VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belgaum - 590018

A Project report on

"Emotion Detection and Suicidal Intention Prediction of Depressed Patients"

submitted in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

INFORMATION SCIENCE & ENGINEERING

by

1CR19IS149 Shreya Soni

1CR19IS150 Shruti Chaubey

1CR19IS156 Suchita Parira

Under the Guidance of

Dr Senthil Velan S

Professor,

Department of ISE, CMRIT, Bengaluru

CMR INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

#132, AECS Layout, IT Park Road, Bengaluru – 560037

2022-23

CMR INSTITUTE OF TECHNOLOGY

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

#132, AECS Layout, IT Park Road, Bengaluru - 560037

Certificate

This is to Certified that the project work entitled **"Emotion Detection and Suicidal Intention Prediction of Depressed Patients"** carried out by **Shreya Soni (1CR19IS149), Shruti Chaubey (1CR19IS150) and Suchita Parira (1CR19IS156)** in partial fulfillment for the award of Bachelor of Engineering in **Information Science & Engineering** of the Visvesvaraya Technological University, Belgaum during the year **2022-23**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated inthe Report deposited in the departmental library.

The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

Signature of Guide	Signature of HoD	Signature of Principal
Dr Senthil Velan S	**Dr M Farida Begam**	**Dr Sanjay Jain**
Professor	Professor & Head	Principal
Department of ISE	Department of ISE	CMRIT,
CMRIT	CMRIT	Bengaluru - 37

External Viva

	Name of the Examiners	Institution	Signature with Date
1.	.....	.....	.....
2.	.....	.....	.....

CMR INSTITUTE OF TECHNOLOGY
DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING
#132, AECS Layout, IT Park Road, Bengaluru - 560037

Declaration

We, **Shreya Soni (1CR19IS149), Shruti Chaubey (1CR19IS150) and Suchita Parira (1CR19IS156)** bonafide students of **CMR Institute of Technology**, Bangalore, hereby declare that the dissertation entitled, **"Emotion Detection and Suicidal Intention Prediction of Depressed Patients"** has been carried out by us under the guidance of **Dr Senthil Velan S**, Professor, CMRIT, Bangalore, in partial fulfillment of the requirements for the award of the degree of Bachelor of Engineeringin Computer Science & Engineering, of the Visvesvaraya Technological University, Belgaum during the academic year 2022-2023. The work done in this dissertation report is original and it has not been submitted for any other degree in any university.

SHREYA SONI (1CR19IS149)
SHRUTI CHAUBEY (1CR19IS150)
SUCHITA PARIRA (1CR19IS156)

Acknowledgement

The satisfaction and euphoria that accompany a successful completion of any task would be incomplete without the mention of people who made it possible. Success is the epitome of hard work and perseverance, but steadfast of all is encouraging guidance.

So, it is with gratitude that we acknowledge all those whose guidance and encouragement served as beacon of light and crowned our effort with success.

We would like to thank Dr Sanjay Jain, Principal, CMRIT, Bangalore, for providing anexcellent academic environment in the college and his never ending support for the B.E program.

We would like to express our gratitude towards Dr M Farida Begam, Professor and HOD, Department of Information Science & Engineering CMRIT, Bangalore, who provided guidance and gave valuable suggestions regarding the project.

We consider it a privilege and honour to express our sincere gratitude to our internal guide Dr Senthil Velan S, Professor, Department of Information Science & Engineering, CMRIT, Bangalore, for their valuable guidance throughout the tenure of this project work.

We would like to thank all the faculty members who have always been very cooperative and generous. Conclusively, we also thank all the nonteaching staff and all others who have done immense help directly or indirectly during our project.

SHREYA SONI (1CR19IS149)
SHRUTI CHAUBEY (1CR19IS150)
SUCHITA PARIRA (1CR19IS156)

Abstract

Facial expressions play an important role in conveying emotions, especially in human-machine interaction. Automatic facial expression recognition (FER) systems have numerous potential applications, such as detecting mental disorders, understanding human behavior, and generating synthetic human expressions. However, achieving high recognition rates remains a challenging task. In the literature, two popular approaches for automatic FER are based on geometry and appearance. The FER process typically consists of four stages, namely pre-processing, face detection, feature extraction, and expression classification. In our project, we utilized various deep learning techniques, specifically convolutional neural networks, to detect seven essential human emotions: anger, disgust, fear, happiness, sadness, surprise, and neutrality. Furthermore, our aim was to predict suicidal tendencies based on the detected emotions since depression is the primary cause of suicide. Detecting emotions in depressed individuals could facilitate their monitoring and help prevent suicide risk by forecasting the rate of suicidal intentions based on their emotional state.

Keywords—*Emotion detection, Suicidal intention prediction, Depression*

Table of Contents

Abstract ii

Table of Contents iii

List of Figures v

1 Preamble 1

1.1 Introduction 1
1.2 Existing System 2
1.3 Proposed System 3
1.4 Plan of Implementation 4
1.5 Problem Statement 4
1.6 Objective of the Project 4

2 Literature Survey 5

3 Theoretical Background 6

3.1 DeepFace 7
3.2 Numpy 8
3.3 Pandas 9
3.4 Machine Learning 10
3.5 Decision Tree Algorithm 11

4 System Requirements Specification 13

4.1 Functional Requirement 14
4.2 Non Functional Requirements 14

5 System Analysis 18

5.1 Feasibility Study 18
5.2 Analysis 19

6 System Design **21**
6.1 System Development Methodology 21
6.2 Data Flow Diagram 24

7 Implementation **25**
7.1 Emotion Detection Model 25
7.2 Suicidal Intention Prediction Model 26
7.3 Emotion Detection and Suicidal Prediction based on the Emotion 27
7.4 Code of Implementation 28

8 Testing **29**
8.1 Testing Methodologies 29
8.2 Unit Testing 32
8.3 System Testing 32
8.4 Quality Assurance 33
8.5 Quality Factor 33
8.6 Functional Test 34

9 Results and Performance Analysis **35**
9.1 Input format description 35
9.2 Output 35
9.3 Result Discussion 36

10 Conclusion **38**

References **39**

List of Figures

6.1 Data Flow Diagram……24
8.1 White Box Testing 30
8.2 Black Box Testing..... 31

9.1 User Input of the Face..... 35
9.2 Output Analysis Figure..... 36

Chapter 1

Preamble

1.1 Introduction

The field of mental health has always been a complex and challenging area to address due to its subjective nature and the stigma often associated with it. However, with advancements in technology and the increasing availability of digital data, there is a growing interest in leveraging machine learning techniques to develop innovative solutions for mental health assessment and intervention. One such promising area is emotion detection and suicide prediction using machine learning. Emotions play a crucialrole in our daily lives, influencing our thoughts, behaviors, and overall well-being. The ability to accurately detect and understand emotions can provide valuable insights into an individual's mental state and help identify early warning signs of potential mental health issues. Additionally, the alarming rise in suicide rates worldwide has emphasized the need for proactive interventions and predictive models to identify individuals at risk. Emotion detection and suicide prediction are critical areas of research in the field of mental health. The use of machine learning techniques has gained significant attentionin recent years due to its ability to analyze large datasets and detect patterns that may not be visible to humans. In this technical paper, we present a novel approach to emotion detection and suicide prediction using machine learning algorithms. We propose a comprehensive framework that incorporates various features such as facial expressions, speech patterns, and physiological signals to accurately detect emotions and predict suicidal tendencies. Our framework leverages state-of-the-art machine learning techniques such as deep learning and ensemble methods to achieve high accuracy in emotion detection and suicide prediction. The paper also includes a detailed description of the dataset used, feature extraction methods, and model training and evaluation techniques. Our approach has the potential to revolutionize the field of mental health byproviding early detection and intervention strategies for depressed patients at risk of suicide.

