BRNO UNIVERSITY OF TECHNOLOGY

Faculty of Electrical Engineering
and Communication

SEMESTRAL THESIS

Brno, 2020

Roman Santa

BRNO UNIVERSITY OF TECHNOLOGY

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF TELECOMMUNICATIONS

ÚSTAV TELEKOMUNIKACÍ

NÁSTROJ PRO AUTOMATICKOU SEGMENTACI NAHRÁVEK ŘEČI

SEMESTRAL THESIS
SEMESTRÁLNÍ PRÁCE

AUTHOR
AUTOR PRÁCE
Roman Santa

SUPERVISOR
VEDOUCÍ PRÁCE
Ing. Daniel Kováč

BRNO 2020

BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

Semestral Thesis

Bachelor's study program **Telecommunication and Information Systems**

Department of Telecommunications

Student: Roman Santa ***ID:*** 187421

Year of study: 3 ***Academic year:*** 2020/21

TITLE OF THESIS:

Automatic speech recordings segmentation tool

INSTRUCTION:

The main objective of the thesis is to implement the tool in one of the programming languages, such as Python, Matlab, C ++ or C #. It will automatically create annotation data for speech recording, according to which the segmentation in the WaveSurfer editing program is performing. To detect the beginning and the end of the spoken word, your voice activity detector will be programmed and tested (eg based on speech energy, etc.). It will be further tested and compared with a detector based on Google WebRTC Voice Activity Detection. In the next step, the recognition of various sentences, words, etc. will take place, for which the method of dynamic time warping will be used along with the already segmented speech recordings that will be available. The tool will expect a WAV recording at its input and a configuration file (at discretion). It will export an annotation file in WaveSurfer format on its output. As part of the semestral project, the first part will be completed to detect the beginning and the end of the spoken word in the recording.

RECOMMENDED LITERATURE:

[1] Google WebRTC Voice Activity Detection (VAD) module [online]. [cit. 2020-09-09]. Dostupné z: https://www.mathworks.com/matlabcentral/fileexchange/78895-google-webrtc-voice-activity-detection-vad-module

[2] H.MANSOUR, Abdelmajid, Gafar ZEN ALABDEEN SALH a Khalid A. MOHAMMED. Voice Recognition using Dynamic Time Warping and Mel-Frequency Cepstral Coefficients Algorithms. International Journal of Computer Applications. 2015, 116(2), 34-41. DOI: 10.5120/20312-2362. ISSN 09758887. Dostupné také z: http://research.ijcaonline.org/volume116/number2/pxc3902362.pdf

Date of project specification: 2.10.2020 ***Deadline for submission:*** 11.12.2020

Supervisor: Ing. Daniel Kováč

prof. Ing. Jiří Mišurec, CSc.
Chair of study program board

WARNING:

The author of the Semestral Thesis claims that by creating this thesis he/she did not infringe the rights of third persons and the personal and/or property rights of third persons were not subjected to derogatory treatment. The author is fully aware of the legal consequences of an infringement of provisions as per Section 11 and following of Act No 121/2000 Coll. on copyright and rights related to copyright and on amendments to some other laws (the Copyright Act) in the wording of subsequent directives including the possible criminal consequences as resulting from provisions of Part 2, Chapter VI, Article 4 of Criminal Code 40/2009 Coll.

DECLARATION

I declare that I have written the semestral project titled "Automatic speech recordings segmentation tool" independently, under the guidance of the advisor and using exclusively the technical references and other sources of information cited in the project and listed in the comprehensive bibliography at the end of the project.

As the author I furthermore declare that, with respect to the creation of this semestral project, I have not infringed any copyright or violated anyone's personal and/or ownership rights. In this context, I am fully aware of the consequences of breaking Regulation § 11 of the Copyright Act No. 121/2000 Coll. of the Czech Republic, as amended, and of any breach of rights related to intellectual property or introduced within amendments to relevant Acts such as the Intellectual Property Act or the Criminal Code, Act No. 40/2009 Coll., Section 2, Head VI, Part 4.

Brno . . . .

