TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií

Robustní odhad odstupu řeči od šumu pomocí hlubokých neuronových sítí

Diplomová práce

Studijní program: N2612 – Elektrotechnika a informatika
Studijní obor: 1802T007 – Informační technologie

Autor práce: **Bc. Michal Mužíček**
Vedoucí práce: Ing. Jiří Málek, Ph.D.

Liberec 2016

TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies

Robust estimation of speech to noise ratio using deep neural networks

Diploma thesis

Study programme: N2612 – Electrical Engineering and Informatics
Study branch: 1802T007 – Information Technology

Author: **Bc. Michal Mužíček**
Supervisor: Ing. Jiří Málek, Ph.D.

Liberec 2016

TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií
Akademický rok: **2015/2016**

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Michal Mužíček**
Osobní číslo:	**M14000172**
Studijní program:	**N2612 Elektrotechnika a informatika**
Studijní obor:	**Informační technologie**
Název tématu:	**Robustní odhad odstupu řeči od šumu pomocí hlubokých neuronových sítí**
Zadávající katedra:	**Ústav informačních technologií a elektroniky**

Zásady pro vypracování:

1. Seznamte se s metodikou odhadu odstupu řeči od šumu (Speech to Noise Ratio - SNR) z řečových záznamů pořízených v reálném prostředí.
2. Důležitou součástí mnoha metod pro odhad SNR je detektor řečové aktivity (Voice Activity Detector - VAD). Seznamte se s detektory postavenými na modelu neuronové sítě.
3. Natrénujte robustní VAD, umožňující rozpoznat řečové úseky v zarušené nahrávce. Uvažujte několik druhů reálných ruchů (např. ruch ulice, šum větráku, ruch v kavárně).
4. Vytvořte aplikaci (volitelně na mobilním zařízení), která bude umožňovat odhad SNR (volitelně v reálném čase) a bude používat Vámi natrénovaný VAD. Vyhodnoťte přesnost odhadu pomocí objektivních kritérií.

Rozsah grafických prací:	**Dle potřeby dokumentace**
Rozsah pracovní zprávy:	**cca 40-50 stran**
Forma zpracování diplomové práce:	**tištěná/elektronická**

Seznam odborné literatury:

[1] **M. Vondrášek, P. Pollák, "Methods for Speech SNR Estimation: Evaluation Tool and Analysis of VAD Dependency", Radioengineering, vol. 1, 2005.**

[2] **Zhang, Xiao-Lei, and Ji Wu. "Deep belief networks based voice activity detection." Audio, Speech, and Language Processing, IEEE Transactions on 21.4 (2013): 697-710.**

[3] **Torch, Scientific computing for LuaJIT, [online 21.9.2015], http://torch.ch/.**

Vedoucí diplomové práce:	**Ing. Jiří Málek, Ph.D.** Ústav informačních technologií a elektroniky
Konzultant diplomové práce:	**Ing. Lukáš Matějů** Ústav informačních technologií a elektroniky
Datum zadání diplomové práce:	**14. září 2015**
Termín odevzdání diplomové práce:	**16. května 2016**

prof. Ing. Václav Kopecký, CSc.
děkan

Technická univerzita v Liberci
L.S.
- 14 -

prof. Ing. Zdeněk Plíva, Ph.D.
vedoucí ústavu

V Liberci dne 14. září 2015

Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 16.5.2016

Podpis:

Poděkování

Rád bych poděkoval svému vedoucímu, Ing. Jiřímu Málkovi, PhD., za jeho trpělivost a veškerou jeho pomoc při tvorbě této diplomové práce.

Abstrakt

Práce se zabývá tvorbou neuronové sítě, která je schopná, i přes výskyt různorodého šumu, odhadnout, kde se v řečové nahrávce vyskytuje řeč. Jako vstupní data pro trénování neuronové sítě slouží databáze aditivní směsi šumu a čistých řečových nahrávek. Data zpracovaná neuronovou sítí jsou následně předána algoritmu, který vypočítá odhad odstupu řeči od šumu. Správnost výstupu navrženého algoritmu je hodnocena dle porovnání s konkurenční metodou WADA. Výsledné hodnoty naznačují, že využití neuronových sítí pro detekci přítomnosti řeči a následného odhadu SNR úrovně jsou reálnou alternativou existujícím metodám.

Klíčová slova

neuronové sítě, VAD, Voice Activity Detector, SNR, Signal To Noise Ratio, odstup řeči od šumu

Abstract

This documentation describes a creation of a neural network that is capable of locating the location of speech in audio sample. Database containing additive mixture of noise and speech signals is used as an input for training of the neural network. Output from this network is then processed by an algorithm, which computes an estimation of signal to noise ratio. Performance of this algorithm is then compared against performance of WADA, a conventionally used software. Results suggest that using neural networks for detecting presence of speech in a signal and estimating speech to noise ratio from it, is an effective alternative to the existing methods.

Keywords

neural networks, VAD, Voice Activity Detector, SNR, Signal To Noise Ratio

Obsah

Seznam obrázků

Seznam tabulek

Seznam grafů

Seznam zkratek

DFT	Discrete Fourier Transformation - diskrétní Fourierova transformace
DBN	Deep Belief Network
DNN	Deep Neural Network - hluboká neuronová síť
GSNR	Global Sinal to Noise Ratio - globální odstup řeči od šumu, vztahující se obvykle delšímu časovému úseku signálu
LSNR	Local Sinal to Noise Ratio - lokální odstup řeči od šumu, vztahující se obvykle k 1 vzorku
NN	Neural Network - neuronová síť
SNR	Sinal to Noise Ratio - odstup řeči od šumu
MFCC	Mel-frequency Cepstral Coefficients
MSE	Mean Square Error - průměrná hodnota kvadrátu chyby
VAD	Voice Activity Detection - detekce řečové aktivity
ZCR	Zero Crossing Rate - rychlost průchodů nulou
WADA	Waveform Amplitude Distribuion Analysis - analýza amplitudové distribuce signálu

Úvod

Každý reálný signál je součet užitečné(pro mojí aplikaci) komponenty a neužitečné komponenty (označujeme jako šum, interference). Jedním ze základních problémů při zpracovávání signálů v oblasti rozpoznávání řeči je pak zjištění, jak moc zašuměná je zpracovávaná nahrávka. Tedy zjistit jestli se v nahrávce vyskytuje užitečná komponenta, nebo jestli je přehlušena neužitečnou komponentou do takové míry, že již samotná užitečná informace není zřetelná. Hovoříme o odstupu řeči od šumu (Signal to Noise Ratio, dále jen SNR), které přímo určuje poměr energií užitečné komponenty vůči neužitečné komponentě v digitální nahrávce. Čím větší SNR, tím lépe je užitečná informace rozlišitelná od šumu a naopak. Zjistit přesně SNR lze pouze v laboratorních podmínkách a v reálném světě je třeba odstup užitečné informace od šumu odhadnout, protože nemáme jednotlivé komponenty ale pouze jejich směs. V této práci je užitečnou komponentou řeč a neužitečnou šum.

Velmi často se odhad odstupu řeči od šumu provádí za pomocí segmentů, kde je v nahrávce aktivní řeč (tedy segmenty se mohou skládat pouze ze šumu nebo ze šumu a řeči). Cílem je zjistit, ve kterých úsecích digitální nahrávky se tyto segmenty s řečí vyskytují. K řešení uvedeného problému se používají tzv. detektory řečové aktivity (Voice Activity Detectors, dále jen VAD). Jedná se o algoritmy, které určí (s jistou mírou tolerance), kde se v dané nahrávce nachází řečová aktivita.

K jednodušším dnes používaným detektorům řečové aktivity patří například energetický detektor [1], který klasifikuje řeč a šum v nahrávce pomocí prahování okamžitého výkonu směsi, případně detektor používající kombinaci energie a rychlosti průchodů nulou [2]. Dále pak VAD pracující s frekvenčním spektrem [3] a případně se speciálními časovými příznaky zvanými kepstrální příznaky [4]. Mezi komplexnější (a většinou efektivnější) detektory patří například detektor založený na statistických vlastnostech řečové a šumové komponenty [5], který je klasifikuje na základě pravděpodobnosti získané ze statistického modelu.

K účelu rozpoznání přítomnosti řeči v nahrávce lze tedy použít velké množství charakteristických vlastností řeči. Mezi ně patří i harmonická struktura řeči, kterou dobře odrážejí logaritmické frekvenční příznaky z frekvenční oblasti signálu, jež jsou použity právě v této práci. Jedná se o nízkoúrovňové příznaky, které jsou schopny dobře reprezentovat digitální signál pomocí poměrně malého množství dat.

Problematiku odstupu řeči od šumu řeší i Dan Ellis ve svém programu WADA(Waveform Amplitude Distribution Analysis) [6], který odhaduje úroveň SNR pomocí statistických metod.

Motivace Práce navrhuje a experimentálně testuje robustní odhad SNR využívající detekce řečové aktivity. Nejprve je popisován použitý VAD, který je implementován jako hluboká neuronová síť, jejíž parametry jsou trénované na rozsáhlé množině řečových a šumových signálů. Díky svým vlastnostem se neuronové sítě stávají efektivní alternativou dosavadních VAD metod. Neuronové sítě se svojí funkcí snaží napodobit schopnost mozku rychle zpracovávat velké množství vstupních dat pomocí navzájem propojených neuronů. Existuje více druhů neuronových sítí, které se liší svým zaměřením na charakter dat (respektive charakterem vnitřních funkcí). Typ sítě použité v této práci se zaměřuje na klasifikaci vstupních dat do výstupních kategorií (řeč/šum). V druhém kroku pak neuronová síť svůj výstup předá algoritmu pro odhad odstupu řeči od šumu, který za pomocí adaptivního odhadu výkonu šumu vypočítá hodnotu globálního SNR (viz. kapitola 1.3), což je cílem předkládané práce.

Odhad SNR je často používán jako jedna z komponent v rozsáhlejším systému pro zpracovávání signálu. Například v úloze, kdy potřebujeme vyextrahovat co nejvíce užitečné řeči z velké databáze nahrávek, nám informace o SNR usnadní proces hledání vhodných zvukových stop pro zpracování, čímž se zkrátí výpočetní čas potřebný k vykonání úlohy.

1 Teoretické základy

1.1 Úvod do zpracovávání signálů (Signal Processing)

Jedná se o technický obor, který se zabývá veškerou manipulací se signály. Signálem se rozumí sekvence dat, která obnáší užitečnou informaci. Tato sekvence může být analogová (např. zvuky v reálném světě, tedy vibrace), nebo digitální (zpravidla analogový signál převedený do formátu, se kterým je počítač schopný pracovat). V současné době je rozšířenější digitální zpracování signálu, které probíhá hlavně v elektronických systémech (např. počítače).