1.2 Existing System

The existing system for emotion detection and suicide prediction using machine learning has seen significant advancements in recent years. Researchers and experts have made considerable progress in developing models and techniques to address these critical areas of mental health. Although the field is still evolving, several noteworthy approaches and frameworks have been established.

1.Emotion Detection Systems:

Emotion detection systems analyze various data sources, such as text, speech, facial expressions, and physiological signals, to identify and classify emotions accurately. Existing approaches often utilize machine learning algorithms, such as deep learning models (e.g., Convolutional Neural Networks, Recurrent Neural Networks), to extract features and make predictions. These models have shown promising results in accurately recognizing emotions in real-time scenarios.

2. Sentiment Analysis:

Sentiment analysis, a subset of emotion detection, focuses on extracting subjective information from textual data. It involves classifying text into categories such as positive, negative, or neutral sentiments. Machine learning algorithms, including Support Vector Machines (SVM), Naive Bayes, and Long Short-Term Memory (LSTM), are commonly employed to perform sentiment analysis. These techniques enable the identification of emotional states expressed in written content, such as social media posts or online forums.

3. Suicide Risk Assessment:

Predicting suicide risk is a complex task that requires careful analysisof various risk factors. Existing systems employ machine learning algorithms to identify patterns and indicators associated with suicidal tendencies. These systems analyze a range of data, including demographic information, social media posts, medical records, and self- reported surveys, to predict the likelihood of suicide attempts. Feature selection, feature engineering, and ensemble methods are commonly used techniques in building effective suicide prediction models.

4. Integrated Approaches:

To enhance the accuracy and reliability of emotion detection and suicide prediction systems, researchers have explored integrated approaches that combinemultiple data modalities. For example, combining facial expressions, speech analysis, and textual data can provide a more comprehensive understanding of an individual's emotionalstate and potential suicide risk. Fusion techniques, such as multimodal fusion or late fusion, are utilized to combine information from different sources and improve overall system performance.

1.2.1 Drawbacks

While existing systems have made significant progress in emotion detection and suicide prediction using machine learning, there are still challenges to overcome.
These challenges include limited labeled data availability, ensuring privacy and ethical considerations, and the need for more comprehensive and accurate prediction models.

Nonetheless, these systems serve as a foundation for further research and development inthe field of mental health assessment and intervention. All the existing systems focus on suicide prediction based on text or image and there is no system to predict the suicide intentions based on the emotion of the person.

1.3 Proposed System

1. The goal of introducing suicidal intention prediction system is to monitor thedepressed patients from remote to prevent suicides.
2. The dataset is trained using Machine Learning algorithm on jupyter notebook.
3. The Decision tree algorithm is used to train the model.
4. The system is given the image of the person with various other inputs of the patient details to detect the emotion and predict the suicidal intentions of the person based on the emotion detected.
5. After taking the input from the user, the system predicts the tendency of the suicidal intention of the person i.e., high/low risk of suicide.
6. Deepface python library is used to detect the emotion of the person which usesCNN model of Deep Learning concept.
7. Decision tree algorithm of Machine Learning is used to predict the suicidal intentions of the patient based on the emotion detected by the emotion detection model.

1.4 Plan of Implementation

The project can be broken down into 5 Major states:

1. The first stage of project is to find a good database for emotion detection. The data should consist of images under each category of emotion.
2. The second stage of project is to train the dataset using the most efficient model to detect the emotion accurately.
3. The third stage is to create a dataset for suicide prediction based on various research and case studies.
4. The fourth stage is to train the suicide prediction data using the most efficient model.
5. The final stage is to integrate both the models to achieve the goal of suicide prediction based on the emotion detected by the emotion detection model.

1.5 Problem Statement

- **Problem:** The suicides are very common in depressed patients and there is no existing system to predict the suicidal intentions of depressed patients based on their emotions andthe portfolio.
- **Affects:** The emotions that require attention of the depressed patients get unnoticed as manual monitoring cannot be continuous and it may increase suicidal risks.
- **Impact:** Ignorance of critical mental health conditions may lead to increase in suicide rates.
- **Solution:** Creation of an efficient model to detect the emotion and predict the suicidalintention of the depressed patients using ML algorithm. It can prevent the suicide by predicting the rate of risk based on the emotion of the patient while monitoring the patient from remote.

1.6 Objective of the Project

The objectives of the “Emotion detection and suicidal intention prediction of depressed patients” can be stated as follows:

- It is a contribution to the mental health organizations for a better monitoring ofthe depressed patients.
- It can detect a person’s emotion and predicting his/her suicidal intentions based on the emotion detected and the details of the patient in the dataset.

Chapter 2

Literature Survey

In this section, we present a literature survey of the existing research on emotion detection and suicide prediction using machine learning techniques.

1. Emotion Detection

In-depth research has been done on emotion recognition utilising a variety of methods, including machine learning algorithms. Facial expression analysis is one of the methods frequently employed for the identification of emotions. Using deep learning, Liu et al.'s study from 2019 [1] suggested a framework for recognising facial expressions. On the Facial Expression Recognition (FER) 2013 dataset, they had an accuracy of 94.68%. A multimodal emotion identification system that incorporates facial expression, speech, and physiological signals was proposed in a different study by Zhang et al. (2020) [2]. The Affect Net dataset showed a 78.8% accuracy for the suggested method.

2. Suicide Prediction

Given the complexity of the issue, predicting suicide is a difficult undertaking. Models for predicting suicide risk have been created using machine learning approaches. NLP is one of the methods frequently used to predict suicide. A machine learning-based technique that employs NLP to predict suicide risk from social media data was proposed in a paper by Coppersmith et al. (2018) [3]. On the dataset from the Suicide Watch subreddit, the suggested method has an accuracy of 80.6%. A model for predicting suicide risk that integrates demographic and clinical data with machine learning algorithms was suggested in a different study by Cao et al. (2021) [4]. On a sizable Chinese dataset, the suggested model has an accuracy of 92.9%.

3. Emotion Detection and Suicide Prediction:

Emotion detection and suicide prediction are closely related areas of research in the fieldof mental health. Several studies have proposed frameworks that combine multiplemodalities to detect emotions and predict suicidal tendencies. A study by Dinesh et al.(2019) proposed a deep learning-based framework that combines facial expressions, speech, and physiological signals for emotion detection and suicide prediction. Theproposed framework achieved an accuracy of 86.8% on the Affect Net dataset and anaccuracy of 93.4% on the Suicide Prevention dataset.

Chapter 3

Theoretical Background

Below is some theoretical background information that highlights various subjects pertinent to the project effort. The description includes a number of issues that merit discussion and highlights some of their shortcomings, which motivate continuing to look for solutions, as well as some of their benefits, which explain why these topics and their features are used in this project.

3.1 Deep Face

DeepFace is a face recognition and facial attribute analysis library developed using Python and Keras. It is built on top of various deep learning frameworks like TensorFlow, Theano, and Caffe.

DeepFace provides a unified API for face detection, alignment, and recognition. It is capable of identifying the faces of people in images or videos with high accuracy, and can also extract various facial attributes such as age, gender, and emotion.

The library provides pre-trained models that can be used for different tasks such as face recognition, face verification, and facial attribute analysis. The pre-trained models are trained on large datasets and can achieve high accuracy when used on new data.

