM

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

FIG.5.1.2 USE CASE DIAGRAM OF DETECTOR.

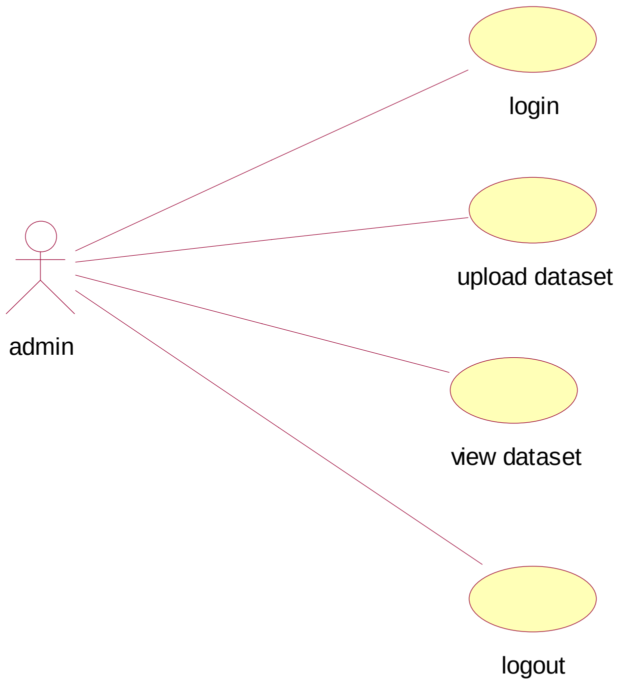


FIG.5.1.1 USE CASE DIAGRAM OF ADMIN.

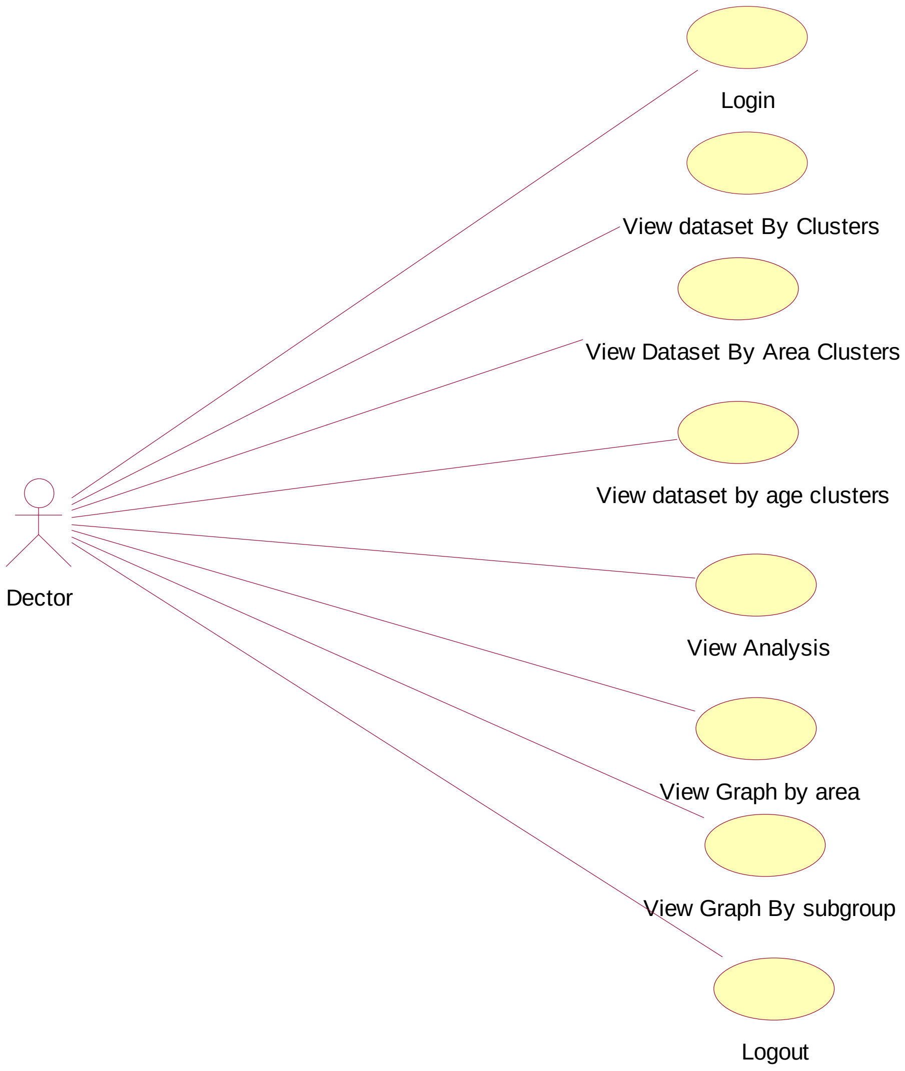


FIG.5.1.2 USE CASE DIAGRAM OF DETECTOR.