1.2 Signál

Signál je (matematická) funkce, která reprezentuje informaci o vývoji nějaké fyzické veličiny. Jak vhodně vyjádřil B. Porat [7], signály, se kterými se setkáme v reálném životě, jsou většinou spojité jak na časové ose, tak na amplitudové ose. Takové signály nazýváme analogové signály a existuje jich velké množství. Mezi nejběžnější patří:

- Elektrické signály: napětí, proudy, elektrická pole, magnetická pole
- Mechanické signály: lineární posunutí, úhly, rychlosti, úhlové rychlosti, síly, momenty
- Akustické signály: vibrace, zvukové vlny, lidská řeč
- Signály související s fyzickými vědami: tlaky, teploty, koncentrace

1.2.1 Diskrétní signál

Diskrétní signál se od analogového liší tím, že není spojitý na časové ose, ale nabývá hodnot v časových (vzorkovacích) intervalech. Diskrétním signálem tedy nazýváme indexovanou nekonečnou posloupnost reálných nebo komplexních čísel.

Pokud z analogového signálu získáme jeho okamžité hodnoty v přesných časových intervalech, získáme vzorkovaný (diskrétní) signál. Pokud signál může nabývat pouze konečného počtu hodnot, pak se jedná o kvantovaný signál. Kombinací těchto dvou kritérií získáme digitální signál.

1.2.2 Řečový signál

Časový průběh akustického tlaku vyvolaného hlasivkovým ústrojím člověka nazýváme řečovým signálem. Frekvence lidské řeči se ve většině případů pohybuje mezi 300 Hz a 3 kHz. Obecně platí, že mužský hlas má znatelně nižší základní frekvenci než hlas ženský. Tento fakt je využíván v oblasti rozeznávání řečníka v hlasové nahrávce. Ovšem najdou se lidé, jejichž hlasový aparát se svými vlastnostmi liší natolik, že značně stíží správné rozpoznání

řečníka (např. muž, který je chybně rozeznán jako žena, kvůli vysokému tónu jeho hlasu). Této vlastnosti lze ovšem využít i pro detekci přítomnosti řeči, pokud se algoritmus zaměří právě na zmíněnou frekvenční oblast při analýze signálu.

1.3 Odstup řeči od šumu (SNR Signal to Noise Ratio)

Každý záznam řeči je v praxi zatížen nějakým šumem (šum pozadí, mikrofonu, kvantovací atd.) Formální zápis tohoto vztahu vypadá takto:

$$x[n] = s[n] + v[n] \tag{1}$$

SNR je kvantitativní kritérium, které měří míru přítomnosti šumu v reálném řečovém záznamu. Je dán jako poměr energií řeči a šumu.

Globální SNR SNR je velmi rozšířené kritérium v oblasti zpracovávání řeči [8]. Pokud se obecně vztahuje k delším úsekům zvukových stop, tak takové SNR označujeme jako globální (GSNR).

$$GSNR = 10 \log \frac{\sigma_s^2}{\sigma_v^2} \tag{2}$$

kde σ_s^2 je energie řečového signálu a σ_n^2 je energie šumu

Pro řeč není takto definované SNR vhodné, protože řeč je aktivní jen někdy, což tuto míru vychyluje. Standardní SNR definice optimalizovaná pro řečové signály je založena na počítání GSNR pouze z řečových segmentů analyzovaného signálu. Rovnice (2) pak může být v takovém případě přepsána jako:

$$GSNR = 10 \log \frac{\sum_{n=0}^{N-1} s^2[n] \cdot vad[n]}{\sum_{n=0}^{N-1} v^2[n] \cdot vad[n]} \tag{3}$$

kde $s[n]$ je n-tý řečový vzorek, $v[n]$ je n-tý šumový vzorek a $vad[n]$ je binární signál popisující řečovou aktivitu v n-tém vzorku signálu a N je délka signálu

Ovšem u signálů, které jsou velmi dynamické, nemá GSNR takovou informační váhu, jelikož se jedná o průměrnou hodnotu. V takových případech získáme více informací z průběhu lokálního SNR.

Lokální SNR Lidská řeč je kvazi-stacionární signál, to znamená, že ačkoli je nestacionární jako celek, tak při analýze v malých časových oknech se jeví stacionárně (jednotlivé hlásky ve větách mají po určitou dobu stejný frekvenční a amplitudový charakter). Proto se řeč zpracovává hlavně v krátkých rámcích (například o délce 30 ms). Lokální SNR (LSNR) je

tedy definováno pro krátké intervaly/segmenty v signálu jako:

$$SNR_i = 10 \log \frac{\sum_{n=0}^{L-1} s_i^2[n]}{\sum_{n=0}^{L-1} v_i^2[n]} = 10 \log \frac{E_{s,i}}{E_{v,i}} \tag{4}$$

kde $s_i[n]$ a $v_i[n]$ jsou řečové a šumové vzorky v i-tém segmentu analyzovaného signálu, L je velikost segmentu a $E_{s,i}$ a $E_{v,i}$ je výkon řeči a šumu v i-tém rámci respektive

Jelikož se již pohybujeme v oblasti energie z vybrané části signálu, tak hovoříme o výkonu signálu.

1.4 Výpočet SNR

Jelikož se v běžné digitální nahrávce nikde nevyskytuje údaj o SNR úrovni, je třeba ho vypočítat. Mohou nastat 2 případy:

Signál s referencí V některých případech máme jak zarušený signál, tak i referenční signál (např. zvukovou stopu čisté řeči). V tu chvíli stačí pouze odečíst referenční hodnotu od zarušeného signálu, čímž získáme k dispozici 2 čisté signály (řečový a šumový). V tu chvíli jsme schopni spočítat jak GSNR (2), tak LSNR (4).

Signál bez reference V praxi ovšem referenční signál nemáme a musíme hodnotu SNR odhadovat. Na tuto problematiku se v této práci zaměřuji.

Existuje několik praktik, které se k tomuto účelu používají. Většina z nich je založena na VAD prvku, který určí (s jistou mírou tolerance), kde se vyskytuje řeč a kde řeč není. V tu chvíli můžeme aplikovat algoritmus na výpočet energie šumu. Zde je třeba vzít v potaz charakter cílových dat. Jestli je šum stacionární, je možné k výpočtu použít průměr globální energie šumu. Je-li šum nestacionární, pak by globální odhad byl velmi nepřesný. V takovém případě musíme odhad šumu průběžně adaptovat(např. pomocí průměrovacího okénka) tím, že budeme měnit hodnotu energie šumu v průběhu signálu, čímž získáme mnohem přesnější odhad.

1.5 Detekce řečové aktivity (VAD Voice Activity Detection)

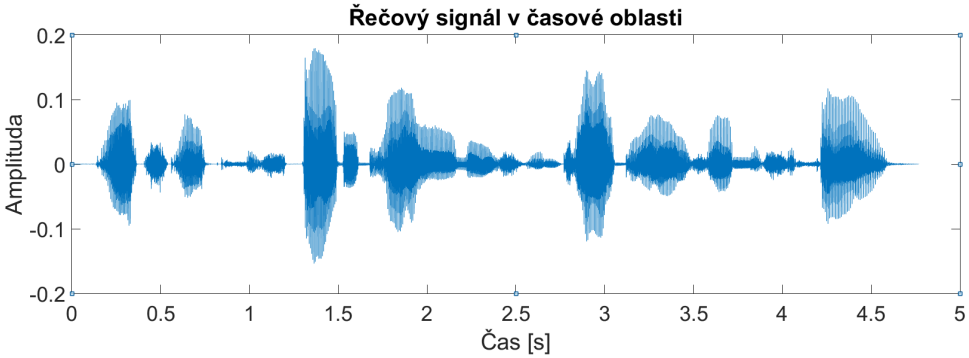
Detekce řečové aktivity patří mezi základní operace při zpracovávání řečových signálů. Existuje několik různých praktik, které se zaměřují na různé vlastnosti lidské řeči, kterým je pak detekce přizpůsobena [9]. Z velké části se disciplína detekce řeči soustředí na časovou nebo frekvenční oblast signálu (případně u komplexnějších VAD lze využít obojí).

1.5.1 Ideální detektor

Jako ideální detektor označujeme velmi přesný VAD, který vznikne manuálním označením skutečných segmentů řečové aktivity. Alternativně jej lze získat pomocí zvukové stopy čisté řeči bez jakéhokoli šumu (nebo referenční stopou pouze s šumem), což je v reálných podmínkách velmi obtížné získat. Konkrétní detekce pak může být založena na jakémkoli níže popsaném algoritmu. Nejjednodušší je např. detektor meze energie, kde se pouze stanoví limit pro energii řeči, čímž nastavíme intenzitu detekované řeči.

1.5.2 Detekce řečové aktivity v časové oblasti signálu

Vývojem signálu v časové oblasti rozumíme změnu amplitudy (akustického tlaku) v závislosti na čase. Jedná se o přímý výstup z A/D převodníku (převodník z analogového signálu na digitální) viz Graf č. 1.



Graf 1: Ukázka signálu obsahující řeč zobrazeného v časové oblasti

Mez energie (Energy threshold) Velmi jednoduchý detektor, kde se pro každý vzorek signálu spočítá jeho energie pomocí vzorce (5).

$$E_x = \sum_{n=0}^{N-1} x[n]^2 \tag{5}$$

Pak už je jen třeba získat referenční hodnotu výkonu šumu $\hat{E}_v$ (6), která se většinou získá jako průměrná hodnota výkonu prvních M vzorků, u kterých se předpokládá, že neobsahují řeč. U statického šumu je toto dostačující, ale pokud se může jednat o dynamický šum, je třeba tuto hodnotu adaptivně měnit. V obou dvou případech se vzorek označí za řečový, pokud splní podmínku nastaveného prahu $\mathrm{E_p}$ (7).

$$\hat{E}_v = \frac{\sum_{n=0}^{M-1} x[n]^2}{M} \tag{6}$$

$$E_p = \alpha \cdot \hat{E}_v \tag{7}$$

$$VAD(n) = \begin{cases} 1, & \text{pokud } E_x(n) \geq E_p(n) \\ 0, & \text{pokud } E_x(n) < E_p(n) \end{cases} \tag{8}$$

Ve vzorci (7) α udává výši rozhodovacího prahu. Například pokud by byla $\alpha = 1.5$, tak by byla stanovena podmínka, že výkon řečového segmentu musí být minimálně o 50% větší než výkon šumu, a rozhodnutí o přítomnosti řeči se pak řídí dle vztahu (8).

