#Sentiment analysis using  audio

import os

import librosa

import soundfile

import numpy as np

import joblib

from google.colab import drive

drive.mount('/content/drive')

# Function to extract features from a sound file

def extract\_feature(file\_name, mfcc, chroma, mel):

    with soundfile.SoundFile(file\_name) as sound\_file:

        X = sound\_file.read(dtype="float32")

        sample\_rate = sound\_file.samplerate

        if chroma:

            stft = np.abs(librosa.stft(X))

        result = np.array([])

        if mfcc:

            mfccs = np.mean(librosa.feature.mfcc(y=X, sr=sample\_rate, n\_mfcc=40).T, axis=0)

            result = np.hstack((result, mfccs))

        if chroma:

            chroma = np.mean(librosa.feature.chroma\_stft(S=stft, sr=sample\_rate).T, axis=0)

            result = np.hstack((result, chroma))

        if mel:

            mel = np.mean(librosa.feature.melspectrogram(y=X, sr=sample\_rate).T, axis=0)

            result = np.hstack((result, mel))

    return result

# Load the pre-trained model using its file path

model\_path = '/content/speech-emotion.sav'

loaded\_model = joblib.load(model\_path)

# Function to make predictions using the pre-trained model

def predict\_emotion(file\_path):

    feature = extract\_feature(file\_path, mfcc=True, chroma=True, mel=True)

    feature = feature.reshape(1, -1)

    prediction = loaded\_model.predict(feature)

    return prediction[0]

# Example usage

file\_path = "/content/03-01-02-02-01-02-03.wav"

predicted\_emotion = predict\_emotion(file\_path)

print("Predicted Emotion:", predicted\_emotion)