To use DeepFace, you need to install the library and its dependencies. Once installed, you can import the library and use its functions to perform face recognition and facial attribute analysis on images or videos.

Some of the key features of DeepFace include:

1. Face detection: the library can detect faces in images or videos and return the bounding boxes around the faces.
2. Face alignment: the library can align faces so that they are all in a similar position and scale, making it easier to compare them.
3. Face recognition: the library can recognize faces of people in images or videos and return their identities.
4. Face verification: the library can verify if two faces belong to the same person or not.
5. Facial attribute analysis: the library can extract various facial attributes such as age, gender, and emotion from faces.

Advantages of DeepFace

DeepFace is a powerful and versatile tool for face recognition and facial analysis, with high accuracy, efficiency, and ease of integration. Its open-source nature also makes it an accessible choice for researchers and developers looking to work with facial data. It fulfills all the custom needs and requirements. Besides these, DeepFace offers many more advantages as mentioned below

1. High accuracy: DeepFace uses deep learning models that have been trained on large datasets, resulting in high accuracy when detecting faces and recognizing individuals. In fact, DeepFace has achieved state-of-the-art results in face recognition benchmarks.

2. Versatility: DeepFace can perform a wide range of tasks related to facial analysis, including face detection, face recognition, and facial attribute analysis. This makes it a versatile tool for a variety of applications, including security, marketing, and entertainment.

3. Efficiency: DeepFace is designed to be efficient, allowing it to process large amountsof data quickly. It also has a small memory footprint, making it suitable for use on resource-constrained devices like smartphones.

4. Open source: DeepFace is an open-source library, which means that its source codeis freely available for anyone to use, modify, and distribute. This makes it a popularchoice for researchers and developers working on face recognition projects.

5. Integration: DeepFace can be easily integrated with other Python libraries and frameworks, including TensorFlow, Keras, and OpenCV. This allows developers to build custom applications that leverage the power of deep learning for face recognition and analysis.

Disadvantages of DeepFace

While DeepFace has several advantages, there are also some potential disadvantages to consider, including:

1. Bias: DeepFace, like all facial recognition systems, is susceptible to bias, particularlywith regard to gender and race. If the training data used to develop the models is not diverse enough, the system may not perform equally well for all individuals.
2. Privacy concerns: Facial recognition technology, including DeepFace, raises concerns around privacy and surveillance. The use of facial recognition in public spaces has sparked debates around civil liberties and the potential for misuse.

3. Resource-intensive: DeepFace relies on deep learning models, which can be resources- intensive to train and run. This can be a barrier for individuals ororganizations

Without accessing to high-performance resources.

4. Limited data availability: While there are large datasets available for face recognition training, there may be limitations on the availability of specific datasets for certain applications or populations.
5. Limitations in performance: While DeepFace has achieved state-of-the-art performance in several benchmarks, its accuracy may be limited in certain scenarios, such as when images have low resolution or poor lighting condition.

3.2 Numpy

NumPy (Numerical Python) is a powerful open-source library for the Python programming language, used for scientific computing and data analysis. NumPy provides an efficient arraydata structure for handling large multi-dimensional arrays and matrices, along with a wide range of mathematical and logical functions for manipulating the arrays. It also provides tools for working with linear algebra, Fourier analysis, and random number generation.

Some key features of NumPy include:

1. nd-array: The nd-array (N-dimensional array) is the primary data structure in NumPy, which provides a fast and memory-efficient way of storing and manipulating multi-dimensional arrays. The nd-array can be created using the np.array() function or by readingdata from a file.

2. Broadcasting: NumPy provides broadcasting, which is a powerful mechanism for performing arithmetic operations on arrays of different shapes and sizes.

3. Mathematical functions: NumPy provides a wide range of mathematical functions, including trigonometric functions, logarithmic functions, exponential functions, and more.

4. Linear algebra: NumPy provides tools for working with matrices and linear algebra, including matrix multiplication, matrix inversion, and eigen decomposition.

5. Fourier analysis: NumPy provides functions for performing Fourier analysis on arrays, including the fast Fourier transform (FFT).
6. Random Number generation : NumPy provides functions for generating random numbers and arrays, including normal distribution, uniform distribution, and more.

NumPy is widely used in scientific computing, data analysis, machine learning, and other domains that require handling of large datasets and complex mathematical operations. It isa foundational library for many other scientific Python libraries, including Pandas, Scikit-learn, and TensorFlow.

3.3 Pandas

Pandas is a popular open-source library for the Python programming language used for datamanipulation, analysis, and cleaning. It provides a powerful data frame structure for handling tabular data, similar to spreadsheets or SQL tables, along with a wide range of functions for manipulating and transforming data.

Some key features of Pandas include:

1. Data frames: The data frame is the primary data structure in Pandas, which provides a two-dimensional table-like structure with rows and columns. Data frames can be created from a variety of sources, including CSV files, Excel spreadsheets, SQL databases, and more.

2. Data manipulation: Pandas provides a wide range of functions for manipulating and transforming data, including filtering, sorting, grouping, merging, and more. These functions can be used to clean and preprocess data, prepare it for analysis, and transform it into different formats.

3. Data analysis: Pandas provides functions for performing statistical analysis on data, including calculating summary statistics, correlations, and more. It also provides tools for visualizing data using built-in plotting functions or by integrating with external visualizationlibraries like Matplotlib and Seaborn.

4. Missing data handling: Pandas provides functions for handling missing or incomplete data, including filling in missing values, dropping rows with missing values, and more.

5. Time series analysis: Pandas provides functions for working with time series data, including resampling, rolling window calculations, and more.

Pandas is widely used in data science, finance, and other domains that require handling and analyzing large amounts of data. It is a foundational library for many other Python data analysis libraries, including NumPy, Scikit-learn, and TensorFlow.

3.4 Machine Learning

To In particular, the creation of algorithms and models that allow computers to learn and make predictions or judgements without being explicitly programmed is the focus of the area of machine learning, a branch of artificial intelligence (AI). In order to train systems on big datasets and enable them to automatically improve over time, it entails the application of statistical approaches and computing algorithms.

There are various types of machine learning algorithms, including:

1. Supervised Learning: In this approach, the algorithm is trained on labeled data, where the input features are mapped to known output labels. The goal is to learn a mapping function that can accurately predict labels for new, unseen data.

2. Unsupervised Learning: Here, the algorithm learns patterns and structures in unlabeled data without any predefined output labels. It aims to discover hidden patterns, groupings, or relationships within the data.

3. Semi-supervised Learning: This combines elements of both supervised and unsupervised learning. It uses a small amount of labeled data alongwith a larger amount of unlabeled data to improve the learning process.

4. Reinforcement Learning: This learning paradigm involves an agent interacting with an environment and learning through a trial -and-error process. The agent receives feedback (rewards or penalties) based on its actions and adjusts its behavior to maximize the rewards.

Machine learning has been successfully applied to various domains, including image and speech recognition, natural language processing, recommendation systems, fraud detection, healthcare, finance, and more. It has revolutionized industries by enabling automation, improving decision-making processes, and extracting valuable insights from large and complex datasets.
However, machine learning also comes with challenges such as data quality, bias, interpretability, and ethical considerations. Addressing these challenges is crucial for building robust and trustworthy machine learning systems.

Overall, machine learning is a powerful tool that empowers computers to learn from data and make intelligent predictions or decisions. Its applications have the potential to transform industries, improve efficiency, and contribute to scientific advancements.

