

Package: audio.vadwebrtc (via r-universe)

October 21, 2024

Type Package

Title Voice Activity Detection using the 'webrtc' Toolkit

Version 0.2

Maintainer Jan Wijffels <jwi jffels@bnosac.be>

Description Voice Activity Detection using the 'webrtc' toolkit.
Identify the locations in audio files where there is an active voice. This is done based on a Gaussian Mixture Model implemented in the 'webrtc' framework.

License MPL-2.0

URL <https://github.com/bnosac/audio.vadwebrtc>

Encoding UTF-8

Depends R (>= 2.10)

Imports Rcpp (>= 0.11.5), utils

Suggests av, audio

LinkingTo Rcpp, abseil

RoxygenNote 7.1.2

SystemRequirements C++14 or greater

Repository <https://r-multiverse-staging.r-universe.dev>

RemoteUrl <https://github.com/bnosac/audio.vadwebrtc>

RemoteRef c3de76ddc738dbb4a80ac7ec67ac9da3cba093ba

RemoteSha c3de76ddc738dbb4a80ac7ec67ac9da3cba093ba

Contents

is.voiced	2
VAD	3
VAD_channel	4

Index	7
--------------	----------

is.voiced	<i>Get from a Voice Activity Detection (VAD object) the segments which are voiced</i>
-----------	---

Description

Postprocessing the Voice Activity Detection whereby sequences of voiced/non-voiced segments are collapsed by

1. first considering all non-voiced segments which are small in duration (default < 1 second) voiced
2. next considering voiced segments with length less than a number of seconds (default < 1 second) non-voiced

Usage

```
is.voiced(x, channel = 0, units = "seconds", ...)
```

Arguments

x	an object of class VAD as returned by VAD or VAD_channel
channel	integer with the channel, showing the voiced section of that channel only. Only used for segments extracted with VAD_channel
units	character string with the units to use for the output and thresholds used in the function - either 'seconds' or 'milliseconds'
...	further arguments passed on to the function

Value

A data.frame with columns vad_segment, start, end, duration, has_voice indicating where in the audio voice is detected

Examples

```
file <- system.file(package = "audio.vadwebrtc", "extdata", "test_wav.wav")
vad <- VAD(file, mode = "normal", milliseconds = 30)
vad$vad_segments
voiced <- is.voiced(vad, silence_min = 0.2, voiced_min = 1)
voiced
voiced <- is.voiced(vad, silence_min = 200, units = "milliseconds")
voiced
```

Description

Detect the location of active voice in audio. The Voice Activity Detection is implemented using a Gaussian Mixture Model from the "webrtc" framework. It works with .wav audio files with a sample rate of 8, 16 or 32 Khz and can be applied over a window of either 10, 20 or 30 milliseconds.

Usage

```
VAD(
  file,
  mode = c("normal", "lowbitrate", "aggressive", "veryaggressive"),
  milliseconds = 10L,
  type = "webrtc"
)
```

Arguments

file	the path to an audio file which should be a file in 16 bit with mono PCM samples (pcm_s16le codec) with a sampling rate of either 8Khz, 16KHz or 32KHz
mode	character string with the type of voice detection, either 'normal', 'lowbitrate', 'aggressive' or 'veryaggressive' where 'veryaggressive' means more silences are detected
milliseconds	integer with the number of milliseconds indicating to compute by this number of milliseconds the VAD signal. Can only be 10, 20 or 30. Defaults to 10.
type	character string with the type of VAD model. Only 'webrtc' currently.

Value

an object of class VAD which is a list with elements

- file: the path to the file
- sample_rate: the sample rate of the audio file in Hz
- channels: the number of channels in the audio - as the algorithm requires the audio to be mono this should only be 1
- samples: the number of samples in the data
- bitsPerSample: the number of bits per sample
- bytesPerSample: the number of bytes per sample
- type: the type of VAD model - currently only 'webrtc-gmm'
- mode: the provided VAD mode
- milliseconds: the provided milliseconds - either by 10, 20 or 30 ms frames
- frame_length: the frame length corresponding to the provided milliseconds