U adaptivního přístupu se pak průměrná hodnota $\hat{E}_v$ mění v závislosti s každým vzorkem, který je označen klasifikátorem jako neřečový. To lze například realizovat pomocí adaptivního okénka o určité délce s faktorem zapomínání, kde hodnoty šumu nejblíže aktuálnímu vzorku mají největší váhu a naopak hodnoty nejdál změní průměr jen minimálně. Výpočet odhadu výkonu šumu pro n-tý vzorek signálu je vidět v rovnici (9).

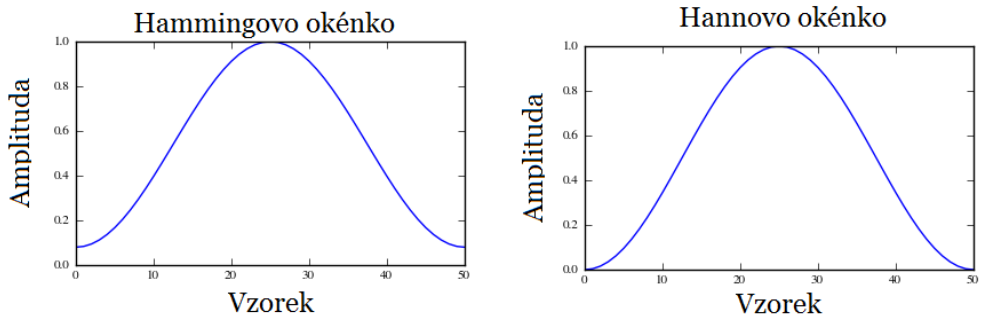
$$\hat{E}_v[n] = \frac{\sum_{i=0}^{L-1} \alpha^i \cdot x[n-i]^2}{L} \tag{9}$$

kde α je zmíněný faktor zapomínání, který se pohybuje v rozsahu (0;1>, kdy při 1 se hodnoty vůbec nezmenšují a jedná se o klasický vážený průměr a naopak je-li α blízko nule, tak se hodnoty velmi rychle snižují (jsou zapomenuty) a prakticky se jedná pouze o aktuální hodnotu výkonu vydělenou velikostí okna. Index i představuje vzdálenost od aktuálního vzorku.

Další možností, je-li znám celý signál, je použít nekauzální adaptivní odhad (kdy pracujeme i s budoucími hodnotami) pomocí tzv. okénkové funkce. Na Obrázku č. 1 je vidět ukázka takových funkcí. Aktuální hodnota vzorku je přesně uprostřed, takže nejbližší okolí vzorku má největší váhu.

Takto se pak výkon rámce spočítá jako součet výkonu M okolních rámců, kdy každý z nich je vynásobený okénkovou váhou (viz. vzorec (10)). Jedná se tedy o jistý průměr, kde se rychlé a velké změny ve velké míře potlačí (v závislosti na velikosti a typu okénka).

$$E_{s,i} = \sum_{n=0}^{M-1} x_i[n]^2 \cdot w[n] \tag{10}$$



Obrázek 1: Ukázka okénkových funkcí

Hlavní rozdíl mezi Hammingových a Hannovým okénkem je, že Hannovo okénko koncové hodnoty potlačuje úplně a Hammingovo je pouze snižuje. Jejich použití závisí na potřebách algoritmu.

Rychlost průchodů nulou (Zero Crossing Rate) ZCR je také jednoduchý detektor, který je založený na frekvenci signálu. U signálu rozděleného na rámce se u jednotlivých rámců počítá jejich ZCR (11). Tato hodnota vypovídá o tom, jak rychle signál v daném rámci prochází nulou a charakterizuje tedy frekvenci signálu. Čím větší ZCR, tím větší frekvence, s kterou signál prochází nulou.

Tato informace se pak používá k detekci řečových segmentů dle předpokládaného charakteru šumu. Jak již bylo řečeno lidská řeč se pohybuje mezi 300 Hz a 3kHz a šum má typicky větší frekvenci, takže lze stanovit hranici, která bude efektivně oddělovat řeč od šumu.

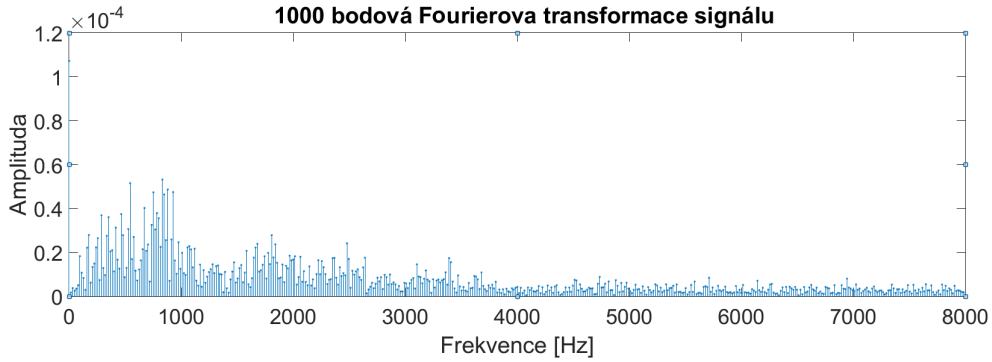
$$ZCR = \frac{1}{2} \sum_{n=0}^{N-1} |\text{sgn } x[n] - \text{sgn } x[n-1]| \tag{11}$$

kde sgn je signum funkce, definovaná jako (12)

$$\text{sgn } x = \begin{cases} 1, & \text{pokud } x > 0 \\ 0, & \text{pokud } x = 0 \\ -1, & \text{pokud } x < 0 \end{cases} \tag{12}$$

1.5.3 Analýza signálu ve frekvenční oblasti

Z časové oblasti signálu se do frekvenční dostaneme úpravou signálu pomocí diskrétní Fourierovy transformace (DFT), jejíž aplikací na řečový signál získáme jeho spektrum. Ukázka spektra je zobrazena v Grafu č. 2 vygenerovaného pomocí prostředí Matlab. V této oblasti pak získáváme komplexnější informace o charakteru signálu (např. jeho harmonické frekvence, ze kterých je složen).



Graf 2: Ukázka signálu obsahující řeč zobrazeného ve frekvenční oblasti

V grafu je vidět, že skutečně největší amplitudu mají frekvence z rozsahu zhruba 300 Hz až 3000 Hz, což by odpovídalo lidské řeči. Ačkoli se jedná o nahrávku čisté řeči, Fourierova transformace ukazuje i složky u vysokých frekvencí, ačkoli malé. Tyto složky jsou většinou způsobené nepřesností nahrávacích prvků, nicméně pro lidské ucho jsou tyto zvuky přehlušeny řečí a jsou prakticky nerozlišitelné lidským uchem.

Detekce aktivních frekvenčních pásem Zjištění aktivních frekvenčních komponent v signálu patří mezi základní informace, které můžeme o signálu získat. Zde lze opět využít frekvenčního rozsahu lidské řeči pro klasifikaci, jestli se v signálu složka z daného rozmezí objevuje ve větší míře či nikoli.

Kepstrální detektor Mezi komplexnější detektory patří kepstrální detektor [4]. Koncept detekce pomocí kepstrálních příznaků vznikl kvůli snaze odstranit limitace jednoduchých detektorů, které jsou závislé na úrovni amplitudy a množství šumu v nahrávce. Díky kepstrální analýze signálu jsme schopni najít skryté charakteristiky lidské řeči, které pomáhají s efektivnější detekcí řeči v nahrávce.

V rovnici (13) je vidět výpočet kepstrálních příznaků pro kepstrální integrální detektor [9]. Příznaky jsou získány jako reálná část inverzní Fourierovy transformace z logaritmu spektra signálu.

$$c_i[k] = Re\{IDFT\{log\,|DFT\{x_i[n]\}|\}\} \tag{13}$$

kde index i znamená i-tý rámec vstupního signálu a $c_i[k]$ představuje kepstrum signálu i-tého rámce v čase k

Tyto příznaky se pak používají k odhadnutí kepstrální vzdálenosti od průměrného kepstra šumu. A podle této vzdálenosti se určí, zda je řeč přítomna či nikoli.

1.6 Neuronové sítě

Umělá neuronová síť (Artificial Neural Network) je výpočetní model, jejímž vzorem je chování biologických nervových systémů, jako je mozek, při zpracovávání informací. Jedná se o jeden ze základních konceptů umělé inteligence, kdy se počítač snaží sám naučit jak vyřešit neznámou problematiku.

Neuronové sítě - vznik Koncept neuronových sítí vznikl již v roce 1943 [10], kdy se vytvořily dva vědecké proudy. Jeden se zaměřil na biologické procesy v mozku a druhý na aplikaci neuronových sítí pro umělou inteligenci. Ačkoli byl koncept neuronových sítí znám velmi dlouho, až v dnešní době se začínají používat ve velké míře. Jedním z důvodů, proč se neuronové sítě nepoužívaly, bylo, že dosud nebyl k dispozici dostatečný výkon techniky, aby byly sítě efektivní. Díky pokrokům v oblasti výpočetní techniky jsme nyní schopni trénovat neuronové sítě v reálném (konečném) čase a s přijatelnými výsledky.

Jako vstupní data mohou posloužit nízkoúrovňové příznaky (jako jsou například logaritmické frekvenční příznaky), se kterými se síť učí pokročilejší klasifikaci charakteru digitální stopy, čímž se VAD stane univerzálnějším. Nebo předem připravené příznaky (například MFCC příznaky [11]), které již v sobě nesou velmi specializovanou informaci, na potenciální úkor robustnosti.

Neuronové sítě obsahují vstupní, výstupní a případně skryté vrstvy. Každá vrstva je tvořena neurony, které jsou složeny z váhy a biasu.

1.6.1 Váhy (Weights) a Bias

Tyto hodnoty jsou nejdůležitější z hlediska učení. Síť si právě tyto hodnoty nastavuje tak, aby z daného vstupu dostala daný výstup. Když vrstva dostane vstupní data, tak je nejdříve vynásobí váhou a pak k výsledku přičte bias.

1.6.2 Dopředná topologie sítě se zpětnou propagací chyb (Feedforward NN with Backpropagation)

V síti s dopřednou topologií signál prochází pouze jedním směrem ze vstupu přes skyté vrstvy do výstupu. Jinými slovy neurony jsou spojeny pouze s bezprostředně předchozími a následujícími neurony a netvoří cykly.

