Project Title: Speech Emotion Detection

Project Description

This project focuses on developing a machine learning system capable of detecting emotions from human speech. Using a publicly available speech emotion dataset from Kaggle or any other data Sources, the system will classify emotions such as **happy, sad, angry, neutral, surprised, or scared**.

Avoiding any code plagiarism. **As an optional bonus**, the system can be extended to identify the speaker's name using voice features.

The project is designed for **teams of 3 students**, promoting collaboration and division of tasks.

Objectives

1. **Use a publicly available Kaggle dataset** containing audio files labeled with emotions.
2. **Extract audio features** such as MFCCs, pitch, energy, and spectral features.
3. **Develop machine learning or deep learning models** to classify emotions.
4. **Evaluate the system's performance** using accuracy, F1-score, and confusion matrix.
5. **Build a simple demo interface** for live or recorded audio input.
6. **Optional bonus**: Implement a speaker identification module to recognize the speaker.

Methodology

Dataset Preparation: Download a Kaggle dataset such as RAVDESS or TESS. Explore and preprocess audio files, normalizing volume and trimming silence.

Feature Extraction: Use Python (Librosa) or other libraries to extract MFCCs, chroma, pitch, energy, and other relevant audio features.

Model Training:

Emotion Detection: Train ML or DL models like CNN, RNN or LSTM.

Model Evaluation: Measure emotion detection accuracy, F1-score, and confusion matrix. For speaker detection, use top-n accuracy.

Deployment (Optional): Create a small web or desktop interface for real-time emotion recognition.

Expected Outcomes

A fully functional emotion detection system using a Kaggle dataset.

Audio preprocessing, feature extraction, and machine learning.

An interactive demo for testing emotion detection in real time or with pre-recorded audio.

Deliverables

Dataset: Original or Kaggle dataset with proper preprocessing applied.

Working Code: Complete, independently written code for preprocessing, feature extraction, model training, and evaluation.

User Interface: Web or desktop interface for real-time or offline emotion detection.

Presentation: Clear presentation summarizing objectives, methodology, results, and demo.

Team Size

3 students per team.

Key Notes

No code cheating: All preprocessing, feature extraction, and model implementation must be your own work.

Dataset can be taken from Kaggle or created by the team, but all code must be original.

Bonus: speaker recognition if time allows.