CLASS DIAGRAM

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

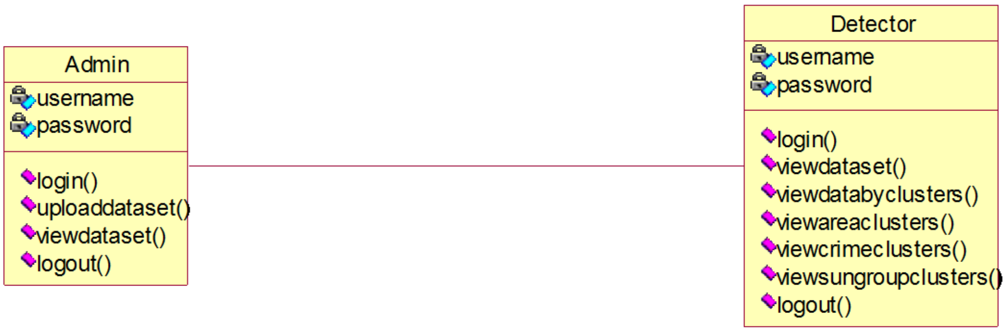


FIG.5.1.3 CLASS DIAGRAM

SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

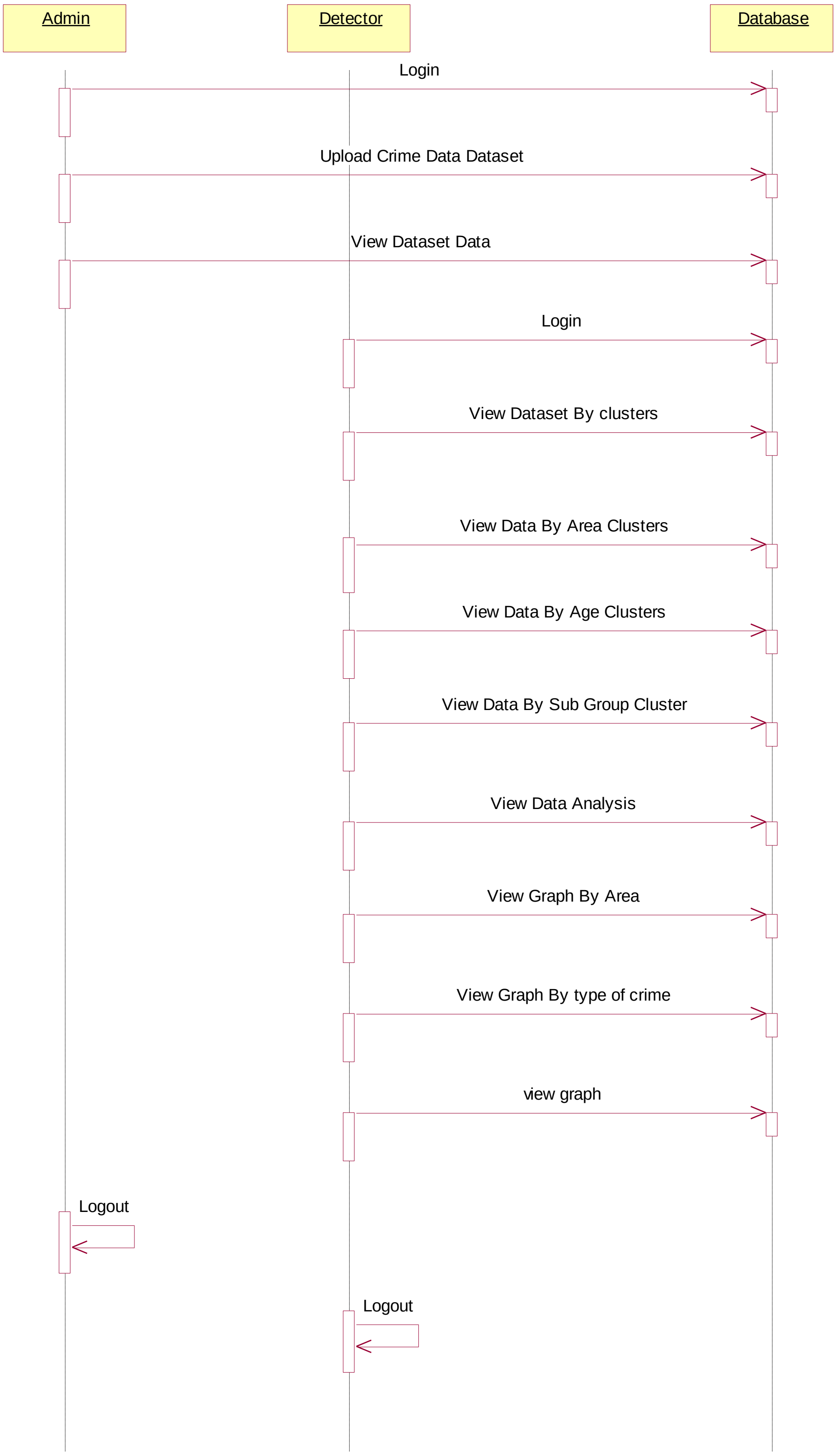

Admin
Detector
Database
Login
Upload Crime Data Dataset
View Dataset Data
Login
View Dataset By clusters
View Data By Area Clusters
View Data By Age Clusters
View Data By Sub Group Cluster
View Data Analysis
View Graph By Area
View Graph By type of crime
view graph
Logout
Logout

FIG.5.1.3SEQUENCEDIAGRAM

COLLABORATION DIAGRAM

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular **use** case and define the role of each object.

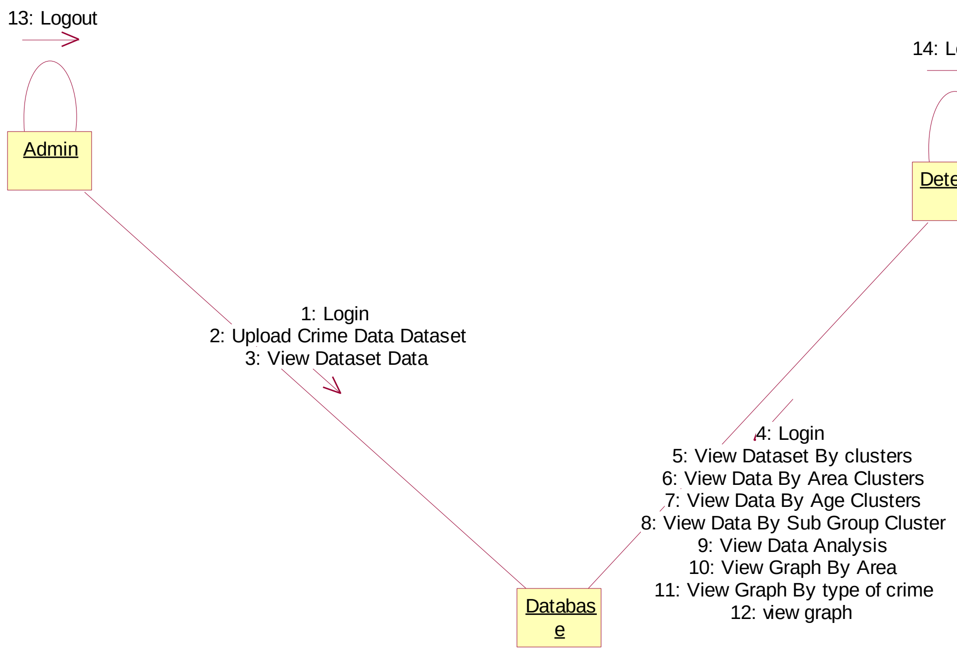

13: Logout
14: Logout
Admin
Detector
1: Login
2: Upload Crime Data Dataset
3: View Dataset Data
4: Login
5: View Dataset By clusters
6: View Data By Area Clusters
7: View Data By Age Clusters
8: View Data By Sub Group Cluster
9: View Data Analysis
10: View Graph By Area
11: View Graph By type of crime
12: view graph
Databas
e

FIG.5.1.4 COLLABORATION DIAGRAM

6. PROJECT CODING

6.1 TECHNOLOGY

Java Technology

Java technology is both a programming language and a platform.

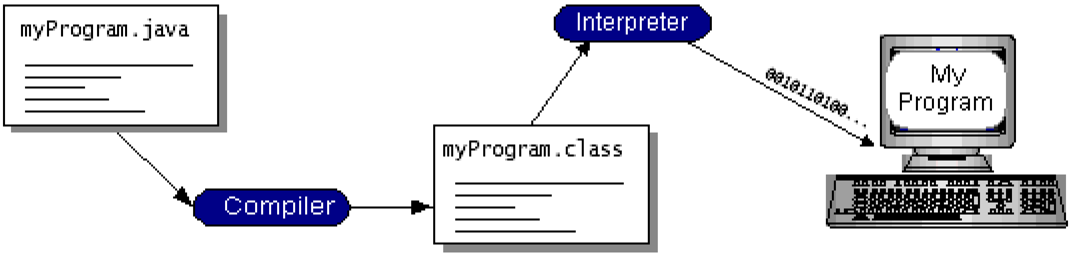
The Java Programming Language

The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

- With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.

.



You can think of Java byte codes as the machine code instructions for the *Java* Virtual Machine (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make "write once, run anywhere" possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.

What can Java Technology do?

The most common types of programs written in the Java programming language are applets and applications. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser. However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs. An application is a standalone program that runs directly on the Java platform. A special kind of application known as a *server* serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a *servlet*. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server. How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

The essentials: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.

- **Applets**: The set of conventions used by applets.
- **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
- **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
- **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
- **Software components**: Known as JavaBeans^{TM}, can plug into existing component architectures.
- **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).

Java Database Connectivity (JDBC™): Provides uniform access to a wide range of relational databases. The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK. How Will Java Technology Change My Life? We can't promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

- **Get started quickly**: Although the Java programming language is a powerful object-oriented language, it's easy to learn, especially for programmers already familiar with C or C++.
- **Write less code**: Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.
- **Write better code**: The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extendible API let you reuse other people's tested code and introduce fewer bugs.
- **Develop programs more quickly**: Your development time may be as much as twice as fast versus writing the same program in C++. Why? You write fewer lines of code and it is a simpler programming language than C++.
- **Avoid platform dependencies with 100% Pure Java**: You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure Java™ Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.
- **Write once, run anywhere**: Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.
- **Distribute software more easily**: You can upgrade applets easily from a central server. Applets take advantage of the feature of allowing new classes to be loaded "on the fly," without recompiling the entire program.

ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a de facto standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made the choice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needs suddenly change. Through the ODBC Administrator in Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN. The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you setup a separate database application, such as SQL Server Client or Visual Basic 4.0. When the ODBC icon is installed in Control Panel, it uses a file called ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand-alone program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and each maintains a separate list of ODBCdatasources. From a programming perspective, the beauty of ODBC is that the application can be written to use the same set of function calls to interface with any data source, regardless of the database vendor. The source code of the application doesn’t change whether it talks to Oracle or SQL Server. We only mention these two as an example. There are ODBC drivers available for several dozen popular database systems. Even Excel spreadsheets and plain text files can be turned into data sources. The operating system uses the Registry information written by ODBC Administrator to determine which low-level ODBC drivers are needed to talk to the data source (such as the interface to Oracle or SQL Server). The loading of the ODBC drivers is transparent to the ODBC application program. In a client/server environment, the ODBC API even handles many of the network issues for the application programmer. The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn’t as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow. Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of “plug-in” database connectivity modules, or drivers. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC’s framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution. JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after. The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java. The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. **SQL Level API**

   The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to “generate” JDBC code and to hide many of JDBC’s complexities from the end user.

2. **SQL Conformance**

   SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard functionality in a manner that is suitable for its users.

3. **JDBC must be implemental on top of common database interfaces**

   The JDBC SQL API must “sit” on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

4. **Provide a Java interface that is consistent with the rest of the Java system**

   Because of Java’s acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

5. **Keep it simple**

   This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

6. **Use strong, static typing wherever possible**

   Strong typing allows for more error checking to be done at compile time; also, less error appear at runtime.

7. **Keep the common cases simple**

   Because more often than not, the usual SQL calls used by the programmer are simple SELECT's, INSERT's, DELETE's and UPDATE's, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible.

