Real-Time Speech-to-Sentiment Analysis

This project combines speech recognition and sentiment analysis to understand human emotions in real-time conversations. It uses OpenAI's Whisper for speech recognition and various emotion detection models for sentiment analysis.

Features:

- Real-time speech-to-text transcription
- Multiple emotion detection modes:
  - Basic (Positive/Negative)
  - Standard (7 basic emotions)
  - Detailed (17 emotions from GoEmotions)
- Audio file upload and processing
- Video file audio extraction and analysis
- Audio recording and playback
- Visual audio level meter
- Real-time emotion visualization
- Configurable analysis modes (segment or full transcript)
- Save and manage recordings

Prerequisites:

- Python 3.8 or higher
- A working microphone
- Git (for cloning the repository)
- FFmpeg (required for audio processing)

Installing FFmpeg:

macOS: brew install ffmpeg brew install portaudio

Windows: Download from https://ffmpeg.org/download.html

Ubuntu/Debian: sudo apt-get install ffmpeg sudo apt-get install libportaudio2

Installation Steps:

1. Clone the repository: git clone cd speech-sentiment
2. Create and activate a virtual environment: python -m venv venv source venv/bin/activate # On Windows: venv\Scripts\activate
3. Install the required packages: pip install -r requirements.txt
4. Set up Hugging Face access: pip install --upgrade huggingface_hub huggingface-cli login

Enter your access token from https://huggingface.co/settings/tokens

Configuration:

The project uses two main configuration files in the config directory:

config.yaml:

- Controls model selection and basic settings
- Choose emotion detection mode:
    - "basic": Simple positive/negative
    - "standard": 7 basic emotions
    - "detailed": 17 detailed emotions

emotions.yaml:

- Defines available emotions for each mode
- Configures colors and thresholds
- Sets number of emotions to display (top_k)

Usage:

1. Start the application: python run.py

2. Real-time Recording:

- Click "Start Recording" to begin
- Speak into your microphone
- Recording automatically stops after 5 seconds
- View transcription and emotion analysis
- Choose to:
    - Start a new recording (clears previous)
    - Continue recording (adds to existing)

3. File Processing:

- Click "Upload Audio File" for audio files
- Click "Upload Video File" for video files
- Supported formats: wav, mp3, m4a, aac, flac, mp4, mov, avi, mkv

4. Analysis Modes:

- "Analyze Current Segment": Shows emotions for latest recording
- "Analyze Full Transcript": Shows emotions for entire session

5. Recording Management:

- Play Recording: Listen to current recording
- Save Recording: Save audio and transcript with timestamp

Project Structure:

speech-sentiment/ ├── config/ | ├── config.yaml # Main configuration | └── emotions.yaml # Emotion definitions ├── src/ | ├── main.py # GUI and main application | ├── speech_recognition/ | | └──

whisper_client.py # Speech recognition | ├── sentiment_analysis/ | | └── bert_model.py # Emotion detection | └── utils/ | └── config_manager.py # Configuration handling ├── recordings/ # Saved recordings directory ├── requirements.txt # Dependencies └── run.py # Entry point

Emotion Detection Modes:

1. Basic Mode:

- POSITIVE (Green)
- NEGATIVE (Red)

2. Standard Mode:

- Joy (Green)
- Surprise (Yellow)
- Neutral (Gray)
- Sadness (Blue)
- Anger (Red)
- Fear (Purple)
- Disgust (Turquoise)

3. Detailed Mode:

- All standard emotions plus:
- Love, Admiration, Excitement
- Gratitude, Pride, Optimism
- Relief, Confusion, Remorse
- Disappointment

Troubleshooting:

1. Audio Issues:

- Check microphone connection
- Verify microphone permissions
- Run python -m sounddevice to list devices

2. Model Loading Issues:

- Ensure Hugging Face login is complete
- Check internet connection
- Verify token permissions

3. File Processing Issues:

- Ensure FFmpeg is installed
- Check file format compatibility
- Verify file permissions

Contributing:

1. Fork the repository

2. Create a feature branch
3. Commit your changes
4. Push to the branch
5. Create a Pull Request

License: This project is licensed under the MIT License - see the LICENSE file for details.