Hlavním předpokladem zpětné propagace chyb je, že výstupní funkce, aktivační funkce a chybová funkce musí mít derivaci, jelikož hodnoty jejich derivací jsou použity k výpočtu jednotlivých gradientů vah. Backpropagation znamená, že když se data dostanou až na výstup, tak síť porovná tento výstup s tím, jak má ve skutečnosti vypadat (tzv. supervised training,

viz. kapitola 2.2) a algoritmus pak prochází zpátky a pomocí derivace chybové funkce a derivace příslušných aktivačních funkcí získá gradient chyby pro každou váhu v síti. Nová hodnota váhy pak je rozdílem aktuální hodnoty váhy a hodnoty gradientu vypočítaného pro danou váhu, která je případně ještě vynásobená koeficientem učení (viz. níže).

Tímto způsobem projde celou síť a přenastaví všechny hodnoty vah a případně biasu. Jakmile jsou hodnoty vah aktualizovány, průchod se ukončí a začíná nové kolo učení.

Optimalizační kritérium Síť již během trénování vyhodnocuje svoji účinnost. To, jakým způsobem svoji účinnost hodnotí, říká funkce optimalizačního kritéria. Ladění sítě se provádí pomocí této funkce, jelikož zjišťujeme, jak změnou parametrů sítě dosáhne síť menší hodnoty této chybové/kriteriální funkce.

Koeficient učení Koeficient učení představuje velikost trénovacího kroku. Příliš velká hodnota může způsobit alternování sítě, kdy efektivita není optimální a naopak příliš malá hodnota způsobí, že se síť bude učit příliš pomalu a může skončit v nějakém lokálním minimu kriteriální funkce. Koeficient učení se může nastavit manuálně, kdy se obvykle začíná s velkou hodnotou, a když se síť přestane zlepšovat, tak se hodnota koeficientu sníží, čímž docílíme, že se síť postupně ustálí ve své efektivitě okolo určité hodnoty.

1.6.3 Mělké neuronové sítě (Shallow neural network)

Mělká neuronová síť se vyznačuje tím, že má pouze 1 skrytou vrstvu (oproti hluboké, která jich má víc). Je ideální pro práci s jednoduchými úlohami (velmi triviální příklad je trénování sítě pro výpočet funkce f(x) = 5x), jelikož taková síť je rychlá a účinná (je-li správně natrénovaná). Ovšem pro komplexnější problematiky se stává neúčinnou, jelikož se není schopna adaptovat pro hlubší spojitosti v datech. V takových případech je třeba využít hlubokých neuronových sítí.

1.6.4 Hluboké neuronové sítě (Deep neural network)

Základní koncept hlubokého učení neuronových sítí (DNN) byl navržen již v roce 1965 [12]. Tyto sítě jsou velmi silný nástroj pro extrakci vlastností. Jsou schopny najít skryté spojitosti v datech, které by mělké sítě nezvládly objevit.

Výpočetní náročnost sítí závisí na počtu neuronů, jelikož každý neuron má svoji váhu a bias, které se při trénování přepočítávají. A hluboké sítě, které mají více skrytých vrstev, mají i obecně větší počet neuronů (záleží na nastavení sítě). Zároveň čím více skrytých vrstev síť obsahuje, tím více zpravidla potřebuje epoch, než začne konvergovat (neboli mít tendenci se ustálit). Dále se pak řeší například počáteční inicializace vah (viz. níže) a tzv. Problém mizícího gradientu (Vanishing gradient problem) [13], což označuje proces, kdy s velkým

počtem vrstev se u zpětné propagace chyb gradient velmi zmenší (zmizí). To je problém, protože se pak síť není schopná správně učit. Tomuto problému lze většinou předejít pomocí vhodné počáteční inicializace, anebo použitím vhodných aktivačních funkcí (viz. kapitola 2.5).

Jistou odnoží hlubokých sítí jsou tzv. Deep belief sítě (DBN) [14]. Ty se liší tím, že síť je nejdříve speciálně trénovaná předem na malé trénovací sadě metodou učení bez učitele. Cílem tohoto postupu je vytvořit vhodné inicializační hodnoty vah a biasu, které urychlí konvergenci sítě. Zjistilo se ovšem, že s dostatečně velkými daty a náhodnou inicializací vah lze toto prakticky zanedbat.

1.6.5 Neuronové sítě pro robustní odhad SNR

K řešení problematiky odhadu úrovně SNR pomocí neuronových sítí vedly 2 hlavní cesty. Buďto zhotovit síť, která rozpozná, kde se v nahrávce vyskytuje řeč, a následně pomocí algoritmu odhadnout SNR. Nebo natrénovat síť přímo na přibližný odhad SNR. Zvolil jsem první přístup, jelikož se díky tomu problematika rozdělí na 2 menší problémy a zároveň pak lze zmíněnou síť použít i jako samostatný modul pro jiné rozpoznávací účely.

Další důležitá výhoda prvního přístupu je, že algoritmus získá pro každý vzorek jeho odhadovaný výkon šumu a energie. To nám umožňuje dobře odhadnout GSNR, které je definované jako poměr energie řeči a šumu v signálu.

V případě druhého přístupu je toto velmi obtížné, jelikož by algoritmus měl k dispozici pouze informaci o LSNR (natrénovat síť na odhad GSNR je prakticky nereálné) a správně odhadnout GSNR z posloupnosti LSNR je velmi náročná úloha.

2 Navržený algoritmus pro odhad SNR

2.1 Konfigurace trénovací i testovací sady

Pro trénovací sadu jsem měl k dispozici 6000 řečových nahrávek (celkem 5.1 hodin zvukových stop, viz. kapitola 3.1), kde jsem každou nahrávku postupně sečetl (viz. níže) se 3 variantami aditivního šumu (z prostředí autobusu, kafeterie a chodníku). A každá takto zašuměná stopa byla vytvořena se 4 variantami různých hladin GSNR (-10,0,5 a 10 dB SNR). Tedy celkem 72000 trénovacích stop.

V rámci testování se této množině říká validační sada, která slouží ke zkoušce funkčnosti. Pokud síť nefunguje ani na validačních datech, tak nemá cenu pokračovat k testovací sadě a naopak pokud síť funguje pro validační sadu, tak to ještě neznamená, že bude efektivní pro testovací (tzv. problém overfittingu, viz. kapitola 2.5).

Pro účely testování jsem použil zbylých 256 řečových nahrávek (zhruba 13 minut audio stop). V případě testování se známými daty jsem je opět sloučil se 3 variantami aditivního šumu (ovšem použil jsem pouze soubory s šumem, které jsem nepoužil při trénování). A pro testování s neznámými daty jsem použil poslední typ šumu (Ulice).

Tyto nahrávky k dispozici jsem rozdělil do rámců o velikosti 512 vzorků s překryvem 256 vzorků a z těchto rámců jsem spočítal jejich vektor 39 frekvenčních příznaků.

Jako vstupní data jsem pak zvolil tento vektor spolu s kontextem 5 rámců před a 5 rámců za aktuálním rámcem. Tedy jeden vstupní vektor má velikost $11/cdot39 = 429$ příznaků.

V krajních případech, kdy rámec neměl 5 předchůdců nebo následovníků, jsem mezery vyplnil nulami.

Výstupní vektor má velikost 2 dle počtu kategorizačních tříd (řeč, šum). Jednička označovala příslušnost do dané kategorie a 0 naopak. To, jestli rámec je označen za řečový či nikoli, se řídilo na základě lokálního SNR při skládání řečové a šumové množiny. Pokud lokální SNR bylo větší jak -5 dB, tak byl tento rámec označen jako řečový.

2.2 Příprava signálu na zpracování sítí

Prvním krokem je vytvořit trénovací množinu, nad kterou máme úplnou kontrolu, co se SNR úrovně týče. Jak již bylo zmíněno, jednotlivé nahrávky se rozdělí na rámce, ke kterým se vypočítá charakteristický vektor frekvenčních příznaků. Tímto dostaneme sérii vektorů, představující celou digitální nahrávku, které poslouží jako vstupní data pro trénování (a testování) sítě.

Vstupní data

Pro tyto účely je vhodné mít množinu nahrávek s čistou řečí a množinu nahrávek s šumem. V tu chvíli jsme schopni naprosto přesně ovládat úroveň SNR ve výsledné nahrávce pomocí sečtení řečové a šumové nahrávky z těchto dvou množin s tím, že prvky šumové množiny jsou vynásobeny speciálním koeficientem k pro nastavení úrovně SNR.

Tento koeficient se počítá pro každou nahrávku zvlášť pomocí rovnice (14).

$$k = 10^{\frac{SNR}{-20}} \cdot \sqrt{\frac{\sigma_s^2}{\sigma_v^2}} \tag{14}$$

kde SNR je požadovaná úroveň globálního SNR, σ_s^2 je celková energie řeči a σ_s^v je celková energie šumu.

Tímto koeficientem pak vynásobíme každý vzorek šumu. Výslednou množinu šumu sečteme s množinou čisté řeči, čímž získáme zarušenou nahrávku řeči s exaktním globálním SNR.

Kategorizace rámců dle řečové aktivity určité úrovně

Síť využívá metody učení s učitelem (supervised training), což je učení, kdy síti předáváme krom vstupních dat i cílová data, tedy jak má vypadat výstup sítě při daných vstupních datech. Kvůli tomu potřebujeme mimo vstupních dat ještě i příslušná cílová data. To v kontextu VAD sítě znamená údaj, říkající jestli se v daném zvukovém rámci nachází řeč nebo ne.

Je třeba si tedy zvolit hranici LSNR (navržený algoritmus používá hranici -5 dB), kdy zarušenou řeč ještě považujeme skutečně za řeč. A následně pomocí algoritmu s touto hranicí sestrojit VAD vektor obsahující námi chtěnou informaci o výskytu řeči. Čímž dostáváme kategorizační vektor pro výstupní množinu sítě. Tedy v tomto případě máme 2 kategorie (řečový rámec a neřečový rámec).

2.3 Logaritmické frekvenční příznaky signálu

Analýzu signálu nám značně ulehčuje výběr charakteristické vlastnosti z frekvenčního spektra. Pro tento účel jsem vybral logaritmické frekvenční příznaky. Signál se v časové oblasti rozdělí na rámce o velikosti L s překryvem o délce O, které se vynásobí okénkovou funkcí. Pro tyto rámce se pak získává charakterizující frekvenční vektor C_f příznaků o délce K. Ten se vypočítá pomocí logaritmu diskrétní Fourierovy transformace absolutní hodnoty daného rámce a následného vážení trojúhelníkovými okénky.

$$C_i = log(|DFT(x_i)|) \tag{15}$$

kde x_i je i-tý rámec vstupního signálu

Ačkoli je možné využít celý vektor C_i, který má velikost stanovenou velikostí DFT (L/2 + 1, L je velikost rámce), tak je více než dostačující použít prvních 39 příznaků pro charakteristiku daného rámce.