3.5 Decision Tree Algorithm

A decision tree algorithm is a supervised learning method used in machine learning and datamining to build a model that predicts the value of a target variable based on several input features. It is a type of predictive model that uses a tree-like structure to represent a set of decision rules and their possible outcomes.

The decision tree algorithm works by recursively splitting the data based on the values of the input features until a stopping criterion is met. At each split, the algorithm selects the feature that best separates the data based on some criterion, such as information gain or Gini index. This process continues until the data is completely classified or until a stopping criterion, such as a maximum depth or minimum number of samples at a leaf node, is reached.

The resulting tree is a set of decision rules that can be used to classify new instances of the data. To classify a new instance, the algorithm follows the decision path from the root of the tree to a leaf node, where the class label is assigned based on the majority class of the training examples at that node.

Advantage of Decision Tree Algorithm

Decision trees have several advantages, including:

1. Interpretable: The resulting decision tree is easy to interpret and can provide insights into the underlying data and decision-making process.

2. Non-parametric: Decision trees do not make assumptions about the distribution of the data and can be used with both categorical and continuous input features.

3. Robust: Decision trees can handle missing values and outliers in the data and are resistant to overfitting if appropriate pruning techniques are used.

Disadvantage of Decision Tree Algorithm

However, decision trees also have some limitations, including:

1. Overfitting: Decision trees are prone to overfitting if the tree is too deep or if the training data is noisy or biased.

2. Instability: Small variations in the training data can result in a different decision tree, making the model less stable.

3. Bias: Decision trees can be biased if certain input features are more heavily weighted in the decision-making process.

4. Limited expressiveness: Decision trees are limited in their ability to express complex relationships between input features.

Chapter 4

System Requirements Specification

Prior to beginning any actual design or development work, a system requirement specification (SRS) is essentially an organization's understanding of a customer's or potential client's system requirements and dependencies at that specific time. A document that outlines a list of requirements is created using the data acquired during the study. It provides a concise explanation of the services the system should offer as well as the limitations that the system must abide by. SRS is typically a document that exhaustively outlines what the proposed software should accomplish without outlining how it will achieve it. It is a two-way insurance policy that ensures that at any given time, the client and the organisation are both aware of the other's requirements from that perspective.

A requirement is an aspect of a capability or condition that the system must meet. Requirement management is a methodical process for gathering, organising, and clearly articulating the system requirements together with the relevant Attributes. The elusive challenges of requirements can arise from a variety of sources and are not always visible.

4.1 Functional Requirement

A functional requirement specifies how a software system should operate and respond to particular inputs or circumstances. Calculations, data processing, and other specialised functionality may be among them. The functional specifications for this system are as follows: -

1. The entire code must be written in Python.

2. All dependencies of machine learning and Python must be properly installed and withupdated version.

3. Deep Face library must be imported and work properly for predicting the emotions of human being.

4. The Decision Tree trained model must work properly while predicting the result.

4.2 Non-Functional Requirements

Non-functional requirements are those that have no direct bearing on the particular function that the system provides. Instead of describing particular behaviours, they define the standards that can be used to evaluate how a system works. They might be connected to characteristics of emergent systems including dependability, response time, and store occupancy. Non-functional requirements can be caused by external reasons like: - Budgetary restrictions, organisational regulations, the need for interoperability, or user demands. - Non-functional requirements can also be caused by:

- Product Requirements
- Organizational Requirements
- User Requirements

4.2.1 Product Requirements

1. **High accuracy:** The emotion detection and suicidal intention prediction algorithms should have high accuracy, so that the product can provide reliable insights into the patient's emotional state and risk of suicidal ideation.

2. **Real-time processing:** The product should be able to process data in real-time, so that it

can provide immediate feedback to healthcare professionals and intervene in emergency situations.

3. **Large dataset:** The product should be trained on a large dataset of emotional expressions and suicidal ideation, so that it can detect a wide range of emotions and predict suicidal risk accurately.

4. **User-friendly interface:** The product should have a user-friendly interface that is easy to use and understand for healthcare professionals, including psychiatrists, psychologists, and social workers.

5. **Integration with electronic health records (EHRs):** The product should be able to integrate with EHRs to access patient data, including demographics, medical history, and medication usage.

6. **Privacy and security:** The product should ensure privacy and security of patient data, and comply with data protection regulations, including HIPAA.

7. **Customization:** The product should provide customization options, so that it can be adapted to different patient needs and preferences.

8. **Alerts and notifications**: The product should provide alerts and notifications to healthcare professionals when a patient is at risk of suicidal ideation or behavior, so that they can intervene promptly.

9. **Collaboration:** The product should foster collaboration between different healthcare professionals, including psychiatrists, psychologists, and social workers, to ensure coordinated care.

10. **Continuous improvement:** The product should be continuously improved based on user feedback and new insights from research and development, to provide the most accurate and reliable emotion detection and suicidal intention prediction capabilities for depressed patients.

Non-functional requirements are frequently referred to as a system's attributes. Execution quality and evolution quality are two categories for these criteria. Execution qualities, which are noticed during system operation, include security and usability, while evolution qualities include testability, maintainability, extensibility, or scalability.

4.2.2 Organizational Requirements

Expertise: The organization should have a team of experts in data science, machine learning, computer vision, and NLP to develop, implement, and maintain the product.

Data management: The organization should have a robust data management system to collect, store, process, and analyze large amounts of patient data, including emotional expressions, medical history, and medication usage.

Infrastructure: The organization should have a strong infrastructure to support the product, including hardware and software resources, cloud computing capabilities, and networking capabilities.

Regulatory compliance: The organization should ensure that the product complies with applicable laws, regulations, and ethical standards related to data privacy, security, and patient confidentiality, including HIPAA.

Integration with electronic health records (EHRs): The organization should ensure that the product can integrate with EHRs to access patient data.

Testing and validation: The organization should have a process for testing and validating the product, including user testing and validation against ground truth data, to ensure accuracy and reliability.

Clinical validation: The organization should conduct clinical validation studies and safety of the product, and ensure that it can be used by healthcare professionals in a clinical setting.

Collaboration: The organization should including psychiatrists, psychologists, and social workers, to ensurecoordinated care and treatment for depressed patients.

4.2.3 User Requirements

1. User must able to open and use this software successfully.
2. User is some health organization who will monitor the patients.
3. User must have enabled camera to use this software to capture the photo.
4. User must Incorporate this software on their system to monitor/predict the patient intensions.
5. User must able to give the captured image to get the emotion of patient.
6. User must enter the details of patients like, his/her age, gender, social interactions level, Depression level, working status, admission status and drugs consumption status.
7. User must able to get the accurate prediction of the intension as a result.

4.2.1 Basic Operational Requirements

With special emphasis on the operator as the key customer. Operational requirements will define need and, at a minimum, will be related to these following points:

Mission profile or scenario: It details the steps used to achieve the mission's goal. Additionally, it determines how effective or efficient the system is.
Utilization environments: It explains how the steps taken to achieve the mission's goal were carried out. It determines the system's effectiveness or efficiency as well.
Operational life cycle: It defines the system lifetime.

4.2.2 System Configuration

H/W System Configuration:

Processor - Core I3 Speed -

2.0 Ghz

RAM - 8GB(min)

Hard Disk - 40 GB

Key Board - Standard Windows Keyboard
Mouse - Two or Three Button Mouse Monitor
- SVGA

S/W System Configuration:

Operating System: Windows 8/8.1/10 Coding
Language: Python
Tools: Jupyter Notebook, VS Code, Machine and Deep Learning

Chapter 5

System Analysis

System analysis involves identifying and understanding the current issues, defining objectives and requirements, and assessing potential solutions to find the optimal resolution to a problem. This approach requires a thoughtful examination of the organization and the challenges it faces, along with the use of various technologies to address these issues. The feasibility study is a crucial aspect of the system analysis process, as it sets the parameters for the design and development phases.

