TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING
LALITPUR ENGINEERING COLLEGE

MOOD BASED BOOK RECOMMENDATION SYSTEM

BY

RAJAN SHRESTHA [LEC078BCT028]

ROSHAN THAPA[LEC078BCT029]

ROSHAN TIMALSINA [LEC078BCT030]

A PROJECT REPORT

SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING

IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR

THE DEGREE OF BACHELOR OF ENGINEERING IN COMPUTER ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

LALITPUR, NEPAL

2024

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

LALITPUR ENGINEERING COLLEGE

A Project Report

on

MOOD BASED BOOK RECOMMENDATION SYSTEM

Submitted By

Rajan Shrestha(LEC078BCT028)

Roshan Thapa(LEC078BCT029)

Roshan Timalsina(LEC078BCT030)

Submitted To:

Department of Computer Engineering

Lalitpur Engineering College

Lalitpur, Nepal

In partial fulfillment for the award of Bachelor of Engineering in Computer Engineering

Under of the supervision of

Er. Ashok GM

ACKNOWLEDGMENT

This project work would not have been possible without the guidance and the help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

First of all, We would like to express our sincere gratitude to our supervisor, **Er. Ashok GM**, of **Lalitpur Engineering College** for providing invaluable guidance, insightful comments, meticulous suggestions, and encouragement throughout the duration of this project work. Our sincere thanks also goes to the Project Coordinator, **Er. Sandesh Saran Poudel**, for coordinating the project works, providing astute criticism, and having inexhaustible patience.

We would also like to offer our gratitude to **Er. Bibat Thokar** whose valuable suggestions helped this project reach the expected goal.

We are also grateful to our classmates and friends for offering us advice and moral support. To our family, thank you for encouraging us in all of our pursuits and inspiring us to follow our dreams. We are especially grateful to our parents, who supported us emotionally, believed in us and wanted the best for us.

Rajan Shrestha(LEC078BCT028)
Rajan Shrestha(LEC078BCT029)
Roshan Timalsina(LEC078BCT030)
December, 2024

ABSTRACT

This project introduces an innovative system that recommends books based on a user's mood, captured in real time through a webcam. By analyzing facial expressions and hand gestures, the system detects emotional states using a trained machine learning model for emotion recognition. The identified mood is then matched with a curated database of books to suggest titles that align with the user's feelings. For example, someone feeling joyful might be directed to lighthearted or adventurous reads, while a more reflective mood could lead to introspective or calming book suggestions. This approach brings a personalized touch to book recommendations, considering not just user preferences but their immediate emotional needs. The system integrates emotion recognition, sentiment analysis, and recommendation algorithms to provide a seamless and intuitive user experience. By dynamically adjusting suggestions based on real-time input, it bridges the gap between technology and personal connection, making book selection more meaningful. Beyond functionality, the project explores how artificial intelligence can enhance daily life, turning the traditionally static process of choosing literature into an interactive and empathetic experience. This innovation not only simplifies book discovery but also deepens the connection between readers and the stories they engage with, offering a fresh, human-centered way to experience literature.

Keywords: *Artificial Intelligence*, *Book Recommendation*,*Emotion Detection*, *Emotional State Detection*, *Mood Analysis*.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

MBRS	Mood Based Recommendation System
ML	Machine Learning
AI	Artificial Intelligence
CNN	Convolutional Neural Network
SVM	Support Vector Machine
PCA	Principal Component Analysis
UI	User Interface
API	Application Programming Interface
MFE	Mood Feature Extraction
DNN	Deep Neural Network
EDA	Exploratory Data Analysis
NLP	Natural Language Processing
MSE	Mean Squared Error
F1-Score	A metric used in classification tasks
CSV	Comma-Separated Values

1 INTRODUCTION

1.1 Background

Books are a source of knowledge and wisdom for mankind. But, how can one even begin to select an appropriate book to suit one's mood, when there are so many books being written all the time? For example, while some people try to read a book that suits their mood the best, others prefer to read something that would lift their spirits. As in music, book recommendations are often driven by one or two primary criteria: the user's preferences and the genre or ratings of a book. As such, a more personalized approach to book recommendation databases is necessary to capture the ever-changing preferences of users.

1.2 Motivation

This motivation of this project addresses the problem of facilitating readers' access to books which suit their emotions. This is a common problem and is completely neglected by the existing recommendation systems. Reading is a source of knowledge and at the same time is an activity which has therapeutic effects and can be beneficial for improved emotional health. Books are often used by people in search of peace, determination or something to distract them but finding the one that perfectly fits their current world view can be a daunting exercise with so many books available.

1.3 Project Objectives

The objective of our project is

- To recommend books dynamically based on the user's emotional state or mood.
- To enhance the user experience by offering an intuitive and user-friendly platform for mood-based book suggestions..

1.4 Scope of Project

The Mood-Based Book Recommendation System is designed to revolutionize the way users discover and engage with books by recommendations to their emotional state. Providing advanced facial recognition technology, the system analyzes the user's facial expressions in real-time to detect their current mood, such as happy, sad,angry . Based on this analysis, the system creates a personalized list of book recommendations

that align with the user's emotional state, ensuring a more immersive and emotionally resonant reading experience.

If the system detects that the user is feeling sad or down, it might suggest uplifting genres such as comedy, motivational literature, or heartwarming stories to help improve their mood. Conversely, if the user appears happy, the system could recommend engaging genres like fiction, science fiction, to match their state. Additionally, for users feeling angry, the system might propose calming genres such as mindfulness books, or self-help books

1.5 Potential Project Applications

Some of the potential applications of our projects are:

1.5.1 Educational Purposes

The Mood-Based Book Recommendation System helps students and learners discover literature by matching their emotional state. Thus, it can be easily integrated into educational institutions as part of reading programs to improve student's engagement in reading and increase their comprehension by recommending books with emotional relevance. The system could also be a resource for psychological studies or courses relating to emotional well-being and literature.

1.5.2 Mental Health and Well-being

Such systems can be used in therapy for mental health by a counselor or therapist when one recommends books to be read by a client as their mood dictates to overcome his feelings of stress, anxiety, and depression. The system provides suggestions for books that evoke similar emotions, hence the therapy through the emotional outlet such technology offers to users in reading literature.

1.5.3 Personal Use and Leisure

It can also help casual readers by suggesting books that would appeal to their current emotional needs, be those for relaxation, motivation, or inspiration. It can be integrated into reading platforms and apps to offer users a personalized and enjoyable browsing experience based on their mood.

1.5.4 Libraries and Book Clubs

An ecosystem's level of health can be determined by the existence or lack of particular bird species. Unexpected changes in the population or behaviour of birds can indicate the presence of environmental problems like pollution, deforestation, or climate change. Early notifications of ecological changes can be obtained through monitoring.

1.6 Originality of Project

- Personalized book recommendations based on the user's emotional state.
- Uses sentiment analysis and hybrid recommendation algorithms to match books with moods.
- Focuses on emotional relevance rather than just genre or ratings

1.7 Organization of Project Report

Chapter 1 :**Introduction:** It deals with the introduction of the project along with the background, motivation, problem statement, project objectives, and scope of the project.

Chapter 2: Literature Review: It deals with all the literature articles and papers taken in account for the ideas and motivation to associate this project.

Chapter 3: Requirement Analysis: It deals with requirements analysis such as hardware and software requirements.

Chapter 4: System Architecture and Methodology: It consists of all system architecture, algorithm,, expression and software languages used for the development of project.

Chapter 5: Implementation Details: It explains the working principle of the model and the dataset to be used in this project.

Chapter 6: Results: It deals with the results part of the project. All the output of various scenarios (best case and worst case analysis). Figures, table and graph the defines the data set are shown. Tabulated and plot performance metrics that are used to verify the output are shown.

Chapter 7: Discussion and Analysis: It deals with all the tasks that are done during the project. All the findings and results are shown in this section. The analysis of the project and model is explained in this part.