Machine Learning

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 neighbour 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.

6.2 MACHINE LEARNING ALGORITHMS

DECISION TREE

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. The decisions or the test are performed on the basis of features of the given dataset. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions. It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure. In order to build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm. A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into sub trees.

RANDOM FOREST

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of over fitting

NAIVE BAYES CLASSIFIER

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems. It is mainly used in text classification that includes a high-dimensional training dataset. Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions. It is a probabilistic classifier, which means it predicts on the basis of the probability of an object. Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

SUPPORT VECTOR MACHINE

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

6.3 CODE IMPLEMENTATION

```
<%

String username=request.getParameter("uname");

String password=request.getParameter("pass");

try{

  String query="select * from admin where username='"+username+"'and
password='"+password+"'";

   ResultSet r=Queries.getExecuteQuery(query);

   if(r.next()){

    %>

    <script type="text/javascript">

        window.alert("admin login success..!!");

        window.location="AdminHome.jsp

/script>

    <%

   }else{

%>

    <script type="text/javascript">

        window.alert("admin login Failed..!!");

        window.location="Admin.jsp"
    </script>

    <%

}

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>Crime Analysis Mapping</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

  <div id="header">

        <center>

      <h2>Crime Analysis Mapping, Using Data Mining</h2>

      </center>

    <br class="clear" />

  </div>
```

```
</div>

<!--
#####################################################################
################################## -->

<div class="wrapper col2">

  <div id="topbar">

    <div id="topnav">

      <ul>

        <li><a href="index.html">Home</a></li>

        <li class="active"><a href="Admin.jsp">Admin</a></li>

        <li class="last"><a href="Intrusion_Detection.jsp">Detection</a></li>

      </ul>

    </div>

    <br class="clear" />

  </div>

</div>

<!--
#####################################################################
################################## -->

<div class="wrapper col3">

  <div id="intro">
```

```
    <div class="fl_left"><a href="#"><img src="images/Capture.PNG" width="400" height="230" alt="" /></a></div>

    <div class="fl_right">

      <h2>About This Project</h2>

      <p>

            Crime Analysis with crime mapping helps in understanding the concepts and practice of Crime Analysis in assisting police and helps in reduction and prevention of crimes and crime disorders.

      </p>

    </div>

    <br class="clear" />

  </div>

</div>

<!--
###########################################################################
################################### -->

<div class="wrapper col4">

  <div id="container">

    <div id="content">

      <h2>Abstract</h2>

      <p align="justify">
```

Data Mining plays a key role in Crime Analysis. There are many different algorithms mentioned in previous research papers, among them are the virtual identifier, pruning strategy, support vector machines, and apriori algorithms. VID is to find relation between record and vid. The apriori algorithm helps the fuzzy association rules algorithm and it takes around six hundred seconds to detect a mail bomb attack. In this research paper, we identified Crime mapping analysis based on KNN (K – Nearest Neighbor) and ANN (Artificial Neural Network) algorithms to simplify this process. Crime Mapping is conducted and Funded by the Office of Community Oriented Policing Services (COPS). Evidence based research helps in analyzing the crimes. We calculate the crime rate based on the previous data using data mining techniques. Crime Analysis uses quantitative and qualitative data in combination with analytic techniques in resolving the cases. For public safety purposes, the crime mapping is an essential research area to concentrate on. We can identity the most frequently crime occurring zones with the help of data mining techniques. In Crime Analysis Mapping, we follow the following steps in order to reduce the crime rate: 1) Collect crime data 2) Group data 3) Clustering 4) Forecasting the data.

```
   </p>

 </div>

 <div id="column">

   <div class="flickrbox">

       <h3>Admin Login</h3>

       <form action="AdminLoginAction.jsp" method="post">

           <table>

              <tr><th>UserName</th><td><input type="text" name="uname" required=""></tD></tr>

              <tr><th>Password</th><td><input type="password" name="pass" required=""></tD></tr>

              <tr><th></th><td><input type="submit" value="Login"></tD></tr>
              </table>

          </form>

        <br class="clear" />

      </div>

    </div>

    <br class="clear" />

  </div>

</div>

<!--
##########################################################################
#################################### -->

<div class="wrapper col5">

  <div id="foot”>

    <br class="clear" />

  </div>

</div>

<!--
##########################################################################
################################### -->

<div class="wrapper col6">

  <div id="copyright">
```

```
    <p class="fl_left">Crime Analysis Mapping, Intrusion Detection - Using Data Mining</p>

    <p class="fl_right"></p>

    <br class="clear" />

  </div>

</div>

</body>

</html>

}catch(Exception e){

  out.println(e);

}

%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.Connection"%>

<%@page import="com.database.Dbconnection"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="com.database.Queries"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>
```

```
<head>

<title>Crime Analysis Mapping</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

  <div id="header">

      <center>

      <h2>Crime Analysis Mapping, - Using Data Mining</h2>

      </center>

    <br class="clear" />

  </div>

</div>

<!--
##############################################################################
###################################### -->

<div class="wrapper col2">

  <div id="topbar">

    <div id="topnav">
```

```
      <ul>

        <li><a href="AdminHome.jsp">Home</a></li>

        <li class="active"><a href="Clustering.jsp">Clustering  </a></li>

        <li class="last"><a href="Crime_Analysis.jsp">Crime Analysis</a></li>

       <li class="last"><a href="Admin.jsp">Logout</a></li>

      </ul>

    </div>

    <br class="clear" />

  </div>

</div>

<!--
#######################################################################
################################### -->



<!--
#######################################################################
################################### -->

<div class="wrapper col4">

  <div id="container">

    <div id="content">

      <h2 style="margin-bottom:50px;">Different Types Of Clusters
```

```
        <p style="float:right;"><a href="Clustering.jsp">Back</a></p>

      </h2>

      <style>

    tr td{

        text-align: center

    }

    tr th{

      text-align: center;

    }

</style>

      <h3>Area_Cluster</h3>

      <table border="1">

        <tr><th><font color="red">Area_Name</font></th>

         <th>Year</th>

        <th>Subgroup</th>

        <th>Rape Case Reported</th></tr>

       <%

       try{

           Connection con=Dbconnection.getcon();
```

```
            Statement s=con.createStatement();

            String query="select distinct Area_Name from dataset";

            ResultSet r=Queries.getExecuteQuery(query);

            while(r.next()){

              String Area_Name=r.getString("Area_Name");

              ResultSet r1=s.executeQuery("select * from dataset where
Area_Name='"+Area_Name+"'");

            W

hile(r1.next()){

              %>

             <td><font color="red"><%=r1.getString(2)%></font></td>

                <td><%=r1.getString(3)%></td>

                  <td><%=r1.getString(4)%></td>

                   <td><%=r1.getString(5)%></td>

         </tr>

         <%>

            }

            }

        }catch(Exception e){

           out.println(e);
```

```
     }

     %>

   </table>

  </div>

  <div id="column">

   <div class="flickrbox">

    <br class="clear" />

   </div>

  </div>

  <br class="clear" />

 </div>

</div>

<!--
#########################################################################
#################################### -->

<div class="wrapper col5">

 <div id="footer">

  <br class="clear" />

 </div>

</div>
```

```
<!--
########################################################################
################################### -->

<div class="wrapper col6">

  <div id="copyright">

    <p class="fl_left">Crime Analysis Mapping, Intrusion Detection - Using Data
Mining</p>

    <p class="fl_right"></p>

    <br class="clear" />

  </div>

</div>

</body>

</html>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.Connection"%>

<%@page import="com.database.Dbconnection"%>

<%@page import="java.sql.ResultSet"%>

<%@page import="com.database.Queries"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>



<!DOCTYPE html>
```

```
<html>

<head>

<title>Crime Analysis Mapping</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

  <div id="header">

      <center>

      <h2>Crime Analysis Mapping, - Using Data Mining</h2>

      </center>

    <br class="clear" />

  </div>

</div>

<!--
#####################################################################
################################## -->

<div class="wrapper col2">

  <div id="topbar">
```

```
    <div id="topnav">

      <ul>

        <li><a href="AdminHome.jsp">Home</a></li>

        <li class="active"><a href="Clustering.jsp">Clustering  </a></li>

        <li class="last"><a href="Crime_Analysis.jsp">Crime Analysis</a></li>

       <li class="last"><a href="Admin.jsp">Logout</a></li>

      </ul>

    </div>

    <br class="clear" />

  </div>

</div>

<!--
########################################################################
#################################### -->

<!--
########################################################################
#################################### -->

<div class="wrapper col4">

  <div id="container">

    <div id="content">

      <h2 style="margin-bottom:50px;">Different Types Of Clusters
```

```
          <p style="float:right;"><a href="Clustering.jsp">Back</a></p>

      </h2>

      <style>

    tr td{

        text-align: center;

    }

    try th{

      text-align: center;

    }

</style>

      <h3>Area_Cluster</h3>

      <table border="1">

          <tr><th><font color="red">Area_Name</font></th>

           <th>Year</th>

          <th>Subgroup</th>

          <th>Rape Case Reported</th></tr>

         <%

         try{

             Connection con=Dbconnection.getcon();
             Statement s=con.createStatement();

             String query="select distinct Area_Name from dataset";

             ResultSet r=Queries.getExecuteQuery(query);

             while(r.next()){

               String Area_Name=r.getString("Area_Name");



               ResultSet r1=s.executeQuery("select * from dataset where Area_Name='"+Area_Name+"'");

             while(r1.next()){

               %>

          <tR>

              <td><font color="red"><%=r1.getString(2)%></font></td>

                  <td><%=r1.getString(3)%></td>

                    <td><%=r1.getString(4)%></td>

                     <td><%=r1.getString(5)%></td>

          </tr>

          <%

             }

             }

         }catch(Exception e){
```

```
            out.println(e);

          }

          %>

      </table>

    </div>

    <div id="column">

      <div class="flickrbox">

        <br class="clear" />

      </div>

    </div>

    <br class="clear" />

  </div>

</div>

<!--
######################################################################
################################## -->

<div class="wrapper col5">

  <div id="footer">

    <br class="clear" />

  </div>
```

```
</div>

<!--
#####################################################################
################################## -->

<div class="wrapper col6">

  <div id="copyright">

    <p class="fl_left">Crime Analysis Mapping, Intrusion Detection - Using Data Mining</p>

    <p class="fl_right"></p>

    <br class="clear" />

  </div>

</div>

</body>

</html>
```

