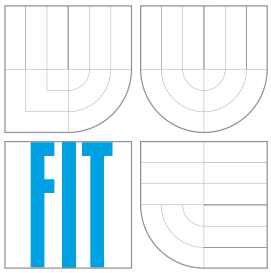
BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ



FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

MUSIC, SPEECH, CRYING, SINGING DETECTION IN AUDIO (VIDEO)

IDENTIFIKACE HUDBY, ŘEČI, KŘIKU, ZPĚVU V AUDIO (VIDEO) ZÁZNAMU

MASTER'S THESIS
DIPLOMOVÁ PRÁCE

AUTHOR
AUTOR PRÁCE

Bc. MICHAL DANKO

SUPERVISOR
VEDOUCÍ PRÁCE

Ing. IGOR SZŐKE, Ph.D.

BRNO 2016

Zadání diplomové práce/17411/2015/xdanko02

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií Akademický rok 2015/2016

Zadání diplomové práce

Řešitel: **Danko Michal, Bc.**
Obor: Informační systémy
Téma: **Identifikace hudby, řeči, křiku, zpěvu v audio (video) záznamu**
Music, Speech, Crying, Singing Detection in Audio (Video)
Kategorie: Zpracování řeči a přirozeného jazyka

Pokyny:

1. Seznamte se s přístupy pro detekci akustických událostí v audiu a také pro klasifikaci řeč / neřeč.
2. Nad dodanými daty vytvořte trénovací a evaluační sety. Implementujte základní algoritmus pro detekci řeči a vyhodnoťte ho.
3. Implementujte pokročilé algoritmy postavené na strojovém učení (například umělé neuronové sítě). Rozšiřte množinu tříd (například na hudba, zpěv, ruch ulice, střelba, ...)
4. Otestujte úspěšnost pokročilých algoritmů.
5. Zhodnoťte dosažené výsledky a navrhněte směry dalšího vývoje.
6. Vyrobte A2 plakátek a cca 30 vteřinové video prezentující výsledky vaší práce.

Literatura:

- Dle pokynů vedoucího

Při obhajobě semestrální části projektu je požadováno:

- Body 1, 2 a část bodu 3 ze zadání.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese http://www.fit.vutbr.cz/info/szz/

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci dřívějších projektů (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Szőke Igor, Ing., Ph.D.**, UPGM FIT VUT
Datum zadání: 1. listopadu 2015
Datum odevzdání: 25. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
612 66 Brno, Božetěchova 2

doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstract

This thesis follows the trend of last decades in using neural networks in order to detect speech in noisy data. The text begins with basic knowledge about discussed topics, such as audio features, machine learning and neural networks. The network parameters are examined in order to provide the most suitable background for the experiments. The main focus of the experiments is to observe the influence of various sound events on the speech detection on a small, diverse database. Where the sound events correlated to the speech proved to be the most beneficial. In addition, the accuracy of the acoustic events, previously used only as a supplement to the speech, is also a part of experimentation. The experiment of examining the extending of the datasets by more fairly distributed data shows that it doesn't guarantee an improvement. And finally, the last experiment demonstrates that the network indeed succeeded in learning how to predict voice activity in both clean and noisy data.

Abstrakt

Tato práce navazuje na trend posledních desetiletí ve využívaní neuronových sítí za účelem odhalení řeči v zašuměných datech. Text začíná základními poznatky o probíraných tématech, jako jsou audio příznaky, strojové učení a neuronové sítě. Síťové parametry jsou zkoumány s cílem poskytnout nejvhodnější zázemí pro experimenty. Hlavní úkol experimentů je sledovat vliv různých zvukových událostí na detekci řeči na malé a různorodé databáze. Přičemž se ukázalo, že nejvýhodnější jsou zvukové události v korelaci s řečí. Kromě toho, přesnost akustických událostí, dříve použita pouze jako doplněk k přesnosti řeči, je také součástí experimentování. Experiment zkoumání datových sad rozšířených o více spravedlivě rozděleny data ukázal, že samotné rozšíření nezaručuje zlepšení. Na závěr, poslední experiment demonstruje, že síti se skutečně podařilo naučit, jak předpovědět hlasové aktivity v obou případech čistých i zašuměných dat.

Keywords

Neural networks, speech, noise, acoustic event detection, voice activity detection, Theano, multi-task networks

Klíčová slova

Neurální sítě, řeč, šum, detekce zvukových událostí, detekce hlasové aktivity, Theano, víceúčelové sítě

Reference

DANKO, Michal. *Music, Speech, Crying, Singing Detection in Audio (Video).* Brno, 2016. Master's thesis. Brno University of Technology, Faculty of Information Technology. Supervisor Szőke Igor.

Music, Speech, Crying, Singing Detection in Audio (Video)

Declaration

By this I declare that I have written this master's thesis by myself under the supervision of Ing. Igor Szőke, Ph.D. All the sources that I have used for this project are listed in the reference section.

.....
Michal Danko
May 25, 2016

Acknowledgements

I would like to express my gratitude to my thesis supervisor, Ing. Igor Szőke, Ph.D., for his guidance and inspiration to learn something in addition to my specialization and to the rest of BUT Speech Recognition Group for their assistance and shared knowledge. And last, but not least to Klara for her unconditional support at the most difficult times and her assistance with the English language.

Contents

Chapter 1

Introduction

Since the invention of a computer, scientists tried to use its computing power to achieve things unimaginable before. Computers dramatically accelerated technology researches in various industries, e.g. medical, military and space industry.

However, some fields didn't seem as promising as was hoped for. Specifically, computer vision and speech recognition, which proved to be too difficult to be solved by regular rule-based systems. To achieve this, it would require to simulate the activity of human brain, specifically neural networks. There was an idea of creating an artificial neural network, but it required much higher performance than the technology back then offered.

As the time passed, the technology allowed to develop far more powerful computers every year. Therefore, a great progress in the field of speech (voice, generally) recognition thanks to the using the artificial neural networks in the last decade was noted. These achievements can improve life in many ways, such as voice remote control and speech-to-text processing.

Despite the progress, the efforts for better results still continue and many different approaches of improving the accuracy are studied. The main motivation behind this thesis is to contribute to this topic by experimenting with parallelism in the neural networks in order to better the results. Details of the objectives will be described in the following section.

The structure of this thesis is logically divided into three more chapters, besides the introduction.

The second chapter is dedicated to the theoretical background of this thesis, more specifically a basic knowledge of the detection process. This chapter describes the feature extraction, some basic division of classification methods used in machine learning with the emphasis on the neural networks and finally the post-processing of the outputs. The last section summarizes the implementation details, the programming language and tools used in this thesis.

The third chapter is the core chapter, since all the important experiments are included.

In the final chapter is the recapitulation of results and achieved objectives with the ultimate conclusion.

1.1 Aim of the thesis

The main aim of this thesis is to experiment with neural networks in the matter of the detection of acoustic events. This aim consists of several lesser steps, objectives. Which are described in the following sections.

1.1.1 Voice activity detection

The first and also primary objective is to detect the presence of human speech sequences in the given input signal with emphasis on noisy data. Voice activity detection (VAD) [11], also known as speech activity detection, is a process, which determines the presence of human speech in an input audio signal.

VAD has a wide field of use in communication, such as speech coding, speech enhancement, speech recognition or real-time VoIP applications.

The approach to this matter can differ according to the quality of the input signal. While receiving a clean input, recorded in a quiet environment, it is easier to successfully detect speech, compared to noisy record capturing a dialogue, with a city traffic acoustic events as its background. Which means, that especially in the second case, VAD is not as trivial task as it seems and most of the VAD algorithms, fail with the increased amount of noise.

1.1.2 Acoustic event detection

Next step is to extent the classification set (so far composed of the speech and the non-speech class) by additional groups of sound events. This extension includes groups correlated to the speech class, e.g. the conversational tone of speech, and on the other hand, the type of noise or music during the non-speech sequence. The aim of the acoustic event detection (AED) is to identify the sequential segments of of sound events present in audio input.

The neural network will be adjusted to be able to detect classes of each group simultaneously, which is also the second objective of this thesis.

1.1.3 Influence on VAD

If these objectives are successful, then the third objective will be to determine, how is the primary task, the speech classification, influenced by the secondary tasks, the detection of other acoustic events. And, finally, whether they improve the accuracy of the speech detection and whether it is regarding the relation between the primary and the secondary task.

1.1.4 Noisy and clean evaluation

Moreover, these experiments will be conducted on a diverse data, therefore it will be also possible to divide data into noisy and clean datasets and compare results produced from both types of data. And the last, but not least objective will be to decide whether is the speech detection successful on the noisy data.

Chapter 2

Detection process

This chapter is addressed to the process of detection in the matter of speech and acoustic events. In the last years, the speech and acoustic event detection is increasingly being used in many fields. Therefore, there exists many different methods. In the following sections, two main detection steps - feature extraction and classification, are described. The last step, although optional, is described in the last section.

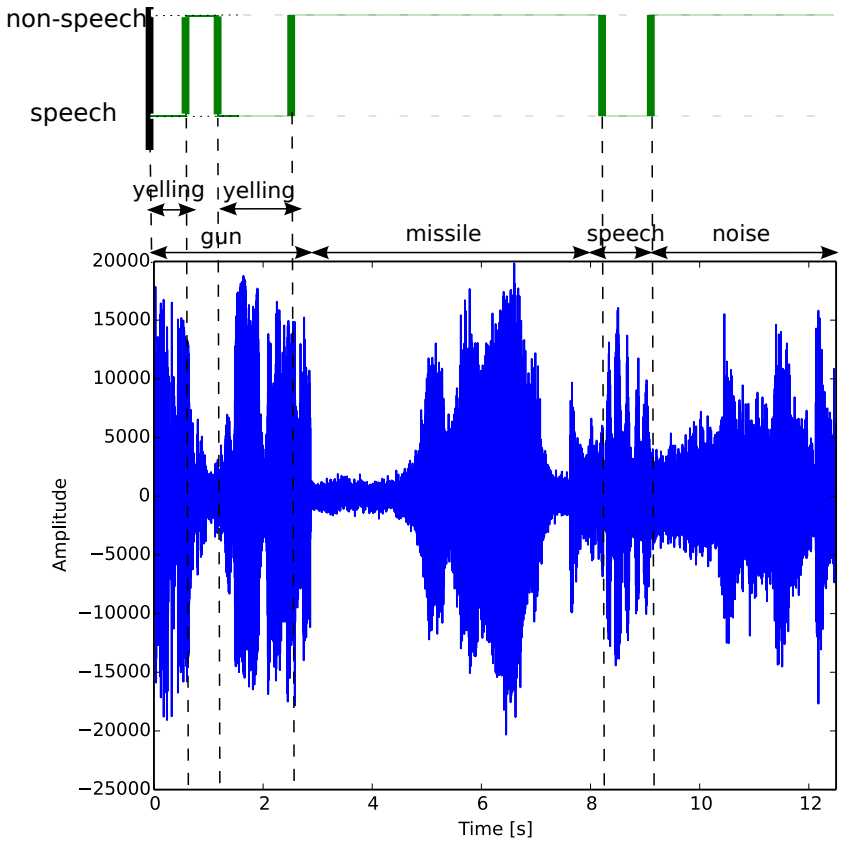


Figure 2.1: An example of the detection of speech in the input audio signal with present various background sounds.

2.1 Feature extraction

The first procedure is a feature extraction, which means acquiring the acoustic features from the audio signal, so that it can be processed by a classification algorithm.

Because of the constant changes in the audio signal, the signal is divided into frames, where presumably the vocal-tract parameters change less, compared to the whole signal. These frames are overlapping segments, generally several tens long. Next step is to compute a power spectrum of each frame. This periodogram estimate identifies which frequencies are present in the frame.

The difference between two close frequencies is hardly distinguishable, which is getting even more noticeable with higher frequencies. This is why a Mel-scale filterbank[16] is applied. Each of filters middle frequency is placed in the way, so that they follow the Mel scale. Which means that the filterbank shows different perceptual effects at different frequency bands.

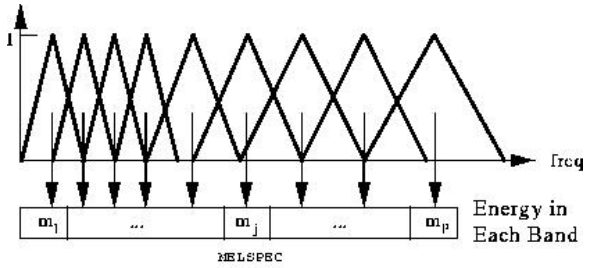


Fig. 5.3 Mel-Scale Filter Bank

Figure 2.2: Mel-scale filterbank[1].

The length of output is corresponding to the number of used filters. And after applying the logarithm on these outputs, the result is log frequency filter bank parameters (FBANK). Which is the first of two types of features widely used in speech recognition systems, which have been proven to be a good representation of speech spectral structure.

