Real-Time Speech-to-Sentiment Analysis

This project combines speech recognition and sentiment analysis to understand human emotions in real-time conversations using OpenAI's Whisper for speech recognition and DistilBERT for sentiment analysis.

Prerequisites

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

Installation

1. Install FFmpeg:

For macOS: brew install ffmpeg

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

For Ubuntu/Debian: sudo apt-get install ffmpeg

2. Clone the repository: git clone cd speech-sentiment

3. Create and activate a virtual environment:

For Windows: python -m venv venv venv\Scripts\activate

For macOS/Linux: python -m venv venv source venv/bin/activate

4. Install the required packages: pip install -r requirements.txt

5. Set up Hugging Face access: pip install --upgrade huggingface_hub huggingface-cli login

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

Project Structure

speech-sentiment/ ├── requirements.txt ├── src/ | ├── **init**.py | ├── main.py | ├── speech_recognition/ | | └── whisper_client.py | └── sentiment_analysis/ | └── bert_model.py

Usage

1. Make sure your microphone is connected and working.

2. Run the main script: python src/main.py

3. The program will:

    - Start listening for speech (5-second intervals by default)
    - Transcribe the speech using Whisper
    - Analyze the sentiment (positive/negative) using DistilBERT
    - Display the results in the console

4. To stop the program, press Ctrl+C

Output Format

The program will output:

- The transcribed text from your speech
- A sentiment label (POSITIVE/NEGATIVE) with a confidence score

Troubleshooting

Common Issues:

1. **FFmpeg not found:**
    - Make sure FFmpeg is installed and accessible from your command line
    - Try running `ffmpeg -version` to verify the installation
2. **Microphone not detected:**
    - Ensure your microphone is properly connected
    - Check if it's set as the default input device
    - Try running `python -m sounddevice` to list available devices
3. **Hugging Face authentication:**
    - If you get authentication errors, ensure you've logged in with `huggingface-cli login`
    - Verify your token has 'read' permissions
    - Try regenerating your token at https://huggingface.co/settings/tokens
4. **CUDA/GPU errors:**
    - The project works with CPU by default
    - For GPU support, install the appropriate CUDA toolkit version for your system
5. **Memory issues:**
    - Try using a smaller Whisper model by modifying the model_size parameter in WhisperTranscriber
    - Available sizes: "tiny", "base", "small", "medium", "large"

Audio Device Selection

If you have multiple audio input devices, you can specify which one to use by modifying the `record_audio` method in `whisper_client.py` to include the device ID:

audio = sd.rec(int(duration * self.sample_rate), samplerate=self.sample_rate, channels=1, device=DEVICE_ID # Add your device ID here)

Acknowledgments

- OpenAI Whisper for speech recognition
- Hugging Face Transformers for DistilBERT implementation
- The DistilBERT model fine-tuned on SST-2 dataset for sentiment analysis