7. PROJECT 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.

TYPES OF TESTS

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.

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.

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.

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.

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.

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.

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

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.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

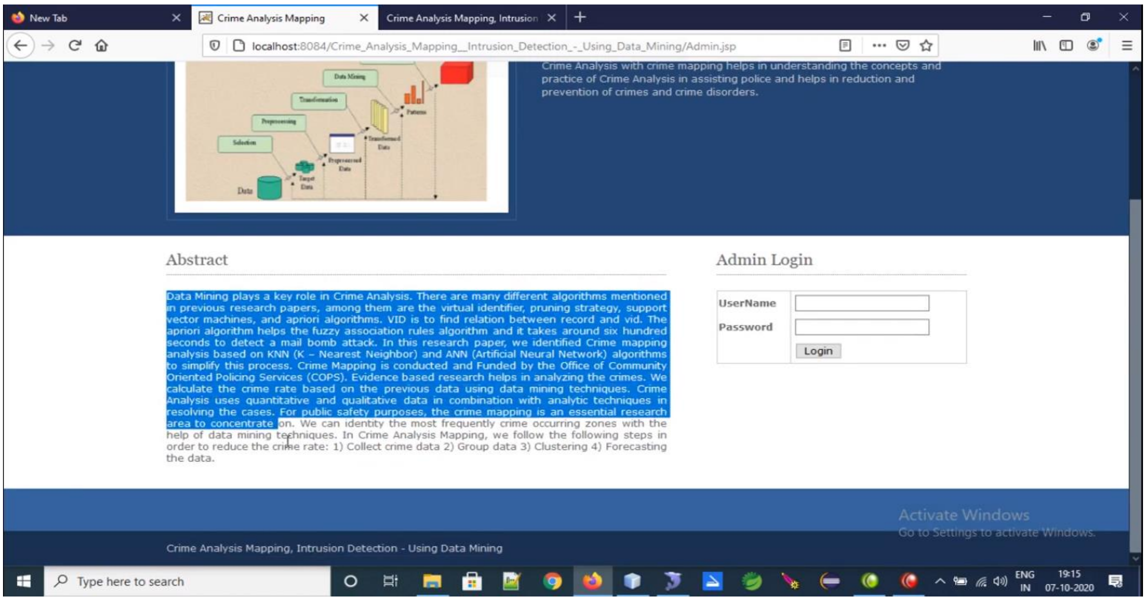
Test Results: All the test cases mentioned above passed successfully. No defects encountered.

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.

8. OUTPUT SCREENS



Crime Analysis with crime mapping helps in understanding the concepts and practice of Crime Analysis in assisting police and helps in reduction and prevention of crimes and crime disorders.

Abstract

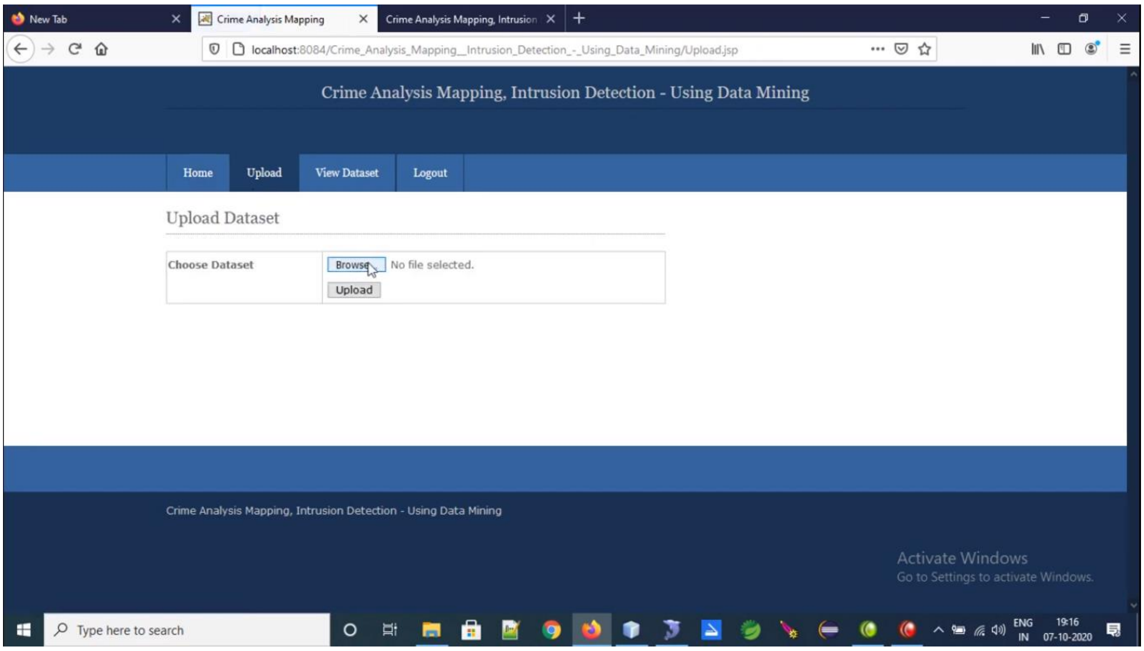
Data Mining plays a key role in Crime Analysis. There are many different algorithms mentioned in previous research papers, among them are the virtual identifier, pruning strategy, support vector machines, and apriori algorithms. VID is to find relation between record and vid. The apriori algorithm helps the fuzzy association rules algorithm and it takes around six hundred seconds to detect a mail bomb attack. In this research paper, we identified Crime mapping analysis based on KNN (K – Nearest Neighbor) and ANN (Artificial Neural Network) algorithms to simplify this process. Crime Mapping is conducted and Funded by the Office of Community Oriented Policing Services (COPS). Evidence based research helps in analyzing the crimes. We calculate the crime rate based on the previous data using data mining techniques. Crime Analysis uses quantitative and qualitative data in combination with analytic techniques in resolving the cases. For public safety purposes, the crime mapping is an essential research area to concentrate on. We can identity the most frequently crime occurring zones with the help of data mining techniques. In Crime Analysis Mapping, we follow the following steps in order to reduce the crime rate: 1) Collect crime data 2) Group data 3) Clustering 4) Forecasting the data.

