Speech Emotion Recognition

Yahia Ibrahim AlKaranshawy, Hossam Osama Iraqi, Ali Hassan Ali

May 9, 2025

# Problem statement

It’s required to Create the Feature Space in the time domain, or in the frequency domain and Convert the audio waveform to a mel spectrogram and compare the performance of the CNN models in the two approaches using [Speech Emotion Recognition (en)](https://www.kaggle.com/datasets/dmitrybabko/speech-emotion-recognition-en)[.](https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction)

## Dataset Loading and visualize

The CREMA-D dataset contains audio files labeled with six emotion classes: **Angry, Disgust, Fear, Happy, Neutral, and Sad**. Each filename follows a specific format that encodes the emotion (e.g., 1001\_DFA\_ANG\_XX.wav where ANG = Angry).

**Key Steps in Data Loading:**

1. **File Parsing**
   * Scan the dataset directory for .wav files.
   * Extract emotion labels from filenames using the embedded emotion codes (e.g., ANG, DIS).
2. **Audio Loading**
   * Use librosa.load() to read audio files at a standard sampling rate (e.g., 22050 Hz).
   * Ensure uniform duration by padding/trimming signals to 3 seconds (66150 samples).
3. **Emotion Mapping**
   * Convert emotion codes (ANG, DIS, etc.) to full names (Angry, Disgust, etc.).

A screenshot of a computer screen

AI-generated content may be incorrect.

## Dataset Splitting

* 1. Split the data into 70% training and validation and 30% testing.
* 2. Use 5% of the training and validation data for validation.

# MEL

For each audio file:

* The waveform is loaded with a sampling rate of 22050 Hz and clipped to 3 seconds duration.
* If shorter than 3 seconds, zero-padding is applied.
* The Mel-spectrogram is computed with 128 Mel bands.
* Spectrograms are converted to log scale (dB).
* Temporal length is padded or truncated to a fixed size (130 frames).

**2.2 Data Augmentation**

The function augment\_audio() applies random transformations:

* **Pitch shifting** (±1 or ±2 semitones),
* **Time stretching** (between 0.9x to 1.1x speed),
* **Additive Gaussian noise**.

These augmentations are only applied to training data to increase diversity and reduce overfitting.

**2.3. Preprocessing Steps**

* Extract Mel-spectrograms for all .wav files.
* Filter out files with unknown emotion labels.
* Encode emotion labels using LabelEncoder and one-hot encoding.
* Add channel dimension to spectrograms to be compatible with CNN input requirements.

**2.4. Train-Test Split**

A deep CNN is designed to extract spatial features from the Mel-spectrograms. The model includes:

* **Three convolutional blocks**:
  + Each block has two Conv2D layers with ReLU, Batch Normalization, MaxPooling, and Dropout.
  + Filter sizes grow from 64 → 128 → 256 to increase representational capacity.
* **Fully connected layers**:
  + Dense layers with 512 and 256 units, followed by Batch Normalization and Dropout.
* **Output layer**:
  + A softmax layer to predict emotion probabilities.

**Model Characteristics**

* Strong regularization via L2 penalties and Dropout.
* Batch Normalization accelerates convergence and stabilizes training.
* Designed to be robust against overfitting.

**2.5. Result**

Classification Report:

precision recall f1-score support

angry 0.70 0.72 0.71 382

disgust 0.52 0.56 0.54 381

fearful 0.53 0.46 0.50 381

happy 0.59 0.53 0.56 382

neutral 0.58 0.63 0.60 326

sad 0.55 0.58 0.57 381

accuracy 0.58 2233

macro avg 0.58 0.58 0.58 2233

weighted avg 0.58 0.58 0.58 2233

A screenshot of a diagram

AI-generated content may be incorrect.

A graph of a graph of a graph

AI-generated content may be incorrect.

**3. Time-Domain Features Extraction**

For each audio file:

* The waveform is loaded with a sampling rate of **22050 Hz** and clipped to a **3-second duration**.
* If it is shorter than 3 seconds, **zero-padding** is applied.
* **Time-domain features** are extracted:
  + **Zero Crossing Rate (ZCR):** The rate at which the signal changes from positive to negative.
  + **Energy:** The sum of squared amplitudes of the signal, representing its strength.

**Key Steps:**

1. **ZCR Calculation:**
   * Computed over frames of 2048 samples with a hop length of 512 samples.
   * Represents the rate of sign changes in the waveform.
2. **Energy Calculation:**
   * Calculated as the sum of squared amplitudes in each frame.
   * Reflects the intensity of the signal over time.
3. **Padding:**
   * Both ZCR and Energy are padded to match length.

**3.1 Preprocessing Steps**

* Extract **ZCR** and **Energy** for all .wav files.
* Filter out files with unknown emotion labels.
* Encode emotion labels using LabelEncoder and one-hot encoding.
* Pad features to a fixed size to match CNN input requirements.

**3.2 Train-Test Split**

The dataset is split into training and testing sets:

* 70% for training and 30% for testing.
* Stratified sampling is applied to maintain class balance.

**3.3 1D CNN Model Design**

A deep **1D CNN** is designed to extract temporal features from the time-domain feature space.

**Model Architecture:**

* **Convolutional Blocks:**
  + Three blocks, each with:
    - Two Conv1D layers with ReLU activation, Batch Normalization, MaxPooling, and Dropout.
    - Filter sizes grow from 64 → 128 → 256 for increased representational capacity.
* **Fully Connected Layers:**
  + Dense layers with 128 units, followed by Batch Normalization and Dropout.
* **Output Layer:**
  + A softmax layer to predict emotion probabilities.

**3.4 Results**

**A screenshot of a graph

AI-generated content may be incorrect.**

**A graph of a line and a line graph

AI-generated content may be incorrect.**