5.1 Feasibility Study

All systems are feasible when provided with unlimited resource and infinite time. In reality, the ideal conditions for system analysis may not always exist, making it crucial and wise to assess the feasibility of a system as early as possible. Feasibility and risk analysis are closely related, as high project risks can lower the feasibility of creating high-quality software. Therefore, three important factors to consider in the feasibility analysis are:

· ECONOMICAL FEASIBILITY

· TECHNICAL FEASIBILITY

· SOCIAL FEASIBILITY

5.1.1 Economical Feasibility

The purpose of this study is to assess the financial implications that the system will have on the organization. Due to limited resources available for research and development, it is essential to ensure that any expenses incurred are justifiable. Therefore, the economic impact of the system must be thoroughly examined to determine the feasibility of investing in it.

the developed system as well within the budget and this was achieved because mostofthe technologies used are freely available. Only the customized products had to be purchased.

5.1.2 Technical Feasibility

The purpose of this study is to evaluate the technical feasibility of the system, which involves assessing its technical requirements. It is crucial that the system does not place excessive demands on the available technical resources, as this could lead to client overload. Therefore, it is essential to ensure that the system has modest technical requirements, allowing for easy implementation with minimal or no changes required.

5.1.3 Social Feasibility

The goal of the study is to determine how much the user accepts the system. This includes the instruction needed for the user to operate the system effectively. The system shouldn't make the user feel threatened; instead, they should view it as a need. The techniques used to inform and acquaint the user with the system are the only factors that affect the level of acceptance by the users. As the system's ultimate user, his confidence must be increased so that he may offer some helpful criticism, which is encouraged.

5.2 Analysis

5.2.1 Performance Analysis

The project is run with the aid of a strong networking infrastructure to ensure full operation. To determine the viability of the suggested system, performance analysis is conducted. The process of performance analysis and definition must be carried out concurrently.

5.2.2 Technical Analysis

System is only useful if it can be transformed into information systems that will satisfy the technical requirements of the organization. Simply put, this test of feasibility investigates if the system will function as intended when installed and investigates whether there are any significant obstacles to deployment. There are various things to pay attention to with regard to all these technical analysis on:-

- **Changes to bring in the system:** All changes should be in positive direction,there will be increased level of efficiency and better customer service.

- **Required skills:** Platforms tools used in this project are widely used. So theskilled manpower is readily available in the industry.

- **Acceptability:** The structure of the system is kept feasible enough so thatthere shouldnot be any problem from the user's point of view.

5.2.3 Economical Analysis

Economic analysis is performed to evaluate the development cost weighed against the ultimate income or benefits derived from the developed system. For running this system, we need not have any routers which are highly economical. So the system is economically feasible enough.

Chapter 6

System Design

A design is a useful technical depiction of a future construction. It is the most important stage in a system's development. The process of translating requirements into a software representation is known as software design. at software engineering, design is encouraged at the design phase. The new system must be designed based on the user needs and a thorough study of the current system. The system design phase is currently underway. The best technique to faithfully transfer a customer's requirement into the final software solution is through design. Design develops a representation or model and gives information on the architecture, interfaces, and components required to implement a system. The system design that was reached logically as a result of systems analysis is converted into physical system design.

6.1 System Development Methodology

The system development method is a procedure used to finish a product or solve a problem with a product. A software development process is defined as a series of stages, steps, and procedures that results in a finished product. It follows a sequence of processes that are utilised to advance the product. System Design for ML-Based Suicide Prediction and Emotion Detection.

Data Collection

- Identify and collect relevant datasets for emotion detection and suicide prediction. This may include textual data from social media posts, clinical records,or audiovisual data for multimodal analysis.

- Implement data preprocessing steps such as cleaning, normalization, and feature extraction to prepare the data for analysis.

- Ensure the dataset is appropriately labeled for emotion categories and suicide risklevels.

Feature Extraction

- Extract relevant features from the collected data. For text-based analysis, techniques like TF-IDF, word embeddings (e.g., Word2Vec, GloVe), or deeplearning-based approaches (e.g., recurrent neural networks) can be used.

- For audiovisual data, extract features such as acoustic features (e.g., pitch, intensity) from speech signals and facial expressions (e.g., Action Units) from images or videos.

- Consider combining multiple modalities for a multimodal approach, leveraging both textual and audiovisual features.

Emotion Detection

- Train the model using the labeled dataset, validating and fine-tuning the model using appropriate evaluation metrics (e.g., accuracy, F1 score).

- Optimize hyperparameters to improve the model's performance.

Suicide Prediction

- Develop a machine learning model for suicide prediction. This can involve binary classification models to predict suicide risk (e.g., logistic regression, SVM, or deeplearning models).

- Extract relevant features related to suicide risk factors such as language patterns,sentiment, social network analysis, or clinical indicators.

- Train the model using the labeled dataset, applying suitable techniques for databalancing if required.

- Evaluate the model's performance using appropriate metrics, such as accuracy, precision, recall, or area under the receiver operating characteristic curve (AUC-ROC).

Integration and Real Time Analysis

- Develop a system that integrates the trained emotion detection and suicideprediction models.
- Create an interface or API for real-time analysis, allowing users to input textualor audiovisual data for emotion detection and suicide risk assessment.
- Implement real-time processing of input data using the trained models toprovide instant predictions or risk scores.

Model Evaluation and Monitoring

- Continuously evaluate and monitor the performance of the deployed models toensure their effectiveness and reliability.
- Collect feedback from users and domain experts to further refine and improvethe system.
- Perform regular updates and retraining of the models as new data becomesavailable or as the system evolves.

Privacy and Ethical Consideration

- Address privacy concerns by ensuring the security and confidentiality of thecollected data.
- Consider ethical implications and potential biases in the models and take stepsto mitigate them.

By following this system design, we can develop an effective and robust system for emotion detection and suicide prediction using machine learning techniques. It's important to adapt the design to the specific requirements and constraints of your project,considering factors such as available data, computational resources, and the target and application environment.

6.2 Data Flow Diagram

The bubble chart is another name for the DFD. It is a straightforward graphical formalism that may be used to depict a system in terms of the data that is fed into it, the different operations that are performed on it, and the data that is generated as a result of those operations.