[1] http://www.ee.columbia.edu/ln/rosa/doc/HTKBook21/node54.html

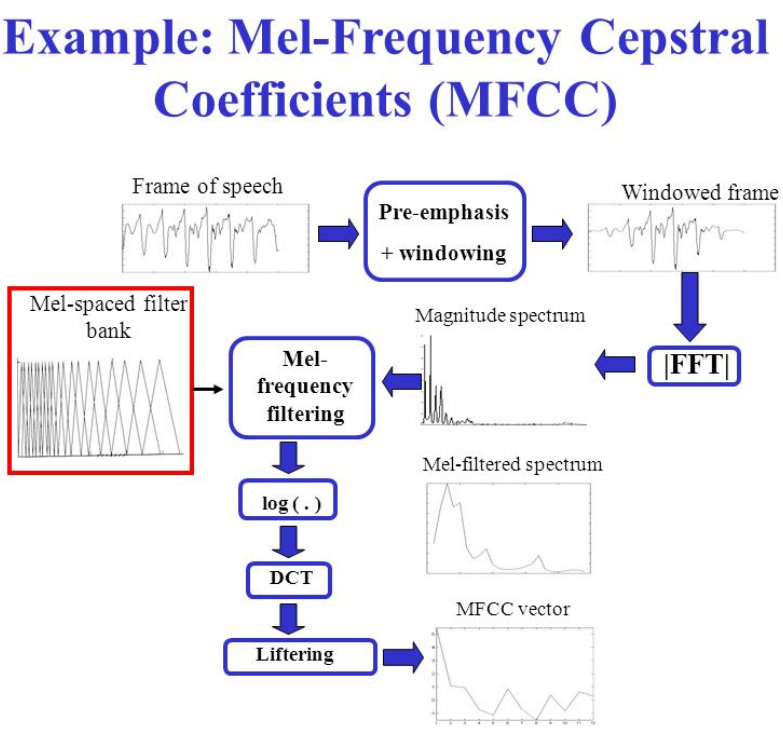


Figure 2.3: The steps of extraction of the MFCC features[2].

Although, often are desired cepstral parameters, so the next step is to calculate Mel-Frequency Cepstral Coefficients (MFCCs) by using the Discrete Cosine Transform on the log filterbank amplitudes.

FBANKs and MFCCs are the most common types of features used in speech recognition and AED.

2.2 Classification methods

Next step is the classification. There are several techniques based on different approaches [11], while they can be divided into two main groups:

- Statistical modelling
- Machine learning

Techniques of both approaches are sensitive to noise, because they are based on the learning from the training data, where it is not possible to contain all noisy scenarios.

In the following section will be shown examples of each group, however only the technique of neural networks from the machine learning group will be discussed in detail.

2.2.1 Statistical modelling

A statistical modelling system focuses on inferring the process how has been the given data collected. It uses probability functions to determine the most likely output. The most

[2] http://slideplayer.com/slide/4966500/

common model is Hidden Markov Model [18].

The Markov model is a finite state automaton consisting of states, which directly correspond to observable events, inputs and transitions with probabilities between them. However, in the Hidden Markov model, the states are abstract, separated from observations, where observations are probabilistic functions of the state. The HMM is specified by number of states, number of distinct observation symbols per state, in this case speech alphabet size. Besides that, it also includes model measures: a state transition probability distribution, an observation symbol probability distribution in specific state and an initial state probability distribution. Hidden Markov models face and answers three questions:

- Evaluation problem - calculating the probability of an observation sequence
- Decoding problem - determining an optimal corresponding state sequence to an observation sequence
- Training problem - updating the model measures to maximize a probability of an observation sequence

This HMM method can be used in speech recognition with the goal of finding the most likely model according to the speech observation sequence. Where a is state representing a specific speech unit, e.g. a word.

This method is used for experimenting in the fields of speech recognition - for example speech emotion [19] and the detection of common acoustic events in a real-life [15].

2.2.2 Machine learning

The last approach is a machine learning. As opposed to the statistical modelling, the machine learning emphasizes on how to predict possible future data, instead of studying the process which was the given data generated by.

First technique belonging to this group is support vector machine (SVM) [11]. Which is a non-probabilistic binary classifier. Its goal is constructing a hyperplane in the feature space, which maximizes the margin between classes. SVMs are also especially used as a classification technique for speech and language detection [3].

Another machine learning method is an artificial neural network, which is described in the following section 2.3. Nowadays, various types of neural networks are widely used in the applications performing the acoustic events detection. In the presence, the highest utilization is in the field of speech recognition [7], which is based on the current trend of the developing voice-user interfaces for computers, smartphones and other devices and therefore competition between the leading companies in this field. Besides, the speech enhancement is another task implemented by neural network [6].

2.3 Neural networks

This whole section is dedicated to the basic knowledge about neural networks, which are implemented as the classification algorithm used in experiments of this thesis.

2.3.1 Biological neural networks

For the beginning, it would be most suitable to start with the explanation, where does the idea of algorithm based on neural networks come from. Simply put, this concept comes (both figuratively and literally) from a human brain.

The study of artificial neural networks is based on the successfully working biological systems [12]. The reason why, is that these biological systems have several significant capabilities.

- The brain consists of numerous nerve cells called neurons, that work massively in parallel.
- The neural networks aren't explicitly programmed, they are using a learning procedure according to training samples.
- The result of this learning is a high fault tolerance against noisy signals, because of the capability to generalize and associate data, which helps to find solutions for similar problems.

Human nervous system consists of the central nervous system and the peripheral nervous system.

The peripheral nervous system consists of nerves outside brain and spinal cord. They form a network, which is throughout the whole body. The central nervous system is formed by the brain and the spinal cord. This system stores and manages all information received from senses.

As mentioned before, the base cell unit of the brain is called a neuron. The number of neurons in the human brain is approximately 10^{11} units. These neurons have connections to other neurons and their function is to send and receive nerve signals.

The direction of spreading the electrical information in the neuron starts with the dendrites. which are structures in a tree-like form (dendrite tree), branching from the neuron's cell body (soma), where they afterwards transfer the received electrical signals. Neurons receive incoming neural pulses from the other neurons with connections called the synapses located at the dendrites.

The cell nucleus is accumulating received signals until they reach a certain threshold value. Then the soma activates a electrical signal, which is transmitted to the surrounding connected neurons.

The transferring to the neurons is accomplished due to the axon. The axon is a long and thin projection of the cell nucleus, which leads to dendrites.

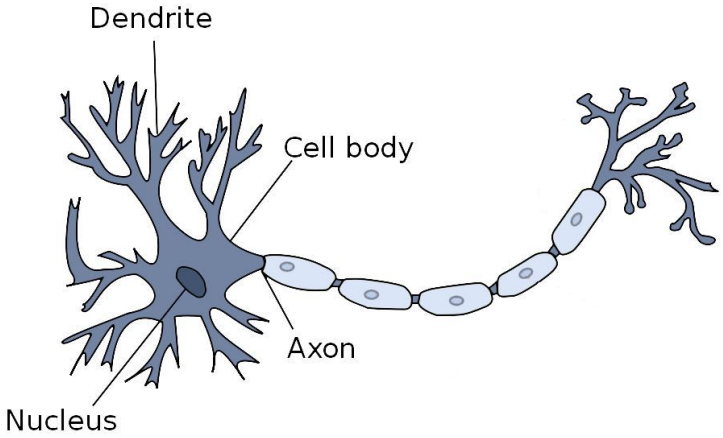


Figure 2.4: A biological neuron with the most important parts: cell body (soma), nucleus, dendrites and axon [12].

The process of learning means that the synapses' effectiveness is changed, which also changes the influence of one neuron on another.

2.3.2 Artificial neural networks

So how are the biological neural networks related to the artificial ones? An artificial neural network is a radically simplified version of the biological one. The scientists try to simulate the essential fundamentals of neurons and their connections.

Structure of neural network

The neural network consists of layers with neurons. There are three layers: input, hidden and output layer. The leftmost, input layer contains the input neurons, which receive input values. The rightmost, output layer is formed by the output neurons (in this case by a single neuron) and the hidden layer between them, consists of neurons which are neither input or output. The neural network can have multiple hidden layers.

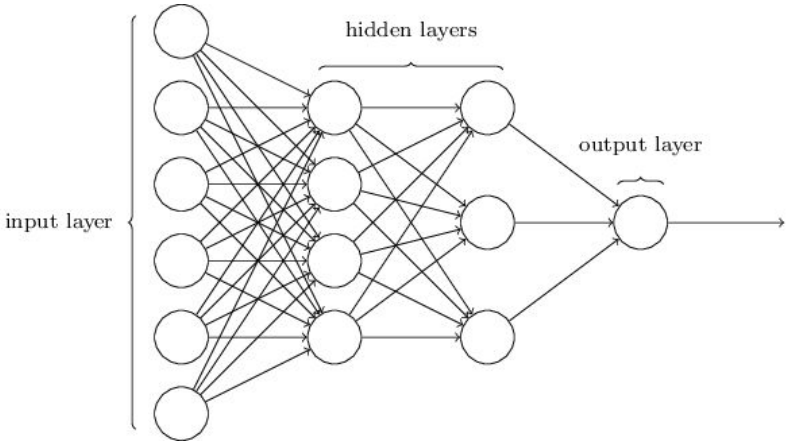


Figure 2.5: An illustration of an artificial neural network consisting of neuron layers: an input layer, two hidden layers and an output layer [17].

There are two basic types of neural network architectures based on the way how neurons interact witch each other [10]:

- Feedforward architecture - in the feedforwarding neural networks, where are connection only between the neurons from the adjacent layers, they do not keep a memory of previous outputs and states
- Feedback architecture - in the recurrent neural networks, the neuron output also depends on the previous states, the connections are between neurons from different layers and also connections in form of feedback loops

Artificial neuron

The most basic artificial neuron is called a perceptron [17]. On the input of his kind of neuron are binary inputs and on the output is a single binary output (0 or 1). The output value is determined by comparing the weighted sum of inputs to the threshold value.

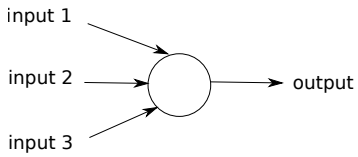


Figure 2.6: An example of perceptron with three inputs and a single output [17].

To simplify the condition of determining the output value, the threshold can be moved to the other side of inequality, forming a bias. In a relation to the biological neuron it can be imagined as a measure how easy it is to get the perceptron to activate the transmission of a signal. In the technological view of point, it is a measure how easy it is that perceptron outputs a 1. This conditioning is called a step function with the output y computed [17]:

$$\mathrm{y} = \begin{cases} 0 & w \cdot x + b \leq 0 \\ 1 & w \cdot x + b > 0 \end{cases} \tag{2.1}$$

where x is input and w, b means weight, respectively bias.

The process of learning is to changing the weights or biases to improve the accuracy of classification. The problem of perceptrons is that when the change of weights and bias flips the value of the certain output, it may change the behaviour for the rest of outputs.

The modified version of the perceptron is called a sigmoid neuron. This kind of neuron can input and output real number values belonging to $< 0, 1 >$. This means, that the output can have more different values. Therefore, the alteration of weights and bias doesn't cause such a massive difference in changing the output, like when flipping between binary values.

In the contrast with the perceptron's step function, the sigmoid neuron uses a sigmoid function σ [17]:

$$\sigma(z) \equiv \frac{1}{1 + e^{-z}} \tag{2.2}$$

Both step and sigmoid functions determine output with taking to account inputs, weights and bias, what is generally called the activation function.

The difference between these two functions can be easily understood by comparing their graphs (Figure 2.7). It can be seen, that the shape of the sigmoid function is a smoothed version of the step function. Which is what is really important, because thanks to this smoothness the relation between the changes of the weights and bias and the changes of output is more adequate.

The learning process

How does the neural network learning process work? It learns from a given input dataset called a training dataset [17].