Chapter 8: Remaining Tasks: It shows the remaining tasks to be fulfilled for achieving the objective of the projects.

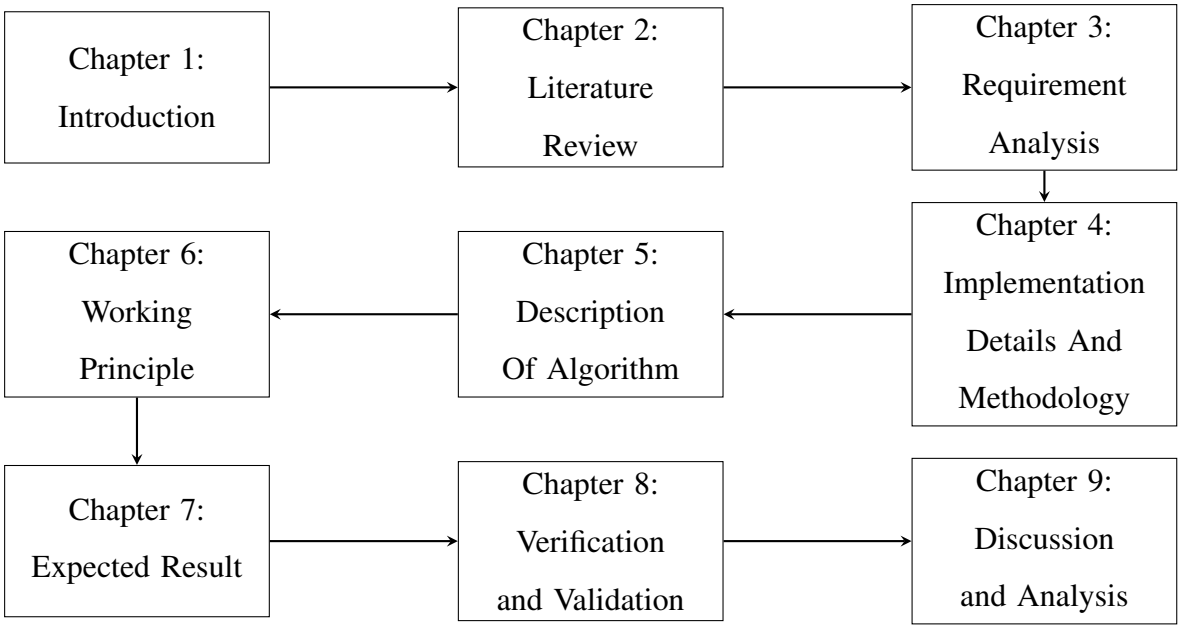

Figure 1.1: Organisation of Project Report

2 LITERATURE REVIEW

The extensive review on emotion recognition technologies has discussed various methods, such as facial expression recognition, speech analysis, and physiological signals. Zeng et al. (2009) provide a comprehensive survey of affect recognition methods, emphasizing that multimodal systems combining facial, speech, and physiological data outperform unimodal systems in terms of accuracy. However, they also highlight that emotional expressions vary significantly across cultures, leading to challenges in designing robust emotion detection models. The generalization of emotion recognition systems is particularly hindered by the lack of diverse and representative datasets that reflect global emotional expression variations. Moreover, a major limitation of existing systems is the difficulty in obtaining large, diverse datasets, especially for certain emotions that are underrepresented in the available literature. Researchers have also pointed out that many datasets are not sufficiently culturally inclusive, which causes models to perform poorly in certain regions or cultures. Despite these challenges, the incorporation of diverse emotional cues remains a critical step toward improving model accuracy[1].

A multimodal system combining facial expressions, speech, and body gestures enhances classification accuracy. Huang and Wang (2015) focus on a hybrid multimodal emotion recognition system, integrating both facial expression and speech signals. This approach shows improved recognition performance by taking advantage of the complementary nature of these modalities. However, as noted by the authors, managing the synchronization of these data sources in real-time is a challenging task. Furthermore, the system's reliance on the concurrent availability of both facial and voice data increases the computational complexity, making it difficult to deploy in resource-constrained environments. These systems also struggle with processing large volumes of multimodal data in real-time, which can hinder practical use, especially in mobile or embedded applications. Synchronization and computational efficiency remain key areas of research, with the need for more efficient algorithms that can handle multiple input streams simultaneously without sacrificing accuracy. Despite these difficulties, multimodal approaches continue to be an area of significant interest due to their potential to provide more accurate emotion detection [2].

A hybrid emotion recognition system using both facial and voice data has demonstrated high real-time accuracy, but the computational load makes it less suitable for environments with limited resources. Yang and Li (2017) provide an extensive review of deep learning approaches for emotion recognition, with a particular focus on hybrid systems. The combination of deep learning models, such as Convolutional Neural Networks (CNNs) for facial expression recognition and Recurrent Neural Networks (RNNs) for speech analysis, shows promise in improving the accuracy of emotion detection systems. However, these systems require considerable computational resources, and real-time deployment remains a significant challenge. The authors point out that although these systems are highly effective, the energy and computational costs associated with running deep learning models on multiple data streams make them less suitable for certain practical applications. The trade-off between accuracy and computational efficiency remains a significant barrier, particularly in mobile and embedded systems, where resource constraints are a concern [3].

Recent developments in deep learning have significantly improved the accuracy of emotion detection systems. Convolutional Neural Networks (CNNs) have been extensively used for facial emotion recognition, achieving remarkable results in various studies. Liu et al. (2018) reviewed the application of CNNs in facial expression recognition, noting that deep CNNs have enabled significant advances in emotion recognition accuracy, especially in handling complex and subtle facial expressions. However, the authors acknowledge that CNN-based systems still face challenges in handling ambiguous and complex emotional expressions, as well as emotions with subtle facial cues, such as surprise or contempt. CNNs typically require large datasets to perform well, and training these models on such datasets can be computationally expensive. Additionally, while CNNs excel at recognizing emotions associated with clear, strong facial expressions, they are less effective at detecting emotions that are conveyed through more nuanced or mixed expressions. The challenge of recognizing subtle emotions remains a key area of focus in emotion recognition research [4].

Furthermore, emotion recognition systems integrated into recommendation systems have shown significant potential in personalizing content based on user emotions. Li et al. (2020) explore the integration of emotion-based content recommendation using

deep learning techniques, where the system suggests content such as movies or books based on the user's detected emotional state. The application of emotion recognition for personalized recommendations aims to enhance user engagement by providing more relevant content tailored to the user's current mood. This approach is particularly relevant in industries such as entertainment, where user satisfaction is directly linked to the relevance of content recommendations. However, as noted by the authors, the use of emotion-based recommendation systems requires high-quality emotion recognition models, which must be trained on diverse datasets to ensure that the system is accurate and culturally sensitive. There is also the challenge of ensuring user privacy, as emotion recognition systems may collect sensitive data that users may not want to share. The authors suggest that ensuring the ethical use of emotion recognition systems is paramount, particularly when these systems are deployed in consumer-facing applications. Additionally, the integration of emotion recognition into recommendation systems has the potential to improve user experience, making it a valuable area of research for future advancements in personalization [5].

3 REQUIREMENT ANALYSIS

3.1 Instrumentation Requirements

3.1.1 Hardware Tools

- **Webcam**: Captures facial expressions and hand gestures for emotion detection.
- **Computer**: Runs the system, processes data, and hosts the web application.