Admin Login

UserName
Password
Login

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

FIG.8.1.1 ADMIN LOGIN PAGE


New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/Upload.jsp
Home
Upload
View Dataset
Logout
Browse...
Upload
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
IN
19:16
07-10-2020

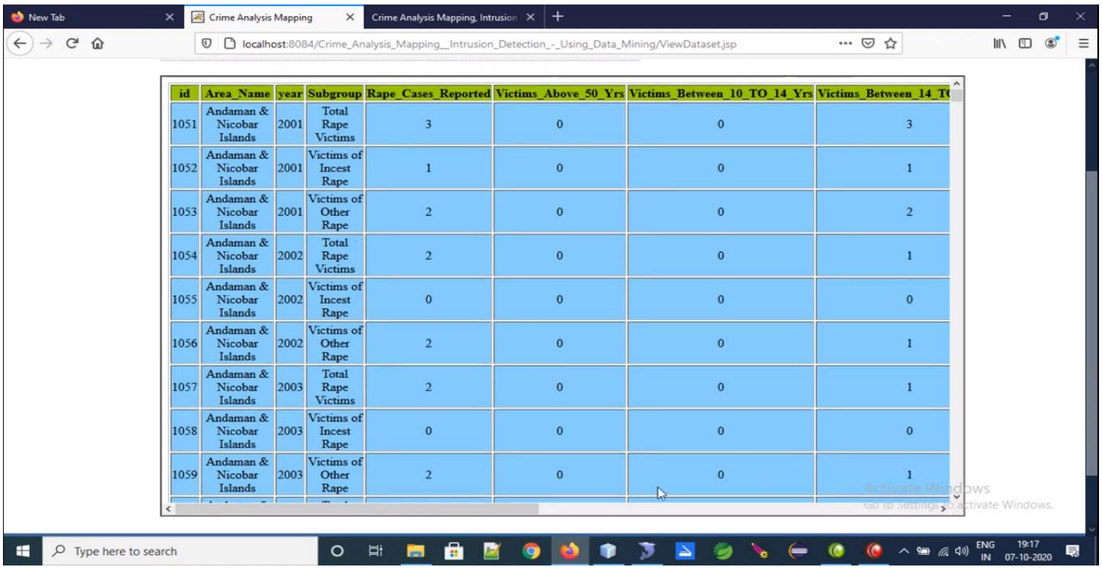
Crime Analysis Mapping, Intrusion Detection - Using Data Mining

Upload Dataset

Choose Dataset — No file selected.

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

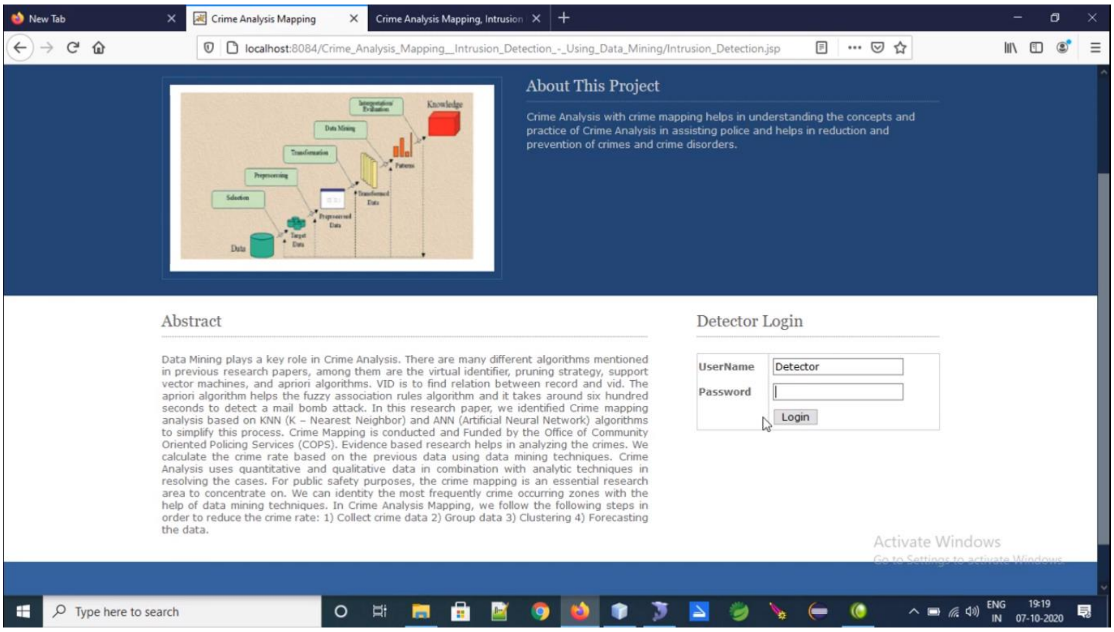
FIG 8.1.2 UPLOADING CRIME DATASET



New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/ViewDataset.jsp
Type here to search
ENG IN
19:17
07-10-2020

id	Area_Name	year	Subgroup	Rape_Cases_Reported	Victims_Above_50_Yrs	Victims_Between_10_TO_14_Yrs	Victims_Between_14_TO
1051	Andaman & Nicobar Islands	2001	Total Rape Victims	3	0	0	3
1052	Andaman & Nicobar Islands	2001	Victims of Incest Rape	1	0	0	1
1053	Andaman & Nicobar Islands	2001	Victims of Other Rape	2	0	0	2
1054	Andaman & Nicobar Islands	2002	Total Rape Victims	2	0	0	1
1055	Andaman & Nicobar Islands	2002	Victims of Incest Rape	0	0	0	0
1056	Andaman & Nicobar Islands	2002	Victims of Other Rape	2	0	0	1
1057	Andaman & Nicobar Islands	2003	Total Rape Victims	2	0	0	1
1058	Andaman & Nicobar Islands	2003	Victims of Incest Rape	0	0	0	0
1059	Andaman & Nicobar Islands	2003	Victims of Other Rape	2	0	0	1

FIG 8.1.3 VIEW CRIME DATASET



About This Project

Crime Analysis with crime mapping helps in understanding the concepts and practice of Crime Analysis in assisting police and helps in reduction and prevention of crimes and crime disorders.

Abstract

Data Mining plays a key role in Crime Analysis. There are many different algorithms mentioned in previous research papers, among them are the virtual identifier, pruning strategy, support vector machines, and apriori algorithms. VID is to find relation between record and vid. The apriori algorithm helps the fuzzy association rules algorithm and it takes around six hundred seconds to detect a mail bomb attack. In this research paper, we identified Crime mapping analysis based on KNN (K – Nearest Neighbor) and ANN (Artificial Neural Network) algorithms to simplify this process. Crime Mapping is conducted and Funded by the Office of Community Oriented Policing Services (COPS). Evidence based research helps in analyzing the crimes. We calculate the crime rate based on the previous data using data mining techniques. Crime Analysis uses quantitative and qualitative data in combination with analytic techniques in resolving the cases. For public safety purposes, the crime mapping is an essential research area to concentrate on. We can identity the most frequently crime occurring zones with the help of data mining techniques. In Crime Analysis Mapping, we follow the following steps in order to reduce the crime rate: 1) Collect crime data 2) Group data 3) Clustering 4) Forecasting the data.

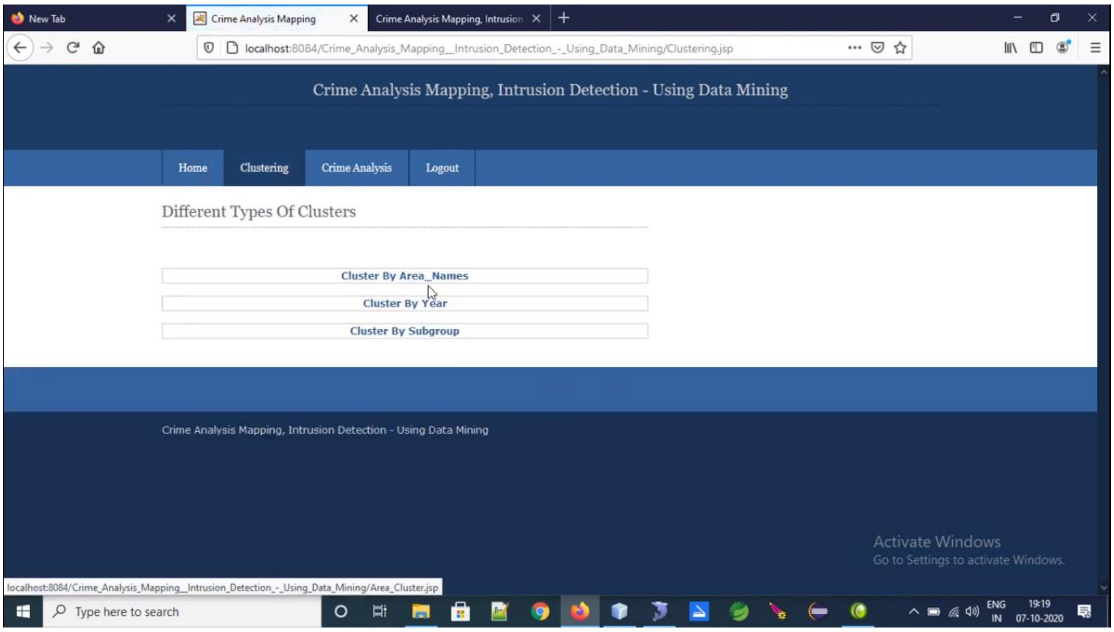
Detector Login

UserName: Detector

Password:

Login

FIG 8.1.4 DETECTOR LOGIN PAGE


New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/Clustering.jsp
Activate Windows
Go to Settings to activate Windows.
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/Area_Cluster.jsp
Type here to search
ENG IN
19:19
07-10-2020

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

Home | Clustering | Crime Analysis | Logout

Different Types Of Clusters

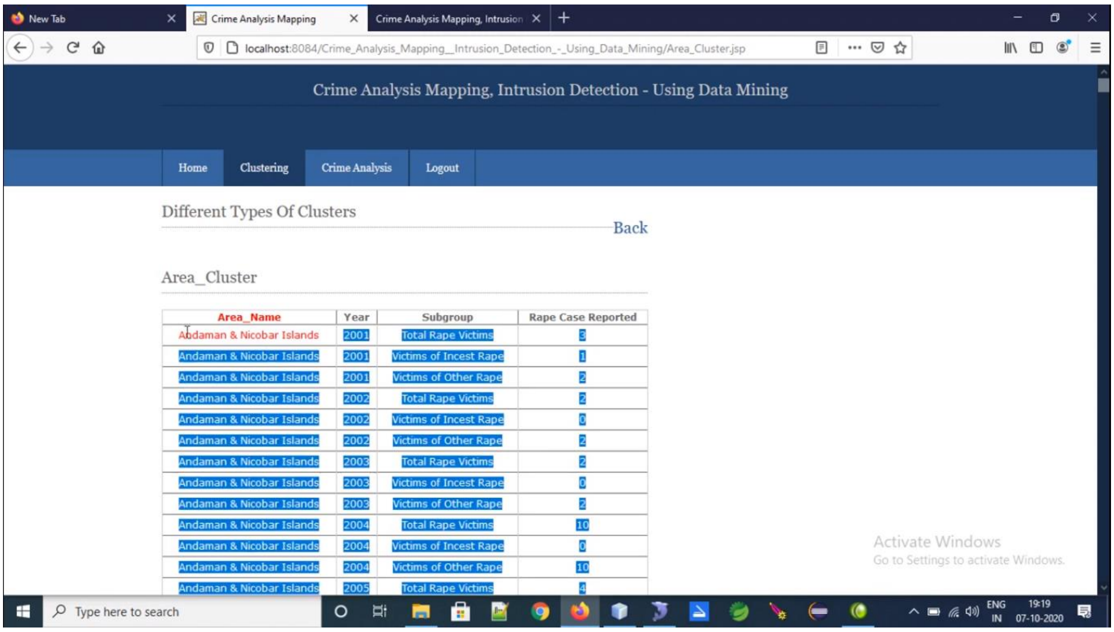
Cluster By Area_Names

Cluster By Year

Cluster By Subgroup

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

FIG 8.1.5 DIFFERENT TYPES OF CLUSTERS


New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/Area_Cluster.jsp
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG IN
19:19
07-10-2020

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

Home | Clustering | Crime Analysis | Logout

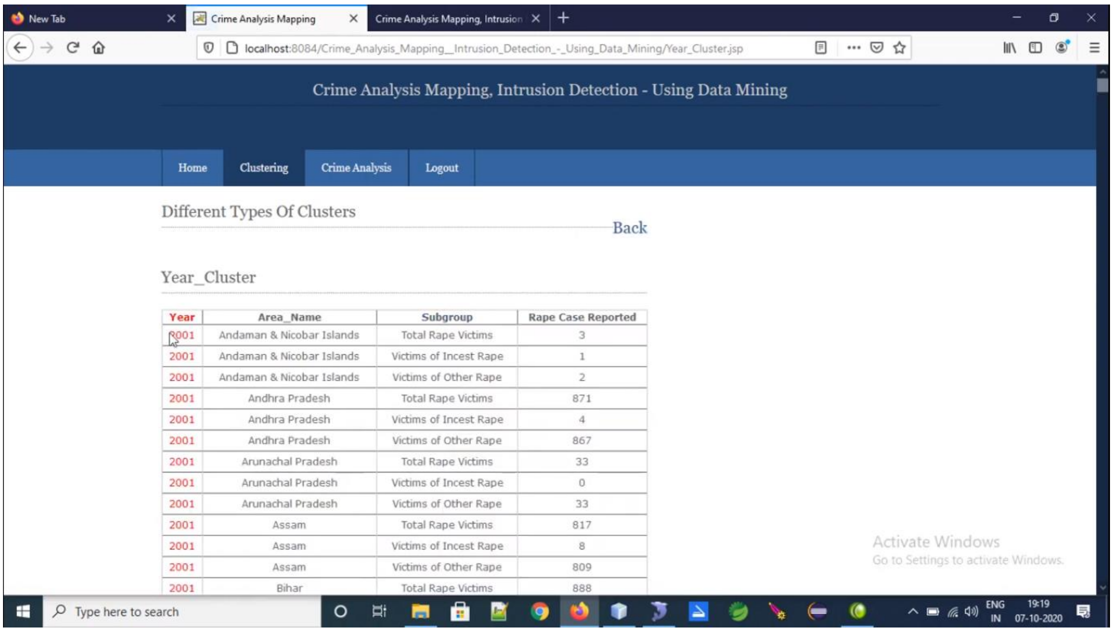
Different Types Of Clusters

Back

Area_Cluster

Area_Name	Year	Subgroup	Rape Case Reported
Andaman & Nicobar Islands	2001	Total Rape Victims	3
Andaman & Nicobar Islands	2001	Victims of Incest Rape	1
Andaman & Nicobar Islands	2001	Victims of Other Rape	2
Andaman & Nicobar Islands	2002	Total Rape Victims	2
Andaman & Nicobar Islands	2002	Victims of Incest Rape	0
Andaman & Nicobar Islands	2002	Victims of Other Rape	2
Andaman & Nicobar Islands	2003	Total Rape Victims	2
Andaman & Nicobar Islands	2003	Victims of Incest Rape	0
Andaman & Nicobar Islands	2003	Victims of Other Rape	2
Andaman & Nicobar Islands	2004	Total Rape Victims	10
Andaman & Nicobar Islands	2004	Victims of Incest Rape	0
Andaman & Nicobar Islands	2004	Victims of Other Rape	10
Andaman & Nicobar Islands	2005	Total Rape Victims	4

FIG 8.1.6 AREA CLUSTER


New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/Year_Cluster.jsp
Home
Clustering
Crime Analysis
Logout
Back
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG IN
19:19
07-10-2020

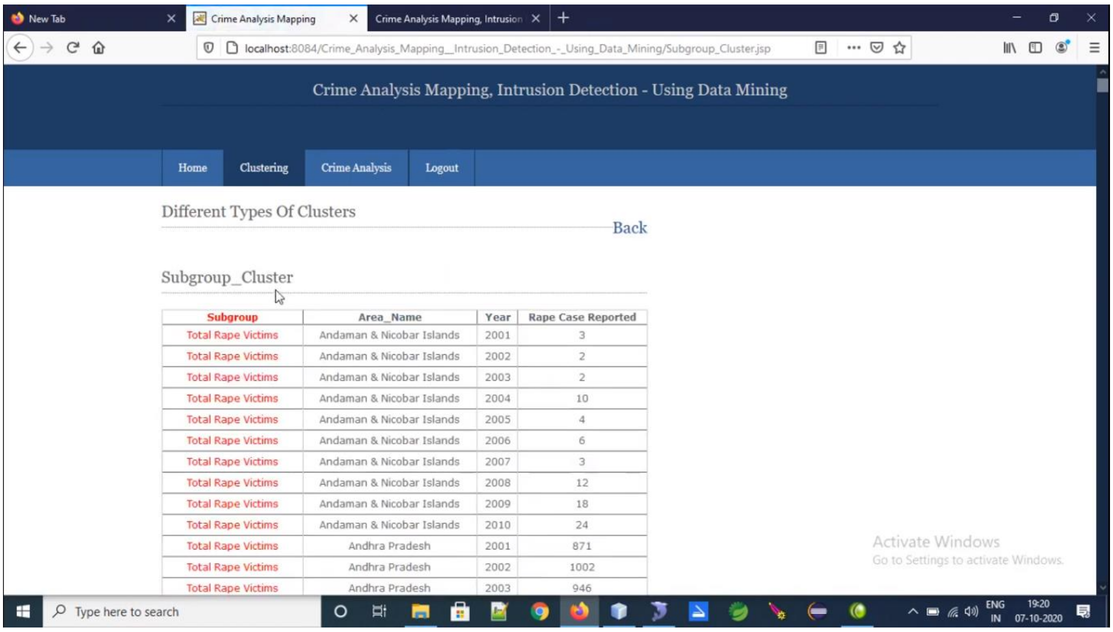
Crime Analysis Mapping, Intrusion Detection - Using Data Mining