The task is to find the right weights and bias, which would allow the neural network to successfully approximate the outputs. Therefore the training is the process of approximating the most suitable weights and bias. To evaluate the fitness of current weights and bias, there is defined a quadratic cost function:

$$C(w, b) \equiv \frac{1}{2n} \sum_{x} \|y(x) - a\|^2. \tag{2.3}$$

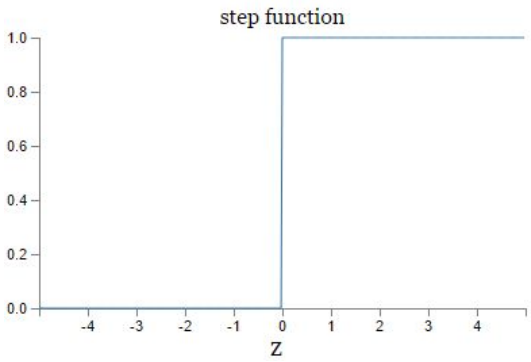


Figure 2.7: Graphs of a step function

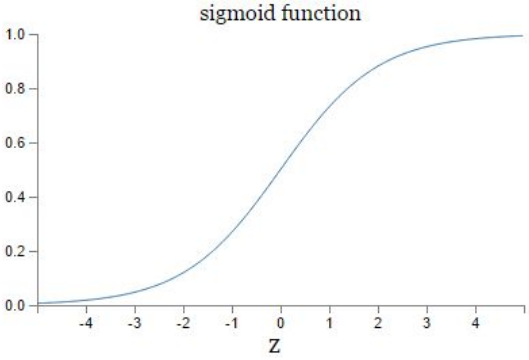


Figure 2.8: Graphs of a sigmoid function [17].

where, w denotes the collection of all weights in the network, b all the biases, n is the total number of training inputs, a is the vector of outputs from the network when x is input, and the sum is over all training inputs.

The desired situation is when the neural network outputs are approximately equal to the desired training outputs. In this case the cost approximately equals a zero. To find a set of weights and bias that result in the most possible cost, it is needed to find a minimum of a function with a large number of parameters. To do that, there is an algorithm called a gradient descent [13].

This algorithm starts with a randomly initialized set of parameter values and iteratively updates these parameters, getting closer to the values, which minimize the function. The nature of this update is computed by iterating in the opposite direction of the gradient. To make the gradient descent work correctly, it is required to set correctly a positive parameter called a learning rate.

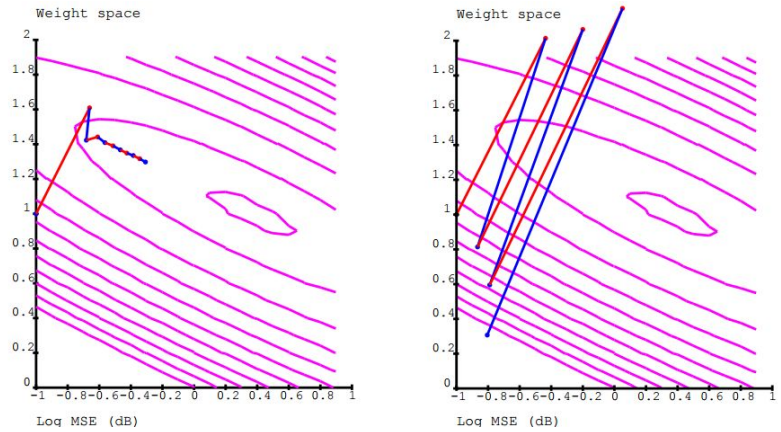


Figure 2.9: An example of a gradient descent in order to adequately (left) and too high (right) set learning rate [13].

For a fast computing of gradients, there is an algorithm known as a backpropagation [13].

2.3.3 Multi-task neural networks

Multi-task learning [4], is a mechanism, which purpose is to enhance the generalization performance. It is achieved by training all tasks simultaneously while they share network's representation. More specifically, it uses shared hidden layers trained in parallel.

The cost is calculated for each task separately. The gradient is computed from all costs of all tasks and then backpropagated through the nodes of tasks. Therefore the correlation between tasks improves their learning ability.

The main idea of multi-task networks is demonstrated on comparison of the Figure 3.8, which illustrates the intention of using the same input for different tasks and the Figure 3.9 with the structure of an actual multi-task network.

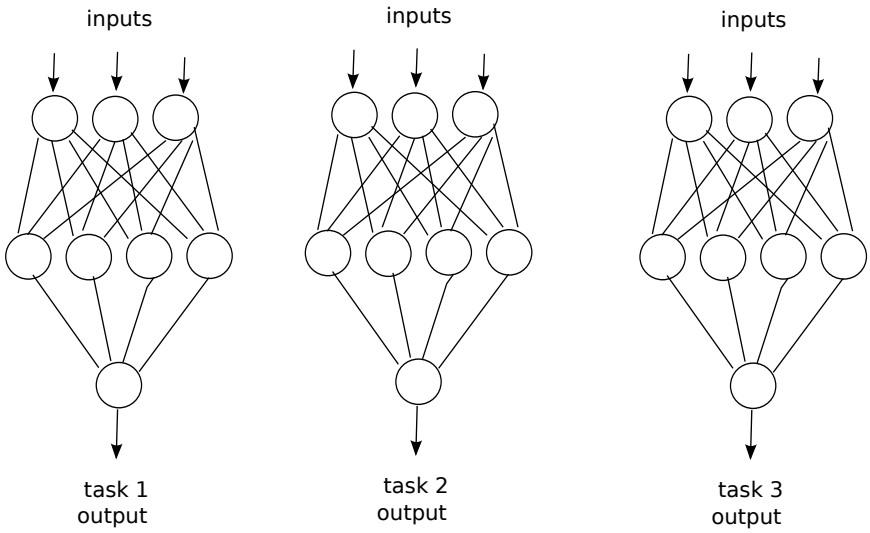


Figure 2.10: An illustration of single task neural networks with the same inputs.

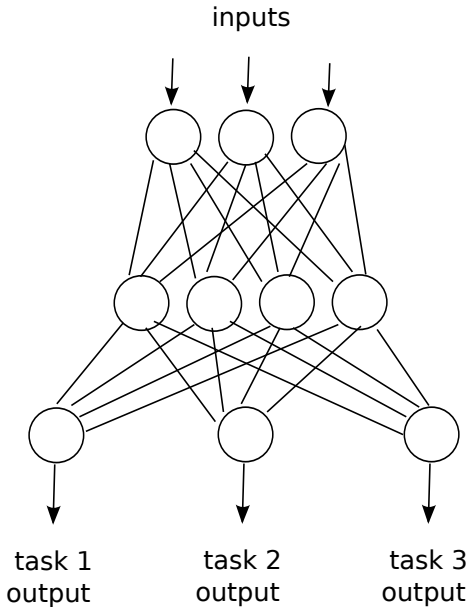


Figure 2.11: An illustration of a multi-task network with the same input for every task.

One of the recent works regarding multi-task networks [9] shows that in the speech recognition, the multi-task network with supportive models of broad acoustic units outperforms the conventional network by decreasing the error rate up to 10.7%. Another research [14] proposes a method of robust voice activity detection under non-stationary noises, which is an important problem, because the most o systems doesn't work accurately with noises from the real-life environment. However, not only the speech detection, but also the acoustic event detection is topic of research. The paper [2] examining the polyphonic detection of overlapping sound events from a real-life recordings shows overall frame accuracy 63.8% and an 19% improvement compared to the result of the system using HMM.

2.4 Post-processing

The network output is a matrix of posterior probabilities, therefore a post-processing of outputs needs to be done necessarily.

This problem can be solved by decoder implementing Viterbi algorithm [8]. The input is the vector of logarithm probabilities of classes. One specific path maximizes the probability of reaching the desired state. The probability is influenced by a parameter called insertion penalty, which is added to the current accumulated value in the case of transition between two states. And the optimal state sequence is formed by states along this path. This algorithm saves the calculation time expense when finding this particular path, because if several paths converge at a specific state at the time, for calculating the next step (from this state to the following one), it continues with calculation only with the most likely path, because it is sure, that there is only one the most likely path for each state at the time.

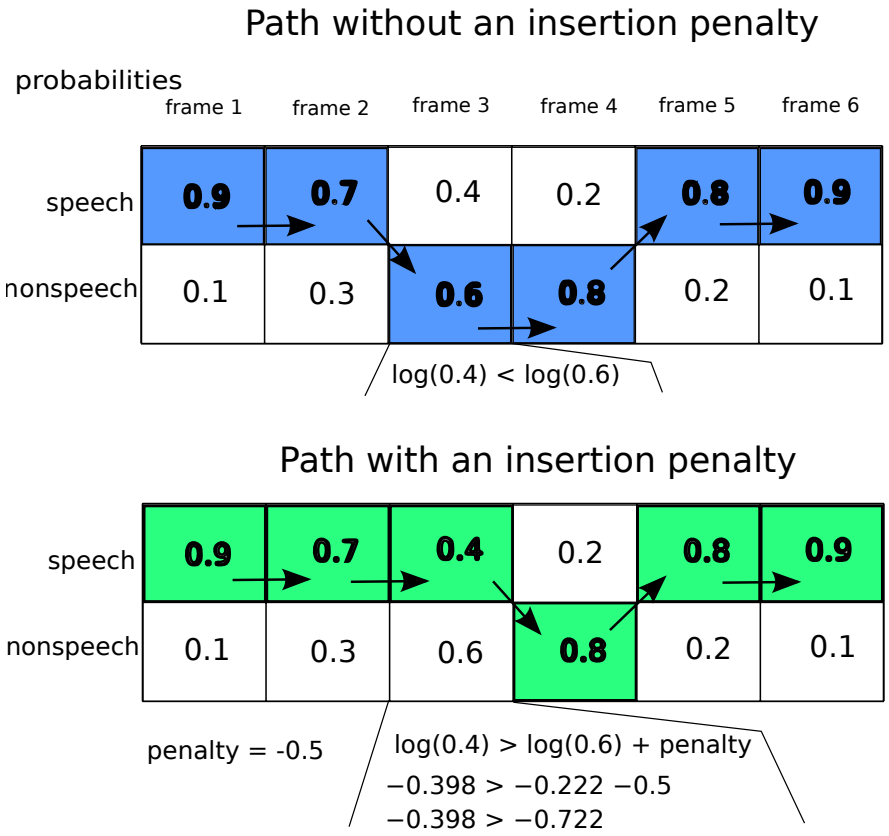


Figure 2.12: An illustration of Viterbi decoder - finding an optimal path with choosing only the most likely option, first case without any penalty, the second with its value -0.5.

The output of the neural network is vector of classes predicted for each frame (10 ms). These outputs might include sequences with the length of only few frames, for example music with the duration of 0.05 seconds. However, in the recordings of the real world situations acoustic events doesn't last this shortly and therefore in the case of a such prediction, it is more likely that it is an incorrect output than a successful prediction of such a short sequence.

The manipulation with the insertion penalty value helps with the elimination with such outputs. Lowering this penalty (to negative values) means, that there will be less surplus transitions, but more missing transitions according to the labels.

2.5 Implementation

All the algorithms used for this term project, including data preprocessing, feature extraction, neural network and evaluation, are written in Python language. These algorithms are using mathematical tool-kits and libraries.

Specifically speaking, tools Sound eXchange[3] (SoX) and FFmpeg[4] were used for the conversion and resampling to the desired audio format. Moreover, HTK Speech Recognition Toolkit [20], which was used for manipulation with audio files, feature files and labels in

[3] http://sox.sourceforge.net/
[4] https://ffmpeg.org/

Python. Another one was the Theano [1] library, providing tools necessary for neural networks. This library interacts with other two used packages, NumPy[5] and SciPy[6], which are package offering multi-dimensional arrays, resp. library for scientific computing.

The implementation is based on algorithms and libraries created and provided by BUT Speech Processing Group[7].

[5] http://www.numpy.org/
[6] https://www.scipy.org/
[7] http://speech.fit.vutbr.cz/

Chapter 3

Experimentation

This is the core chapter of the thesis. It describes the performed experiments, the evaluation of results and conclusions. As mentioned before, there were used two approaches differing in the type of used neural network. Whereas a simple network was sufficient for the first approach, the advanced approach required a multi-task network. All experiments were used on two different databases.

3.1 Databases

The following sections describe datasets used in experiments. Source audio/video files with corresponding annotation files were gathered, annotated and provided for experimental purposes by BUT Speech Processing Group. From these files were created two databases, used later in the experiments.

3.1.1 Diverse Audio Database

The purpose of the experiments on this database is to train the network to detect the speech and later other acoustic events especially on the data with high amount of noise. The first database originally consists of 169 audio files with each recording lasting 180 seconds and with overall duration of 5 hours and 45 minutes.

Description

This database has a great variety of recordings with different acoustic events. The majority of recordings are news reports and amateur footages capturing warfare in the Middle East. The language spoken in these recordings varies as well, starting with Asian (mostly Arabic), continuing with several European (e.g. English, French, Dutch) and ending with African dialects. These footages are rich in both stationary (e.g. babbling) and non-stationary noises (gunfire, explosions, etc.) and also greatly differ in the audio quality. In addition, there are occasional music sequences with or without singing. The database also includes samples from documentary films with the presence of animal and nature sounds. This database is rather small, but very diverse in the matter of acoustic events.

The desired 16-bit, 8kHz mono waveform files were extracted from the source audio/video files in AVI format.

Table 3.1: Table with enumeration and description of all Diverse Audio Database sound events and characteristics, which can possible occur during the classification process.

