DOCUMENTATION - VIBETRACK

PROJECT : VibeTrack, a voice driven emotion detector and mood tracker.
SCOPE : This document includes model choice, low level explanation of algorithm, preprocessing used.

1. **SUMMARY :**
    - **Selected Model :** Custom 1D Convolutional Neural Network (CNN) trained on MFCC features extracted from audio.
    - **Preprocessing :** Librosa based MFCC extraction with silence trimming, normalization, and fixed length padding or truncation.
    - **Data Augmentation :** Applied additive noise, pitch or time shifts, time stretching and spectrogram masking during training. Augmentation reduces overfitting and improves generalisation to real world audio.
    - **Why This Choice ?** MFCC's compress important spectral info thats perfectly suitable for emotion recognition type of tasks that are analytic; 1D CNN learn local patterns efficiently and are well suited for low latency inference (for faster though efficient outputs).

2. **FEATURES OF THE PROJECT :**
    - Real time microphone emotion detection.
    - .wav file upload support
    - Custom CNN architecture trained on MFCC features
    - Emotion prediction with confidence scores
    - Mood journal to track emotional patterns
    - Personalized suggestion based on detected emotion
    - Clean UI with waveform and probability visualisations
    - Emotion aware chatbot with conversation memory, tone adaptation and visual insights.

3. **HIGH LEVEL WORKING :**
    - User can record via microphone or can upload .wav file.
    - The audio is trimmed, resampled and converted into MFCC features.
    - Data is augmented to increase robustness to real audio files.
    - Processed MFCCs are passed into 1D CNN that learns patterns and corelates them to 6 emotional states(Happy, Sad, Fear, Neutral, Angry, Disgust).
    - Sofmax probabilities are returned with emotional label and confidence score.
    - Predictions are logged to mood journal and used to adapt emotion aware chatbot responses.

4. **DETAILS OF PREPROCESSING :**

    Mel Frequency cepstral coefficients are the core feature representation used in VibeTrack. They are used to summarise the spectral envelope of speech, and capture prosodic and timbral cues strongly tied to emotional expression.

    PIPELINE:

    - **Audio Loading :** audio clips are loaded using librosa.load(default sampling rate here -> 22050 Hz)
    - **MFCC Extraction :** around 30 mfcc coefficients per frame are considered, they act as compact representations of speech spectrum. Visualisation is also used for checks during development.
    - **Padding :** sequences are paddedto fixed length to ensure equal shape across samples.
    - **Normalisation :** basic scaling already applied during dataset preparation.

5. **DATA AUGMENTATION :**

To avoid overfitting and increase robustness, multiple waveform level augmentations were applied (during training time only).

- **Noise addition :** added small amounts of Gaussian noise to the audio signal. Only small randomised amplitude is chosen as much noise can cause confusion in emotion recognition.
- **Time Streching :** Slight speeding and slowing down the audio without altering pitch. Used short time fourier transform + phase vocoding to preserve spectral consistency while scaling time.
- **Shifting :** Rolls the waveform to a little forward and backward by a small random offset, implemented using simple circular shift on waveform array.
- **Pitch :** Raises or lowers pitch of voice by 0.7 semitones. Now the model learns to generalise across different genders and vocal ranges, avoiding bias to a narrow pitch distribution.

6. **MODEL CHOOSING : 1D CONVOLUTIONAL NEURAL NETWORK :**

Why 1D CNN on MFCCs ?

- If we treat MFCC frames as a sequence (time * coefficients) and use 1D conv along the time axis, model learns temporal patterns in prosody (rise or fall or pitch modulations) and treats coefficients at each time step as channels.
- Compared to 2D CNNs, 1D CNNs are computationally cheap and map naturally to streaming or real time inputs where the temporal axis matters.
- 1D CNNs allow parallel training and perform equal or better than RNNs on SER datasets.