author's signature

Contents

Introduction **7**

1 Speech signal pre-processing **8**

1.1 Signal Sampling . . . 8

1.2 Pre-emphasis filtering . . . 8

1.3 Segmentation . . . 9

2 Voice Activity Detectors **10**

2.1 Short-Term Energy Detector . . . 10

2.1.1 Volume Detector . . . 10

2.2 Likelihood Ratio Test . . . 10

2.3 Mel-Frequency Cepstral Coefficients . . . 11

2.4 Google WebRTC VAD . . . 12

3 Python implementation **13**

3.1 Used libraries . . . 13

3.2 VAD structure . . . 13

4 Testing **14**

4.1 Database . . . 14

4.2 Segmentation . . . 15

4.3 ROC analysis . . . 15

4.4 Errors in detectors . . . 16

5 Results **18**

6 Discussion **21**

Conclusion **22**

Bibliography **23**

List of abbreviations **24**

List of Figures

1.1 Frequency response of a filter with α=0.95 8

2.1 Cepstral coefficients 11

4.1 Wavesurfer interface 15

4.2 ROC Curve of Energy VAD 16

List of Tables

4.1 Speech tasks 14

4.2 Error type table 16

5.1 Detector quality for Czech and Hungarian recordings 18

5.2 Healthy Control CZ Male 19

5.3 Parkinson Disease CZ Male 19

5.4 Healthy Control CZ Female 19

5.5 Parkinson Disease CZ Female 19

5.6 Healthy Control HU Male 20

5.7 Parkinson Disease HU Male 20

5.8 Healthy Control HU Female 20

5.9 Parkinson Disease HU Female 20

Introduction

Human voice has become a powerful source of information in many science departments. Alongside informatics, voice analysis has advanced rapidly during last decade. A lot of new algorithms have been introduced by engineers around the world. From Short-Term Energy algorithm used for voice detection to advanced neural networks extracting various data for further analysis(e.g. emotion recognition, disease detection).

Voice Activity Detection (VAD) is a necessity in speech analysis. It detects beginning and the end of voiced parts ignoring unvoiced and noisy samples. Signal to noise ratio (SNR) plays crucial role in VAD. Heavy noised samples are hard to analyze and advanced techniques are needed to satisfy further requirements. Therefore, good microphone quality and quiet room lead to higher SNR and more satisfying results. There are plenty of VAD methods taking different approach. One of the fastest and modern nowadays is Google WebRTC VAD used for real-time communications. Short-term energy VAD is one of the most trivial but delivering good results when adapted to a specific task. Energy is a common property extracted from speech samples used in VAD. In speech recognition, our goal is to extract voice samples from sound clips as precise as possible to avoid unnecessary error in further analysis. Detector should take recording with other parameters as input and returns text file with .lab extension for further analysis with Wavesurfer application. Output file should contain time stamps in seconds of beginning and the end of spoken parts.

The purpose of this paper is to compare different detectors including Google WebRTC VAD and detector based on energy. According to results for given recordings, the most accurate will be chosen as the segmentation tool. Recordings contain speech tasks – from continuous speech to one breath word repetition.

This paper describes how speech pre-processing works as a first step before specific VAD method application. Including sampling, filtering and segmentation. Also, how detectors work and what types of detectors are used for testing. Google's detector is examined and briefly explained. Implementation of detectors is described, as well as used libraries and methodology. Testing was performed for specified database as described later in this paper. Results were analyzed with manually segmented recordings and errors were computed for each input data. In the end, best detector is chosen and further steps are suggested for better results.

1 Speech signal pre-processing

Continuous-time signal is processed to computer-like form as discrete-time values. Pre-processing tools are used to improve the results of further analysis.

1.1 Signal Sampling

Real-world signals such as speech or audio signal in general are continuous. A process of converting continuous-time signal to a discrete sequence of numbers is called **sampling**. It is a process of picking values of a signal at specific time intervals depending on sampling frequency. **Sampling frequency** used for audio signals is often 44 100 Hz but for speech signals, 16 kHz is sufficient enough. Where most speech samples are bellow 3 kHz, giving speech samples space up to 8 kHz by the **Sampling theorem** [1].

1.2 Pre-emphasis filtering

In speech processing, many analysis methods tend to process a signal based on their high intensity spectrum. In general, most of the speech energy is bellow 1 kHz so in order not to lost information on higher frequencies pre-emphasis filter comes in hand. **Pre-emphasis** is a high-pass FIR filter used to flatten the spectrum of a signal [2]. It is often used as a first step in speech analysis. Pre-emphasis raises speech energy by a variable amount increased with frequency. Transfer function of pre-emphasis filter:

$$H(z) = 1 - \alpha z^{-1} \tag{1.1}$$

where α is a coefficient usually between 0.9 and 1.

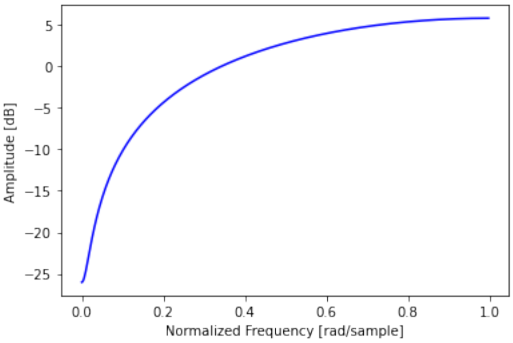

Fig. 1.1: Frequency response of a filter with α=0.95

Pre-emphasis is an **optional** pre-processing step because it can affect the result by enhancing high frequency noise. This noise can be reduced by good conditions during recording(e.g. microphone quality, silence room, etc.).