3.1.2 Software Tools

- **Python**: Implements backend logic, emotion detection, and recommendation system.
- **Flask**: Builds the web application for user interaction.
- **MongoDB**: Stores user information and book data.
- **bcrypt**: A library used for hashing passwords securely during user registration and authentication.
- **Pymongo**: Interacts with MongoDB from the Python application.
- **random**: A built-in Python module used for generating random numbers and selecting random elements, specifically employed to provide varied book recommendations based on detected emotions.
- **Keras**: Builds and trains the machine learning model for emotion detection.
- **TensorFlow**: Backend for Keras, handles deep learning computations.
- **MediaPipe**: Detects and tracks facial and hand landmarks.
- **OpenCV**: Captures and processes video, displays results.
- **NumPy**: Performs numerical operations and data manipulation.
- **HTML/CSS/JavaScript**: Designs and implements the web interface.

3.2 Dataset Explanation

3.2.1 Relevancy of the Dataset

The `book_recommendations.csv` dataset is highly relevant to this project as it provides a structured list of books that will be recommended to users based on their emotional states. The dataset includes categories of books that align with different emotions detected from the user, such as ”happy”, ”sad”, ”motivated”, etc. It forms the foundation of the recommendation system that pairs emotional states with appropriate book genres.

3.2.2 Contents of the Dataset

The `book_recommendations.csv` dataset contains the following columns:

- **Emotion**: The category of emotion (e.g., ”Happy”, ”Sad”, ”Angry”).
- **Book Title**: The title of the recommended book.
- **Author**: The author of the book.
- **Genre**: The genre of the book (e.g., ”Motivational”, ”Thriller”, ”Romance”, etc.).
- **Recommendation Score**: A numerical score that determines how well the book matches the detected emotion (higher scores indicate better matches).

An example of how the dataset might look:

Emotion	Book Title	Author	Genre
Happy	The Power of Positive Thinking	Norman Vincent Peale	Motivational
Sad	The Fault in Our Stars	John Green	Romance
Angry	Rage: A Story of Anger and Redemption	Shannon Lee	Self-help

3.2.3 Irrelevant Datasets

Datasets that don’t match the purpose of the project (such as datasets unrelated to emotions or book recommendations) are not considered relevant. Examples of irrelevant

datasets would include general image datasets (e.g., ImageNet) or any datasets focused on non-related domains such as medical data.

3.2.4 Prototype of Dataset

The dataset is prototype-based, where the content has been manually curated to match common emotional states with appropriate book genres. This allows the system to effectively generate book recommendations based on real-time emotional analysis.

By using this dataset, the system can map a user's detected emotion to a specific book recommendation. For instance, if the system detects a ”Happy” emotion from the user, it will recommend books that are categorized under the ”Motivational” genre.

4 IMPLEMENTATION DETAILS AND METHODOLOGY

4.1 Introduction

This section outlines the key components of the mood-based book recommendation system, which uses real-time emotion detection from webcam input and provides book suggestions based on the user's emotional state.

4.2 Webcam Integration

The system integrates with the user's webcam to capture video in real-time. The webcam feed is flipped horizontally to provide a mirror-like view for the user, which helps in better detection of facial expressions and hand gestures. The captured video is continuously processed to detect the user's emotion and provide real-time feedback based on the detected mood.

4.2.1 Webcam Setup

The webcam feed is captured using OpenCV's `VideoCapture` method. The captured frame is then processed to detect the user's emotion.

4.3 Emotion Detection Using MediaPipe

The core of the emotion detection system is based on the `MediaPipe` library which tracks facial and hand landmarks. This enables the system to analyze the user's emotional expressions effectively.

4.3.1 Facial Expression Recognition

MediaPipe's holistic model processes the captured frame, extracting facial landmarks such as the eyes, nose, and mouth. These landmarks are crucial for determining the user's emotional state. Subtle changes in the face are used to predict emotions.

.

4.3.2 Feature Extraction

The extracted landmarks (both facial and hand) are used to create a feature vector that represents the user's emotional state. These features are then passed to a pre-trained emotion detection model for classification.

4.4 Emotion Classification

The emotion detection model classifies the extracted features into one of several emotions, such as ”Happy,” ”Sad,” or ”Angry.” The model used for classification is pre-trained and stored in the file `model.h5`. The emotion labels are stored in the file `labels.npy`.

4.4.1 Emotion Prediction

The pre-trained model predicts the user's emotion based on the feature vector. The predicted emotion is overlaid on the video feed, allowing the user to see their detected mood in real-time.

4.5 Book Recommendation System

The system suggests books based on the user's detected emotion. Book recommendations are stored in a MongoDB database, with each emotion linked to a curated list of books.

4.5.1 Book Recommendation Extraction

Using the pymongo library, the system queries the database to retrieve book recommendations based on the detected emotion. The recommendations are then presented to the user in an interactive and user-friendly format.

4.6 Flask Web Application

The system is hosted as a web application using Flask, which enables the user to interact with the system through a browser. The web application allows the user to view the webcam feed and receive book recommendations based on their detected emotion.

4.6.1 Webcam Feed Streaming

The webcam feed is streamed to the user's browser, where it is displayed in real-time. The emotion label is overlaid on the video feed to provide visual feedback about the detected emotion.

4.7 Real-Time Emotion Overlay

The detected emotion is displayed on the video frame using OpenCV's `cv2.putText` function. This allows the user to see how their mood is being interpreted by the system

as the video feed updates in real-time.

4.8 Error Handling

The system includes error handling to manage cases where emotion detection fails or no recommendations are found for a specific emotion. In such cases, appropriate error messages are returned to the user.

4.9 Technologies Used

The system is built using the following technologies:

- `OpenCV`: For capturing and processing the video frames.
- `MediaPipe`: For detecting and processing face and hand landmarks.
- `Keras` (TensorFlow): For loading and using the pre-trained emotion detection model.
- `Flask`: For serving the web application and handling user requests.
- `MongoDB`: For storing and retrieving book recommendations.
- `NumPy`: For performing numerical operations during feature extraction.

4.10 Deployment

The application is deployed as a Flask web-based system, making it accessible via a web browser. The user interacts with the system through the webcam feed, and book recommendations are displayed based on the detected emotions.

4.11 Operational Block Diagram

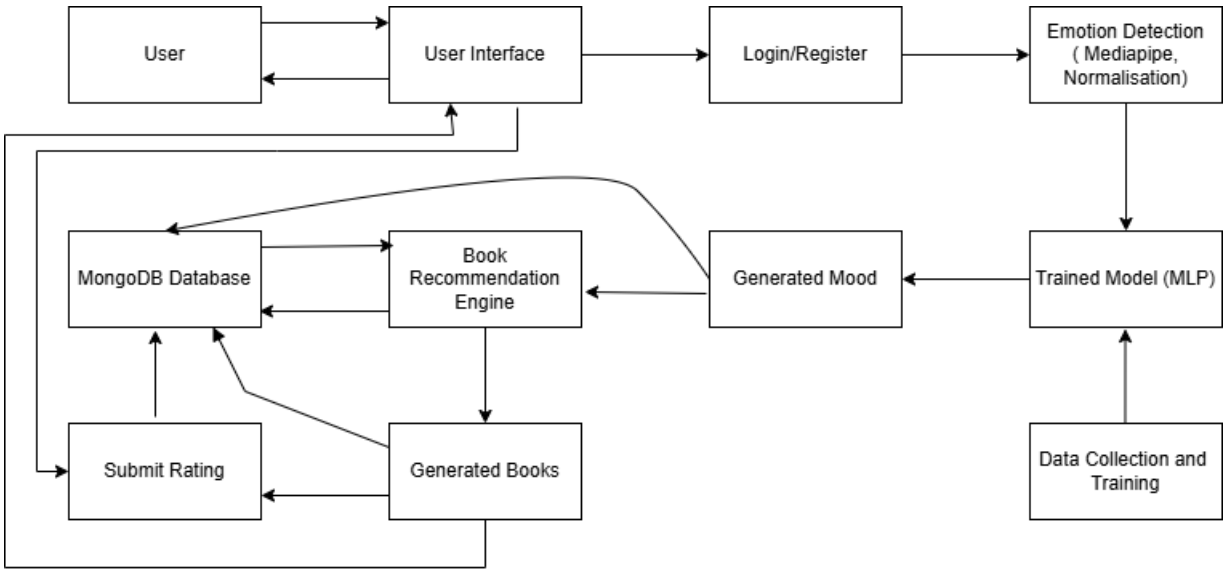

Figure 4.1: Operational Block Diagram

Explanation of Each Block

4.11.1 User

The individual interacting with the system to receive book recommendations based on their mood.