Different Types Of Clusters

Year_Cluster

Year	Area_Name	Subgroup	Rape Case Reported
2001	Andaman & Nicobar Islands	Total Rape Victims	3
2001	Andaman & Nicobar Islands	Victims of Incest Rape	1
2001	Andaman & Nicobar Islands	Victims of Other Rape	2
2001	Andhra Pradesh	Total Rape Victims	871
2001	Andhra Pradesh	Victims of Incest Rape	4
2001	Andhra Pradesh	Victims of Other Rape	867
2001	Arunachal Pradesh	Total Rape Victims	33
2001	Arunachal Pradesh	Victims of Incest Rape	0
2001	Arunachal Pradesh	Victims of Other Rape	33
2001	Assam	Total Rape Victims	817
2001	Assam	Victims of Incest Rape	8
2001	Assam	Victims of Other Rape	809
2001	Bihar	Total Rape Victims	888

FIG 8.1.7 YEAR CLUSTER


New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/Subgroup_Cluster.jsp
Home
Clustering
Crime Analysis
Logout
Back
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG IN
19:20
07-10-2020

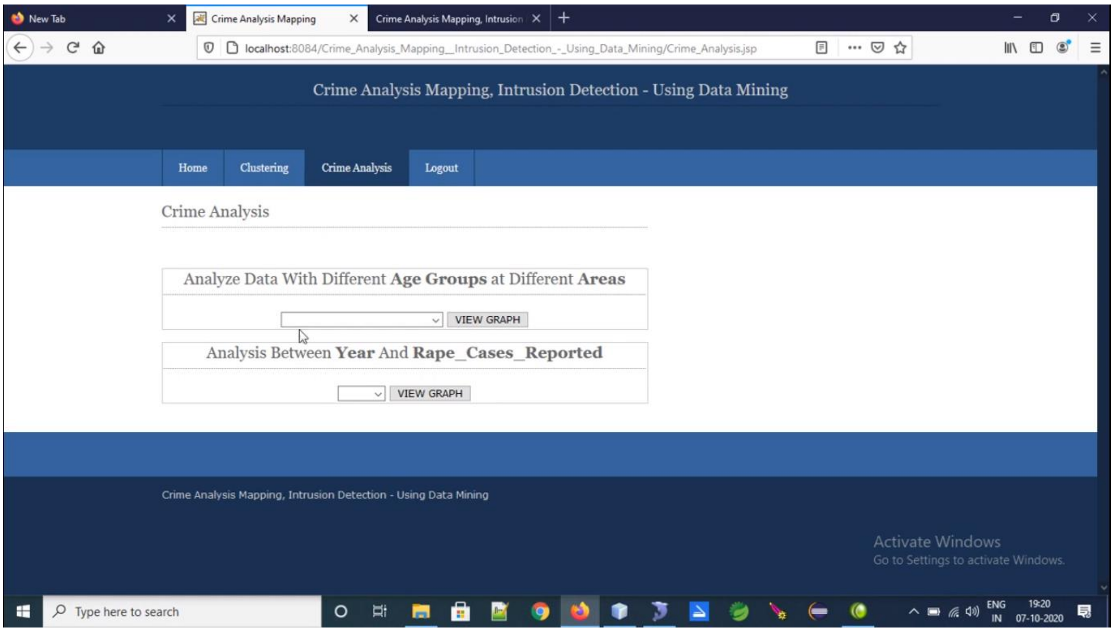
Crime Analysis Mapping, Intrusion Detection - Using Data Mining

Different Types Of Clusters

Subgroup_Cluster

Subgroup	Area_Name	Year	Rape Case Reported
Total Rape Victims	Andaman & Nicobar Islands	2001	3
Total Rape Victims	Andaman & Nicobar Islands	2002	2
Total Rape Victims	Andaman & Nicobar Islands	2003	2
Total Rape Victims	Andaman & Nicobar Islands	2004	10
Total Rape Victims	Andaman & Nicobar Islands	2005	4
Total Rape Victims	Andaman & Nicobar Islands	2006	6
Total Rape Victims	Andaman & Nicobar Islands	2007	3
Total Rape Victims	Andaman & Nicobar Islands	2008	12
Total Rape Victims	Andaman & Nicobar Islands	2009	18
Total Rape Victims	Andaman & Nicobar Islands	2010	24
Total Rape Victims	Andhra Pradesh	2001	871
Total Rape Victims	Andhra Pradesh	2002	1002
Total Rape Victims	Andhra Pradesh	2003	946

FIG 8.1.8 SUBGROUP CLUSTER



New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/Crime_Analysis.jsp
Home
Clustering
Crime Analysis
Logout
VIEW GRAPH
VIEW GRAPH
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
IN
19:20
07-10-2020