1.3 Segmentation

After sampling, there is a frequency rate, which the signal is sampled with and signal values as the result of previous step. In speech analysis, signal is often split into **segments** of defined size to evaluate behavior over short period of time. Each spoken vowel has its characteristic properties such as length and frequency. **Length** of each segment is set to 10-30ms as the average vowel length in speech is in the interval. In some cases, the segment will include signal from middle of the vowel duration, so when later analyzed, the information about the whole vowel is lost. To prevent this scenario, **overlap** is used. Segments overlap up to 50% of their size. Each segment now contains 25% of the end of previous and 25% of the beginning of the next segment. There is almost double the data used in computing using 50% overlap, so it is important to select overlap with caution [3].

To amplify original(non-overlapped) segment data, window function is applied. **Hamming window**:

$$H(\theta) = 0.54 + 0.46 \cdot cos[(2\pi N)n] \tag{1.2}$$

have sinusoidal shape. Hamming window is characteristic by not touching zero value at both ends and its good results in DTFT(Discrete-Time Fourier Transform). After multiplying every value of each segment with corresponding window value, output contains respective values to their importance in further analysis.

2 Voice Activity Detectors

Voice Activity Detectors are widely used in current digital age. From simple detector in our favorite communication application to complex detectors used in army to detect suspicious communication. In this paper our aim is to chose the best detector to extract speech from given recording.

Detectors react to change in signal and trigger beginning and end of speech part. This change can be spotted with basic signal volume, its energy, Signal-to-Noise ratio and so on. Energy is the most common feature for speech/silence detection. However this feature loses its efficiency in noisy conditions especially in lower SNRs[4].

There are also other detectors like Mel-Frequency Cepstral Coefficient detector or Likelihood Ratio Test detector, that are more complex. Google WebRTC detector is used widely on the internet, therefore it was included and looked at.

2.1 Short-Term Energy Detector

Energy detector is based on difference between total signal energy and short-term energy of each segment.

Recording is sampled and filtered with pre-emphasis. Total energy of the signal is computed as:

$$E(n) = \frac{1}{n} \cdot \sum^{n} s(n)^2 \tag{2.1}$$

where n is signal over which energy is calculated. As a percentage value of total energy, **threshold** is set. Threshold is usually between 10 to 20% of total energy, but in special cases another value is set. Then, signal is divided into segments which contains nearly stationary wavelet(1.3). Energy of each segment is computed with (2.1) and compared with threshold. If the short-term energy is bigger than threshold it indicates voiced segment.

2.1.1 Volume Detector

Similar to Energy Detector but using absolute values of the signal instead of power.

2.2 Likelihood Ratio Test

Statistical likelihood ratio test is a common used voice activity detection method, in which the likelihood ratio of the current frame is compared with adaptive threshold. Threshold is dependent on previous and current frame SNRs. Probability that current frame is speech or noise is computed from geometric mean of individual frequency band likelihood ratios[5].

2.3 Mel-Frequency Cepstral Coefficients

Extracting MFCC features from the signal includes applying pre-emphasis, segmentation, applying STFT (Short-time Fourier transform), taking the log of amplitude at preset Mel frequencies and applying DCT(Discrete Cosine transform).

A Mel is a measure unit based on human hearing. It is approximately linearly spaced bellow 1kHz and logarithmic above. Formula for converting from frequency to Mel scale is:

$$f_Mel = 1125.0 * \log(1.0 + f/700.0) \tag{2.2}$$

where f is a physical frequency in Hz and f_{Mel} is frequency in Mel scale[6].

Pre-emphasis and segmentation is trivial(1), after applying STFT to each frame, spectrum is returned. Human ear cannot recognize change in two close frequencies, so taking bunch of similar spectral energy values to sum them up and adding them to one bin. This is performed multiple times, depending on size of Mel filterbank, to cover important frequency range, that is to get an idea how much energy exists in each. This bank consists of narrow triangular filters that widens as the frequency go higher to get less concerned about varitions. Human do not hear loudness on a linear scale, so once energy is stored in bins, logarithms of them are taken. At last, beacause bins of energy are overlapping, DCT is applied to bins of energy to decorrelate them. Output of MFCC extraction are coefficients that indicates change on different frequencies. Higher coefficients represents fast change that can degrade detectors performance, so only first 10 to 15 coefficients are kept while the others are dropped[7]. In figure 2.1 there is shown presence of three speech parts visible on zero coefficients (brighter color means higher coefficient value).

