



**Name: Muhammad Sudais Khalid**

**Reg.No: BSAI-23F-0050**

**Program: BS Artificial Intelligence**

**Semester: 04**

**Subject: Programming for AI**

**Project: Speech Emotion Analysis for Workplace Wellness**

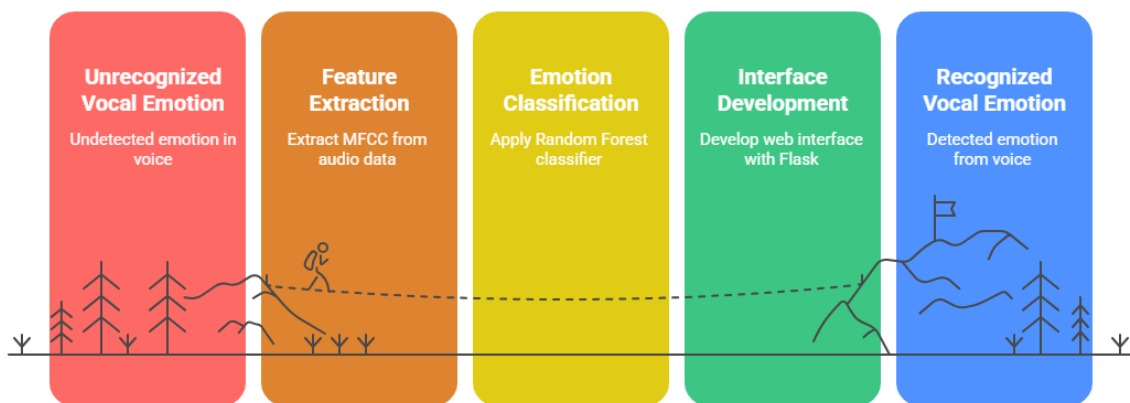
**Date: 26/06/2025**

**Submitted to: Sir Naveed**

## Abstract

In modern workplaces, emotional well-being significantly influences employee productivity, collaboration, and overall mental health. This project proposes a speech emotion recognition (SER) system that uses machine learning to detect and classify human emotions from voice recordings. By extracting Mel Frequency Cepstral Coefficients (MFCC) features from audio data and applying the Random Forest classifier, this system aims to accurately categorize emotions such as happy, sad, angry, or neutral. The trained model can be used for HR wellness assessments, mental health monitoring, or AI-assisted communication platforms. A simple web interface will be developed using HTML, CSS, and Flask to interact with the backend.

Transforming Speech to Recognized Emotion



## **Introduction**

Emotions play a vital role in communication, and speech is one of the most natural means to express emotions. Recognizing emotions from voice can be extremely useful in building empathetic and responsive systems. In the workplace, stress and emotional fatigue often go unnoticed, which can affect productivity and morale. This project addresses this challenge by creating a machine learning-based system that classifies emotions from speech input using Random Forest. The project also includes a basic web interface for user interaction, built with HTML, CSS, and Flask.

## **Problem Statement**

Existing workplace wellness tools rely heavily on surveys and manual inputs, which may not accurately reflect a person's emotional state. There is a need for an automated and intelligent system that can assess emotional well-being through natural interactions such as voice. This project aims to explore whether Random Forest can be effectively used to detect emotions from audio inputs for workplace wellness applications.

## **Objectives**

- Extract MFCC features from speech audio files using Librosa.
- Train and evaluate a Random Forest model for emotion classification.
- Build a user-friendly interface using HTML, CSS, and Flask to allow users to upload audio and view results.
- Provide analytics on prediction performance using accuracy, precision, recall, and F1-score.
- Demonstrate how such a system can be useful in monitoring workplace emotions.

## **Methodology**

1. Data Collection: Use a labelled emotional speech dataset in WAV format.
2. Preprocessing: Convert audio files into numerical features using Librosa.
3. Feature Extraction: Extract MFCC features from each audio file and normalize them.
4. Model Training: Train a Random Forest classifier using Scikit-learn.
5. Evaluation: Validate the model using performance metrics such as accuracy, precision, and F1-score.
6. Frontend Development: Design a basic interface using HTML and CSS.
7. Backend Integration: Use Flask to connect the web interface with the trained machine learning model.

## **Tools and Technologies**

- Python for programming
- Google Colab for model training and experimentation
- Librosa for feature extraction from audio
- Scikit-learn for Random Forest model and evaluation
- HTML and CSS for frontend design

- Flask as the web framework for backend integration
- NumPy and Pandas for data processing and handling

### Expected Outcomes

- A working emotion recognition model trained on speech data
- A web-based interface where users can upload audio and receive predicted emotions
- A comparison of classification performance using Random Forest
- A demonstration of how speech emotion detection can enhance workplace wellness programs

### Project Timeline

Week	Task
Week 1	Dataset collection, MFCC feature extraction, and Random Forest training
Week 2	Model evaluation, accuracy analysis, and frontend design using HTML and CSS
Week 3	Flask backend integration, testing, and final documentation preparation

### References

- Librosa: Audio and music analysis in Python
- Scikit-learn: Machine learning in Python
- RAVDESS Dataset and other open-source emotional speech datasets
- Flask Documentation
- Python Official Documentation