Group	Class	Description
speech	speech	The presence of a human voice (not singing).
	non-speech	The absence of a human voice.
speech type	monolog	Monologue voice tone.
	conversational	Voice tone suggesting an ongoing conversation.
	emotional	An expressive tone (e.g. yelling).
	crosstalk	Presence of multiple speakers at once.
	none	The absence of speech.
music	music	The presence of music (without singing).
	song	The presence of music with singing.
	none	The absence of any sort of music.
microphone	close	Speaker is at a close distance .
	distant	Speaker is located more distantly.
	telephone	Speech on telephone.
	none	The absence of speech.
non-stationary noise	vehicle	Sounds of moving cars, helicopters, etc.
	gun	The sound of a gunfire including explosions.
	animal	The natural sounds of animals.
	human	Human-noise (e.g. coughing).
	other	Sounds not belonging to any of previous classes.
	none	The absence of non-stationary noise.
stationary noise	nature	The sounds of nature.
	babbling	The indistinguishable human voice.
	other	Other types of distracting noise.
	none	The absence of stationary noise.

Classes

The database's annotation files provides 10 different characteristics. Three of them are discarded. The stationary noise level characteristics isn't related to the primary task, speech detection, therefore it isn't beneficial for the purpose of the thesis. The remaining two characteristics are on the other hand related to the speech, the age and the gender of speaker is rather useless in VAD. The class groups chosen for the classification are in the following table:

Notably, the structure of the network's output layer depends on the concept of these groups, therefore it should be chosen wisely. Of course, classes within the same group are mutually excluded, although groups are not. According to the fundamentals of multi-task networks, there must be active one class of every group in the same frame.

Therefore the last class of each group is intended to signalize the absence of the sound event (this class is generally called „none“, with the exception of the speech group, where this class is named „non-speech“). Also, the speech type group is a subgroup of the speech one, which implies that the last class is equivalent to the non-speech class and the remaining classes are subsets of the speech class. Of course, classes within the same group are mutually excluded, however groups are not,

Only the first class group (the speech and non-speech detection) is evaluated in the

simple approach, the advanced focuses on all six groups.

The annotation files, which were provided with the source audio files, were processed into transcription files called Master label files (MLFs), each corresponding to a specific class group.

The purpose of this file is to carry the expected correct classes outputs (referred as labels) according to the audio segment defined by a start and an end frame, which is used during the training and testing process.

The structure of MLF starts with MLF tag o the first line. Afterwards, there are file names of the recording labels in the each line with corresponding segment dictionary, ended by a dot mark. A dictionary segment has three parts: on the left a starting frame in 100 nanoseconds, in the middle an ending frame in 100 nanoseconds and a class assigned to this segment.

speech

```
#!MLF!#
"*/fc9480e70de1ca..._001.lab"
0 9400000 nonspeech
9400000 15600000 speech
15600000 26900000 nonspeech
26900000 38300000 speech
38300000 56800000 nonspeech
56800000 64800000 speech
64800000 64950000 nonspeech
64950000 84650000 speech
84650000 84850000 nonspeech
...
```

music

```
#!MLF!#
"*/fc9480e70de1ca..._001.lab"
0 144700000 none
144700000 236700000 music
236700000 781250000 none
781250000 868520000 music
868520000 1223000000 none
1223000000 1259480000 music
1259480000 1800000000 none
.
"*/39b76b20bafa2b..._003.lab"
...
```

non-st noise

```
#!MLF!#
"*/fc9480e70de1ca..._001.lab"
0 5900000 gun
5900000 15800000 none
15800000 21800000 gun
21800000 42900000 none
42900000 50000000 gun
50000000 97400000 none
97400000 106100000 gun
106100000 121050000 none
121050000 131350000 gun
...
```

Figure 3.1: Examples of MLF of the speech, music and non-stationary noise groups in following format: a MLF tag, a filename, a dictionary (a start frame, an end frame and a class), an end of file tag.

Datasets

The database has to be divided into three datasets, due to the process of classification. The variety of Diverse Audio Database data is not caused by the data variety within individual recordings, but by including diverse recordings in the database. For example, in one recording there might be 10 000 frames of a specific class, however in the following five recordings there is non of them. As a result the database suffers from an unbalanced data distribution. This is a problem for the neural network learning process, because it inflicts over-fitting to the class with the major probability in the dataset.

The best solution would be to acquire more balanced data. On the other hand, the easiest solution is an over-sampling (duplicating suitable recordings) and an under-sampling (removing of unsuitable recordings) [5]. However, the first method cannot be used because of the multiple tasks. More specifically, where the addition of one recording would improve classification of one task, it would also diminish the classification of other tasks. The under-sampling method is performed in order to balance the data distribution and prevent this over-fitting phenomenon. The under-sampling needs to be applied accordingly to every speech group, therefore the number of suitable recordings is significantly lower as can be seen in the Table 3.2. Which is also a potential problem for the learning process, as it might stop before learning anything useful.

Table 3.2: Comparison between the original and processed data duration

Class	Full-size data duration	Under-sampled data duration
all	8h 27m	3h 15m
speech	6h 04m	1h 43m
music	2h 12m	1h 18m
stationary noise	2h 05m	1h 46m

For this reason, one of the experiments will be performed regarding whether the under-sampling actually helps and which datasets to use in the remaining experiments.

Three datasets are created from the full-size data: a training set (105 files), a cross-validation set (30 files) and a testing set (34 files).

Similarly, the under-sampled data is also divided into three sets: a training dataset of 40 recordings, a cross-validation dataset of 12 recordings and a testing dataset of 13 recordings.

Notably, the recordings in the testing datasets were intentionally chosen in the way that they contain both extreme (with absence, resp. rich presence of classes) and average samples for every class group.

Table 3.3: Comparison between the VAD experiment results performed on the full-size and under-sampled training data. The original-size testing set was used for the evaluation of both experiments, because of the higher number of recordings.

Data	FAcc	Hit rate	BAcc
full-size	88.89%	28.51%	10.42%
under-sampled	89.81%	35.01%	17.03%

The results from the Table 3.3 suggest that the under-sampled datasets are indeed more suitable, despite their smaller size. One more experiment will be conducted in order to either confirm or disprove this statement. These results will be from the network using multi-tasking.

Table 3.4: Comparison between the multi-task experiment results performed on the full-size and under-sampled training data. The original-size testing set was used for the evaluation of both experiments, because of the higher number of recordings.

Data	Group	FAcc	Hit rate	BAcc
full-size	baseline	87.82%	25.78%	6.51%
	type	76.02%	26.54%	8.45%
	music	65.86%	26.60%	-129.26%
	microphone	81.98%	28.37%	9.30%
	non-st noise	81.08%	8.76%	3.85%
	st noise	64.63%	13.24%	13.24%
under-sampled	baseline	89.07%	35.29%	17.70%
	type	72.51%	31.89%	5.38%
	music	64.77%	26.60%	-143.09%
	microphone	79.33%	40.42%	14.76%
	non-st noise	80.21%	8.11%	4.42%
	st noise	64.63%	13.24%	13.24%

The comparison shown in the Table 3.3 confirms the previous statement. The under-sampled datasets are more suitable for the upcoming experiments than the original-sized datasets, thus they will be actually used. Only exception is the testing dataset, which will be used from the original-sized, because it contains not only all of the recordings from the down-sampled training set, but also additional 21 unique recordings. Which will provide a more precise evaluation, because the class distribution of set doesn't matter any more in the testing phase.

3.1.2 Clean Radio Database

The second database will be used as a supplement to the noisy data the first database in order to watch how does extending the dataset by more clean data alters the results. Clean Radio Database is formed by 72 audio files with the duration approximately one hour, together giving 71 hours and 41 minutes of recordings.

Description

This database consists of radio broadcast recordings in English, Arabic, Cuba Spanish, Asian and African languages. Which include interviews, reports, songs and music. The amount of noise in this database is significantly lower compared to the Diverse Audio Database, since the recordings were taken from clear radio environment and the major sound events are music and speech sequences.

The motivation of introducing this database with clean data is that even with the VAD in noisy data being the main objective of this thesis, adding fairly distributed clean speech data might prove beneficial as a supplement for the first database datasets.

The audio files were converted from RAW format to 16-bit monophonic audio files in waveform format with 8kHz sample rate.

Classes

Annotation files of this database don't include details about the type of speech and there wasn't present any noticeable non-stationary noise during the radio broadcast, therefore there are five class groups in overall. It also differs in the structure of the group for stationary noise, because the annotation files don't differentiate between the types of this noise, which results in detection either presence or absence of the stationary noise. Therefore, these class groups can be classified in Clean Radio Database:

Datasets

In the contrast with Diverse Audio Database, this database has a balanced data distribution, therefore adjusting datasets is not necessary.

This database has a low amount of noise and additionally has a fair distribution of the speech and the music. Also, its size is approximately nine times larger than the size of Diverse Audio Database. It can be presumed, that only a portion of data will be needed to improve the training capability of Diverse Audio Database from the effectiveness perspective without any drastic reduction of accuracy.

Table 3.5: Table containing acoustic events and characteristics used in the classification process of Clean Radio Database data, as they are divided into groups and their class, also with their description.

Group	Class	Description
speech	speech	The presence of a human voice (not singing).
	non-speech	The absence of a human voice (including singing).
speech type	monolog	Monologue voice tone.
	conversational	Voice tone suggesting an ongoing conversation.
	emotional	An expressive tone (e.g. yelling).
	crosstalk	Presence of multiple speakers at once.
	none	4The absence of speech.
music	music	The presence of music (without singing).
	song	The presence of music with singing.
	none	The absence of any sort of music.
microphone	close	Speaker is at a close distance .
	distant	Speaker is located more distantly.
	telephone	Speech on telephone.
	none	The absence of speech.
st noise	stationary noise	The presence of stationary noise.
	none	The absence of stationary noise.

Table 3.6: Comparison between the original and processed data duration

Class	Full-size data	Reduced data
all	71h 41m	11h 56m
speech	32h 43m	5h 50m
music	33h 22m	6h 28m
stationary noise	4h 53m	1h 03m

Thus, as can be seen on the Table 3.6, the data distribution is persevered in the reduced database and therefore it may suffice as a supplement for the datasets from Database. This statement is the task of the following experiment, in which the database of the original size is split into three sets of the following proportion of a training, a cross-validation and a testing set: 40, 15 and 17 and the data of the reduced size: 8, 2, 2, both in the same way as in the section 3.1.1.

Table 3.7: Comparison between the VAD experiment results performed on the full-size and reduced training data. The original-size testing set was used for the evaluation of both experiments, because of the higher number of recordings.

Data	FAcc	Hit rate	BAcc
full-size	88.89%	28.51%	10.42%
under-sampled	89.81%	35.01%	17.03%

This database does not include a great variety of acoustic events, but provides a better data distribution for the speech and music groups. This extension should improve the speech and music detection, albeit lower the accuracy of the noise detection.

3.1.3 Used datasets

In summary, the databases, which will be used for the upcoming experiments, are:

- The under-sampled noisy and diverse Diverse Audio Database (with overall duration 3h 15m), where the training dataset of 40 recordings, the cross-validation dataset of 12 recordings and the testing dataset of 13 recordings will be used as a primary experiment data.
- The reduced-sized clean and fairly-distributed Clean Radio Database (with overall duration 11h 56m), where the training dataset of 8 recordings, the cross-validation dataset of 2 recordings and the testing dataset of 2 recordings will be used as a supplement data in order to improve results.

Table 3.8: Duration of the database versions used in the experiments - under-sampled Diverse Audio Database and reduced Clean Radio Database.

Database	Group	Duration
	all	3h 15m
Diverse Audio Database	speech	1h 43m
	music	1h 18m
	stationary noise	1h 46m
	all	11h 56m
Clean Radio Database	speech	5h 50m
	music	6h 28m
	stationary noise	1h 03m

3.2 Metrics

There are totally three metrics used during the experimentation phase. First is frame-based and the remaining two are based on segments.

The first metric used for evaluation is a frame accuracy (FAcc). It simply compares the outputs of the network to the labels and represents the percentage of correctly predicted classes on all frames.

The next step is a process called segmentation, which means creating sequences of classes from the output in a vector, where every class instance represents a prediction made from a single frame.