- `vad`: a `data.frame` with columns `millisecond`, `has_voice` and `vad_segment` indicating if the audio contains an active voice signal at that millisecond
- `vad_segments`: a `data.frame` with columns `vad_segment`, `start`, `end` and `has_voice` where the start/end values are in seconds
- `vad_stats`: a list with elements `n_segments`, `n_segments_has_voice`, `n_segments_has_no_voice`, `seconds_has_voice`, `seconds_has_no_voice`, `pct_has_voice` indicating the number of segments with voice and the duration of the voice/non-voice in the audio

Examples

```

file <- system.file(package = "audio.vadwebrtc", "extdata", "test_wav.wav")
vad <- VAD(file, mode = "normal", milliseconds = 30)
vad
vad <- VAD(file, mode = "lowbitrate", milliseconds = 20)
vad
vad <- VAD(file, mode = "aggressive", milliseconds = 20)
vad
vad <- VAD(file, mode = "veryaggressive", milliseconds = 20)
vad
vad <- VAD(file, mode = "normal", milliseconds = 10)
vad
vad$vad_segments

## Not run:
library(av)
x <- read_audio_bin(file)
plot(seq_along(x) / 16000, x, type = "l")
abline(v = vad$vad_segments$start, col = "red", lwd = 2)
abline(v = vad$vad_segments$end, col = "blue", lwd = 2)

##
## If you have audio which is not in mono or another sample rate
## consider using R package av to convert to the desired format
av_media_info(file)
av_audio_convert(file, output = "audio_pcm_16khz.wav",
                  format = "wav", channels = 1, sample_rate = 16000)
vad <- VAD("audio_pcm_16khz.wav", mode = "normal")

## End(Not run)

file <- system.file(package = "audio.vadwebrtc", "extdata", "leak-test.wav")
vad <- VAD(file, mode = "normal")
vad
vad$vad_segments
vad$vad_stats

```

Description

Voice Activity Detection per channel. Transforms the audio file to a wav file with the provided `sample_rate` and perform the voice activity detection per channel.

Usage

```
VAD_channel(file, sample_rate = 16000, channels = c("default", "all"), ...)
```

Arguments

<code>file</code>	the path to an audio file
<code>sample_rate</code>	integer with the <code>sample_rate</code> to convert the file to. Passed on to av_audio_convert
<code>channels</code>	character string - either 'default' or 'all' indicating to do the voice activity detection for each channel independently ('default') or for all channels independently as well as all channels together ('all')
<code>...</code>	further arguments passed on to VAD

Value

an object of class `webrtc-gmm-bychannel` which is a list with elements

- `file`: the path to the file
- `duration_secs`: seconds
- `sample_rate`: the sample rate of the audio file in Hz
- `channels`: the number of channels in the audio
- `samples`: the number of samples in the data
- `bitsPerSample`: the number of bits per sample
- `bytesPerSample`: the number of bytes per sample
- `type`: the type of VAD model - currently only 'webrtc-gmm'
- `mode`: the provided VAD mode
- `milliseconds`: the provided milliseconds - either by 10, 20 or 30 ms frames
- `frame_length`: the frame length corresponding to the provided milliseconds
- `vad_segments`: a `data.frame` with columns `channel`, `vad_segment`, `start`, `end` and `has_voice` where the start/end values are in seconds
- `vad_stats`: a list with elements `channel`, `n_segments`, `n_segments_has_voice`, `n_segments_has_no_voice`, `seconds_has_voice`, `seconds_has_no_voice`, `pct_has_voice` indicating the number of segments with voice and the duration of the voice/non-voice in the audio

Channel 0 means all audio combined in 1 channel.

Examples

```
library(audio)
library(av)
file <- system.file(package = "audio.vadwebrtc", "extdata", "stereo.mp3")
vad <- VAD_channel(file, sample_rate = 32000,
                  mode = "normal", milliseconds = 10, channels = "all")
vad
vad$vad_segments
voiced <- is.voiced(vad, channel = 0, silence_min = 0.2, voiced_min = 1)
voiced
voiced <- is.voiced(vad, channel = 1, silence_min = 0.2, voiced_min = 1)
voiced
voiced <- is.voiced(vad, channel = 2, silence_min = 0.2, voiced_min = 1)
voiced
```

Index

[av_audio_convert](#), [5](#)

[is.voiced](#), [2](#)

[VAD](#), [2](#), [3](#), [5](#)

[VAD_channel](#), [2](#), [4](#)