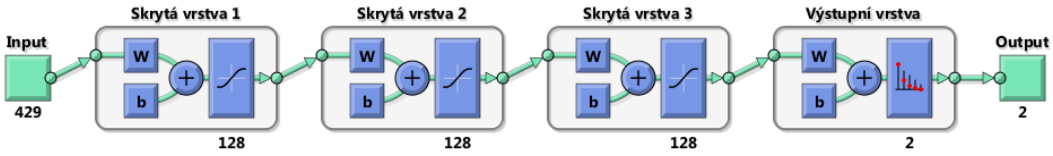
Jak již bylo zmíněno vektor frekvenčních logaritmických příznaků C_i je pak následně zvážen trojúhelníkovými okénky o zmíněné velikosti 39 prvků, čímž dostaneme finální vektor 39 frekvenčních logaritmických příznaků C_i.

V této práci konkrétně používám příznaky z filtrbanky, které získávám pomocí algoritmu Melcepst ze sady Voicebox [15]. Tento algoritmus nad příznaky ještě provádí Diskrétní Kosinovu transformaci (DCT), ovšem já tuto operaci již neprovádím.

2.4 Konfigurace sítě

V této sekci jsou popsány veškeré hyperparametry sítě, které bylo potřeba nastavit pro správnou funkci sítě.

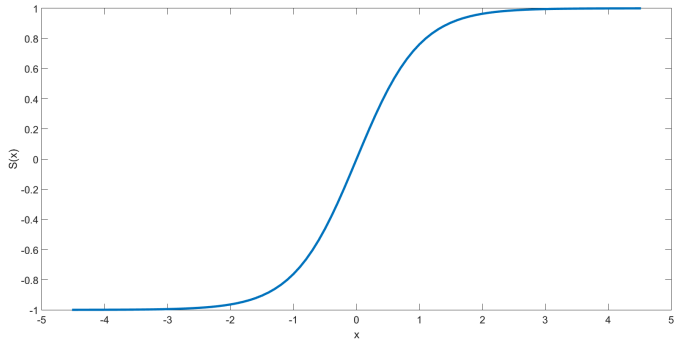
Deep neural network K získání výsledku byla použita hluboká neuronová síť, konkrétně se 4 vrstvami (3 skryté a 1 výstupní). Každá skytá vrstva měla aktivační funkci Tansig a 128 neuronů. Výstupní vrstva měla aktivační funkci Softmax a 2 neurony. Jako optimalizační kritérium bylo použito Cross Entropy a jako trénovací funkce SCG (Scaled Conjugate Gradient, viz kapitola 2.5).



Obrázek 2: Schéma neuronové sítě vygenerované prostředím Matlab

Tansig Pokud ve vstupních datech hledáme nelinearity, tak zpravidla volíme sigmoidní funkce (funkce, který mají sigmoidní průběh viz. Graf č. 3). Tansig funkce má pak rozsah hodnot <-1,1>.

$$S(c) = \frac{2}{1 + e^{-2 \cdot c}} - 1 \tag{16}$$



Graf 3: Průběh sigmoidní funkce Tansig vygenerované prostředím Matlab

Softmax Jelikož výstupem sítě je klasifikace do kategorií, tak je vhodné vybrat takovou výstupní funkci, která právě počítá, s jakou pravděpodobností budou vstupní data patřit do jaké kategorie (tedy součet pravděpodobností dá dohromady 1). Softmax [16] je funkce vhodná pro tento účel.

Cross Entropy Hodnotící kritérium pro kategorizační algoritmus je tzv. Cross Entropy [17], které se snaží minimalizovat negativní logaritmickou pravděpodobnost pro daný výstup, tedy maximalizovat pravděpodobnost správného výstupu pro daný vstup. Důvod proč nepoužijeme MSE (Mean Square Error) je ten, že MSE hodnotí výstup na základě vzdálenosti od cílové hodnoty. My ovšem potřebujeme dávat velkou penalizaci za špatně klasifikovaný výsledek, nikoliv za vzdálenost od cíle.

2.5 Implementační detaily - Jak vybrat hyperparametry sítě

Pod hyperparametry sítě rozumíme veškerá nastavení, která ovlivňují chování sítě. Tedy například počet vrstev, počet neuronů, přechodové (aktivační) funkce apod.

Vrstvy a počet neuronů v nich

Jedním z hlavních hyperparametrů je počet vrstev a jejich neuronů. Jejich volba přímo ovlivňuje schopnost sítě najít skryté souvislosti. Jak bylo možné vidět na obrázku 4, více není vždy lépe. Pokud na jednoduchý problém aplikujeme velmi hlubokou neuronovou síť, může se stát, že síť začne nalézat spojitosti i tam, kde nejsou. Je to dané tím, že velká síť pomaleji konverguje a má větší sklony k tzv. Overfittingu.

Overfitting je problém, kdy se síť „přeučí“ z trénovacích dat a následně na testovacích datech je velmi neefektivní. To je způsobeno tím, že má k dispozici mnoho volných parametrů, aby modelovala i nepodstatné detaily vstupních dat, které pak právě zhoršují efektivitu na testovací množině.

Trénovací funkce

Trénovací funkce ovlivňuje celý proces učení a pro síť se zpětnou propagací chyby se doporučuje použít SCG funkce(Scaled Conjugate Gradient), která je schopná si optimální koeficient učení (tedy velikost trénovacího kroku) vypočítat sama.

Optimalizační kritérium

Síť již během trénování vyhodnocuje svoji účinnost. To, jakým způsobem svoji účinnosti hodnotí, říká funkce optimalizačního kritéria. Jelikož výstupem mé sítě je klasifikace do kategorií, zvolil jsem Cross Entropy, která je právě na tuto problematiku ideální. Ale pokud by výstupem měly být například přepočítané číselné hodnoty, pak by bylo vhodné použít MSE (Mean Square Error), které je navrženo pro počítání vzdálenosti od cílové hodnoty.

Aktivační funkce

Aktivační funkce je velmi důležité nastavení sítě, jelikož jakmile vrstva vynásobí vstup váhovou maticí a přičte matici biasu, tak se výsledek vloží právě do této funkce a vrstva ho předá dál. Má tedy velký vliv na chování celé sítě.

Počet trénovacích epoch

Tento hyperparametr obvykle nelze předem určit, je třeba průběžně hodnotit jednotlivé epochy sítě a v případě, že síť již konverguje k minimu kriteriální funkce, tak je třeba trénink zastavit a pomocí testovací sady vybrat nejefektivnější epochu pro následné použití.

Hyperparametry sítě je nutné zvolit v závislosti na charakteru vstupních dat a očekávaného výstupu sítě. Některé z nich ovšem je nutné zvolit až podle výsledků experimentů.

2.6 Vyhlazení VAD výstupu

Poté, co ze sítě získáme VAD informaci, je možné se ji pokusit ještě zlepšit dalším zpracováním (post-processing). Toho lze docílit například tzv. vyhlazováním. Zlepšení spočívá v tom, že v některých případech VAD mění stavy příliš rychle a není pravděpodobné, že by slova byla tak krátká.

Jedná se o proces, kdy analyzujeme výskyt řečových a šumových segmentů a na základě stanovených kritérií VAD informaci upravíme. V této práci jsem použil jednoduché vyhlazování v podobě filtru klouzavého průměru, který zjistí, zda není nějaký neřečový segment bezprostředně obklopen z obou stran několika řečovými segmenty. V takovém případě je velmi pravděpodobné, že i tento segment bude řečový.

2.7 Kritéria hodnocení efektivity sítě a odhadovacího algoritmu

Pro správné zhodnocení efektivity je vždy potřeba zvolit vhodné kritérium, které ji objektivně a výstižně charakterizuje.

2.7.1 Kritéria VAD sítě

V případě hodnocení efektivity detekce přítomnosti lidské řeči v nahrávce se zabýváme hlavně úspěšností kategorizace jednotlivých segmentů do 2 tříd (řeč/šum).

Pro tento druh úlohy se používají následující kritéria:

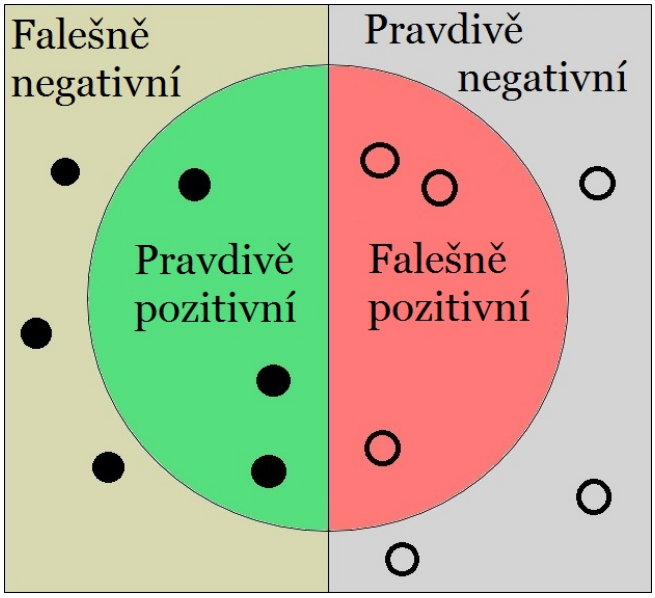
Přesnost (Precision) Určuje, s jakou pravděpodobností síť správně klasifikuje řečový segment. Tedy pokud má VAD prvek vysokou přesnost, znamená to, že nemá problémy rozlišit řeč od šumu a málokdy je zamění.

Sensitivita (Recall) Určuje, kolik správných řečových segmentů vybere ze všech řečových segmentů. Jinými slovy pokud VAD prvek má vysokou sensitivitu, tak byl schopen najít podstatnou část řečových segmentů v nahrávce. Ovšem tento údaj nelze hodnotit sám o sobě, protože pokud by VAD prvek všechny segmenty označil jako řeč (ačkoli by se tam vyskytoval i šum), tak sensitivita by byla 100 %.

Míra shody (Hitrate) Jedná se o počet správně vyhodnocených segmentů (tedy řečových i neřečových) vydělený počtem všech segmentů. Tento údaj je komplementární k přesnosti a sensitivitě. Pokud VAD prvek má vysokou přesnost a sensitivitu, tak bude mít i vysokou míru shody, jelikož správně klasifikoval řeč jako řeč a šum jako šum. Vypočíta se jako zmíněný počet vydělený počtem všech segmentů.