The second metric is a boundary accuracy (BAcc). The boundaries between sequences created by the segmentation process both from predicted outputs and labels are organized into an alignment of pair of boundaries. They are organized in a way that in the case the distance between them is lower than a threshold, they are paired, otherwise they are marked as unpaired. Afterwards, the algorithm counts the number of pairs with the same classes (known as correct hits) *hit*, substitutions of classes *sub*, redundant insertions of the predicted boundaries *ins* and label boundaries without a pair, marked as deletions *del*. The BAcc (%) is defined as:

$$BAcc \equiv 100 \cdot (1 - \frac{ins + del + sub}{hit + del + sub}) \equiv 100 \cdot \frac{hit - ins}{hit + del + sub} \tag{3.1}$$

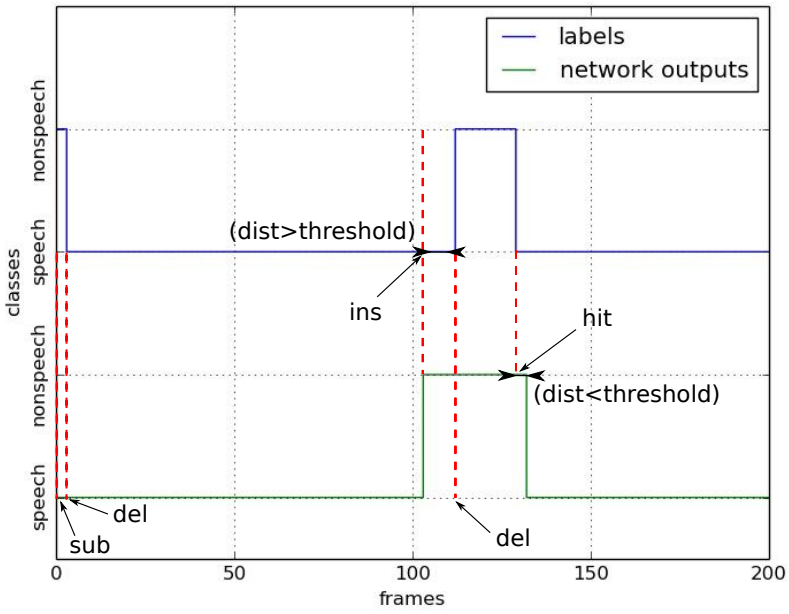


Figure 3.2: An illustration of insertions, a hit, a deletion and a substitution instances during the calculation of BAcc.

Notably, this formula calculates the accuracy with respect to the total number of referenced sequence boundaries, therefore the results might be below 0.0%. Which would mean, that the number of insertions is higher than number of hits. The BAcc results from a successfully learnt neural network are supposed to be positive values, otherwise negative.

Also, as can be seen in Figure 3.2 between 100th and 150th frame, if one output boundary is being misplaced, while having one corresponding label boundary, it is counted both as an insertion and as a deletion this implementation of the BAcc. Therefore the error is markedly higher than in the BAcc variant counting it as only one miss.

And the last used metric is a hit rate. Which is the ratio between the correctly predicted segment boundaries and the all predicted segment boundaries. The hit rate is defined accordingly to the expressions established in the second metric:

$$hitrate \equiv 100 \cdot \frac{hit}{hit + del + sub} \tag{3.2}$$

The goal is to correctly detect the whole segments, therefore the most important is the BAcc metric. Moreover, even high accuracy doesn't ensure the absence of over-fitting or correctly predicted segments and the hit rate of boundaries is a component of the more complex BER. Thus, all the experiment decisions will be made with respect to the BAcc.

3.3 The simple approach

The first objective is to experiment with a neural network as an implementation of VAD algorithm. A simple network with two output nodes is used in this approach. It learns how to predict only one of two classes (speech or non-speech) belonging to the same VAD class

group. The experiments will be performed on both databases described before. These are the baseline experiments for the further experimenting.

This section starts with the experiment of the VAD on this simple neural network. The second experiment will be the VAD on the extended datasets created from Diverse Audio Database and Clean Radio Database all together. And the last experiment will test the influence of the network parameter values. Finally, the results with the conclusion will be discussed in the last subsection.

Notably, the alignment pairing threshold and the Viterbi insertion penalty is implicitly set to 20, respectively -20.0, unless stated otherwise.

The features used as input to the network's first layer were extracted from the input signal, by using 23 Mel-scale filterbank, where the frame window length was 25 ms and with 10 ms overlapping and by applying logarithm function afterwards. Finally, these features (in HTK known as FBANK), now as network's inputs, are pre-processed by applying Hamming DCT of 16 basis functions, which results in input vectors with size 368.

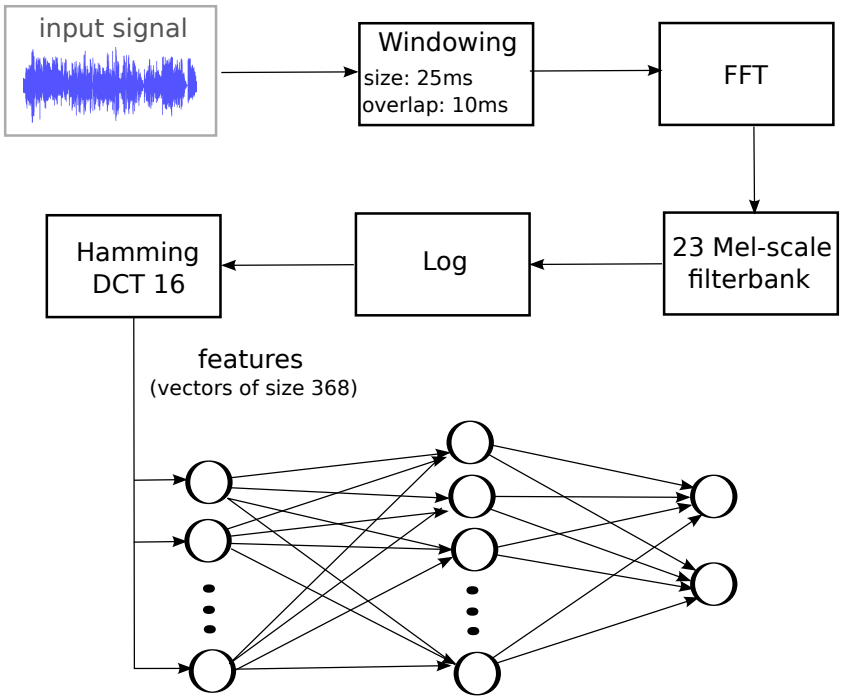


Figure 3.3: The process of the feature extraction.

The structure of neural network used in this experiment is composed of input vectors length 368, as stated above, one hidden layer consisting of 500 units and 2 output units consisting of two classes - speech and non-speech.

3.3.1 Diverse Audio Database voice activity detection experiment

The first experiment is designed to show the efficiency of the neural network in the detection of speech and non-speech sequences. With the Diverse Audio Database being rather small and noisy, the results will be later compared to the results of the second larger database.

With the network parameters set, the network's learning process ended after 11 iterations.

Table 3.9: Frame accuracy and BAcc of the first experiment - VAD on the Diverse Audio Database

FAcc	Hit rate	BAcc
89.81%	35.01%	17.03%

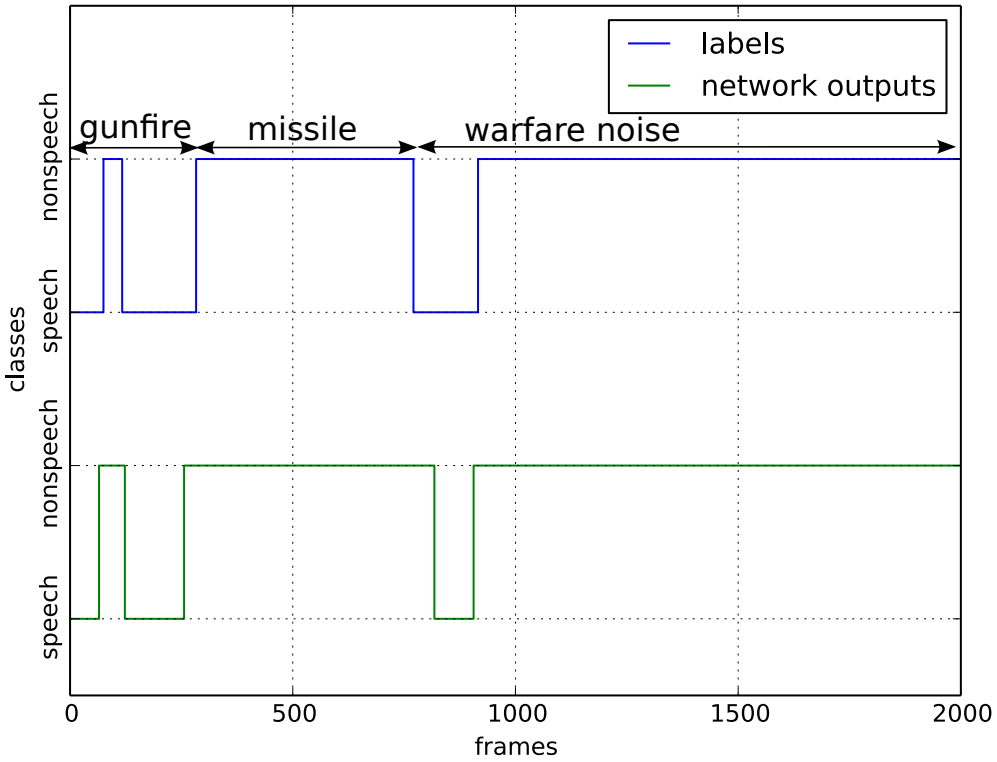


Figure 3.4: An example of comparison between labels and network outputs of the speech detection in a 20 seconds long passage from a warfare footage. This recording includes gunfire, launch of missile and more warfare noise. The results for this input are: FAcc 89.57%, hit rate 8.11% and BAcc -78.38%.

According to the results, which can be seen in Table 3.9 (FAcc 89.81% and BAcc being positive) and an example of visual comparison of a reference and a network output demonstrated in the Figure 3.4, the network successfully adapted VAD on Diverse Audio Database.

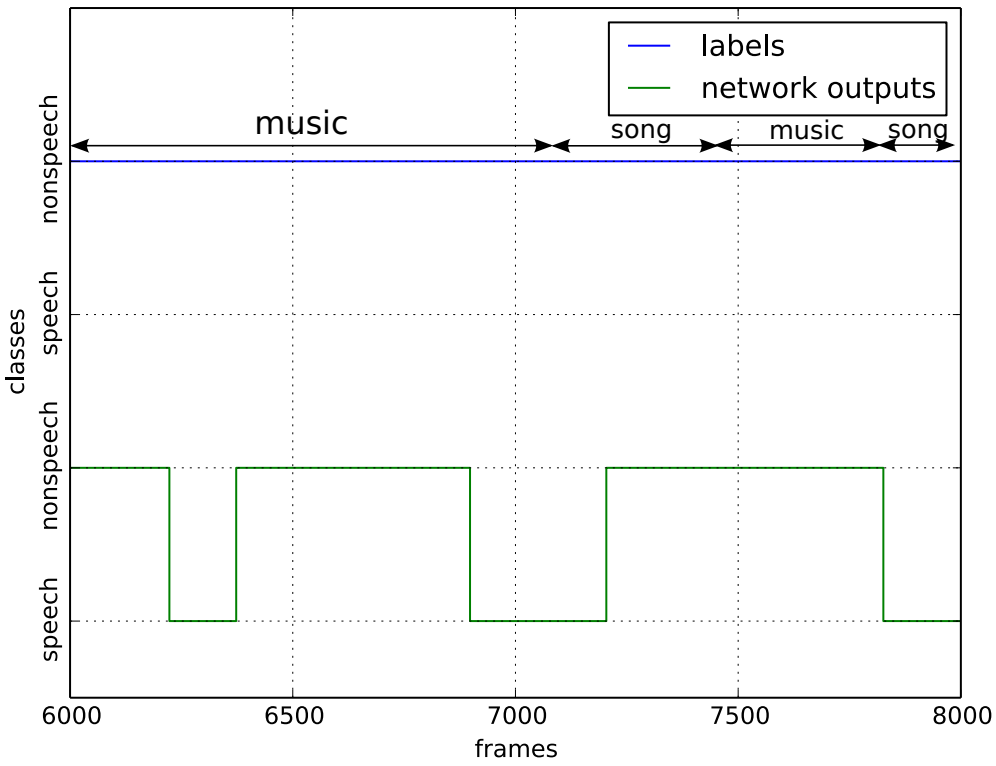


Figure 3.5: An example of comparison between labels and network outputs of the speech detection in a 20 seconds long passage from a radio recording. Whole passage consists of music with singing sequences.

Probably the most problematic issue for the classifier is to correctly detect a non-speech in a song segment. As shown in the Figure 3.5, the classifier mistakenly predicts the sequence of singing as speech. The voiceless music is correctly identified as a non-speech.

