Human Emotion Detection from Voice

Introduction

Speech Emotion Recognition (SER), or human emotion detection from voice, is the task of understanding a person's emotional state based purely on how they speak. It goes beyond the words being said and focuses on vocal cues like tone, pitch, pace, and intensity. By analyzing these features using machine learning and signal processing techniques, we can categorize speech into different emotions such as happiness, sadness, anger, or fear. Understanding emotions through speech plays a key role in areas like virtual assistants, healthcare, and human-computer interaction. This project explores how machine learning techniques can classify emotions based on audio features extracted from voice recordings.

Abstract

The goal of this project was to detect human emotions from speech using audio files from the RAVDESS dataset. Here, I used audio processing techniques to extract meaningful features like MFCCs, Chroma, and Mel spectrograms. A Random Forest classifier was trained to recognize emotions such as happy, sad, angry, and calm. An interactive Streamlit web app was built to allow users to upload audio files and receive real-time emotion predictions.

Tools Used

- Python.
- Libraries: Librosa, Pandas, NumPy, Scikit-learn, Sounddevice, Joblib.
- RAVDESS dataset.
- Streamlit for building the user interface.
- Visual Studio Code as the IDE.

Steps Involved

I. *Dataset Preparation*: Organized the RAVDESS dataset by emotion categories.
II. *Feature Extraction*: Used librosa to extract MFCCs, chroma, and Mel features from audio files.
III. *Data Splitting*: Divided data into training and testing sets for evaluation.
IV. *Model Training*: Trained a Random Forest model on the extracted features.
V. *Testing*: Evaluated model accuracy and plotted the confusion matrix.
VI. *UI Deployment*: Developed a simple Streamlit app for real-time emotion detection from user-uploaded .wav files.

Conclusion

The project successfully demonstrates that emotional states can be identified from voice using machine learning. The model achieved good accuracy, and the web interface made it user-friendly. This project can be extended with deep learning models and multilingual datasets to make it more robust for real-world applications.