Speech Emotion Recognition

Dhruv Varshney
Electrical Engineering, IIT Kanpur
220366, vdhruv22@iitk.ac.in

Dwij Om Oshoin
Electrical Engineering, IIT Kanpur
220386, dwij22@iitk.ac.in

Gautam Arora
Electrical Engineering
220405, garora22@iitk.ac.in

Manas Ranjan
Mathematics & Scientific Computing
220612, manasr22@iitk.ac.in

Sumit Kumar
Mechanical Engineering
221102, sumitkumar22@iitk.ac.in

***Objective*— This report explores a deep learning-based approach for Speech Emotion Recognition (SER) using a Convolutional Neural Network (CNN) with residual connections. The model extracts Mel Frequency Cepstral Coefficients (MFCC) and other spectral features from audio signals to classify emotions accurately. It incorporates batch normalization, dropout regularization, and label smoothing for enhanced stability and generalization. The Adam optimizer with gradient clipping is used to improve training efficiency. Experimental results demonstrate the model's effectiveness in recognizing emotions from speech, making it a valuable tool for applications in human-computer interaction and affective computing.**

Keywords*— Convolutional Neural Network, Mel Frequency Cepstral Coefficients (MFCC), Deep Learning, Residual Connections, Batch Normalization, Adam Optimizer, Feature Extraction

I. Introduction

Speech Emotion Recognition (SER) is an emerging area of artificial intelligence in which human emotions can be understood by analyzing speech using machines. The area has numerous applications, including but not limited to human-computer interaction, diagnosis of mental health, virtual assistants, and automation of customer services. Historically, conventional techniques used hand-designed features and conventional machine learning algorithms; however, recent advancements in deep learning approaches, specifically Convolutional Neural Networks (CNNs), have made tremendous progress.

This work proposes a residual connection-based CNN model for classification of strong emotions from speech. The proposed model employs MFCC & its derivatives, spectral contrast, spectral Centroid, spectral Bandwidth, Spectral Roll-Off, ZCR and the pitch-dependent feature (Chroma) for emotionally diverse features efficiently. Batch normalization, dropout regularization, and label smoothing are employed for better generalization and prevention from overfitting.

The information includes labeled audio samples that have been classified into various emotions, preprocessed to maintain uniformity. Convergence stability and effective training are guaranteed by the use of the Adam optimizer with gradient clipping. The experimental results point to the accuracy and reliability of the model, showing that it can be utilized in real-world applications in affective computing and speech-based AI systems.

II. Feature Extraction & Preprocessing

To make speech data usable for emotion recognition, we extract meaningful features and preprocess them for consistency.

A. Feature Extraction

Librosa loads audio files; resampled to 22,050 Hz, they are then pre-emphasized to improve clarity. Silence is trimmed to focus on speech content . We extract key features:

- **MFCCs & Derivatives (ΔMFCC, ΔΔMFCC):** Record speech patterns and variations.
- **Spectral Features:** Centroid, bandwidth, contrast, and Rolloff, which describe frequency distribution.
- **Zero Crossing Rate (ZCR):** Measures the rate of signal changes, useful for distinguishing sounds.
- **Chroma & Pitch Features:** Capture tonal variations related to emotions.

Each feature is **padded/truncated** to a fixed length (64 frames) for uniformity.

B. Preprocessing & Normalization

In this project, preprocessing plays a critical role in ensuring that the extracted features are clean, consistent, and suitable for training a deep learning model. The pipeline includes several steps aimed at preparing the audio data and features for optimal performance.

Preprocessing Steps:

- **Silence Removal:** The audio files were trimmed to remove silent segments. This step eliminates unnecessary pauses and ensures that the features extracted focus solely on the speech content, which is essential for emotion recognition.
- **Pre-Emphasis-** A pre-emphasis filter was applied to amplify high-frequency components of the audio signals. This is important because higher frequencies often carry significant information about speech patterns and emotions.
- **Standardization**: To normalize the features, we used StandardScaler,

which transforms the data to have a mean of zero and a standard deviation of one. This ensures that all feature dimensions are on the same scale, preventing any single feature from disproportionately influencing the model during training.