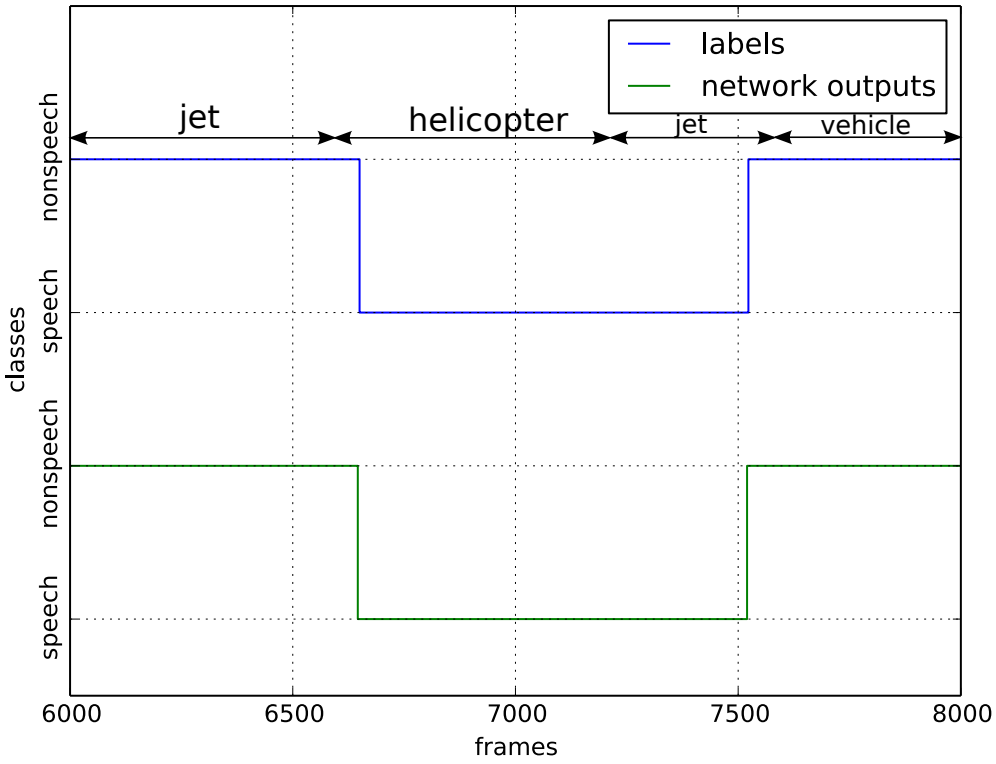


Figure 3.6: An example of comparison between labels and network outputs of the speech detection in a 20 seconds long passage from a warfare footage. This recording includes gunfire, launch of missile and more warfare noise. The results are: FAcc 93.45, hit rate 66.67% and BAcc 54.17%.

On the other hand, the detector seems to respond relatively accurately to the noise. An example can be seen in the Figure 3.6, which demonstrates a report with a sound of jet and helicopter flying and a heavy vehicle moving in the background. In this case, the speech segments are recognized correctly, despite the high noise.

3.3.2 Network parameter experiments

In these experiments, the objective is to watch how does the network size changes the results. The experimenting with network settings is performed on Diverse Audio Database, because a smaller size and a greater diversity are promising more interesting results. The experimentation behind choosing the insertion penalty and threshold to -20.0 and 20 frames respectively is also demonstrated in this subsection.

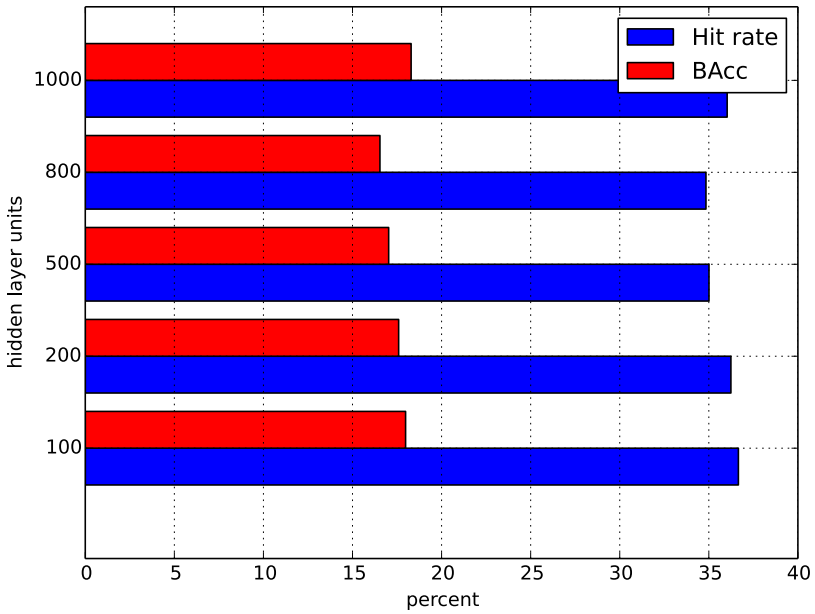


Figure 3.7: Dependency of the number of hidden layer units on the accuracy on the Database 1

As can be seen in the Figure 3.7, the neural network results does not improve until 1000 hidden units, when training on small Diverse Audio Database.

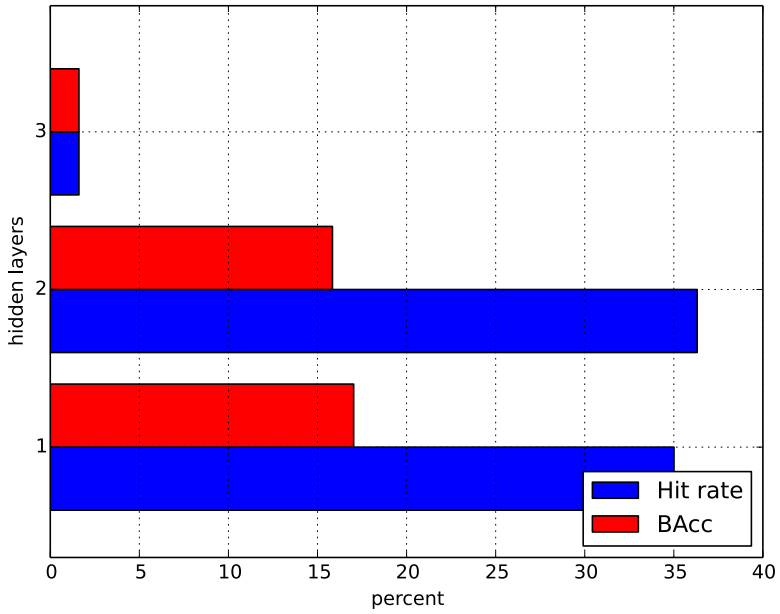


Figure 3.8: Dependency of the number of hidden layers on the accuracy on the Diverse Audio Database

As shown in the Figure 3.8, the network gets over-trained even with the hidden layer size of 2. The hidden layer of size 500 units has been chosen for experimenting with the number of hidden layers.

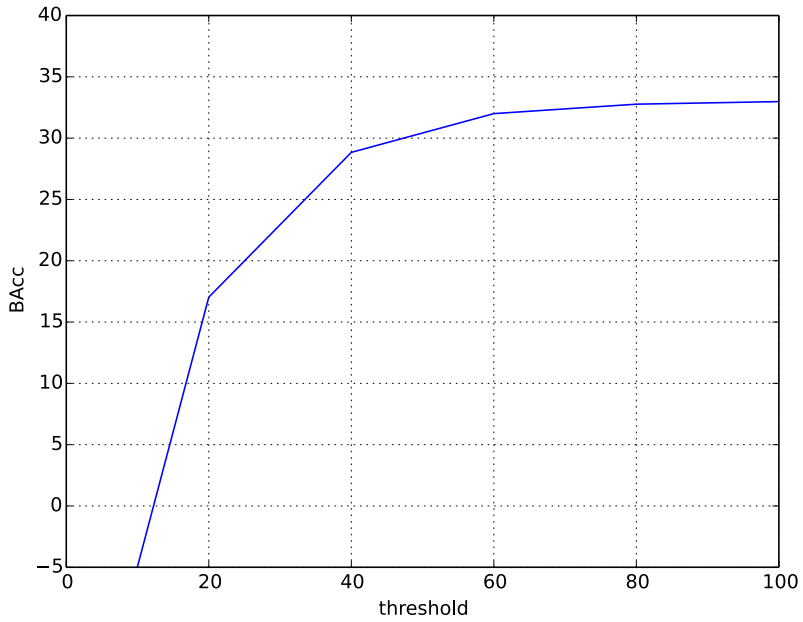


Figure 3.9: Dependency of the threshold on the accuracy on the Diverse Audio Database

Nextly, an illustration of tuning the threshold in order to achieve better BAcc is in the Figure 3.9. Starting on 10 frames (10ms), the results become better with wider threshold, however the acceptable threshold is 20 frames (200ms).

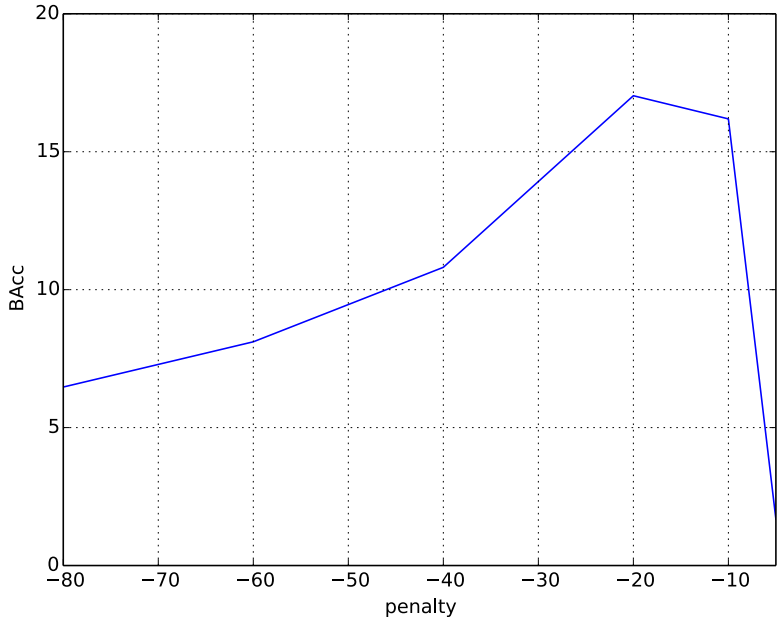


Figure 3.10: Dependency of the insertion penalty on the accuracy on the Diverse Audio Database

The last parameter observation, in the Figure 3.10, is influence of the insertion penalty on the BAcc. The results were slightly improving to the peak at value -20.0, than they started to drastically decrease.

3.3.3 Summary

The simple approach experiments proved that the neural network has successfully learned to detect the speech and the non-speech. Furthermore, it is capable of the VAD for both noisy and clean data. It has problems with sequences of singing, low quality speech recording with reverberation, multiple sources of heavy noise. And in the next analysis, the dataset is too small and prone to over-fitting to use more hidden layers than 1. Lastly, the insertion penalty -20.0 and threshold 20 frames are the mos suitable values of these parameters accordingly to the current scenario.

3.4 The advanced approach

In this section, there will be details about experimentations towards the multi-task neural network. The purpose of first experiment will be comparison of results with and without using more tasks, which will lead to decision whether is this method beneficial for the VAD. The interest of the next experiment will be to observe the accuracy of the multi-task acoustic event detection. Experimenting on the extended database, consisting from Diverse Audio Database datasets and additional Clean Radio Database data, will be approached by the third experiment. The final test will compare the VAD and sound events detection results between noisy and clean data.

The neural network has the same structure as in the simple approach with the exception of varying size of output layer.

3.4.1 Diverse Audio Database multi-task voice activity detection experiment

The aim of this experiment is to show whether is the multi-task beneficial for the speech detection and which of the secondary tasks influences it the most. The results of VAD from the simple approach experiment (the subsection 3.3.1) are used as a baseline.

One of the objectives of this thesis is - does the multi-tasking improve the efficiency of VAD? To answer this question, the experiment begins with the training of the neural network to detect all class groups. Henceforth, the incremental experimenting succeeds. Which consists of training the network always on only one class group in addition to the VAD. This part of the experiment is supposed to unravel which of the class groups influences the primary one the most. The learning process of these experiment ended after sixth iteration in the most cases.

Only the outputs corresponding to the speech detection are kept for the evaluation, the outputs of other groups are discarded.

Table 3.10: Results of the VAD and the difference of BAcc compared to the baseline experiment. Included results are from overall and incremental experimenting.

Task	FAcc	Hit rate	BAcc	BAcc diff
baseline	89.81%	35.01%	17.03%	0.0
all	89.07%	35.29%	17.70%	+0.67%
type	89.84%	36.87%	17.94%	+0.53%
music	87.17%	30.61%	14.41%	-2.62%
microphone	89.70%	37.92%	19.06%	+2.03%
non-st noise	89.73%	34.52%	17.42%	+0.39%
st noise	88.82%	34.17%	16.89%	-0.24%

The results demonstrated in the Table 3.10 indicate, that even if performed on a small dataset, the multi-task learning actually improves the accuracy of VAD, specifically the BAcc by 0.67%. Moreover, most secondary tasks improved the boundary accuracy of the VAD. The highest performance boost can be observed in the case of the microphone distance as a supportive task, not only improving the BER by 2.03%, but also enhancing the hit rate by 2.91%, which is this experiment's maximum.

