Speech Emotion Recognition System

Technical Documentation

By Gulam Mazid

B.Tech Computer Science

Maulana Azad National Urdu University

Table of Contents

Introduction

This document provides a comprehensive technical documentation for the Speech Emotion Recognition System. The system uses deep learning techniques to analyze and classify emotions from speech signals. It implements both LSTM (Long Short-Term Memory) and CNN (Convolutional Neural Network) models for emotion classification.

Key Features

- Real-time voice recording and emotion analysis
- Support for pre-recorded audio file analysis
- Dual model architecture (LSTM and CNN)
- Feature extraction from audio signals
- Emotion classification into multiple categories
- User-friendly command-line interface

System Architecture

High-Level Overview

The system follows a modular architecture with the following main components:

1. Audio Recording Module
2. Feature Extraction Module
3. Preprocessing Module
4. Model Inference Module
5. User Interface Module

Component Interaction

```
[Audio Input] → [Feature Extraction] → [Preprocessing] → [Model Inference] → [Results]
```

Installation Guide

Prerequisites

- Python 3.8 or higher
- Virtual Environment (recommended)
- Git (for cloning the repository)

Dependencies

The following Python packages are required:

```
numpy>=1.19.5
pandas>=1.2.4
librosa>=0.8.1
scikit-learn>=0.24.2
sounddevice>=0.4.2
matplotlib>=3.4.2
torch>=2.0.0
torchaudio>=2.0.0
soundfile>=0.10.3
imbalanced-learn>=0.13.0
resampy>=0.4.3
```

Installation Steps

1. Clone the repository
2. Create and activate virtual environment:

```
python -m venv .venv
.\.venv\Scripts\activate  # Windows
source .venv/bin/activate  # Linux/Mac
```

3. Install dependencies:

```
pip install -r requirements.txt
```

Project Structure

```
speech_emotion_recognition/
├── .venv/                  # Virtual environment
├── __pycache__/           # Python cache files
├── datasets/              # Dataset storage
├── features/              # Extracted features
├── models/                # Trained models
├── recordings/            # Audio recordings
├── __init__.py           # Package initialization
├── eda.py                # Exploratory Data Analysis
├── extract_features.py   # Feature extraction
├── main.py               # Main application entry
├── models.py             # Model definitions
├── predictions.py        # Prediction logic
├── preprocessing.py      # Data preprocessing
├── voice_recorder.py     # Audio recording
└── requirements.txt      # Dependencies
```

Detailed Pipeline

1. Audio Recording Module

The `voice_recorder.py` module handles audio recording:

- Lists available audio input devices
- Records audio for a specified duration

- Saves recordings in WAV format
- Implements error handling for device selection

2. Feature Extraction

The `extract_features.py` module:

- Extracts MFCC (Mel-frequency cepstral coefficients)
- Computes spectral features
- Generates statistical features
- Handles audio resampling and normalization

3. Preprocessing Pipeline

The `preprocessing.py` module implements:

- Audio signal normalization
- Feature scaling
- Data augmentation
- Class balancing using RandomOverSampler
- Train-test splitting

4. Model Architecture

The system implements two models:

LSTM Model

- Input layer with feature dimension
- Multiple LSTM layers
- Dropout for regularization
- Dense output layer
- Softmax activation for classification

CNN Model

- Convolutional layers for feature extraction
- Max pooling layers
- Fully connected layers
- Dropout for regularization
- Softmax activation for classification

5. Prediction Pipeline

The `predictions.py` module:

- Loads trained models
- Processes input audio
- Generates predictions from both models
- Returns emotion classifications

Usage Guide

Running the Application

1. Activate the virtual environment
2. Run the main script:

   ```
   python main.py
   ```

Available Commands

- Press 'R' to record and analyze voice
- Press 'P' to analyze existing audio file

Example Usage

1. Recording Mode:

   ```
   Press 'R' to record or 'P' to predict: R
   Recording for 3 seconds... Speak now!
   Voice recording saved to recordings/myvoice.wav
   Predictions:
   LSTM Model predicts: neutral
   CNN Model predicts: surprised
   ```

2. Prediction Mode:

   ```
   Press 'R' to record or 'P' to predict: P
   Enter the path to your audio file: path/to/audio.wav
   ```

Technical Specifications

Audio Specifications

- Sample Rate: 22050 Hz

- Duration: 3 seconds
- Format: WAV
- Channels: Mono

Feature Specifications

- MFCC Features: 13 coefficients
- Spectral Features: 40 dimensions
- Statistical Features: 5 dimensions

Model Specifications

- LSTM Layers: 2 layers with 128 units
- CNN Layers: 3 convolutional layers
- Dropout Rate: 0.5
- Batch Size: 32
- Learning Rate: 0.001

Future Improvements

1. Real-time emotion visualization
2. Multi-language support
3. Enhanced feature extraction
4. Model ensemble methods
5. Web interface development
6. Mobile application development
7. API integration capabilities
8. Improved error handling
9. Performance optimization
10. Extended emotion categories

Author Information

- **Name:** Gulam Mazid
- **Degree:** B.Tech Computer Science
- **Institution:** Maulana Azad National Urdu University
- **Project:** Speech Emotion Recognition System
- **Date:** March 2024

This documentation is part of the Speech Emotion Recognition System project. For any queries or support, please contact the author.