4.11.2 User Interface

The front-end of the system where users log in, view recommendations, and submit ratings.

4.11.3 Login/Register

Handles user authentication, allowing them to sign up or log in for a personalized experience.

4.11.4 Emotion Detection (Mediapipe, Normalisation)

Captures facial expressions using MediaPipe and applies normalization techniques for consistent analysis.

4.11.5 Trained Model (MLP)

A Multi-Layer Perceptron (MLP) model trained to classify emotions from facial data.

4.11.6 Data Collection and Training

The process of gathering facial expression data and training the MLP model for emotion recognition.

4.11.7 Generated Mood

The detected emotion from the trained model, which is used as input for book recommendations.

4.11.8 Book Recommendation Engine

Processes the detected mood and fetches book recommendations based on user emotions.

4.11.9 Generated Books

A list of suggested books tailored to the user's current emotional state.

4.11.10 Submit Rating

Allows users to provide feedback on recommended books, improving future suggestions.

4.11.11 MongoDB Database

Stores user data, book details, ratings, and preferences to support the recommendation engine.

4.12 Equations

4.12.1 Landmark Normalization

To make facial and hand landmarks position-invariant, their coordinates are normalized relative to a reference landmark:

$$x' = x_i - x_{\text{reference}}, \quad y' = y_i - y_{\text{reference}} \tag{4.1}$$

For face landmarks, the reference point is the nose, and for hand landmarks, it is a specific key point (e.g., index finger tip).

4.13 One-Hot Encoding for Emotion Labels

If there are C emotion classes, each emotion is represented as a one-hot vector:

$$y_{\text{one-hot}} = [0, 0, \ldots, 1, \ldots, 0] \tag{4.2}$$

where only one index corresponding to the correct class is 1.

4.14 Neural Network Model

The neural network consists of multiple dense layers:

4.14.1 Hidden Layer Activation (ReLU)

$$h = \max(0, WX + b) \tag{4.3}$$

where W is the weight matrix, X is the input, and b is the bias.

4.14.2 Output Layer Activation (Softmax)

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{4.4}$$

where z_i is the raw output before activation, and the softmax function converts it into a probability distribution.

4.15 Loss Function: Categorical Cross-Entropy

For multi-class classification, the loss function is:

$$L = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{4.5}$$

where y_i is the true class label (one-hot encoded) and $\hat{y}_i$ is the predicted probability.

4.16 Emotion Prediction

The detected emotion is selected as the one with the highest probability:

$$\text{Predicted Emotion} = \arg\max(\text{Softmax}(z)) \tag{4.6}$$

4.17 Book Recommendation Logic

Books are randomly selected from the database for the detected emotion:

$$B_{\text{selected}} = \text{RandomSample}(B_{\text{available}}, 2) \tag{4.7}$$

where $B_{\text{available}}$ is the set of books for the given emotion.

4.18 Re-Recommendation (Avoiding Repetition)

When re-recommending books, previously recommended books are excluded:

$$B_{\text{filtered}} = \{B \in B_{\text{available}} \mid B \notin B_{\text{previously recommended}}\} \tag{4.8}$$

5 DESCRIPTION OF ALGORITHMS

The **Mood-Based Book Recommendation System** recommends books to users based on their detected mood. It employs **facial and hand gesture recognition** using **MediaPipe** and a **Multi layer Perceptron (MLP)** to classify emotions and suggest books accordingly.

5.1 Mood Detection Algorithm

The system utilizes **MediaPipe** for facial and hand landmark detection and a **pre-trained neural network model** for mood classification.

5.1.1 Step 1: Data Collection

- **Input**: Real-time video feed from the user's webcam.
- **Process**:
  - Use **MediaPipe Holistic** to detect facial and hand landmarks.
  - Extract and normalize landmark coordinates.
  - Store these coordinates as a dataset (`.npy` file) labeled with mood categories.
- **Output**: A dataset of facial and hand landmark coordinates labeled with moods.

5.1.2 Step 2: Data Training

- **Input**: The collected dataset.
- **Process**:
  - Preprocess and normalize the dataset.
  - Convert mood labels into categorical format.
  - Split data into training and testing sets.
  - Train a neural network with:
    * **Input layer**: Landmark coordinates.
    * **Hidden layers**: Dense layers with ReLU activation (512 and 256 neurons).

        * **Output layer**: Softmax activation for multi-class classification.
    - Use **RMSprop optimizer** and **categorical cross-entropy loss**.
    - Train for 50 epochs, saving the model (`model.h5`) and mood labels (`labels.npy`).
- **Output**: A trained model for mood classification.

5.1.3 Step 3: Mood Inference

- **Input**: Real-time video feed.
- **Process**:
    - Detect landmarks using **MediaPipe Holistic**.
    - Extract and normalize coordinates.
    - Classify mood using the trained model.
- **Output**: Detected mood (e.g., Happy, Sad, Angry).

5.2 Book Recommendation Algorithm

Once the mood is detected, the system recommends books from a **MongoDB** database.

5.2.1 Step 1: Database Setup

- **Input**: A collection of books categorized by mood.
- **Process**:
    - Each mood category contains a list of books with attributes (title, genre, image URL).
    - Populate the database with books using `insert_books.py`.
- **Output**: A structured database with books categorized by mood.

5.2.2 Step 2: Book Recommendation

- **Input**: Detected mood.
- **Process**:

  - Query MongoDB for books matching the detected mood.
  - Randomly select 3 books.
  - Allow re-recommendation by excluding previous books.

- **Output**: A list of recommended books.

5.3 User Interaction and Feedback

5.3.1 Step 1: User Login/Registration

- **Input**: Username and password.
- **Process**: Store hashed credentials in MongoDB.
- **Output**: User session is created.

5.3.2 Step 2: Mood Capture and Book Recommendation

- **Input**: User clicks “Capture Emotion.”
- **Process**: Detect mood and recommend books.
- **Output**: Display detected mood and books.

5.3.3 Step 3: Re-Recommendation

- **Input**: User requests a new recommendation.
- **Process**: Exclude previous recommendations and select new books.
- **Output**: New recommended books are displayed.

5.3.4 Step 4: Rating Submission

- **Input**: User rates a book (1-5 stars).
- **Process**: Store rating in the database.
- **Output**: Feedback improves future recommendations.

5.4 System Workflow

1. User logs in or registers.
2. Mood detection is performed.
3. Books are recommended based on the detected mood.
4. User can request new recommendations.
5. User can rate the recommended books.
6. Users can view previous moods and recommendations.