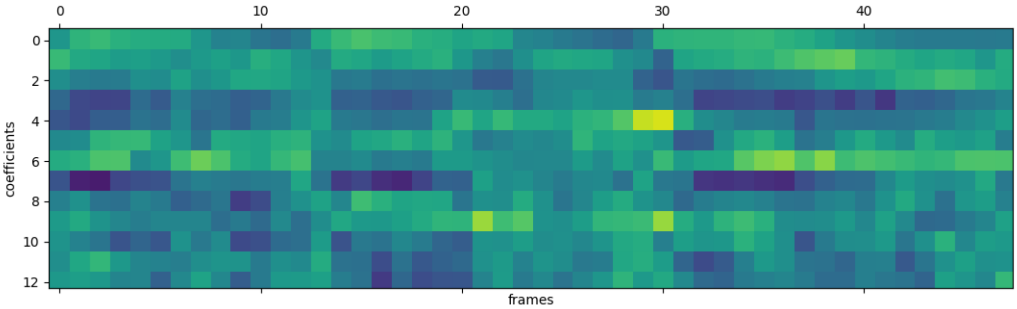

Fig. 2.1: Cepstral coefficients

2.4 Google WebRTC VAD

The VAD that Google developed for the WebRTC project is reportedly one of the best available, being fast, modern and free. It has one *mode* parameter. It represents aggressiveness, which is an integer between 0 and 3. 0 is the least aggressive about filtering out non-speech, 3 is the most aggressive.

First, VAD down-sample input signal to 8kHz to unify process for every recording. Then, uses high-pass filter to remove signal up to 80 Hz. After, it computes logarithms of energy on different frequency intervals(e.g. 80-250Hz)up to 4kHz. With total energy and energies on different intervals it calculates the probabilities for both speech and background noise using Gaussian Mixture Models (GMM). The process combines global likelihood ratio test with local tests for each frequency band. Final decision if frame is voiced is made depending on threshold values set by mode in the beginning. Mode 0 is least aggressive and mode 3 is very aggressive. Each mode has 2 different thresholds (local and global) and everyone of them contains 3 values for different frame length that needs to be set also in the beginning (10,20 or 30ms). Local threshold sets the frame voice or silence indication while main decision does the global threshold that is compared to summary of logarithms of local likelihood ratios[8].

3 Python implementation

Python version 3.8.5 was used for the implementation.

3.1 Used libraries

Various libraries were used to manage interface between OS(operating system) file system and gain access to built-in mathematical functions.

For example:

- os, pathlib, glob: managing OS interface(file access)
- numpy, scipy, math: providing mathematical functions(e.g. fft, dct) and reading wave files

3.2 VAD structure

Each VAD starts with reading wave file getting sample frequency and signal vector. For example:

```
freqRate,signal=wavfile.read(filePath+fileName+fileExt)
```

Continued with with signal processing for better detector results(1.2). From here, detector algorithm is applied. Result is if current frame is voiced or not. If it is, detector sets *VAD* variable to 1 and saves current position in seconds. When it changes back to 0 after several frame cycles, program saves time of beginning from before and current time of triggered *VAD* value change to text file with *.lab* extension as expected result in seconds. This is going for the length of sample vector. At the end all files are closed and result is stored on preset path with same `fileName` as recording but *.lab* extension for future analysis with wavesurfer application. Example of result text file:

```
0.0228212 0.1061188
0.1677361 0.2613032
0.3252027 0.4176287
0.4963620 0.5739542
0.6241609 0.7154459
0.7713579 0.8706303
0.9505047 1.0303790
```

indicating 7 speech parts in 1 second of a recording.

4 Testing

Every detector is adaptive and can be edited to specific purpose. They were tested based on their error rate automatically and manually compared in wavesurfer application.

4.1 Database

Testing speech samples were obtained from The Brain Diseases Analysis Laboratory. They contain speech tasks, which are used for automatic speech analysis of people with neurological diseases. They are about 5 minutes long and contains up to 17 tasks shown in the table bellow.

Tab. 4.1: Speech tasks

Label	**Speech task**	**Description**
TSK1	Monologue	Monolog, at least 90 s long without interruption of a clinician. The participants will be instructed to speak about their hobbies, family, job, actual date activity, etc.
TSK2	Reading	Reading ashorttext.The patient can read the text for her-/himself in advance.
TSK3	Sustained phonation	Approximately 3-s(not longer than 5 s)sustained vowel of /a/ at a comfortable pitch and loudness. Performed on one breath.
TSK4	Sustained phonation	Approximately 3-s (not longer than 5 s)sustained vowel of /i/ at a comfortable pitch and loudness. Performed on one breath.
TSK5	Sustained phonation	Approximately 3-s (not longer than 5 s)sustained vowel of /u/ at a comfortable pitch and loudness. Performed on one breath.
TSK6	Sustained phonation	Sustained phonation of /a/ at a comfortable pitch and loudness as constant and long as possible, at least 5 s. Performed on one breath.
TSK7	Diadochokinetic task	Rapid steady /pa/-/ta/-/ka/ syllables repetition as constant and long as possible, repeated at least 5 times. Performed on one breath.
TSK8-17	Polysyllable word repetition	Repeat 10 polysyllable words according to the clinician. 6 of the words should have at least 3 syllables and CVCV (C –consonant, V –vowel) structure for the first two of them.

