Audio Splitter - Detailed Code Overview

This Python tool records audio from your microphone and automatically splits it into segments based on detected keystrokes, using amplitude spikes and actual keypress logging. It is ideal for acoustic keystroke recognition, biometric research, and sound-based interaction logging.

The script works by capturing live microphone input, detecting keystrokes from audio amplitude spikes, logging actual keypresses using the 'keyboard' module, and splitting the audio into multiple .wav files for each detected segment.

Features

- Real-time audio recording
- Detects keystrokes based on amplitude spikes
- Automatically splits audio at keystroke points
- Saves each segment as a separate .wav file
- Logs actual keyboard keypresses in a .json file
- Customizable thresholds and directories via command-line arguments

Requirements

- Python 3.8+
- Libraries:
  - pyaudio
  - numpy
  - keyboard

To install dependencies, run:

pip install -r requirements.txt

Installation

Clone the repository:

git clone https://github.com/Saronzeleke/Splitter.git

Install dependencies:

pip install -r requirements.txt

Usage

Run the script:

python main.py

This starts:

- Audio recording from your microphone

- Keystroke detection (via amplitude and actual keypresses)

Press ESC to stop recording. After stopping:

- The script splits the audio into chunks

- Saves each chunk as a .wav file in the split_audio/ folder

- Writes keystroke logs to keystroke_log.json

Command-Line Arguments

You can customize behavior using the following options:

python main.py --silence-threshold 600 --keystroke-gap 0.2 --output-dir output --log-file mylog.json

Argument	Description	Default
--silence-threshold	Amplitude level above which audio is considered a keystroke	500
--keystroke-gap	Minimum time (in seconds) between two keystroke detections	0.1
--output-dir	Directory where split audio files are saved	split_audio
--log-file	File to store actual keyboard events with timestamps	keystroke_log.json

Output Files

- split_audio/segment_1.wav, segment_2.wav, ...
- keystroke_log.json: Contains keypress event info (key name, timestamp, frame index)

Example Output

Recording started... Press ESC to stop.

Detected 5 keystrokes by amplitude

Saved segment_1.wav (frames 0-1024)

Saved segment_2.wav (frames 1024-2048)

...

Keystroke log saved to keystroke_log.json

Audio segments saved to 'split_audio' directory

Notes

- The amplitude threshold (--silence-threshold) determines how sensitive the keystroke detection is.
- Frame-based splitting ensures timing alignment between actual and amplitude-based keystrokes.
- All audio is saved in 16-bit PCM WAV format.

Troubleshooting

- No keystrokes detected? Lower the --silence-threshold (e.g., 400)
- Keystrokes logged incorrectly? Check for background noise or adjust --keystroke-gap
- Permission error with keyboard logging? Run as administrator (Windows) or use sudo (Linux)

License