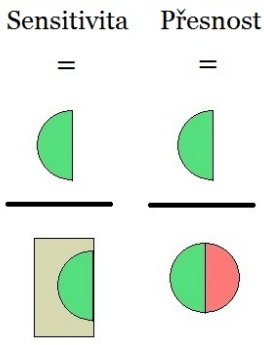
Pro objektivní hodnocení sítě stačí kombinace přesnosti a sensitivity. Ale míra shody je na první pohled mnohdy více vypovídající, jelikož přímo říká správnost všech kategorizací.

Na Obrázku č.4 je ukázáno, jakým způsobem se počítá sensitivita a přesnost pomocí údajů zobrazených na Obrázku č. 3.



Obrázek 3: Přehled klasifikací výsledku

- Pravdivě negativní - Neřečový prvek, který byl skutečně označen jako neřečový.
- Falešně negativní - Neřečový prvek, který byl špatně označen jako řečový.
- Pravdivě pozitivní - Řečový prvek, který byl skutečně označen jako řečový.
- Falešně pozitivní - Řečový prvek, který byl špatně označen jako neřečový.



Obrázek 4: Výpočet přesnosti a sensitivity

2.7.2 Kritéria algoritmu pro odhad SNR úrovně

U odhadu SNR úrovně se již hodnotí vzdálenost od předpokládané hodnoty, tudíž je třeba využít jiných hodnotících kritérií.

Bias

Bias nám udává, jak moc se v průměru liší odhadovaná hodnota od předpokládané. Tedy pokud mají odhady malý Bias, znamená to, že se příliš nevzdálily od předpokládané hodnoty.

$$B(\hat{\theta}) = \frac{\sum(\theta_i - \theta_t)}{R - 1} \tag{17}$$

kde θ_i jsou jednotlivé hodnoty odhadu SNR úrovně, θ_t je očekávaná hodnota odhadu a R je celkový počet odhadů.

Variance

Variance je očekávaná hodnota kvadrátu odchylek vzorků. Používá se k indikaci, jak daleko v průměru se liší jednotlivé odhady od sebe. Tedy odhady s malou Variancí budou k sobě velmi blízko (vytvoří shluk).

$$Var(\hat{\theta}) = \frac{\sum(\theta_i - \overline{\hat{\theta}})^2}{R - 1} \tag{18}$$

kde x_i jsou jednotlivé odhady SNR úrovně, $\overline{\hat{\theta}}$ je průměrná hodnota odhadů a R je celkový počet odhadů. Jedná se tedy o průměrnou hodnotu rozdílu odhadu a průměru.

Odhad s malou Variancí tedy nemusí nutně dávat správný výsledek. Pokud má velký Bias, tak jsou sice odhady blízko sebe, ale jejich hodnota je daleko od cílové.

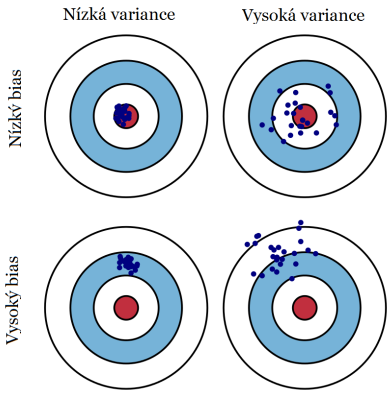
Mean Square Error (MSE)

Toto kritérium je vypočítané z předchozích dvou a udává nám očekávanou hodnotu kvadrátu chyby.

$$MSE(\hat{\theta}) = B(\hat{\theta})^2 + Var(\hat{\theta})^2 \tag{19}$$

Obecně tedy platí, že čím menší hodnota MSE, tím přesnější odhad.

Na Obrázku č. 5 je vidět grafická analogie v podobě střeleckého terče, kde jednotlivé modré tečky představují zásah šípem do terče.



Obrázek 5: Grafická ukázka Biasu a Variance dle Scotta Fortmann-Roe [18]

2.8 WADA

WADA používá statistický algoritmus pro odhad SNR, který je založený na předpokladu, že amplitudová distribuce čisté řeči je přibližně stejná jako Gamma distribuce (s tvarujícím parametrem 0,4). Algoritmus zároveň předpokládá, že aditivní šum je Gaussovým šumem. Za těchto předpokladů je WADA schopný odhadnout úroveň SNR v nahrávce.

Symetrické gamma rozložení je dobrou aproximací amplitudové distribuce velkého řečového korpusu. Konkrétně funkce pravděpodobnostní hustoty může být reprezentovaná následovně:

$$f_x(x|\beta_x) = \frac{\beta_x}{2\Gamma(\alpha_x)}(\beta_x|x|)^{\alpha_x - 1} exp(-\beta_x|x|) \tag{20}$$

kde x je amplituda řeči, parametr α_x udává tvar a β_x udává rychlost gamma distribuce Γ

Samotná hodnota SNR je pak odhadována pomocí vzdálenosti amplitudové distribuce signálu od gamma distribuce.

3 VAD Experimenty s různými parametry neuronové sítě

3.1 Vstupní data

CHiME jako zdroj reálného šumu

Jako podklady pro aditivní šum jsem použil databázi reálného šumu z projektu CHiME challenge [19]. Šumové nahrávky mají vzorkovací frekvenci 16 kHz a každá varianta šumu obsahuje přes 10 hodin reálného šumu.

Jednotlivé nahrávky byly pořízeny pomocí tabletového zařízení, které má 6 integrovaných mikrofonů a nahrávač TASCAM DR-680, který je schopen nahrávat až 24-bitovou informaci při vzorkovací frekvenci 48 kHz. Audio signál byl poté zdecimován na 16-bitovou informaci se vzorkovací frekvencí 16 kHz z důvodu distribuce.

V databázi se vyskytují tyto 4 varianty šumu:

Autobus Nahráno z prostředí autobusu, tento šum má stacionární charakter.

Kafeterie Prostředí kafeterie se jeví jako nejvíce dynamické, jelikož v pozadí je lidská mluva a např. „cinknutí příboru“.

Chodník V těchto nahrávkách se vyskytuje velká míra konverzací v pozadí, ačkoli nejsou velmi zřetelné.

Ulice Jedná se o nahrávky projíždějících aut, tento šum má také stacionární charakter.

TiMIT jako zdroj čisté řeči

Pro přesné vytváření zašuměných nahrávek potřebuji kromě šumových stop i nahrávky s čistou řečí. K tomuto účelu posloužila zvuková databáze TIMIT [20], která obsahuje 6256 různých řečových nahrávek s mnoha různými řečníky (různého pohlaví). Nahrávky byly pořízeny také se vzorkovací frekvencí 16 kHz a mají dohromady něco málo přes 5 hodin řečových nahrávek.

V celé databázi se vyskytuje celkem 2342 vět (o různých délkách), které se opakují. Věty jsou v anglickém jazyce a většina nahrávek trvá asi 4 sekundy, nejdelší pak trvá 8 sekund (kdy řečník mluví cíleně pomalu).

Vytvoření zašuměné nahrávky

Konkrétní zašuměné nahrávky pak byly vytvořeny pomocí součtu nahrávek čisté řeči spolu s reálným šumem (který byl vynásobený vhodným koeficientem pro chtěnou úroveň SNR).

Rozdělení nahrávek na rámce

Každá nahrávka se rozdělí na rámce o délce 512 vzorků s překryvem 256 vzorků (tedy poloviční překryv).

Vytvoření výstupních vektorů

Jelikož jsem měl absolutní kontrolu nad SNR nahrávek, tak jsem si uložil LSNR každého rámce do speciálního vektoru, podle kterého jsem pak rozhodoval o přítomnosti řeči v nahrávce (tzv. ideální VAD). Tuto hranici jsem nakonec zvolil -5 dB pro LSNR (více k volbě hranice viz. kapitola 4.3). Po zpracování signálu touto hranicí, jsem získal cílový vektor pro učení sítě.

Vektor s logaritmickými frekvenčními příznaky

Z každého rámce je následně vyextrahován vektor s 39 frekvenčními příznaky, které reprezentují daný rámec ve frekvenční oblasti.

Odečtení nulové střední hodnoty

Dále bylo třeba normalizovat veškeré digitální nahrávky na tzv. nulovou střední hodnotu (Zero Mean Value). Vektor středních hodnot obsahoval 39 průměrných hodnot frekvenčních příznaků (na první pozici vektoru středních hodnot byl průměr všech příznaků na první pozici apod.). Tento vektor byl pak odečten od veškerých dat, čímž jsem v rámci těchto dat dostal nulovou střední hodnotu.

Tento vektor bylo třeba zachovat, jelikož bylo nutné ho odečíst i od testovací množiny.

Kontext

Dalším krokem je kontextový vektor, který vznikne přidáním vektorů 5 rámců před a 5 rámců za aktuálním rámcem, čímž vznikne vektor o délce 5x39 + 39 + 5x39, tedy 429.

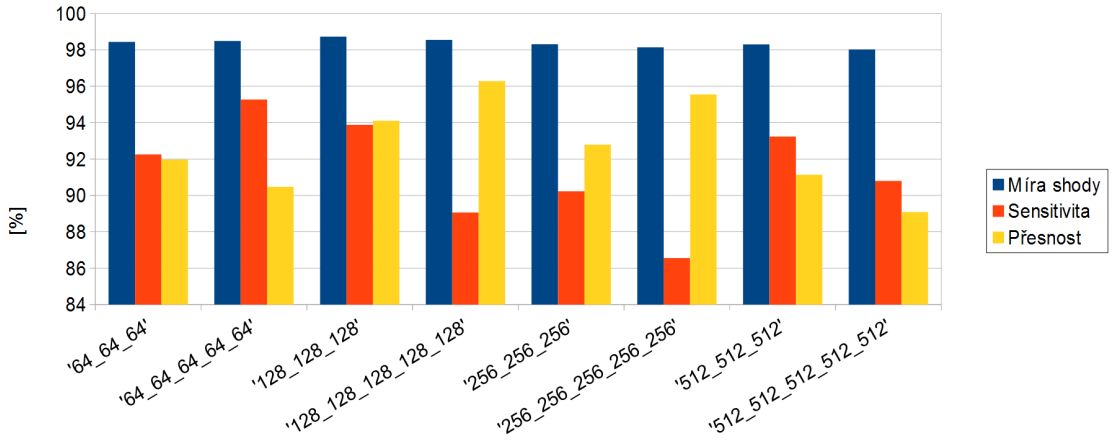
3.2 VAD pro umělý (Gaussův) šum

Než jsem začal přímo s problematikou detekce přítomnosti řeči v reálném zarušeném prostředí, zvolil jsem experimenty s umělým (Gaussovým) šumem pro získání hrubého přehledu, jaké hyperparametry sítě mají pro problematiku VAD nejlepší výsledky. Gaussův šum je stacionární a je pro úlohu VAD jednoduchý případ. Je tedy pravděpodobné, že natrénovaná síť bude mít vysokou míru shody.