Recordings have 48kHz sampling rate containing 32-bit float values and 768 kbps bit rate resulting in approximately 25MB size. That size is problematic in testing and could take a lot of time. This paper is focused on task 7 (labeled TSK7) – diadochokinetic task. It is rapid steady /pa/-/ta/-/ka/ syllables repetition as constant and long as possible, repeated at least 5 times. Performed on one breath. This task is cut off form original recording. TSK7 is about 15 seconds long and for detecting speech, there is no need for 48kHz sampling rate. So, TSK7 was downsampled to 16kHz with 16-bit depth and bit rate of 256kbps resulting in size of average 500kB.

Four recordings were used in Czech language, other four in Hungarian. Half of them have healthy controlled patients, other half patients with Parkinson disease. There is 50% male 50% female samples.

4.2 Segmentation

Eight samples described in previous chapter were manually labeled using spectrogram in wavesurfer application as prototypes. Wavesurfer interface is show in the picture bellow.

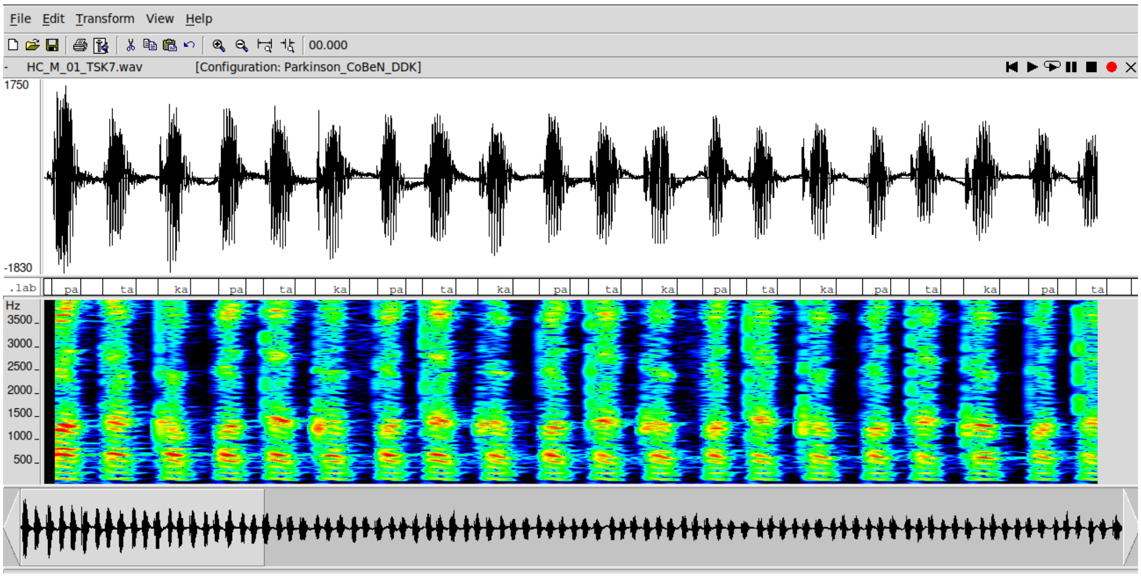
File Edit Transform View Help
00.000
HC_M_01_TSK7.wav
[Configuration: Parkinson_CoBeN_DDK]
1750
-1830
.lab
pa ta ka pa ta ka pa ta ka pa ta ka pa ta ka pa ta ka pa ta
Hz
3500
3000
2500
2000
1500
1000
500

Fig. 4.1: Wavesurfer interface

4.3 ROC analysis

Prototypes manually labeled were compared with automatic labels from detectors with 25ms accuracy. Detectors operating with threshold values were tested throughout the spectrum of available values. For example, Energy VAD with threshold from 0 up to 3 times average energy of the recording was tested. Graph in figure 4.3 shows how precise is the detector with change of threshold. It displays relation between correct speech detection and false speech detection [10].

Best threshold value with minimal error rate(e.g. threshold:0.075, error rate:6.8% for energy VAD) was picked. Then, testing continued with specific error tests counting different relevant errors(4.2).

Some detectors do not offer changing threshold as they have built up functions that have only limited use. For example, Google WebRTC VAD offers only 4 levels of speech detection aggressiveness.