MODEL SUMMARY :

```
    model = tf.keras.Sequential([
    L.Conv1D(512, kernel_size=5, strides=1, padding='same', activation='relu',
input_shape=(x_train.shape[1],1)),
    L.BatchNormalization(),
    L.MaxPool1D(pool_size=5, strides=2, padding='same'),

    L.Conv1D(512, kernel_size=5, strides=1, padding='same', activation='relu'),
    L.BatchNormalization(),
    L.MaxPool1D(pool_size=5, strides=2, padding='same'),
    Dropout(0.2),

    L.Conv1D(256, kernel_size=5, strides=1, padding='same', activation='relu'),
    L.BatchNormalization(),
    L.MaxPool1D(pool_size=5, strides=2, padding='same'),

    L.Conv1D(256, kernel_size=3, strides=1, padding='same', activation='relu'),
    L.BatchNormalization(),
    L.MaxPool1D(pool_size=5, strides=2, padding='same'),
    Dropout(0.2),

    L.Conv1D(128, kernel_size=3, strides=1, padding='same', activation='relu'),
    L.BatchNormalization(),
    L.MaxPool1D(pool_size=3, strides=2, padding='same'),
    Dropout(0.2),

    L.Flatten(),
    L.Dense(512, activation='relu'),
    L.BatchNormalization(),
    L.Dense(6, activation='softmax')
])
```

- **Convolutional layers** : **(5 blocks)** Deep CNN, to extract hierarchical features from coarse pitch variations to fine spectral modulations.

- **Batch Normalisation and Dropout :** To prevent overfitting despite high capcity. Stabilisation of training by normalising layer inputs per mini batch and regularisation.
- **Max Pooling :** To focus on dominating features and also reduce unnecessary computation.
- **Large Filter banks : (512 -> 128)** allow model to learn diverse emotion relevant patterns.
- **Flatten + Dense (512) :** ensures global interactions of features and also useful for emotion classification where context matters.

TRAINABLE PARAMETERS OF OUR POWERFUL+DEEP CNN:

(This extra depth made it very good at understanding emotion patterns in audio data.**)**

- **Conv1 (512 filters, kernel size 5)** -> 3K parameters.
  -> 5 * 1 * 512 + 512 = 3,072
  catches the most basic voice patterns like pitch and energy.
- **Conv2 (512 filters, kernel size 5)** -> 1.3M parameters.
  -> 5 * 512 * 512 + 512 = 1,310, 72
  biggest layer, learns complex emotional structures.
- **Conv3 (256 filters, kernel size 5)** -> 656K parameters.
  -> 5 * 512 * 256 + 256 = 655,616
  shrinks features while keeping important voice details.
- **Conv4 (256 filters, kernel size 3)** -> 197K parameters.
  -> 3 * 256 * 256 + 256 = 196,864
  focuses on small tone and rhythm changes.
- **Conv5 (128 filters, kernel size 3)** -> 98K parameters.
  -> 3 * 256 * 128 + 128 = 98,432
  To make the features more compact and emotion focused.
- **Dense (512 units, after flattening)** -> millions of parameters (exact number depends on input size).
  Brings all together to make final emotion decisions.

- F**inal Dense (6 outputs, Softmax)** -> about 3K parameters.
  -> 512 * 6 + 6 = 3,078
  Gives the final emotion class: Happy, Sad, Angry, Fear, Neutral, or Disgust.

7. **TRAINING AND EVALUATION :**
   - **Loss Function:** Categorical Crossentropy
     Used because this is a multi-class classification problem (6 emotions).
   - **Optimizer:** Adam
     Popular optimizer that adjusts the learning rate automatically for faster and more stable training.
   - **Metrics:** Accuracy
     track how often the predicted emotion matches the true one.
   - **Batch Size:** 64
     Process 64 audio samples at a time for efficient GPU training.
   - **Epochs:** Up to 50
     But training can stop earlier due to Early Stopping.

   - **EarlyStopping:** Stop training if validation accuracy doesn't improve for 10 epochs, prevents overfitting.
   - **ReduceLROnPlateau:** lower the learning rate if the model gets stuck, helps to escape plateaus.
   - **ModelCheckpoint:** saves the best model (highest validation accuracy upto now) during training.

   - The model achieved **99.9% val accuracy** on validation data.

8. **OTHER :**

   Kaggle Dataset : Speech Emotion Recognition English

   Notebook on Google Colab : SER4.ipynb

Let your voice speak your feelings, with **VibeTrack** !