Statistika Pro toto validační měření bylo vybráno celkem 80 zašuměných zvukových stop o hodnotách SNR -10, -5, 0, 5,10 dB. Tedy celkem 400 různých nahrávek.

Vybral jsem několik možných hyperparametrů sítě, na kterých jsem síť natrénoval a následně zjistil efektivitu sítě na validačních datech.

Každá konfigurace sítě proběhla celkem přes 10 trénovacích epoch (iterací). Do Grafu č. 4 jsem vybral epochy s nejlepší efektivitou pro srovnání s ostatními konfiguracemi.

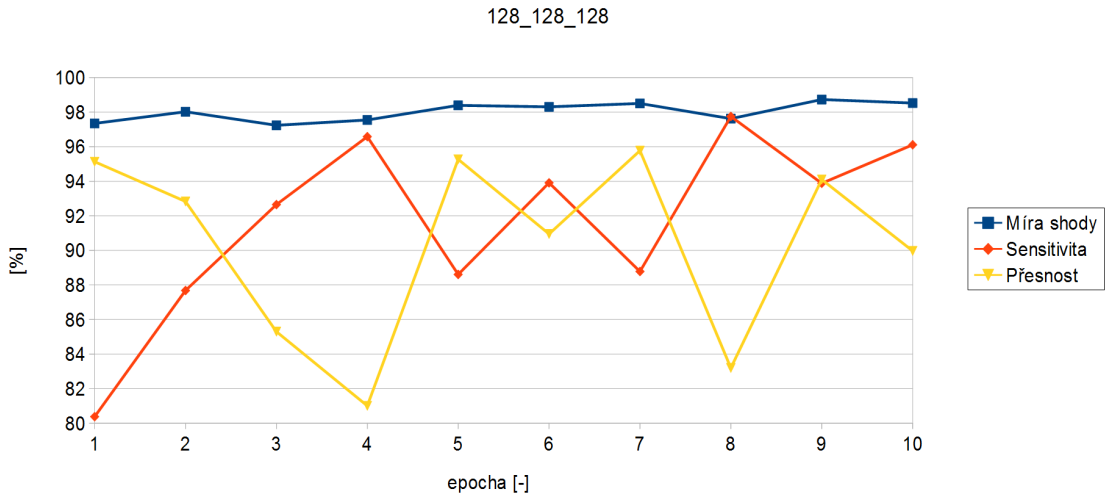


Graf 4: Různé hyperparametry sítě a jejich výsledky

Osa X představuje různé konfigurace, kde počet čísel oddělených podtržítkem označuje počet skrytých vrstev a samotná čísla udávají počet neuronů v příslušné vrstvě. Pro vysvětlivky ohledně hodnotících kritérií viz. kapitola 2.7.

Pro daný problém mají nejlepší míru shody konfigurace s 64 nebo 128 neurony. Nejlépe se umístila konfigurace 3 skrytých vrstev, každá o 128 neuronech s Mírou shody 98,7%, Sensitivitou 93,9% a Přesností 94,1%. Rozšiřování a prohlubování sítě nepřinášelo zlepšení výsledku.

V Grafu č. 5 je zobrazen postupný průběh efektivity jednotlivých trénovacích epoch této konfigurace na validačních datech.



Graf 5: Efektivita jednotlivých trénovacích epoch nejlepší konfigurace

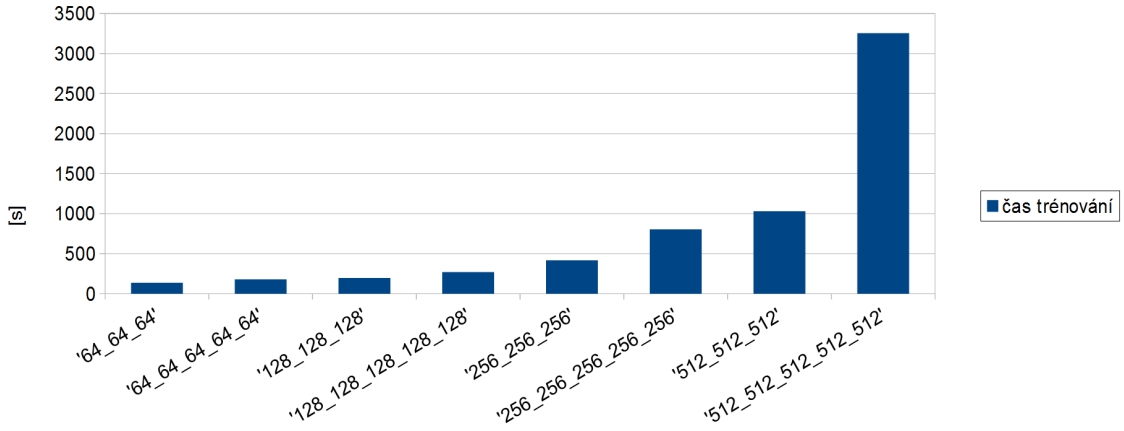
Nejlepších výsledků dosáhla 9. trénovací epocha, která byla zobrazena i na Obrázku č. 4. Zajímavé je, že síť může do jisté míry alternovat mezi efektivním odhadem a ne příliš dobrým odhadem. To je dáno většinou stochaistickým gradientem. Což znamená, že tím, že se síť učí pomocí malých podmnožin trénovací sady (minibatche), tak se může stát, že se naskládají v nevhodném pořadí a kriteriální funkce se může i zhoršit.

Například hned první epocha má relativně špatnou sensitivitu, neboť oproti ostatním epochám nebyla schopna správně rozpoznat tolik řečových segmentů. Oproti tomu její přesnost je velmi vysoká, což znamená, že když už síť segment klasifikovala jako řečový, tak tomu tak s vysokou pravděpodobností skutečně bylo.

Oproti tomu ve 4. epoše síť označovala většinu segmentů jako řečové. Důsledkem byla velmi vysoká sensitivita, jelikož většina skutečných řečových segmentů byla správně vybrána, ale přesnost nám říká, že tak označovala i segmenty šumové.

Z grafu je možné vysledovat, že síť v průběhu epoch začínala mít tendenci se ustálit ve své efektivitě, to znamená, že při větším počtu epoch by se efektivita lišila jen s malou odchylkou, ale spíše by se pohybovala okolo stejné hodnoty.

Ačkoli Míra shody, Sensitivita a Přesnost jsou nejdůležitějšími faktory, trénování sítě lze popsat ještě jedním kritériem, a tím je doba trénování. V Grafu č. 6 můžete vidět porovnání různých konfigurací z časového hlediska. Tyto časy se vztahují k dříve zobrazeným datům, tedy jak dlouho trvalo trénování 10 epoch s příslušnou konfigurací sítě.



Graf 6: Časová náročnost trénovaní daných sítí

Se vzrůstajícím počtem neuronů mají sítě schopnost vyjádřit složitější nelineární souvislosti mezi vstupem a výstupem, ale zároveň roste i výpočetní doba potřebná k natrénování sítě. Je tedy lepší zvolit menší počet vrstev a neuronů, pokud je to možné.

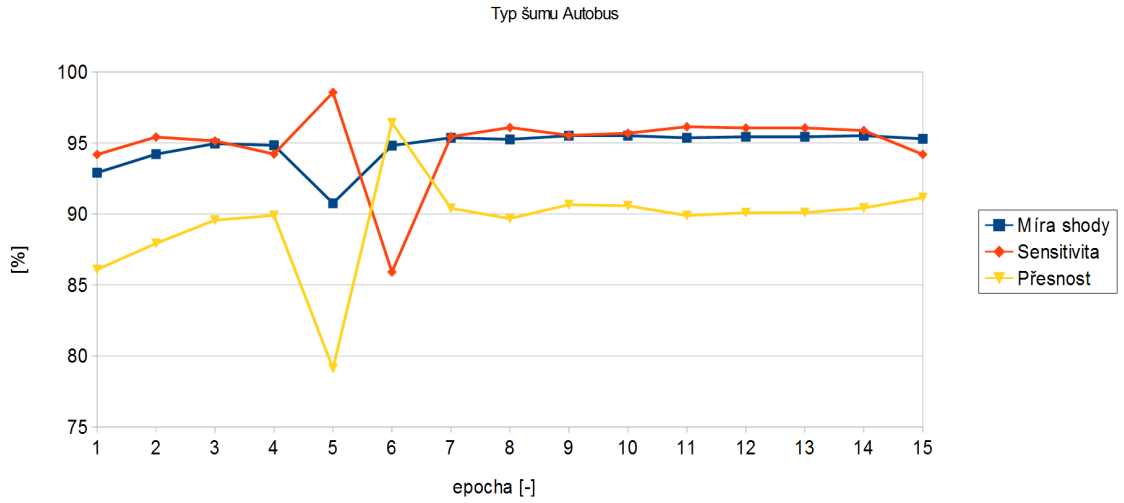
Tyto hodnoty byly získány na stolním počítači se 4 jádrovým procesorem AMD Phenom II X4 965 (3,4 GHz), 16 GB DDR3 RAM pamětí a 64 bitovým operačním systémem Windows 7.

Vzhledem k výsledkům z Obrázku č. 4 a nízkému trénovacímu času z Obrázku č. 6 jsem usoudil, že pro účely detekce přítomnosti řeči v signálu je optimální konfigurace sítě 3 skryté vrstvy se 128 neurony. Tyto parametry jsem tedy použil i pro trénování na reálném šumu.

3.3 VAD pro reálný šum

3.3.1 Validační sada

Pro validační test byly použity veškeré trénovací nahrávky (tedy 6000 zvukových stop, každá pro 3 různé druhy šumů a 4 různé úrovně SNR). Na validačních datech by měla mít síť z principu nejlepší výsledky, jelikož se přesně s těmito daty setkala při trénování. Může se ovšem stát, že kvůli robustnosti se může najít takový typ šumu, který bude mít lepší výsledky než validační data, přestože nebyl viděn při trénování.