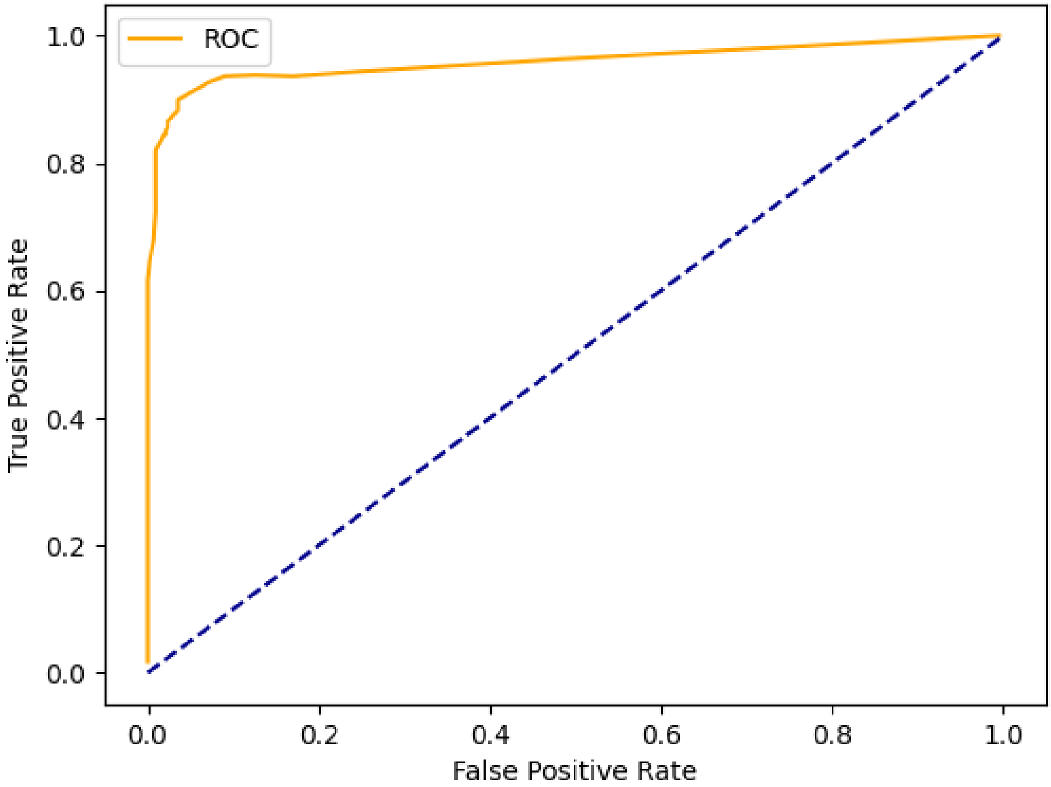

Fig. 4.2: ROC Curve of Energy VAD

4.4 Errors in detectors

For further testing, errors are specified:

Tab. 4.2: Error type table

Error Type	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Activity - 1 Inactivity - 0	**000**	**001**	**010**	**011**	**100**	**101**	**110**	**111**
VAD Decision	**010**	**011**	**000**	**001**	**110**	**111**	**100**	**101**
Name	**NDS**	**FEE**	**WC**	**FEC**	**OVER**	**WB**	**EXT**	**MSC**

Where 0s and 1s represents unvoiced or voiced frames. Error Description:

- **NDS : Noise detected as speech** — when in the original sample is noised frames but VAD detects speech
- **FEE : Front End Extension** — when VAD detects beginning of the speech before it originaly starts

- **WC : Word Clipping** — when VAD skips original speech frames and assign noise instead
- **FEC : Front End Clipping** — when VAD detects speech later than it originaly started
- **OVER: Prolongated detection of speech in noise**
- **WB : Word Blend** — when VAD blends two speech parts together but originaly they are separated with noise
- **BEC : Back End Clipping** — when VAD ends speech before it originally ended
- **MSC : Midspeech Clipping** — when VAD split one speech into two adding noise in between [11]

5 Results

Achieved VADs results are shown in tables starting next page. Each showing number and type of error described in (4.2) as Well as total error rate. Tables are sorted by language and sex starting with Czech speaking healthy male patient following with diseased to differentiate with ease. Knowing Hungarian recordings have very low SNR compared with quality of Czech recordings, table (5.1) is showing the best detector for each category. Word Blend error(4.2) is picked as most important representing quantity of false identified speech. This is occurring because in TSK7, there is rapid syllables repetition, which means quick change between voiced and unvoiced parts.

Tab. 5.1: Detector quality for Czech and Hungarian recordings

Recording	**Czech (high SNR)**		**Hungarian (low SNR)**	
key variable	**average error rate**	**average WB error**	**average error rate**	**average WB error**
Energy VAD	**10.0**	**0**	**11.4**	**19**
Volume VAD	**10.7**	**0**	**11.4**	**13.5**
MFCC VAD	**21.7**	**0.5**	**15.9**	**17.8**
LRT VAD	**12.2**	**1.3**	**15.6**	**16.3**
Google VAD	**37.2**	**2.3**	**19.0**	**18.3**

Tab. 5.2: Healthy Control CZ Male

Error Type	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Total err**
Name	**NDS**	**FEE**	**WC**	**FEC**	**OVER**	**WB**	**EXT**	**MSC**	**%**
Energy VAD	**0**	**7**	**24**	**0**	**14**	**0**	**16**	**0**	**6.8**
Volume VAD	**0**	**3**	**40**	**0**	**8**	**0**	**11**	**0**	**6.8**
MFCC VAD	**0**	**5**	**41**	**0**	**6**	**0**	**19**	**0**	**7.9**
LRT VAD	**3**	**59**	**3**	**0**	**12**	**0**	**29**	**0**	**12.3**
Google VAD	**4**	**85**	**2**	**0**	**0**	**0**	**0**	**0**	**44.8**