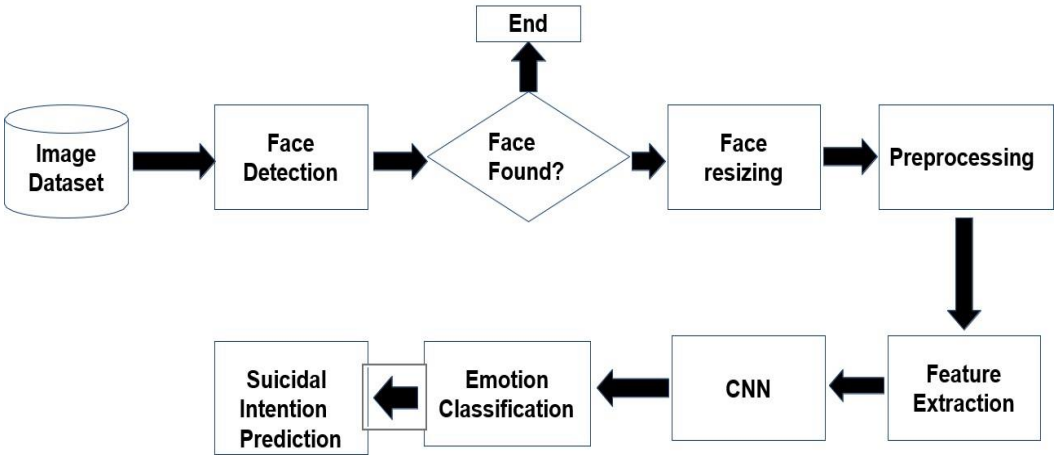

Figure 6.2.1 Data flow diagram

Figure 6.2.1 represents the following:

- Dataset: The image of the face is taken from the user as input with other detailsof the registered patients.
- Face detection: face detection model takes the input image and checks if theimage is valid.
- Face resizing: the image is resized to fit the requirements.
- Preprocessing: The preprocessing of the data is done by converting it to the grey scale image.
- Feature extraction: Feature extraction is a critical step in machine learning and data analysis, where meaningful and informative features are extracted from raw data to represent the underlying patterns and characteristics of the data.
- Emotion classification: Emotion classification using Convolutional NeuralNetworks (CNNs) involves training a CNN model to recognize and classify emotions in images.
- Suicidal intention prediction: The emotion detected is then passed to the intention prediction model to predict the intention of the depressed patient.

Chapter 7

Implementation

7.1 Emotion Detection Model

The data was collected from Kaggle in csv format. There are four basic steps to be performed for implementing emotion detection.

i. Preprocessing
ii. Face registration
iii. Facial feature extraction
iv. Emotion classification

Description about all these processes is given below:

1. Preprocessing:

Preprocessing refers to a set of image operations at the most fundamental level of abstraction, where both input and output are intensity images. Common preprocessing steps involve various operations, including:

a. Reduce the noise
b. Convert the Image To Binary/Grayscale.
c. Pixel Brightness Transformation.
d. Geometric Transformation

2. Face Registration: The computer technology of face registration is utilized in diverse applications to recognize human faces in digital images. In the face registration process, the first step involves identifying faces in the image using a set of landmarkpoints known as "face localization" or "face detection". The next step entails geometrically normalizing the detected faces to align with a template image in a process called "face registration".

3. Facial Feature Extraction: Facial feature extraction is a crucial step in face recognition that involves identifying specific regions, points, landmarks, or contoursin each 2D or 3D range image. This process generates a numerical feature vector

from the registered image. Various features can be extracted during this step, including but not limited to:

a. Lips
b. Eyes
c. Eyebrows
d. Nose tip

4. Emotion Classification: During the third stage of the process, the algorithm endeavors to categorize the faces into one of the seven fundamental emotions. PaulEkman, an American psychologist and professor emeritus at the University of California, San Francisco, is a notable figure in the research of emotions and their correlation with facial expressions. He is credited with developing an "atlas of emotions" that comprises over ten thousand facial expressions.

Our purpose is to improve the accuracy of facial expression classification by using a new CNN architecture. As a first step, after preparing the database we fixed the batch size inputof CNN architecture to 165 × 165 then we trained the architecture with fine tuning by Visual Geometry Group (VGG) model to generate the first model. In the second step to improve the classification we repeat the training of our CNN architecture, but the fine tuning here is achieved with the obtained first model, and finally we get our final model. After detection of the human emotions with and without mask we go on predicting their intentions based on the emotion detected.

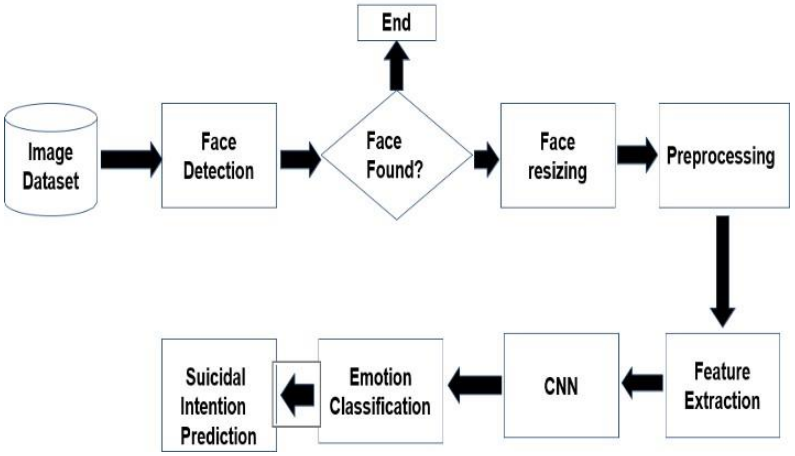

Figure 7.1.1 Data flow diagram

7.2 Suicidal intention prediction model

Following are the steps taken to generate a model using Machine Learning:

1. Dataset creation
2. Data preprocessing

3. Data Visualization

4. Training the data using the most efficient ML algorithm

1. Dataset Creation

Creation of the dataset is the most challenging part of the entire project as a lot of researchwas needed to be done on various case studies of depressed patients. After all the research,the following are the dependent features of the dataset that were proved to be most valuable factors affecting the suicide intentions in depressed patients: emotion, age, relationship status, gender, depression level, working status, no of days of admission in the hospital, social interaction level, whether admitted in hospital, drugs/alcohol consumption.All these are the factors that are considered to predict the rate of suicide risk. The final column in the dataset is the intention column that is the dependent feature. The dependent feature was evaluated based on the knowledge gained by various case studies.

2. Data preprocessing

The data preprocessing is an important step to filter the relevant data from the large dataset. In this step all the fields with null values were replaced by the desired or the mostrelevant values. Label Encoding was done for the categorical data.

3. Data visualization

Data was visualized using the heatmap function of the seaborn library to figure out the most relevant independent features. various graphs were plotted to determine the relation between the columns.

4. Training and testing

The data was spitted into training and testing datasets in 80:20 ratio. The DecisionTreeClassifier model was imported and used to train the training dataset. The Decision Tree model showed the accuracy of 80% on testing the model.

7.3 Emotion Detection and Suicidal Intention Prediction based on the Emotion

The image of the face and various desired data about the depressed patient was takenas input from the user. The image was passed to the emotion detection model to detect the emotion of the person and the resultant dominant emotion with all the other inputs was passed to the intention prediction model to predict the rate of suicidal intention of the person on that specific emotion.

7.4 Code

7.4.1 run.py

```
0]: from deepface import DeepFace

1]: file_path = input("Enter the file path of the input image: ")

    Enter the file path of the input image: images/img1.png

2]: face_analysis = DeepFace.analyze(img_path = file_path)
    emotion=face_analysis[0]["dominant_emotion"];
    print(emotion)

    Action: race: 100%|████████████████████████████████████████████████| 4/4 [00:02<00:00,  1.66it/s]

    happy

3]: age=int(input("age [0 => 15-25, 1 => 26-35, 2 => 36-45, 3 => 46-55, 4 => 56-65]  :  "))
    relationship_status=int(input("relationship_status [0 => unmarried, 1 => married, 2 => in a relationship, 3 => breakup, 4 => divo
    gender=int(input("gender [0 => male, 1 => female, 2 => transgender]  :  "))
    depression_level=int(input("depression level [0 => mild, 1 => moderate, 2 => severe]  :  "))
    working_Status=int(input("working_status [0 => unemployed, 1 => employed]  :  "))
    no_of_days=int(input("no of days  :  "))
    social_interaction_level=int(input("social_interaction_level [0 => No_social_interaction - living alone, 1 => Less social interac
    admission_status=int(input("admission_status [0 => not admitted, 1 => admitted]  :  "))
    drugs_alcohol=int(input("drugs/alcohol [0 => donot consume drugs/alcohol, 1 => consume drugs/alcohol]  :  "))

:   if(emotion=="angry"):
        emotion=0
    elif(emotion=="anxiety"):
        emotion=1
    elif(emotion=="fear"):
        emotion=2
    elif(emotion=="happy"):
        emotion=3
    elif(emotion=="neutral"):
        emotion=4
    elif(emotion=="sad"):
        emotion=5
    else:
        emotion=6

:   import numpy as np
    testdata=[[emotion,age,relationship_status,gender,depression_level,working_Status,no_of_days,social_interaction_level,admission_s
    input_data=np.array(testdata)
    input_data

:   array([[ 3,  1,  3,  1,  1,  0, 10,  1,  1,  0]])

:   # Load the model
    import joblib
    model = joblib.load('decisiontree_model.joblib')

:   prediction = model.predict(input_data)
    if(prediction):
        print("High risk of suicide")
    else:
        print("Low risk of suicide")
```