- **Reshaping Features:** After normalization, the features were reshaped into a 4D format to match the input requirements of the Convolutional Neural Network (CNN). Specifically:
  (a) The data was structured as batches of samples with dimensions corresponding to feature length, width, and channels.
  (b) The features were converted to float32 data type to ensure compatibility with deep learning frameworks.

III. Model Architecture

We have used CNN Model with 15 Hidden layers. The Convolutional block has 9 hidden layers, and the dense layers have 6 hidden layers.

1. **Input Layer: -**
   - It accepts processed audio features as 2D spectrogram-like input
   - Shape: - 82(MFCC + Spectral Features) * 64(time steps) * 1(channel)
   - 20 MFCCs + **20 Δ (deltas)** + **20 ΔΔ (delta-deltas)** = 60
   - Spectral features: - Chroma (12), ZCR (1), Spectral Contrast (6), Spectral Centroid, Spectral Rolloff, Spectral Bandwidth
2. **Convolutional (Conv2D) layers: -**
   - First Conv Layers (32 filters): -

   Scans the "sound picture" with 3×3 magnifying glasses to find small patterns so that each filter learns to detect different features
   - Deeper Conv Layers (64/128 filters): -
     Combine small patterns into bigger ones
3. **Batch Normalization: -**
   - Keeps volume consistent across all audio clips (like auto-adjusting microphone levels) which prevents one loud emotion (e.g., anger) from drowning out others.
4. **ReLU Activation: -**
   - Turns negative values to zero ("mutes" irrelevant noise) and keeps only the noticeable features.
5. **Max Pooling: -**
   - Shrinks the "sound picture" by keeping only the loudest/most noticeable parts every 2×2 area which makes the model focus on important features.
6. **Dropout: -**
   - Randomly ignores 20-40% of detected features during training to prevent over-reliance on any single clue (e.g. not just using "loudness" to detect anger.
7. **Residual (Skip) Connection: -**
   - Provides a shortcut for the original sound features to bypass some layers which ensures the model doesn't "forget" basic info.
8. **Global Average Pooling: -**
   - Averages all features into one summary number per pattern to make it simpler and less prone to errors than flattening.
9. **Dense (Fully Connected) Layers: -**
   - First Dense (256 neurons): - Combines all detected patterns ("high pitch + fast tempo + shaky voice = fear").
   - Second Dense (128 neurons): - Refines the decision with fewer but more precise connections.
10. **SoftMax Output: -**
    - Converts scores into probabilities ensuring probabilities adding up to 1.
11. **Adam Optimizer**: -
    The learning rate controls how much the model updates its weights during training. A value too high can cause overshooting, while too low leads to slow convergence; a default of 1e-3 balances efficiency and stability. Beta_1 (commonly set to 0.9) determines how much past gradients influence updates, smoothing erratic steps and stabilizing training. Beta_2 (typically 0.999) scales updates based on gradient magnitudes, helping the optimizer adapt to rare features effectively.
12. **Clipnorm: -**
    - Clips gradients to a max norm of 1.0.
    - Prevents extreme updates that could destabilize training.
13. **Loss function: -**
    - Cross-entropy loss function has been used.

$$-\sum p(x)\log(p(x))$$

IV. Results

1. Model Performance

- Accuracy: 73.88 % (validation) and demonstrating great generalization.
- Loss: Approaches ~2.5–5.0, indicating convergent training.

2. Confusion Matrix Insights

a) Top Performers: Angry (83%), Calm (83%), Happy (78%) – high accuracy.

b) Weak Classes: Sad (43%), Surprised (18%) – frequently confused with fearful/happy.

c) Key Misclassifications
- Sad → Fearful (30%)
- Surprised → Fearful (27%)
- Disgust → Angry (13%)

3. Key Performance Metrics

- The model is very accurate at spotting angry and calm emotions (over 80% correct) but struggles with sad and surprised emotions (less than half right).

- It reliably detects calm and angry voices but often misses when someone is surprised.