Tab. 5.3: Parkinson Disease CZ Male

Error Type	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Total err**
Name	**NDS**	**FEE**	**WC**	**FEC**	**OVER**	**WB**	**EXT**	**MSC**	**%**
Energy VAD	**8**	**21**	**42**	**0**	**20**	**2**	**10**	**0**	**12.7**
Volume VAD	**6**	**15**	**48**	**0**	**22**	**0**	**9**	**0**	**12.1**
MFCC VAD	**0**	**50**	**8**	**0**	**0**	**2**	**85**	**0**	**14.2**
LRT VAD	**1**	**38**	**21**	**0**	**23**	**2**	**22**	**0**	**10.1**
Google VAD	**0**	**107**	**0**	**0**	**0**	**3**	**1**	**0**	**29.4**

Tab. 5.4: Healthy Control CZ Female

Error Type	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Total err**
Name	**NDS**	**FEE**	**WC**	**FEC**	**OVER**	**WB**	**EXT**	**MSC**	**%**
Energy VAD	**10**	**17**	**39**	**0**	**23**	**0**	**34**	**11**	**14.2**
Volume VAD	**29**	**32**	**33**	**1**	**18**	**0**	**27**	**1**	**17.2**
MFCC VAD	**0**	**5**	**40**	**50**	**3**	**0**	**55**	**0**	**42.5**
LRT VAD	**8**	**68**	**10**	**0**	**24**	**3**	**25**	**2**	**15.1**
Google VAD	**57**	**27**	**62**	**3**	**7**	**6**	**9**	**8**	**33.4**

Tab. 5.5: Parkinson Disease CZ Female

Error Type	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Total err**
Name	**NDS**	**FEE**	**WC**	**FEC**	**OVER**	**WB**	**EXT**	**MSC**	**%**
Energy VAD	**1**	**12**	**16**	**0**	**20**	**0**	**10**	**0**	**6.5**
Volume VAD	**2**	**23**	**9**	**0**	**18**	**0**	**10**	**0**	**6.8**
MFCC VAD	**0**	**30**	**19**	**0**	**0**	**0**	**69**	**0**	**22.0**
LRT VAD	**4**	**54**	**1**	**0**	**19**	**0**	**15**	**0**	**11.4**
Google VAD	**62**	**18**	**36**	**0**	**1**	**0**	**0**	**0**	**41.1**

Tab. 5.6: Healthy Control HU Male

Error Type	1	2	3	4	5	6	7	8	Total err
Name	NDS	FEE	WC	FEC	OVER	WB	EXT	MSC	%
Energy VAD	2	11	1	0	1	12	0	1	16.9
Volume VAD	4	10	0	0	0	12	1	0	17.1
MFCC VAD	3	12	1	0	0	12	0	0	18.6
LRT VAD	1	13	0	0	0	12	0	0	21.3
Google VAD	3	12	0	0	0	12	0	0	19.1

Tab. 5.7: Parkinson Disease HU Male

Error Type	1	2	3	4	5	6	7	8	Total err
Name	NDS	FEE	WC	FEC	OVER	WB	EXT	MSC	%
Energy VAD	19	21	21	17	18	41	19	20	11.9
Energy VAD	5	17	18	0	12	14	17	2	11.9
Volume VAD	4	15	16	0	14	14	13	2	9.3
MFCC VAD	1	53	0	0	0	19	1	9	26.9
LRT VAD	6	36	6	0	5	16	10	0	14.9
Google VAD	8	47	3	0	0	19	0	0	21.0

Tab. 5.8: Healthy Control HU Female

Error Type	1	2	3	4	5	6	7	8	Total err
Name	NDS	FEE	WC	FEC	OVER	WB	EXT	MSC	%
Energy VAD	14	8	25	0	9	11	12	1	7.3
Volume VAD	5	22	13	0	13	11	21	3	6.7
MFCC VAD	18	71	0	0	2	17	0	0	20.5
LRT VAD	10	35	11	2	9	12	23	2	13.0
Google VAD	20	58	4	0	0	17	0	0	22.6

Tab. 5.9: Parkinson Disease HU Female

Error Type	1	2	3	4	5	6	7	8	Total err
Name	NDS	FEE	WC	FEC	OVER	WB	EXT	MSC	%
Energy VAD	2	12	23	0	12	12	7	9	9.4
Volume VAD	5	24	16	0	8	17	3	3	10.0
MFCC VAD	1	45	2	0	1	23	2	9	15.2
LRT VAD	1	48	0	0	0	25	0	0	13.2
Google VAD	1	47	0	0	0	25	0	0	13.1

6 Discussion

This paper sums up looking for ideal speech detection algorithm for given recordings. Testing achieved 10.0% average error rate and 0 average word blend errors for male and female with Energy VAD for Czech recordings (with low SNR). For Hungarian recordings Energy and Volume detectors have similar results (both with 11.4% error rate), but word blend error occurred less in Volume detector, therefore Volume performed as the best for low SNR recordings(5.1). There was no significant difference found in detection between male and female recordings.