Chapter 8

Testing

8.1 Testing Methodologies

There are many different types of testing methods or techniques used as part of the software testing methodology. Some of the important testing methodologies are:

8.1.1 White Box Testing

White box testing, sometimes referred to as structural testing, glass box testing, clear box testing, and transparent box testing, creates test cases based on the internal organisation of the system. It takes programming knowledge to figure out every turn the software makes. The tester determines the appropriate outputs and selects test case inputs to run through multiple code routes. Despite the fact that this type of test design can unearth a staggering amount of test cases, it might not catch unimplemented requirements or sections of the specification, but one can be sure that all possible paths through the test object are taken. White box testing allows us to create test cases that:

1. Ensure that each independent path has been used at least once within a module.
2. Consider both the true and false sides of any logical conclusion.
3. Carry out each loop's operations within and at its limits.
4. Execute internal data structure to assure their validity

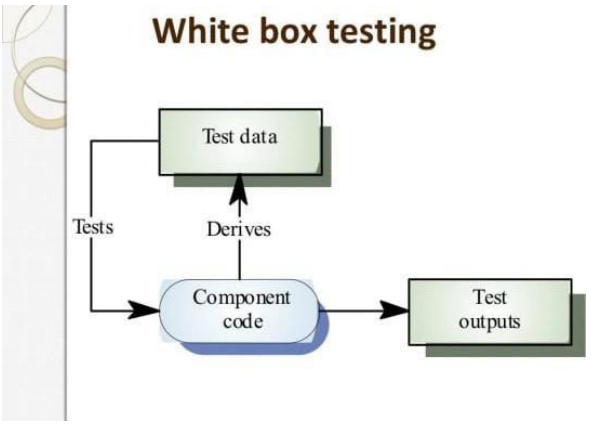

Figure 8.1: White Box Testing

8.1.1.1 Advantages of White Box Testing

1. There is no need to wait for the user interface (UI) to be finished before beginning white box testing of the desired application. It covers all potential code paths, ensuring comprehensive testing.
2. It helps in checking coding standards.
3. The number of test cases that must be run during black box testing can be decreased by asking the tester about how each section is implemented. By doing this, it may be able to remove unused or outdated coding.

8.1.1.2 Disadvantages of White Box Testing

1. To test the software application a highly skilled resource is required to carry out testing who has good knowledge of internal structure of the code which willincrease the cost.
2. Updating the test script is required if there is change in requirement too fre- quently.
3. If the application to be tested is large in size, then exhaustive testing is impos- sible.
4. It is not feasible to verify each and every path or condition of a software programme because this could lead to the omission of coding errors.

5. White box testing is a very expensive type of testing.

6. To test each path or condition, different input conditions may be needed, therefore the tester must produce a range of inputs, which may be time-consuming, in order to test the entire application.

8.1.2 Black box testing

Black box testing concentrates on the software's functional needs. Functional testing is another name for it. It is a method of testing software in which the tester is not aware of how the thing being tested operates internally. For instance, in a "black box" test of software design, the tester is only aware of the inputs and the desired results but not how the program generates them. The tester only needs to be aware of the program's requirements; they are the only things they ever look at in the programming code. It allows us to generate input sets that will completely exercise all of a program's functional needs. White box technique is an alternative to black box testing. Instead, it is a complementary approach that islikely to uncover a different class of errorsin the following categories:-

- Incorrect or missing function
- Interface errors.
- Performance errors
- Initialization and termination errors.
- Errors in objects.

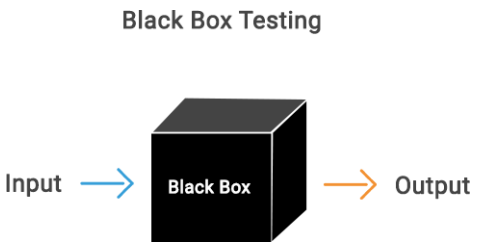

Figure 8.2: Black Box Testing

8.1.2.1 Advantages of Black Box Testing

1. The test is unbiased as the designer and the tester are independent of each other.

2. The tester does not need knowledge of any specific programming languages.

3. The test is done from the point of view of the user, not the designer.

4. Test cases can be designed as soon as the specifications are complete.

8.1.2.2 Disadvantages of Black Box Testing

1. The test inputs need to be from large sample space. That is, from a huge set of data this will take time.

2. Also it is difficult to identify all possible inputs in limited testing time. So writingtest cases is slow and difficult.

3. Chances are more that there will be unidentified paths during this testing.

8.2 Unit Testing

Creating test cases for unit testing guarantees that the fundamental programme logic is operating properly and that inputs into the programme produce legitimate outputs. All decision-making branches and internal code flow should be examined. It involves testing each individual piece of software that makes up the programme. An understanding of how it was constructed is necessary for this intrusive structural test. Fundamental testing are performed at the component level via unit tests, which look at a specific operational process, piece of software, or system configuration. Unit tests provide assurance that each unique business process path complies properly with the published specifications and has clearly specified inputs and outputs.

8.3 System Testing

This knowledge aids in clearing up any confusion regarding the system. For instance, while considering whether to release a product, the decision-makers would need to be aware of the product's current status, including elements like its usability, compliance with any applicable rules, and its conformance to requirements. Software testing offers objective evaluations of how well the system adheres to stated objectives and specifications. System testing verifies all end-to-end processes, just as a user would. By putting the system up in a controlled environment, it can be tested to see if its functionality is correct. Testing of functional and nonfunctional requirements is part of system testing. It is beneficial to confirm and validate validate the system. All components of system should have been successfully unit tested and then checked for any errors after integration

8.4 Quality Assurance

The management tasks of audits and reporting make up quality assurance. The purpose of quality assurance is to give management the information they need to understand product quality, develop understanding, and feel confidence that the product quality is accomplishing its objectives. This "umbrella activity" is utilised at all stages of the engineering process.

8.5 Quality Factor

An important objective of quality assurance is to track the software quality and assess the impact of methodological and procedural changes on improved software quality.The factors that affect the quality can be categorized into two broad groups:

1. Factors that can be directly measured.

2. Factors that can be indirectly measured

- These factors focus on three important aspects of a software product

1. Its operational characteristics

2. Its ability to undergo changes

3. Its adaptability to a new environment.

8.6 Functional Test

Functional tests offer methodical proof that the functions being tested are available in accordance with the technical and business requirements, system documentation, and user manuals. Focus of functional testing is on the following areas:

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

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

3. Functions : identified functions must be exercised.

4. Output : identified classes of application outputs must be exercised.

5. Systems/Procedures: Interfacing systems or procedures must be invoked

With a focus on requirements, crucial functionality, or special test scenarios, functional tests are planned and created. The systematic coverage of data fields, established practices, following processes, and business process flows must all be taken into account during testing. Before functional testing is complete, further tests are discovered, and the value of the currently conducted tests is evaluated.