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

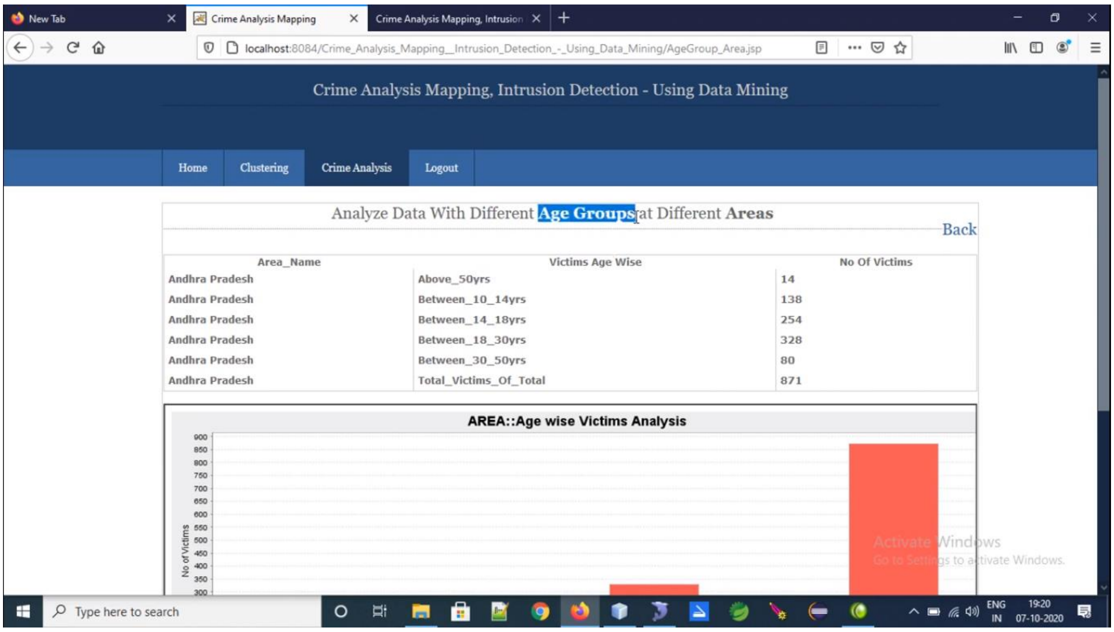
Crime Analysis

Analyze Data With Different **Age Groups** at Different **Areas**

Analysis Between **Year** And **Rape_Cases_Reported**

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

FIG 8.1.9 CRIME ANALYSIS



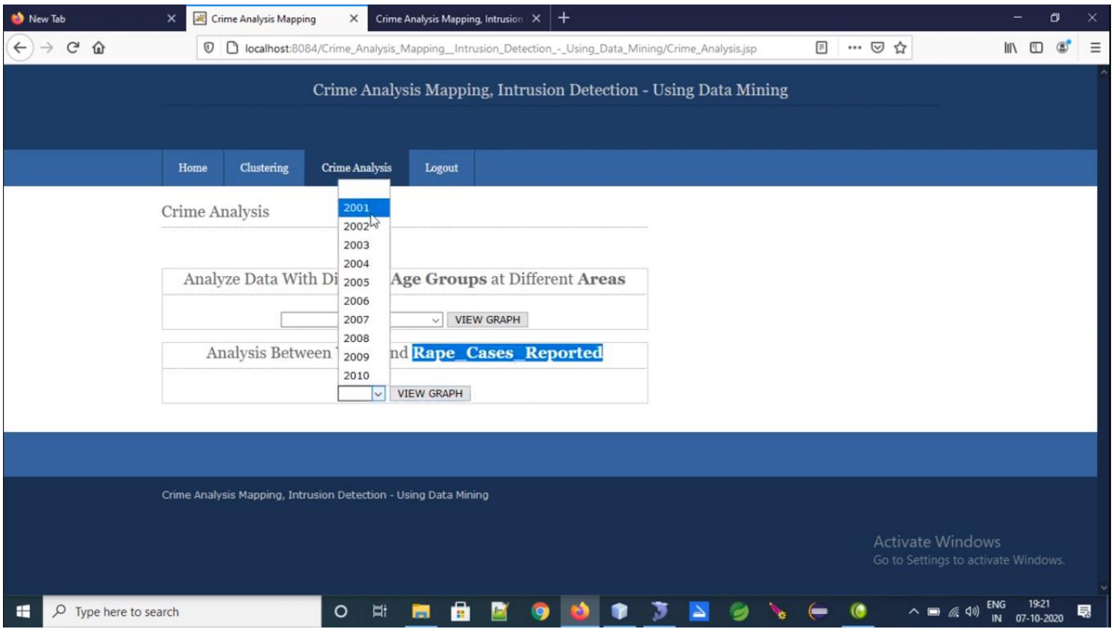
New Tab
Crime Analysis Mapping
Crime Analysis Mapping, Intrusion
localhost:8084/Crime_Analysis_Mapping__Intrusion_Detection_-_Using_Data_Mining/AgeGroup_Area.jsp
Home
Clustering
Crime Analysis
Logout
Back
AREA::Age wise Victims Analysis
No of Victims
900
850
800
750
700
650
600
550
500
450
400
350
300
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
IN
19:20
07-10-2020

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

Analyze Data With Different **Age Groups** at Different **Areas**

Area_Name	Victims Age Wise	No Of Victims
Andhra Pradesh	Above_50yrs	14
Andhra Pradesh	Between_10_14yrs	138
Andhra Pradesh	Between_14_18yrs	254
Andhra Pradesh	Between_18_30yrs	328
Andhra Pradesh	Between_30_50yrs	80
Andhra Pradesh	Total_Victims_Of_Total	871

FIG 8.1.10 ANALYSIS WITH DIFFERENT AGE GROUPS AT DIFFERENT AREAS



Crime Analysis Mapping, Intrusion Detection - Using Data Mining

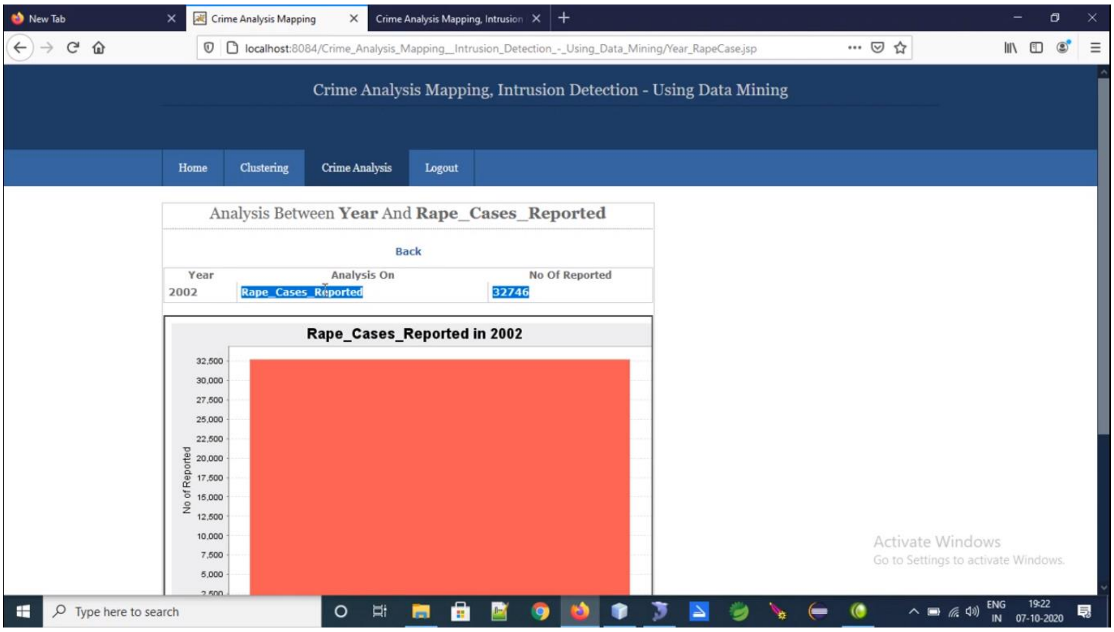
Crime Analysis

Analyze Data With Di Age Groups at Different Areas

Analysis Between Y nd Rape_Cases_Reported

Crime Analysis Mapping, Intrusion Detection - Using Data Mining

FIG 8.1.11 CRIME ANALYSIS IN PARTICULAR YEAR



Crime Analysis Mapping, Intrusion Detection - Using Data Mining

Analysis Between Year And Rape_Cases_Reported

Year	Analysis On	No Of Reported
2002	Rape_Cases_Reported	32746

FIG 8.1.12 CRIME CASES REPORTED IN YEAR 2002

9. CONCLUSION

With the assistance of these devices, the wrongdoing information will be nourished to the information digging device for investigation and afterward comes about for two unique models will be recorded. With the assistance of the SAM instrument/tools, we will maintain a strategic distance from the distinction in the outcome and after that the subsequent information will be utilized for the finding the relations amongst those et cetera. Along these lines we will lessen false positives and false negatives in the field of the interruption identification framework utilizing the information mining in the field of wrongdoing information examination.

10. FUTURE ENHANCEMENT

As we have applied clustering technique of data mining for crime analysis we can also perform other techniques of data mining such as classification. Also we can perform analysis on various dataset such as enterprise survey dataset, poverty dataset, aid effectiveness dataset, etc.

11.BIBILOGRAPHY

1. Chen, Hsinchun, et al. "Crime data mining: a general framework and some examples." computer 37.4 Authorized licensed use limited to: University of Southern Queensland. Downloaded on August 02,2020 at 02:07:40 UTC from IEEE Xplore. Restrictions apply.

2. Ektefa, Mohammadreza, et al. "Intrusion detection using data mining techniques." Information Retrieval & Knowledge Management,(CAMP), 2010 International Conference on. IEEE, 2010.

3. Clifton, Chris, and Gary Gengo. "Developing custom intrusion detection filters using data mining." MILCOM 2000. 21st Century Military Communications Conference Proceedings. Vol. 1. IEEE, 2000.

4. Dickerson, John E., and Julie A. Dickerson. "Fuzzy network profiling for intrusion detection." Fuzzy Information Processing Society, 2000. NAFIPS. 19th International Conference of the North American. IEEE, 2000.

5. Siraj, Ambareen, Susan M. Bridges, and Rayford B. Vaughn. "Fuzzy cognitive maps for decision support in an intelligent intrusion detection system." IFSA World Congress and 20th NAFIPS International Conference, 2001. Joint 9th. Vol. 4. IEEE, 2001.

6. Nath, Shyam Varan. "Crime pattern detection using data mining." Web intelligence and intelligent agent technology workshops, 2006. wi-iat 2006 workshops. 2006 ieee/wic/acm international conference on. IEEE, 2006.

7. Florez, German, S. A. Bridges, and Rayford B. Vaughn. "An improved algorithm for fuzzy data mining for intrusion detection." Fuzzy Information Processing Society, 2002. Proceedings. NAFIPS. 2002 Annual Meeting of the North American. IEEE, 2002.

8. Panda, Mrutyunjaya, and Manas Ranjan Patra. "A comparative study of data mining algorithms for network intrusion detection." Emerging Trends in Engineering and Technology, 2008. ICETET'08. First International Conference on. IEEE, 2008.

9. Vaidya, Jaideep, and Chris Clifton. "Privacy-preserving data mining:

Why, how, and when." IEEE Security & Privacy 2.6 (2004): 19-27.

10. Mukkamala, Srinivas, Guadalupe Janoski, and Andrew Sung. "Intrusion detection using neural networks and support vector machines." Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on. Vol. 2. IEEE, 2002