An interesting observation occurred - while the speech type and the microphone distance are obviously correlated to the speech detection and both have benefiting effect, on the other hand class groups uncorrelated to the speech with the exception of non-stationary noise tend to diminish the results.

Another experiment will be conducted in order to study this observation further. The secondary tasks will be chosen according to whether they are related to the speech. The first multi-task will learn to detect the speech type and the microphone distance in addition to the VAD. The second will focus on the remaining uncorrelated groups together with the VAD.

Table 3.11: Results of the VAD with the correlated and uncorrelated groups as the secondary tasks, and the difference of BER compared to the baseline experiment.

Task	FAcc	Hit rate	BAcc	BAcc diff
baseline	89.81%	35.01%	17.03%	0.0
all	89.07%	35.29%	17.70%	+0.67%
correlated	89.72%	38.13%	18.89%	+1.86%
uncorrelated	88.49%	34.66%	16.61%	-0.42%

An additional experiment, which can be seen in the Table 3.11, was performed in interest of sustaining the previously stated observation. The uncorrelated groups (music and noises) actually have even a negative impact on the VAD resulting in the BAcc lower by 0.42% than the baseline. In contrast, the correlated groups achieved the improvement of the BAcc by 1.86%. These results confirm that the secondary task relation is an important factor in the matter of improving the primary task accuracy.

3.4.2 Diverse Audio Database multi-task acoustic event detection experiment

The accuracy of acoustic events besides the speech is also one of the objectives. Experiments towards this topic are located in this subsection.

The first experiment is performed on the multi-task network with only two tasks. Each test evaluates the accuracy of one secondary task, which has been trained simultaneously only with the primary task.

Table 3.12: Results of acoustic events after the multi-task learning only with the speech.

Task	FAcc	Hit rate	BAcc
type	73.27%	33.02%	6.75%
music	64.47%	27.13%	-136.17%
microphone	79.57%	42.07%	16.81%
non-st noise	82.26%	11.79%	5.90%
st noise	64.53%	13.24%	11.62%

And finally, the experiment, in which the network learned all the tasks in parallel.

Table 3.13: Results of all secondary tasks after the multi-task learning all simultaneously with their comparison to result of training the speech in pair with each another task.

Task	FAcc	Hit rate	BAcc	BAcc diff
type	72.51%	31.89%	5.38%	-0.67%
music	64.77%	26.60%	-143.09%	-6.52%
microphone	79.33%	40.42%	14.76%	-2.05%
non-st noise	80.21%	8.11%	4.42%	-1.48%
st noise	64.63%	13.24%	13.24%	+1.62%

As shown above in the Table 3.13, every task has worse results than the previous experiment (in the Table 3.12) with the exception of the stationary noise, which has actually improved BAcc by 1.62%.

The network also successfully learned to predict the type of speech and the speaker's distance from the microphone. The non-stationary and stationary noise learned to detect only some of their classes, due to the over-fitting. On the contrary, tests shown that the network completely failed to learn the music detection. It is probably caused by that the music class involves both background and foreground music, however the background doesn't have as sharp frequency amplitudes as the foreground and it can be easily misinterpreted as a non-music sequence.

3.4.3 Extended database multi-task acoustic event detection experiment

The dataset from Diverse Audio Database extended by 12 recordings from Clean Radio Database is used in this experiment. The goal is to find out, how does extending the dataset by a clean and more fairly distributed data alters the results.

Both experiments are evaluated on the extended testing set.

Table 3.14: Results of the all acoustic events after the multi-task learning. Note: the insertion penalty used for the music evaluation was -100.0.

Task	FAcc	Hit rate	BAcc
speech	89.27%	37.04%	19.10%
type	72.96%	33.56%	7.22%
music	70.93%	30.11%	-12.37%
microphone	79.57%	42.46%	17.90%
st noise	64.50%	13.96%	9.97%

Table 3.15: Results of the all acoustic events after the multi-task learning. Note: the insertion penalty used for the music evaluation was -100.0.

Task	FAcc	Hit rate	BAcc	BAcc diff
speech	84.07%	19.87%	1.08%	-18.02%
type	72.73%	16.80%	-2.86%	-10.08%
music	60.63%	26.06%	-51.06%	-38.69%
microphone	75.71%	17.48%	-5.46%	-23.36%
st noise	64.12%	14.25%	1.42%	-8.55%

The testing was performed on the same dataset as in the previous experiments. The results from Table 3.15 in comparison with the Table 3.10 are worse. It might be because of the great difference between the diversity and noisiness of the first and second database.

3.4.4 Noisy and clean multi-task acoustic event detection experiment

This subsection aims for the comparison of the comparison between the VAD performed on the noisy and clean data. Two new testing datasets were created specially for the following experiment. The first dataset contains averagely or severely noised data, whereas the second dataset consists of clean data, without any significant noise.

Table 3.16: Results of the all acoustic events after the multi-task learning evaluated on the clean dataset.

Task	FAcc	Hit rate	BAcc
baseline	93.59%	36.12%	21.32%
type	82.02%	39.47%	10.35%
music	51.79%	15.70%	-75.21%
microphone	90.31%	50.90%	23.06%

Testing the non-stationary and stationary class groups doesn't make sense in this case, therefore it was excluded.

Table 3.17: Results of the all acoustic events after the multi-task learning evaluated on the noisy dataset.

Task	FAcc	Hit rate	BAcc
baseline	85.49%	34.78%	15.42%
type	65.00%	27.40%	2.44%
music	75.03%	46.27%	-265.67%
microphone	70.66%	33.72%	9.47%

The results, which are demonstrated in the Table 3.17, confirm that the speech detection is successfully detected both in the clean and noisy data. The detection in the clean data is more accurate, specifically speaking result of every metric is higher: the FAcc by 8.10%, the hit rate by 234% and the BAcc by 5.50%.

3.4.5 Summary

On the basis of the previous experiments, it can be concluded that multi-task system including involved acoustic events is beneficial for the VAD. The first experiment showed that the speech type and the microphone distance characteristics improve the VAD's accuracy. The music and noise in the contrary diminish the results. Which altogether indicated that the correlated events have potential to be beneficial. Henceforth, the consecutive experiment based on comparing the correlated and uncorrelated groups confirmed this statement. Extending the database by clean data was rather unsuccessful, probably because of the lack of noise. Which escalated into decreased accuracy in the noisy testing dataset. In the final analysis, it was proven, that the neural network is capable of performing the VAD both on clean and noisy data.

3.5 The experimentation conclusion

This section summarizes every important observation during the experimentation phase.

The VAD was successfully implemented by using a simple neural network. The network learned how to predict the speech class with the BAcc 10.24%, hit rate 28.51% and FAcc 88.89% despite the small size and the diversity of Diverse Audio Database. After applying an under-sampling method on the original database, the detection even reached the BAcc of 17.03%, hit rate 35.01% and FAcc 89.81%. Moreover, the under-sampled database was more successful also in the multi-tasking with the Bacc 17.70% compared to 6.51% BAcc of thefull-sized data.

However, the size and diversity of data resulted into over-fitting while using more than one hidden layer. While adding the hidden layer units, the network's performance didn't improve until 1000 nodes.

The threshold started to converge to the maximum accuracy at 80 frames and the insertion penalty achieved peak with value -20.0.

In the second phase, the adjusted network also coped with multi-tasking and improved the BAcc to 17.70% and even 19.06%, when using the microphone distance as the only secondary task during the incremental experimenting. On the other hand, two of five secondary tasks, music and stationary noise, even diminished the accuracy. The worst case was registered with the music secondary task, decreasing the BAcc by -2.62%. As the experiments had shown, the most beneficial class groups was the speech type microphone distance, thanks to it's correlation to the speech task, enhancing the baseline BAcc by 2.03%. The follow-up experiment proved that the relation between the primary task and the supplementary task really matters, because the group of correlated tasks bettered the BAcc by 1.86%, on the other side the uncorrelated lowed by 0.24%.

Also the uncorrelated tasks completely failed to be successfully detected. The lowest accuracy was in the case of the music during the full multi-task, where the number of insertion was even higher then hits, resulting into the BAcc -143.09%. Neither of noise events didn't have a negative BAcc, however their hit rates were both under 10%. On the contrary, the best results, the BAcc 16.81% and hit rate 42.07%, had the microphone distance task while trained only with the speech.

The multi-tasking turned out to be a boost only for the stationary noise, improving the BAcc by 1.62%, the accuracy of every other secondary task decreased.

According to the different distribution of noise, extending the training dataset by data from Clean Radio Database didn't prove to be useful, due to the fact that every results decreased.

Finally, the objective achievement of the speech detection was confirmed in the last experiment dividing the testing dataset into the noisy and clean part. Where the baseline achieved the BAcc 21.32%, hit rate 36.12% and FAcc 93.53% in the clean data. Although the results can't be compared due to the different testing sets, the results BAcc 15.42%, hit rate 34.78% and the FAcc 85.49% confirm success in this cas as well.

Chapter 4

Conclusion

The main aim of this thesis was to successfully implement the VAD by neural network. Train it on noisy data and predict the outputs correctly. The observations of the behaviour of the multi-task network were also a part of the objective.

After a chapter dedicated to the theoretical basis, mentioning the feature extraction, classification methods, neural networks and post-processing, the experiments were performed.

The experimentation phase confirmed achieving the outlined objectives. The first experiment showed that the neural network could be used as a voice activity detector. The advanced approach demonstrated how is the multi-tasking beneficial for the speech detection. The best results were achieved with the correlated class group as the secondary tasks. Unfortunately, the experiments also showed that the network wasn't able to learn prediction of music and stationary noise. Other secondary tasks were more or less successful.

The final analysis demonstrated that although with slightly worse results, the network managed to detect the speech both in clean and noisy data. In a word, the main objective of this thesis is accomplished as well.

The future plans of this research begin with collecting more data, preferably with a better class distribution, so it is easier to manipulate with datasets. According to the results of experimentation, the detection of acoustic events with a multi-task neural network seems promising. However, database size is the weakness of otherwise excellent tool such a neural network truly is. There is a potential of satisfyingly accurate results with a wider database with a fair distribution of classes.

Anther option is to try to minimize the diversity between the classes by implementing a more sophisticated method than an under-sampling.

An obvious possible continuation of this research is to perform experiments with new class groups or divide the existing ones into more specific classes.

These suggestion are, however, only a mere insight of all the possible options.

Bibliography

[1] Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian Goodfellow, Arnaud Bergeron, Nicolas Bouchard, and Yoshua Bengio. Theano: new features and speed improvements. [online] http://arxiv.org/pdf/1211.5590v1.pdf, 2012.

[2] Emre Cakir, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen. Polyphonic sound event detection using multi label deep neural networks. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7, 2015. ISSN 2161-4393.

[3] William Campbell, Joseph Campbell, Douglas Reynolds, Elliot Singer, and Pedro Torres-Carrasquillo. Support vector machines for speaker and language recognition. In *Computer Speech I& Language*, volume 20, pages 210 – 229. Elsevier, 2006. ISSN 0885-2308.

[4] Rich Caruana. Multitask learning. Master's thesis, Blekinge Institute of technology, 1997.

[5] Vaishali Ganganwar. An overview of classification algorithms for imbalanced datasets. volume 2, pages 42–47. Citeseer, 2012.

[6] Tian Gao, Jun Du, Yong Xu, Cong Liu, Li-Rong Dai, and Chin-Hui Lee. *Latent Variable Analysis and Signal Separation: 12th International Conference*, chapter Improving Deep Neural Network Based Speech Enhancement in Low SNR Environments, pages 75–82. Springer International Publishing, 2015. ISBN 978-3-319-22482-4.

[7] Geoffrey Hinton et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. volume 29, pages 82–97. IEEE, 2012. ISSN 1053-5888.

[8] Xuedong Huang and Li Deng. An overview of modern speech recognition. In *Handbook of Natural Language Processing, Second Edition*. CRC Press, Taylor and Francis Group, 2010. ISBN 978-1420085921.

[9] Zhen Huang, Jinyu Li, Sabato Marco Siniscalchi, I-Fan Chen, Ji Wu, and Chin-Hui Lee. Rapid adaptation for deep neural networks through multi-task learning. In *Interspeech*, 2015.

[10] Nikola Kasabov. *Foundations of neural networks, fuzzy systems, and knowledge engineering*. Marcel Alencar, 1996. ISBN 0-262-11212-4.

[11] Pham Chau Khoa. Noise robust voice activity detection. Master's thesis, Nanyang Technological University, 2012.