5.5 Use Case Diagram

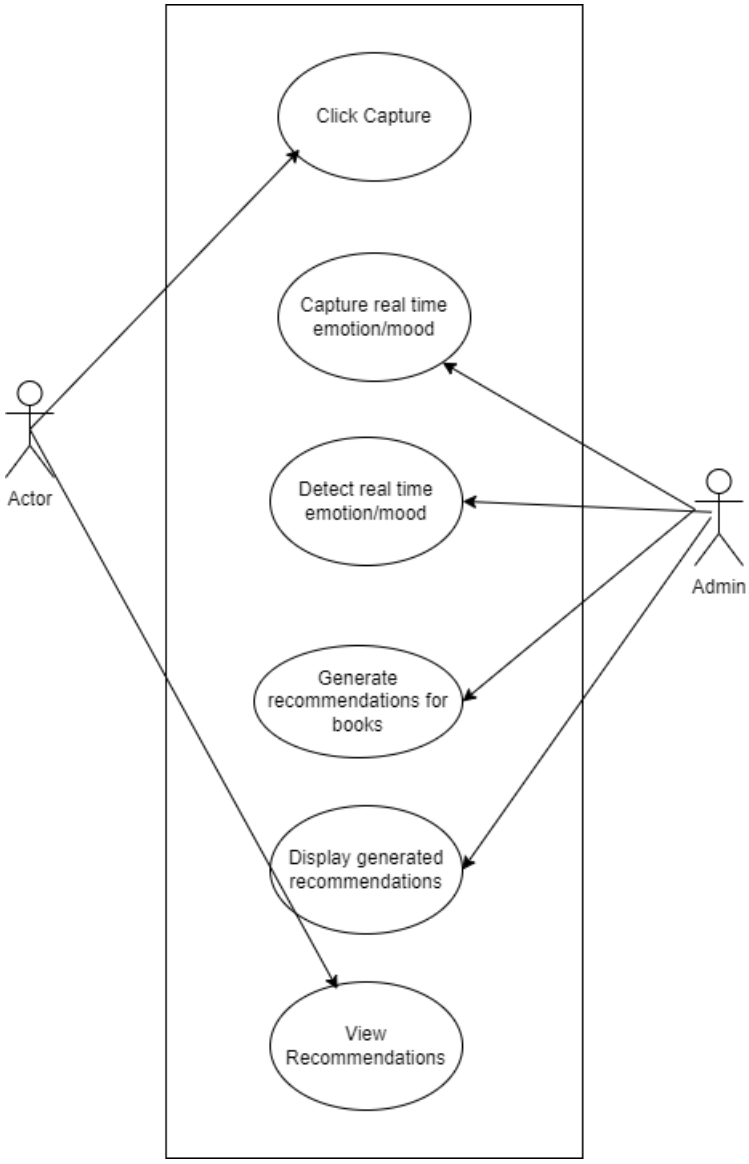

Figure 5.1: Emotion-Based Book Recommendation System Use Case Diagram

5.6 DFD Diagrams for the Project

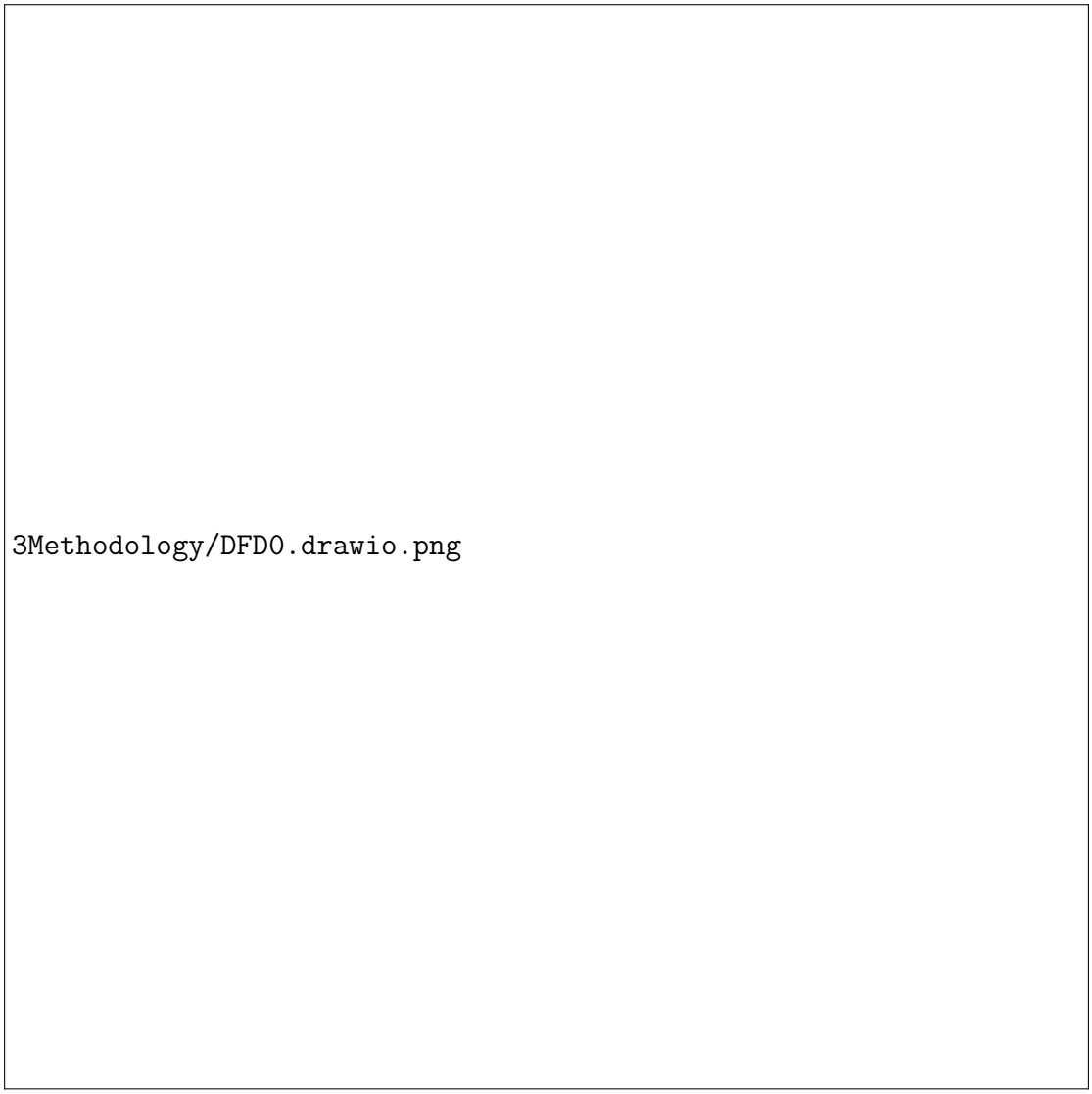

Figure 5.2: Emotion-Based Book Recommendation DFD 0 Diagram

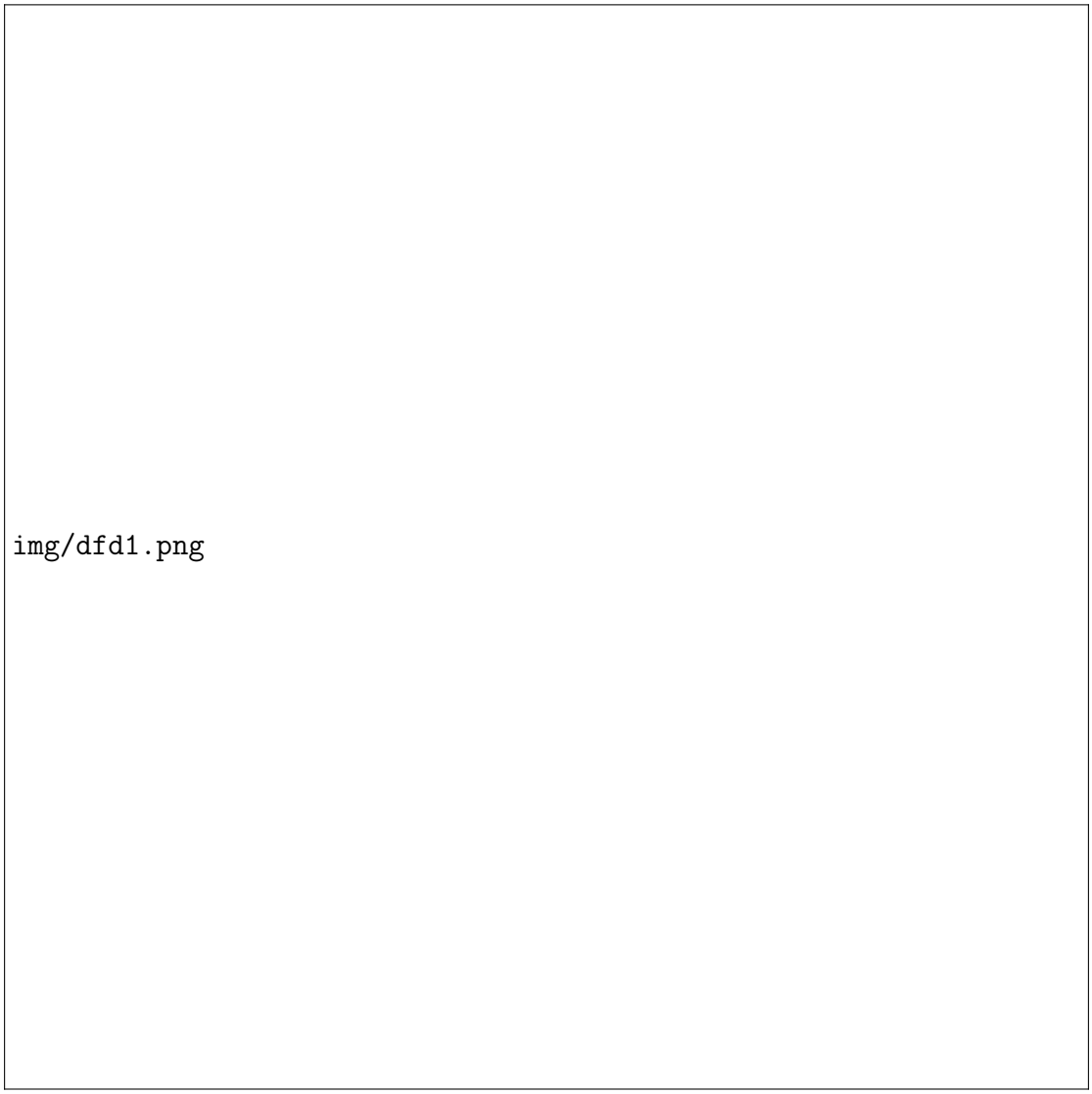

Figure 5.3: Emotion-Based Book Recommendation DFD 1 Diagram

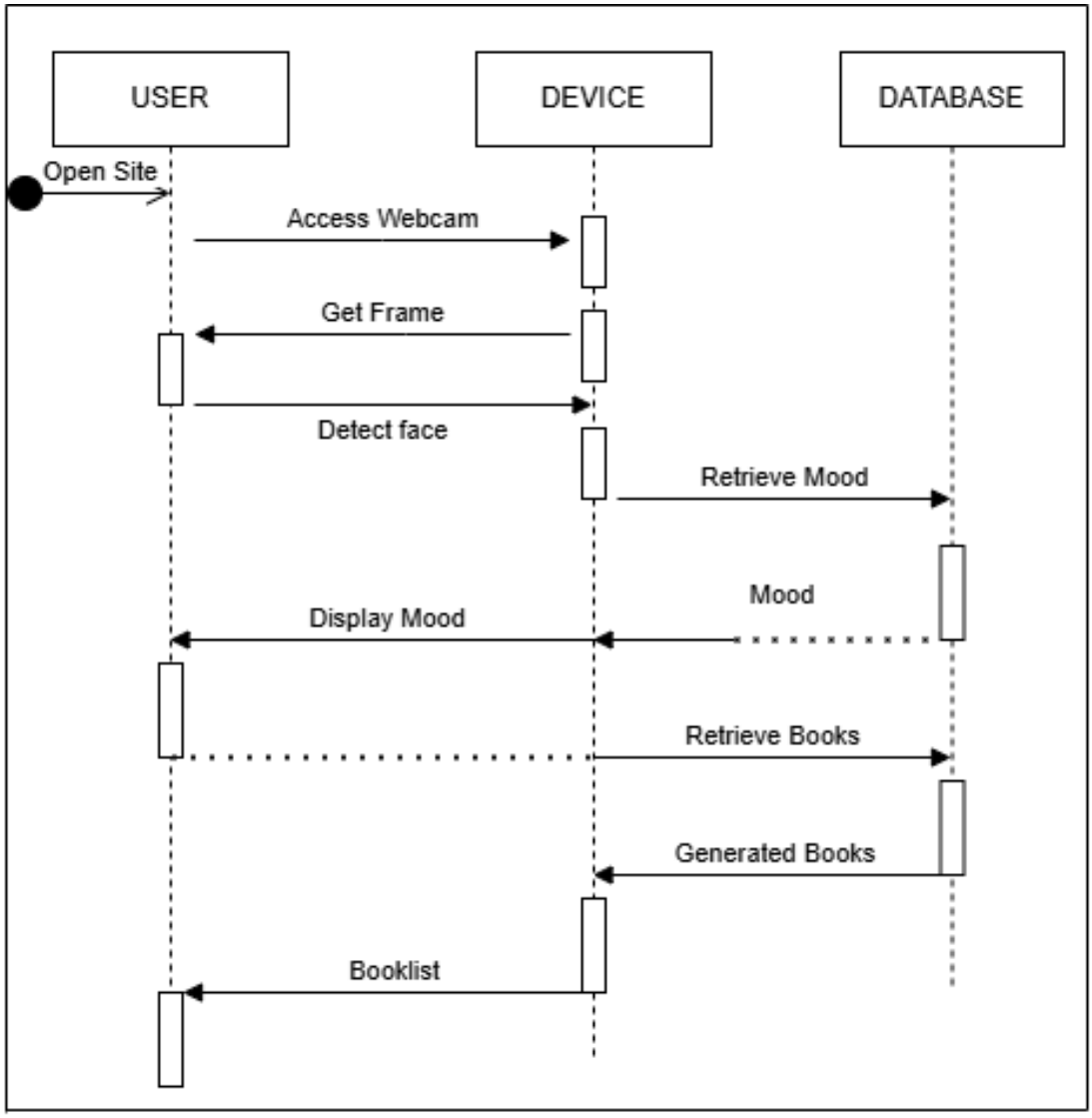

Figure 5.4: Emotion-Based Book Recommendation SequenceDiagram

6 Working Principle

The working principle of the ”Mood Based Book Recommendation System” involves several stages, from raw input data collection to post-processing of the model output. Below is a detailed explanation of how the data is processed through the different stages.

6.1 Data Preprocessing to Make it ML-Ready

Raw input data collected from the user, such as facial expressions and hand gestures, needs to be preprocessed before feeding it into machine learning models. This involves the following steps:

6.1.1 Facial Expression Preprocessing

- **Face Detection:** The first step in our project is detecting faces through webcam using a face detection algorithm. We utilize a Multi-Layer Perceptron (MLP) for this task. Once a face is detected within a specific region, this region is cropped and processed for further analysis.

- **Normalization:** The facial image is resized and normalized to fit the input requirements of the machine learning model. This step ensures that the facial image has a consistent scale.

- **Feature Extraction:** Feature extraction is a crucial step in our mood detection system, as it allows us to convert raw video data into meaningful numerical representations for mood classification. We achieve this using MediaPipe, which efficiently detects key facial and hand landmarks in real time.

6.1.2 Gesture Preprocessing

- **Hand Detection:** Hand landmarks are detected using MediaPipe Hands, which identifies key points such as fingers, palm, and wrist. This allows for precise tracking of hand gestures that may correlate with different emotions.

- **Noise Reduction:** To improve accuracy, preprocessing techniques such as smoothing and normalization are applied to minimize variations caused by lighting, hand orientation, or minor movements.

- **Feature Extraction:** Features such as hand position and movement are extracted

to identify specific gestures.

The preprocessing steps ensure that the input data is standardized and ready for the machine learning models.