8.6.1 Accuracy of Features

1. Accuracy of Logistic Regression model is 75%.

2. Decision Tree Model has an accuracy of 80%.

Chapter 9

Results and Performance Analysis

9.1 Input Format Description

The input is taken from the user in the following format:

- image: image of the face of the patient in jpeg/jpg/png format
- age: age of the patient (numeric)
- relationship status: relationship status of the patient (married/single/divorced)
- gender: gender of the patient (male/female/transgender)
- depression level: depression level of the patient as specified by the doctor (acute/moderate/ severe)
- working status: working status of the patient (employed/unemployed)
- no of days: no. of. days the patient is admitted in the hospital (numeric)
- social interaction level: the level of interaction a person has in his personallife (numeric)
- admission status: whether the patient is admitted in the hospital (yes/no)
- drugs/alcohol: whether the patient is drug/alcohol addicted (yes/no)

9.2 Input from the User

The input from the user is taken in the following format:
photo:

Fig. 9.2.1 user input of the face

- age: 30

- relationship status: unmarried
- gender: male
- depression level: severe

- working status: unemployed
- no of days: 50
- social interaction level: 2
- admission status: yes
- drugs/alcohol: yes

9.3 Output

The output is displayed in the following format:

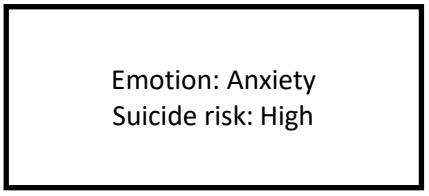

Fig. 9.2.2

The image of the face and various desired data about the depressed patient was takenas input from the user as shown in the Fig. 9.2.2 The image was passed to the emotiondetection model to detect the emotion of the person and the resultant dominant emotion with all the other inputs was passed to the intention prediction model to predict the rate of suicidal intention of the person on that specific emotion. The dominant emotion of the patient is anxiety as detected by the emotion detection model and the suicidal intention rate predicted by the intention prediction model is high.

9.4 Result Discussion

The decision tree and logistic regression algorithms were used to train the suicidal intention prediction model and the decision tree model was found to be more accurate with the testing data as decision tree algorithm fits well with the numerical data. In the context of predicting the suicidal intention of depressed patients, a decision tree algorithm may have some advantages over logistic regression.

First of all, decision trees are capable of handling non-linear interactions between features and the target variable, which may be crucial when working with complex mental health data that may contain such relationships. Decision trees can also deal with missing values in the data, which might be common in datasets related to mental health. In contrast, before the model can be trained for logistic regression, missing values must be imputed or removed. Thirdly, while logistic regression is better suited for categorical data, decision trees can handle both continuous and categorical data. Fourthly, because there may be intricate interactions between various risk factors, decision trees' ability to manage interactions between variables may be crucial when predicting a patient's desire to commit suicide. Finally, decision trees can produce data that are easy to interpret and can aid physicians in locating critical risk factors for suicide intention. Given that it is a linear model and does not offer a clear visual picture of the decision-making process, logistic regression may not offer

as much interpretability.

Of course, the choice of algorithm ultimately depends on the characteristics of the data and the specific problem at hand.

It's important to carefully consider the strengths and weaknesses of each algorithm before making a decision.

Chapter 10

Conclusion

The project "Emotion Detection and Suicidal Intention Prediction of Depressed Patients using ML" aims to develop a machine learning model that can detect emotions and predictsuicidal tendencies in depressed patients based on their facial expressions.
The project utilizes a Convolutional Neural Network (CNN) for emotion detection and a Decision tree algorithm for suicidal intention prediction. The model was trained and tested using the FER-2013 and 90,000 datasets, which include facial images with different emotional expressions. The project has important implications for the field of mental health, as it provides a potential tool for early detection and intervention in depressed patients who may be at risk for suicidal behavior. This project represents an important step towards using machine learning techniques for mental health assessment and intervention. Further research and development are necessary to improve the accuracy and reliability of the models, as well as to address ethical and privacy concerns related to the use of personaldata in mental health assessments. This project can be enhanced further by adding more independent features in the datasetthat have a great impact in the suicide intentions of the people to increase the accuracy. The real time data can be used from the hospital and fed to this model to predict the results. This project has a lot of scope as it can be used to monitor the depressed patients from remote to prevent suicides through CCTV cameras that can record the facial expressions and the model can predict the suicidal intentions based on the emotions detected. The prediction of suicidal risk of depressed patients based on their emotion helps to prevent the suicides that are very common in depressed patients when they are alone. This project aims at providing a tool that will automatically detect the suicide risk rate of aperson based on various emotions. This project can be used to monitor the depressed patients and their suicidal intentions from even remote places.

References

[1] Abir Fathallah, Lofti Abdi, and Ali Douik, " Facial Expression Recognition via Deep Learning".

[2] Dilbag Singh, " Human Emotion Recognition System.

[3] Daun Shin, Kyundgo Kim, Seung-Bo Lee, Changwoo Lee, Ye Seul Bae, Won ik cho, Min Ji Kim, C. Hyung Keun Park, Eui Kyu Chie, Nam Soo Kim, and Young Min Ahn, "Detection of Depression and Sucide Risk Based on Text From Clinical Interviews using Machine Learning"[National Library of Medicine].

[4] Ayaan Haque, "NLP for Sucide and Depression Identification with NoisyLabels.

[5] https://worldpopulationreview.com/country-rankings/depression-rates-by-country.

[6] https://m.economictimes.com/magazines/panache/one-in-seven-indians-between-15-24-years-of-age-feel-depressed-lack-interest-in-doing-things-shows-unicef-report/articleshow/86807417.cms.

[7] https://mindvoyage.in/depression-in-india-latest-statistics/#:~:text=According%20to%20a%20report%2C%20in,the%20cases%20stand%20at%2056%2C675%2C969.

[8] https://www.who.int/india/health-topics/depression.Nie, W., Bao, Y., Zhao, Y., & Liu, A. (2023). Long Dialogue Emotion Detection Based on Commonsense Knowledge Graph Guidance. IEEE Transactions on Multimedia.

[9] Alsubai, S. (2023). Emotion Detection Using Deep Normalized Attention-Base Neural Network and Modified-Random Forest. Sensors, 23(1), 225.

[10] Karilingappa, K., Jayadevappa, D., & Ganganna, S. (2023). Human emotion detection and classification using modified Viola-Jones and convolution neural network. IAES International Journal of Artificial Intelligence, 12(1), 79.

[11] Dessai, A. U., & Virani, H. G. (2023). Emotion Detection and Classification Using Machine Learning Techniques. In Multidisciplinary Applications of Deep Learning-Based Artificial Emotional Intelligence (pp. 11-31). IGI Global.

[12] Hung, L. P., & Alias, S. (2023). Beyond Sentiment Analysis: A Review of Recent Trends in TextBased Sentiment Analysis and Emotion Detection. Journal of Advanced Computational Intelligence and Intelligent Informatics, 27(1), 84-95.

[13] Qaqish, E., Aranki, A., & Etaiwi, W. (2023). Sentiment analysis and emotion detection of post-COVID educational Tweets: Jordan case. Social Network Analysis and Mining, 13(1), 1-11.