MFCC VAD was implemented using only first cepstral coefficient; consequently, performed with significant error. Detector should be updated with use of more coefficients and tested again in the future.

Tests has shown how world-wide used detectors like Google WebRTC can have problem with specific recording where there is frequently changing speech and silence in the recording. In the future, detectors must be changeable to work with more diverse input. Expectations from Google's detector were higher but it should perform better with more generic speech recordings available for more testing. There is much work to do to analyze behaviour of the detectors on more recordings.

Conclusion

Voice detection with low enough error rate is relatively easy to implement for expected input recording. It gets harder for complex recording with more noise, unexpected stops or multiple voices. The goal was to implement speech detector that detects beginning and end of speech parts and saves them in output file for further analysis. The goals were achieved as Energy VAD performed as the best detector for TSK7 with caution of higher word blend error rate for low SNR recordings. Also, Google VAD was tested and compared with detectors for TSK7. Next step is to test available detectors for other speech tasks and implement complex speech detector for recordings of patients performing speaking tasks in order to extract each task without surrounding silence and/or noise. This is to increase efficiency in analysing differences between healthy and diseased patients. In the follow-up work, output time segments from VAD will be used for syllables recognition applying machine learning and using Dynamic Time Wrapping or other method for measuring similarities between syllables.

Bibliography

[1] SHANNON, C.E., 1949. *Communication in the Presence of Noise.* Proceedings of the IRE, 37(1), pp.10–21. Available at: `https://doi.org/10.1109/jrproc.1949.232969`.

[2] SMÉKAL, Z. *Zpracování řeči.* Brno: Vysoké učení technické v Brně, 2012. s. 1-171. ISBN: 978-80-214-4896- 4.

[3] DENG, L., O'SHAUGHNESSY, D. *Speech Processing: A Dynamic and Optimization-Oriented Approach* Signal Processing and Communications, CRC Press, 2018. ISBN: 9781482276237

[4] MOATTAR, M. H., HOMAYOUNPOUR, M. M.: *A simple but efficient real-time Voice Activity Detection algorithm*, 2009 17th European Signal Processing Conference, Glasgow, 2009, pp. 2549-2553.

[5] RAMIREZ, J., SEGURA, J. C., BENITEZ, C., GARCIA, L. and RUBIO, A.: *Statistical voice activity detection using a multiple observation likelihood ratio test* in IEEE Signal Processing Letters, vol. 12, no. 10, pp. 689-692, Oct. 2005, doi: 10.1109/LSP.2005.855551.

[6] X. HUANG, X., ACERO, A. and HON, H. *Spoken Language Processing: A guide to theory, algorithm, and system development*, Prentice Hall, 2001.

[7] On, C. K., Pandiyan, P. M., Yaacob, S. and Saudi, A.: *Mel-frequency cepstral coefficient analysis in speech recognition* 2006 International Conference on Computing & Informatics, Kuala Lumpur, 2006, pp. 1-5, doi: 10.1109/IC-OCI.2006.5276486.

[8] WISEMAN, J., *Python interface to the WebRTC Voice Activity Detector (VAD)*, Available at: `https://github.com/wiseman/py-webrtcvad`.

[9] ENQUING, D., GUIZHONG, L., YATONG, Z. and YU, C. *Voice activity detection based on short-time energy and noise spectrum adaptation*,6th International Conference on Signal Processing, 2002., Beijing, China, 2002, pp. 464-467 vol.1.

[10] FAN, J., UPADHYE, S., WORSTER, A. *Understanding receiver operating characteristic (ROC) curves* Canadian Journal of Emergency Medicine 2006, 8(1), pp. 19-20, doi:10.1017/S1481803500013336

[11] ROSCA, J., BALAN, R., FAN, N.P., BWAUGEANT, C. and GILG, V. *Multichannel voice detection in adverse environments*, Proc. ofEUSIPCO, vol. 1, pp. 251–254, Sept. 2002.

List of abbreviations

VAD	Voice Activity Detector
ROC	Receiver Operating Characteristic
SNR	Signal-to-noise ratio
FIR	Finite Impulse Response
DTFT	Discrete-Time Fourier Transform
WebRTC	Real-time communication for the web
MFCC	Mel-frequency cepstral coefficients
DCT	Discrete Cosine transform
GMM	Gaussian Mixture Model
OS	Operating System
TSK7	Diadochokinetic Task 7 1.1
Wavesurfer	WaveSurfer is an open source tool for sound visualization and manipulation