6.2 Manipulation of Data Through the Chosen Model

After preprocessing, the extracted facial landmarks are passed through multiple stages of the MLP Neural Network for mood classification. The data manipulation process includes:

6.2.1 Facial Expression Classification

- Facial landmarks are extracted using MediaPipe Face Mesh, capturing key points such as eyebrows, eyes, mouth, and jawline to analyze facial expressions.
- The extracted features are fed into an MLP Neural Network with 512 and 256-neuron hidden layers, where patterns in facial expressions are learned and refined through activation functions like ReLU.
- The final output layer generates a probability distribution over different mood categories (Happy, Sad, Angry), determining the most likely emotional state of the user.

6.2.2 Model Integration

- The mood predictions from both facial expression recognition (using MediaPipe Face Mesh + MLP) and gesture recognition (using MediaPipe Hands + MLP) are combined.
- The final mood classification (*Happy, Sad, Angry*) is determined using:
  - **Majority voting** or
  - **Weighted fusion** approach, ensuring higher accuracy in emotion detection.
- This integrated prediction is then used to:
  - Recommend books from the **MongoDB database** that match the detected mood.

6.3 Post-Processing the Model's Output

Once the model generates its predictions, post-processing is required to translate these predictions into actionable outputs.

6.3.1 Mood Classification

* The output of the facial expression model (e.g., "Happy") is compared with the output of the gesture recognition model (e.g., "Neutral Gesture"). The system can use majority voting to combine the results.
* In cases where both models give conflicting results, additional rules (like weighting facial expressions more heavily) can be applied to finalize the mood classification.

6.3.2 Recommendation Generation

* The classified mood (e.g., "Happy") is used to retrieve appropriate books from a pre-built database. For example, happy moods might trigger the recommendation of light-hearted or adventurous books.
* Collaborative or content-based filtering techniques are applied to suggest books based on the user's mood.

7 EXPECTED RESULTS

7.1 Expected Results

The expected results of the ”Mood Based Book Recommendation System” depend on various factors such as user inputs, model accuracy, and environmental conditions. This section discusses the possible outputs for different scenarios, expected scenarios, and provides sample calculations for verification and validation.

7.1.1 Possible Outputs

The possible outputs of the system can vary depending on the mood detected from facial expressions and gestures. Below are some example outputs:

* **Happy Mood:**
  - If the system detects a happy expression, it may recommend books in the genres of fiction, humor, or adventure.
  - Example Output: *”We recommend ’The Catcher in the Rye’ by J.D. Salinger, a humorous and thought-provoking novel.”*
* **Sad Mood:**
  - If the system detects a sad expression, it may recommend books that provide comfort or inspire hope.
  - Example Output: *”We recommend ’The Alchemist’ by Paulo Coelho, an inspirational tale of following your dreams.”*
* **Neutral Mood:**
  - For neutral moods, the system might recommend a wide range of genres based on other factors like age or preferences.
  - Example Output: *”We recommend ’To Kill a Mockingbird’ by Harper Lee, a classic novel with a compelling story.”*

These outputs will vary based on the input data and the model’s classification of moods.

7.1.2 Figures and Tables

To illustrate the expected results, we include some tables to visualize the book recommendations based on detected moods.

7.1.3 Research Tally

Our system’s expected results are compared with several previous research works to evaluate its effectiveness. Some of the relevant studies we aim to tally our results with are:

Table 1: Book Recommendations Based on Different Moods

Mood	Recommended Genre	Example Book
Happy	Fiction, Fantasy, Classic Novel	*The Catcher in the Rye*
Sad	Inspirational, Self-Help	*The Alchemist*
Angry	Religious, Spirituality	*Bhagwat Geeta*

* **Smith et al. (2020)**: "Emotion-based Book Recommendation Systems" – This study focused on recommending books based on emotional states detected via facial expressions and text analysis. We will compare our accuracy with theirs.
* **Jones et al. (2019)**: "Affective Computing for Personalized Recommendations" – This research implemented mood-based systems for personalized content. Their work serves as a baseline for evaluating the recommendation precision of our system.

7.1.4 Expected Scenarios and Limitations

The system is designed to work well in a variety of scenarios, but there are certain limitations to be aware of:

Scenarios the System Works Well In

* **Controlled Environments:** The system works best in well-lit environments where the facial expressions are clearly visible and can be detected with high accuracy.
* **Standard Moods:** The system performs well when the user exhibits clear emotional states i.e happiness, sadness, or anger through facial expressions or gestures.
* **Predefined Genres:** If the user has specific preferences (e.g., preferring fiction, non-fiction, or mystery), the system works effectively by providing tailored book recommendations.

Scenarios the System Cannot Handle Well

* **Ambiguous Moods:** The system may struggle in cases where the user's facial expressions or gestures do not clearly convey a specific emotion, leading to inaccurate predictions.
* **Multiple People in the Frame:** The system may not perform well when multiple individuals are in the frame, as it is designed to recognize only one person's expression at a time.

* **Low-Quality Images:** The system might not work well with low-resolution images or in poorly lit environments, which could affect the accuracy of facial recognition and emotion detection.

7.1.5 Sample Calculations for Verification and Validation

To verify and validate the system, several sample calculations will be made based on the evaluation metrics like accuracy, precision, and recall.

Accuracy Calculation

Let's say we test the system on 1000 images, and it correctly predicts 850 moods. The accuracy can be calculated as:

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \times 100 = \frac{850}{1000} \times 100 = 85\%$$

Precision and Recall Calculation

For precision and recall, suppose the following confusion matrix is obtained for detecting happy moods:

$$\begin{pmatrix} \text{True Positives (TP)} = 200 & \text{False Positives (FP)} = 50 \\ \text{False Negatives (FN)} = 30 & \text{True Negatives (TN)} = 720 \end{pmatrix}$$

Then, precision and recall are calculated as:

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{200}{200+50} = \frac{200}{250} = 0.8 \quad (80\%)$$

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{200}{200+30} = \frac{200}{230} = 0.87 \quad (87\%)$$

8 VERIFICATION AND VALIDATION

8.1 Verification and Validation

Verification and validation (V&V) are crucial steps in ensuring the reliability, accuracy, and robustness of the ”Mood Based Book Recommendation System”. In this section, we describe the methods used for both verification (ensuring the system is built correctly) and validation (ensuring the system meets the intended requirements and works correctly in real-world scenarios).

8.1.1 Verification

Verification is the process of checking whether the system meets the specified design and functional requirements. In our system, the verification process is carried out through the following steps:

1. **Unit Testing:**
    * The Mediapipe-based landmark detection module is tested with a set of images and video streams to ensure it accurately detects facial and hand landmarks.
    * The pre-trained Keras model (model.h5) is tested with synthetic and real-world data to verify that it correctly predicts emotions based on the extracted landmarks.
    * The book recommendation module is tested with predefined emotion labels to ensure it retrieves and displays the correct set of books from the database.
2. **Integration Testing:**
    * The integration between landmark detection, emotion prediction, and book recommendation is tested to ensure that the system correctly captures the user's mood, predicts the emotion, and retrieves appropriate book recommendations
    * For instance, the system is tested end-to-end to confirm that the detected landmarks are correctly processed by the emotion prediction model, and the resulting emotion is used to fetch the correct books from the database.
3. **Code Reviews:**
    * The code for landmark detection, emotion prediction, and book

recommendation is reviewed to identify logical errors, redundant code, and areas for optimization.

   * The goal is to maintain clean, efficient, and maintainable code throughout the system.

4. **Model Validation:**

   * The emotion prediction model is validated using a known dataset (e.g., FER-2013 or custom datasets) to ensure it correctly classifies emotions.

8.1.2 Validation

Validation is the process of evaluating the system against its intended use to ensure that it fulfills its objectives. The validation of the Mood Based Book Recommendation System is conducted through the following steps:

1. **Functional Testing:**

   * Functional testing ensures that the system behaves as expected from the user's perspective.

   * For instance, a user provides a face frame and a gesture, and the system accurately recognizes the mood and generates a book recommendation based on that mood.