[12] David Kriesel. A brief introduction to neural networks. [online] http://www.dkriesel.com/_media/science/neuronalenetze-en-zeta2-2col-dkrieselcom.pdf, 2007.

[13] Yann LeCun, Léon Bottou, Genevieve Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012. ISBN 978-3-642-35288-1.

[14] Valentin Sergeyevich Mendelev, Tatiana Nikolaevna Prisyach, and Alexey Alexandrovich Prudnikov. Robust voice activity detection with deep maxout ne,ural networks. volume 9, page 153. Canadian Center of Science and Education, 2015. ISSN 1913-1844.

[15] Annamaria Mesaros, Toni Heittola, Antti Eronen, and Tuomas Virtanen. Acoustic event detection in real life recordings. In *18th European Signal Processing Conference*, pages 1267–1271, 2010. ISSN 2219-5491.

[16] Lindasalwa Muda, Mumtaj Begam, and Irraivan Elamvazuthi. Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques. *Journal of Computing*, 2010. ISSN 2151-9617.

[17] Michael Nielsen. Neural networks and deep learning. [online] http://neuralnetworksanddeeplearning.com/, 2015.

[18] Mikael Nilsson and Marcus Ejnarsson. Speech recognition using Hidden Markov Model: Performance evaluation in noisy environment. Master's thesis, Blekinge Institute of technology, 2002.

[19] Bjorn Schuller, Gerhard Rigoll, and Manfred Lang. Hidden markov model-based speech emotion recognition. volume 2, pages 401–404, 2003. ISSN 1520-6149.

[20] Young Steven. The HTK Hidden Markov Model toolkit: Design and philosophy. volume 2, pages 2–44, 1994.

Appendices

List of Appendices

Appendix A

Contents of the CD

The attached CD contains:

- PDF version of this document
- LaTeX version of this document
- Python scripts directory
- experimentation results directory
- demonstration video
- A2 poster

Appendix B

Graphical examples of outputs

B.1 Speech detection

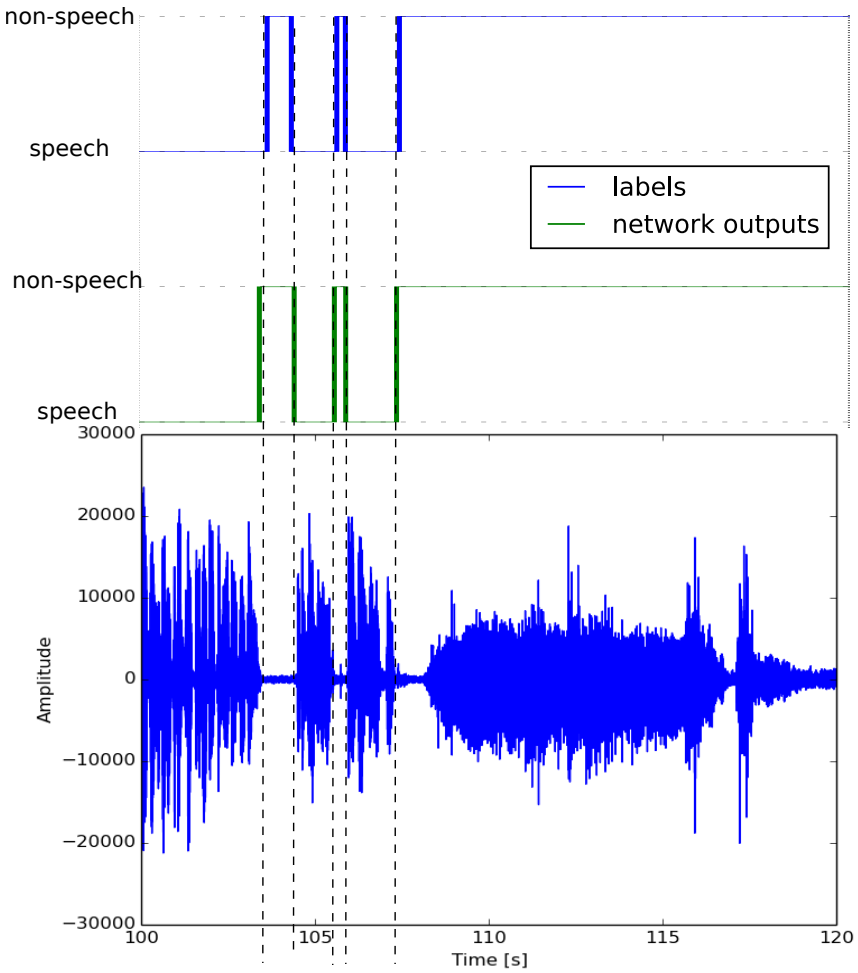


Figure B.1: An example of the speech detection output, labels and audio signal from the 100th second to 120th second. This footage is from documentary film with natural noises, the most noticeable are elephant sounds starting approximately in 108th second.

B.2 Music detection

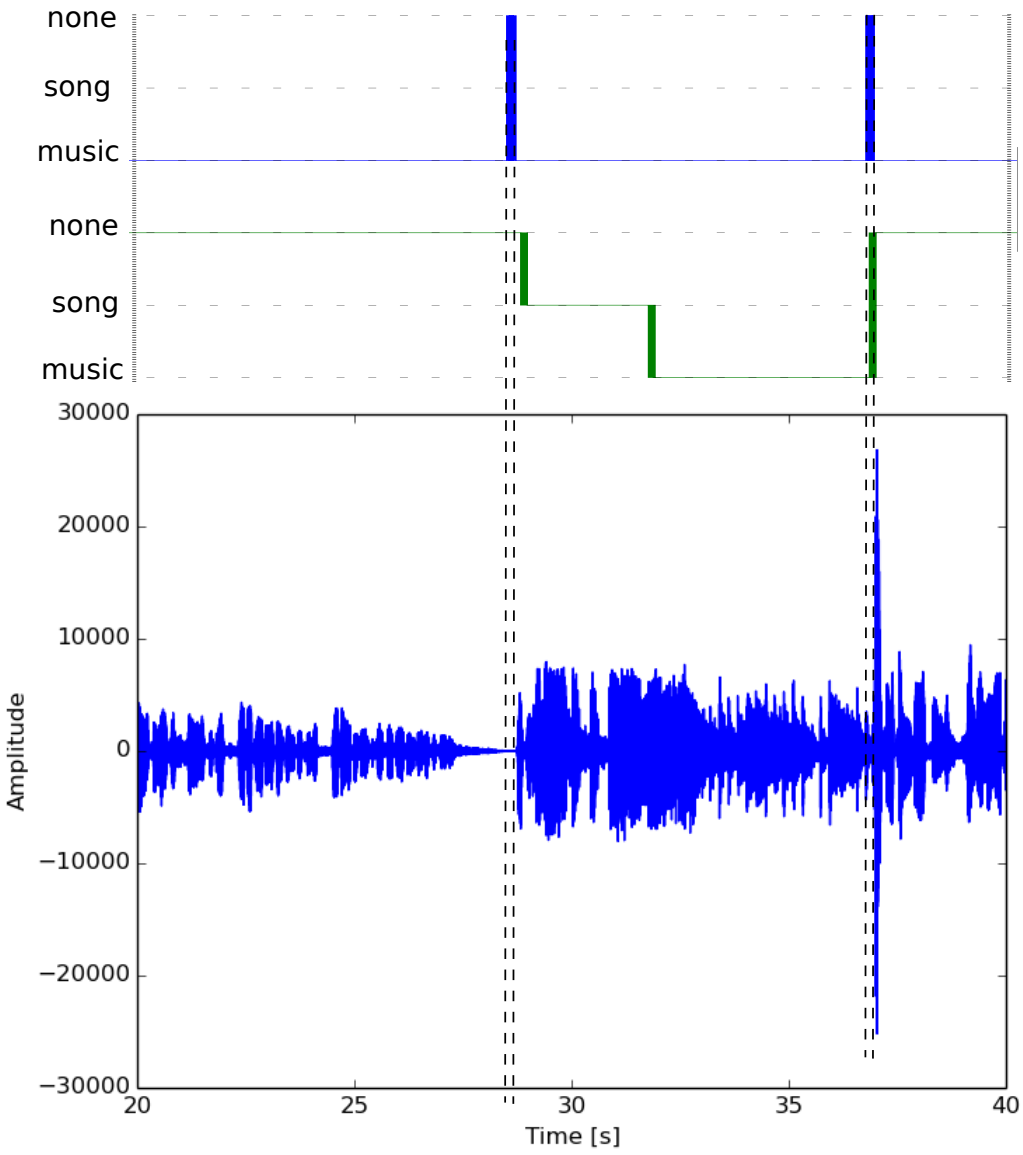


Figure B.2: An example of the music detection output, labels and audio signal from the 20th second to 40th second. This part is from a radio with music, the example illustrates a problem of recognizing background music, the only detected sequence is a noticeable foreground music located around 30th second.

B.3 Music detection

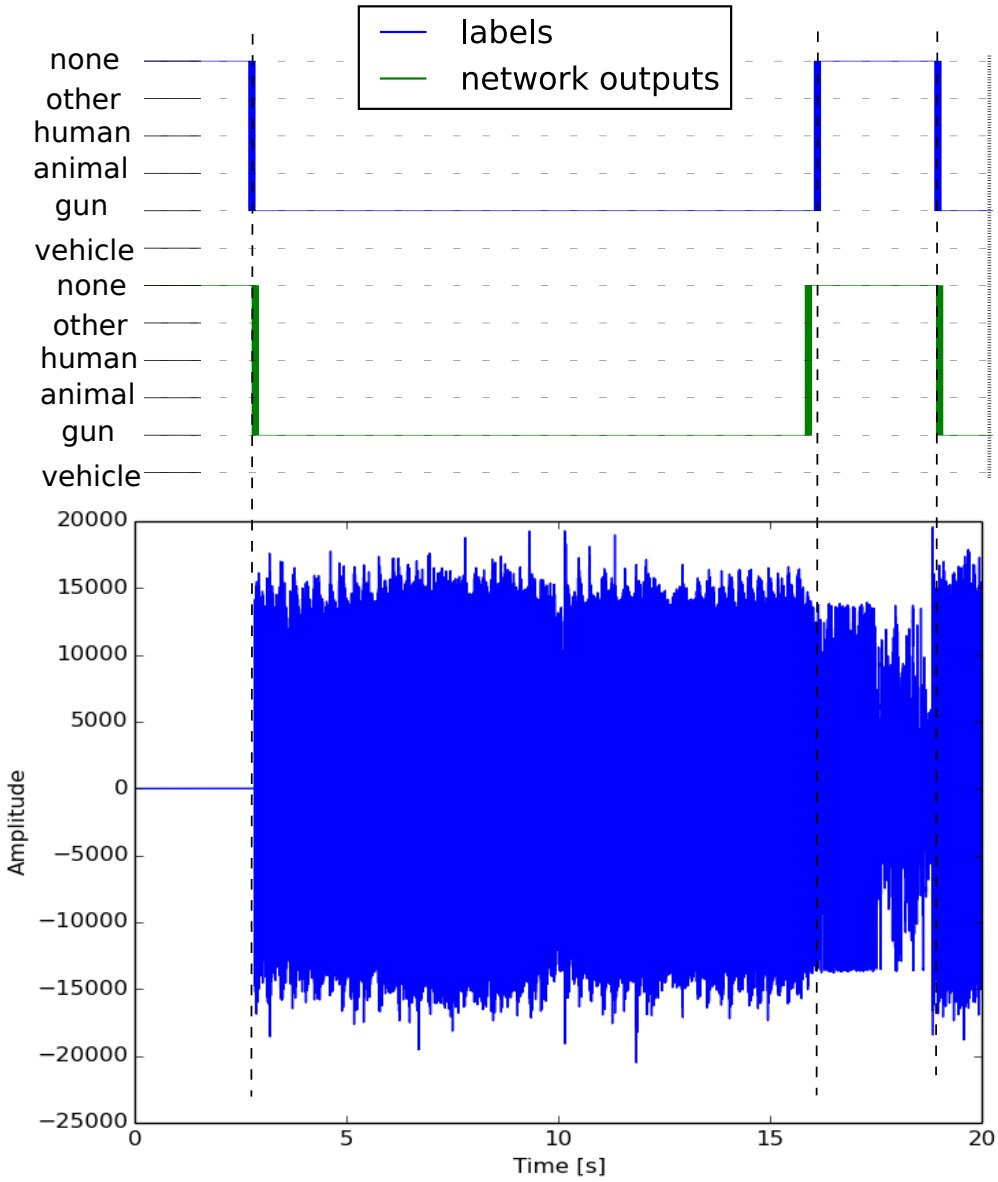


Figure B.3: An example of the non-stationary detection output, labels and audio signal from the first 20 seconds. This is a footage recording a warfare in the Middle East. The class „gun“ correctly classified instances of gunfire and tank firing.