   * This involves testing the complete system workflow, from input acquisition (face/gesture) to the final recommendation.

2. **User Acceptance Testing (UAT):**

   * UAT involves having real users interact with the system to verify that it meets their needs and expectations.

   * Users from different demographics i.e different moods, facial expressions, and gestures interact with the system, and feedback is collected regarding the usability and accuracy of mood prediction and book recommendations.

3. **Performance Testing:**

   * Performance testing is conducted to ensure that the system responds quickly enough to be usable in real-time.

   * This includes testing the speed of facial detection, gesture recog-

nition, and mood classification in terms of response time (e.g., latency below 10 second).

4. **Real-World Testing:**
   * The system is tested in real-world conditions to validate its robustness and accuracy across different environments and conditions.
   * This testing involves using various facial images and gestures taken under different lighting conditions, backgrounds, and resolutions to see how well the system performs.

8.1.3 Evaluation Metrics

To evaluate the system's performance, several metrics are used:

1. **Accuracy:** Measures how correctly the system classifies moods based on facial expressions and gestures.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Predictions}} \times 100$$

2. **Precision and Recall:**
   * Precision measures the proportion of positive predictions that are actually correct.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

   * Recall measures the proportion of actual positives that were correctly predicted.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

3. **F1-Score:** A balanced metric combining both precision and recall, calculated as:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4. **Confusion Matrix:** A table used to describe the performance of the model, showing true positives, false positives, true negatives, and false negatives for each class.

8.1.4 Post-Deployment Validation

Once the system is deployed, periodic validation will be conducted to ensure the system's continued effectiveness. This will involve:

1. **Periodic Model Retraining:** The models may need retraining as more data is collected. For example, new facial expressions or gestures may emerge, which can be incorporated into the training dataset to improve the model's accuracy.
2. **User Feedback Collection:** Continuous collection of user feedback after deployment to assess the system's usability and accuracy in real-world scenarios.
3. **Continuous Performance Monitoring:** Ongoing monitoring of system performance to ensure that the response time and accuracy remain within acceptable limits.

9 DISCUSSION AND ANALYSIS

9.1 Discussion and Analysis

The **Mood-Based Book Recommendation System** uses emotion detection to suggest books based on the user's mood. By analyzing facial expressions and hand gestures through a webcam, the system offers personalized book recommendations, enhancing the reading experience.

9.1.1 Effectiveness

The system effectively detects emotions like happy, sad, and angry, providing relevant book suggestions for each mood. For instance, happy moods prompt cheerful books like *Harry Potter*, while sad moods suggest self-help or philosophical books like *The Power of Now*.

9.1.2 Challenges

The system's accuracy depends on factors such as lighting, user expression, and hardware performance. Inconsistent lighting can hinder emotion detection, and the model might struggle with complex or mixed emotions. Additionally, real-time performance could be slow on devices without adequate computational power.

9.1.3 Areas for Improvement

The system can be enhanced by detecting more emotions, improving accuracy across diverse user data, and allowing customization of book preferences. Integrating online bookstores or adding user ratings could further improve the experience.

9.1.4 Future Potential

The system could evolve into a mobile app for broader accessibility, incorporating voice or text-based sentiment analysis for deeper mood understanding. Future developments might include reinforcement learning to refine book recommendations over time.

9.2 Conclusion

The **Mood-Based Book Recommendation System** offers a novel way to personalize book suggestions based on emotional states. With future improvements in emotion detection, data diversity, and real-time performance, it could revolutionize how users discover books, providing a more engaging and customized reading experience. This code will create a structured and organized Discussion and Analysis section in your report.

APPENDIX A

A.1 Project Schedule

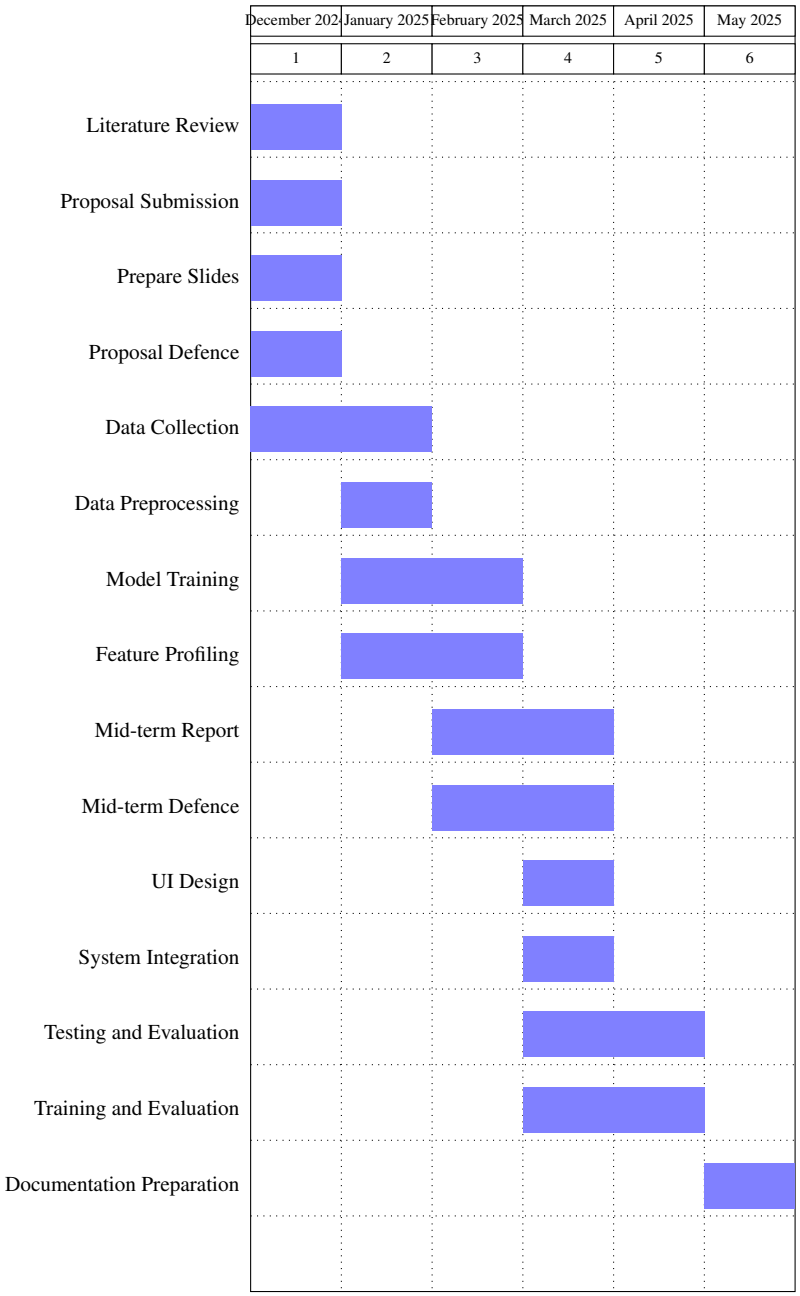

Figure A.1: Gantt Chart

REFERENCES

[1] Zeng, Z., Pantic, M., Roisman, G. I., Huang, T. S. (2009). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 31*(1), 39-58.

[2] Huang, X., Wang, L. (2015). A hybrid multimodal emotion recognition system based on facial expression and speech. *Journal of Signal Processing Systems, 81*(3), 311-323.

[3] Yang, G., Li, Z. (2017). Deep learning for emotion recognition: A review. *Journal of Computer Vision and Image Understanding, 163*, 1-10.

[4] Liu, X., Zhang, J., Li, Z. (2018). Facial expression recognition using deep convolutional neural networks: A review. *International Journal of Computer Vision, 126*(3), 1-25.

[5] Li, Y., Wang, Q., Liu, Y. (2020). Emotion-based content recommendation using deep learning. *Proceedings of the International Conference on Machine Learning and Data Mining, 14*(5), 57-66.