Graf 7: Efektivita VAD epoch validační sady - Autobus

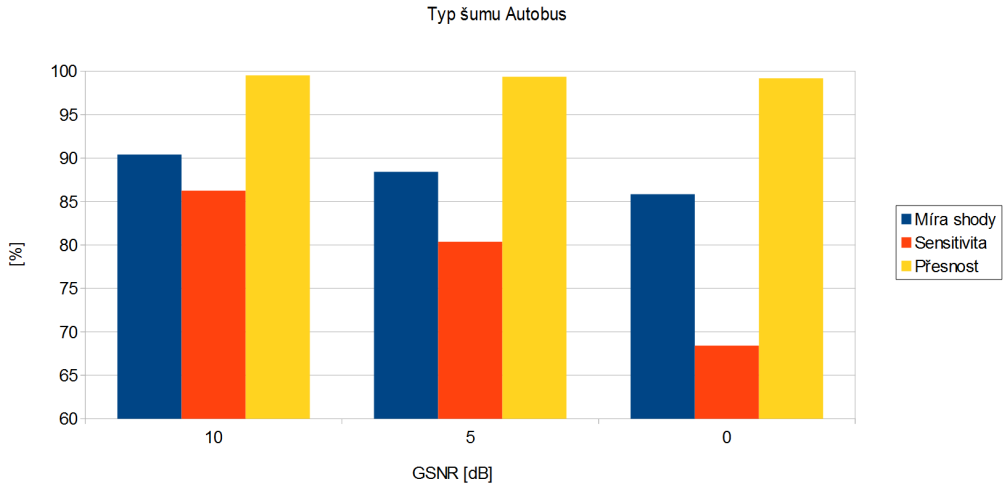
Z Grafu č. 7 je vidět, že je potřeba natrénovat více epoch sítě a pak z nich vybrat tu, která má nejlepší celkové výsledky. Zpravidla není nikdy známo, po kolika trénovacích epochách pro danou problematiku se síť začne blížit k lokálnímu minimu (v lepším případě globálnímu) kriteriální funkce a je třeba tuto hodnotu experimentálně najít. V mém případě se síť od 7. epochy začala pohybovat velmi blízko lokálnímu minimu (je možné, že i globálnímu) kriteriální funkce a již se jen ustáluje.

3.3.2 Testovací sada

Testovací sada se skládá z 256 zvukových nahrávek, které nebyly použity při trénování.

Níže jsou zobrazeny statistiky detekce přítomnosti řeči epochy sítě, která měla nejlepší výsledky (9. epocha) na testovacích datech s typem šumu Autobus, Kafeterie, Chodník a Ulice. Každý šum má svoje charakteristické vlastnosti, které se v grafech projevují.

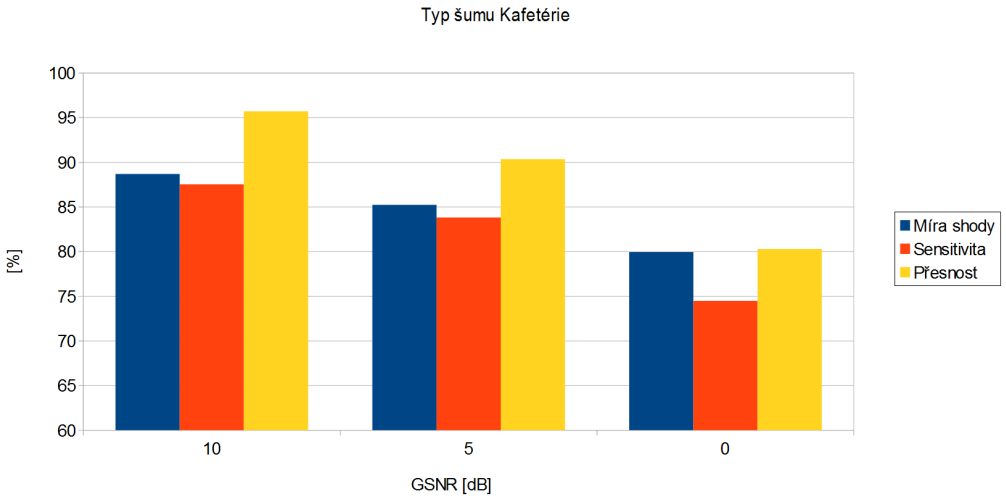
Testování známých dat (Matched conditions) Obsažené šumové signály patří do stejných kategorií jako šumy v trénovací sadě, ale tyto konkrétní signály síť při trénování neviděla.



Graf 8: Efektivita nejlepší VAD sítě na testovací sadě - Autobus

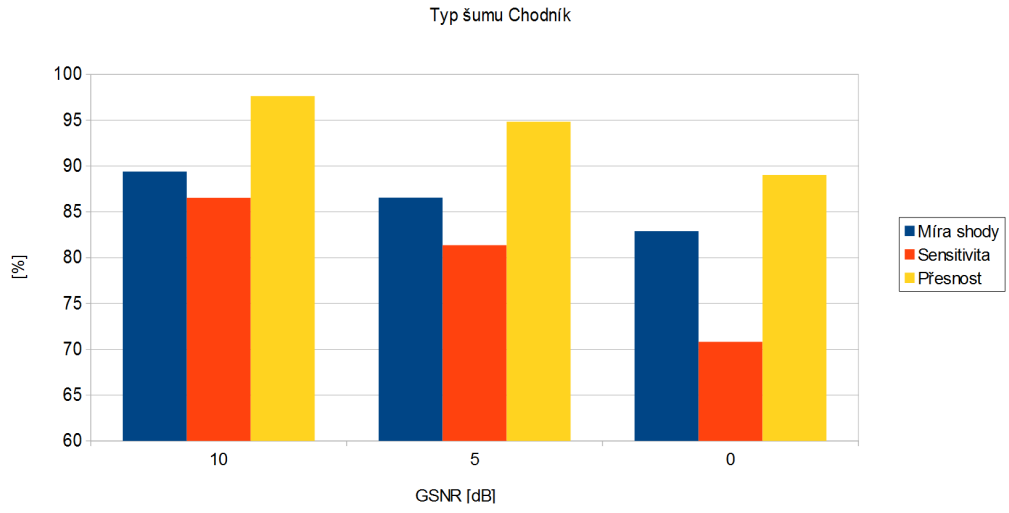
Jelikož šum z prostředí autobusu má velmi stacionární charakter, tak ho síť byla schopna velmi dobře zanalyzovat, což se projevilo ve vysoké hodnotě Míry shody. Je vidět, že s klesající úrovní GSNR nahrávky klesá i sensitivita sítě (ale přesnost zůstává téměř stejná), což znamená, že když síť označí segment za řečový, tak z 98% skutečně řečový je, ale mnoho řečových segmentů síť označila jako šumové. V porovnání s typem šumu Kafeterie a Chodníku jsou dosažené výsledky nejlepší.

Z toho lze usoudit, že síť bude fungovat nejlépe na datech se stacionárním šumem (např. hlučení větráku, zdroje napětí apod.). V dalších experimentech uvádím také příklady nestacionárních šumových signálů, které se v běžném prostředí vyskytují častěji.



Graf 9: Efektivita nejlepší VAD sítě na testovací sadě - Kafeterie

Kafeterie má naopak v porovnání nejhorší výsledky. Hlavním důvodem pro to je pravděpodobně nestacionární charakter šumu, kdy v pozadí lidé povídají a zároveň konzumují jídlo (s čímž jsou spojené další hluky). Lidská řeč v pozadí, která je ve skutečnosti šum vůči užitečné informaci v popředí, je jedním z nejsložitějších problémů, které mohou pro detektory řeči nastat, jelikož se pořád jedná o lidskou řeč, kterou mají být schopny rozpoznat.



Graf 10: Efektivita nejlepší VAD sítě na testovací sadě - Chodník

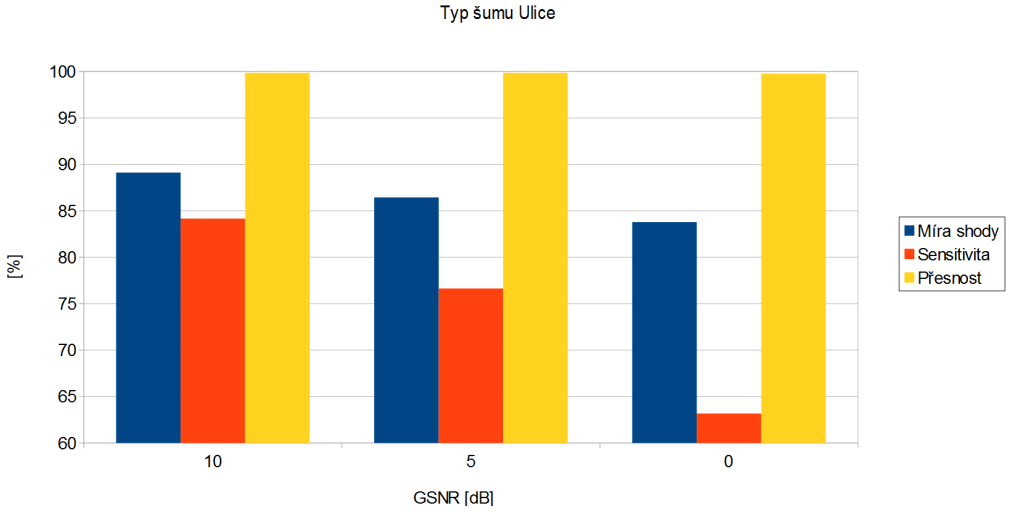
V nahrávkách Chodníku jsou slyšet sice hlavně konverzace lidí v pozadí, ale nejsou tak zřetelné jako v Kafetérii, takže si s nimi síť poradila lépe. Tyto vzorky mají stacionárnější charakter než Kafeterie a v Míře shody sítě to lze zpozorovat. Zajímavý poznatek je, že čím stacionárnější šum, tím větší hodnota Přesnosti sítě.

Z Grafů č. 8, 9 a 10 lze vidět chování sítě při zpracovávání reálných šumů různého charakteru.

Síť má spíše problém se záměnou řečového signálu za šumový, než naopak. Což je lepší varianta (z hlediska následných SNR odhadů založených na řečových rámcích), než kdyby byl šum označován za řeč.

Jelikož se jedná o známou testovací množinu, tak lze očekávat, že tyto výsledky budou o něco zvýhodněny oproti neznámé množině (viz. níže).

Testování neznámých dat (Mismatched conditions) Zde se vyskytly nahrávky s typem aditivního šumu, který při trénování nebyl viděn.



Graf 11: Efektivita nejlepší VAD sítě na testovací sadě - Ulice

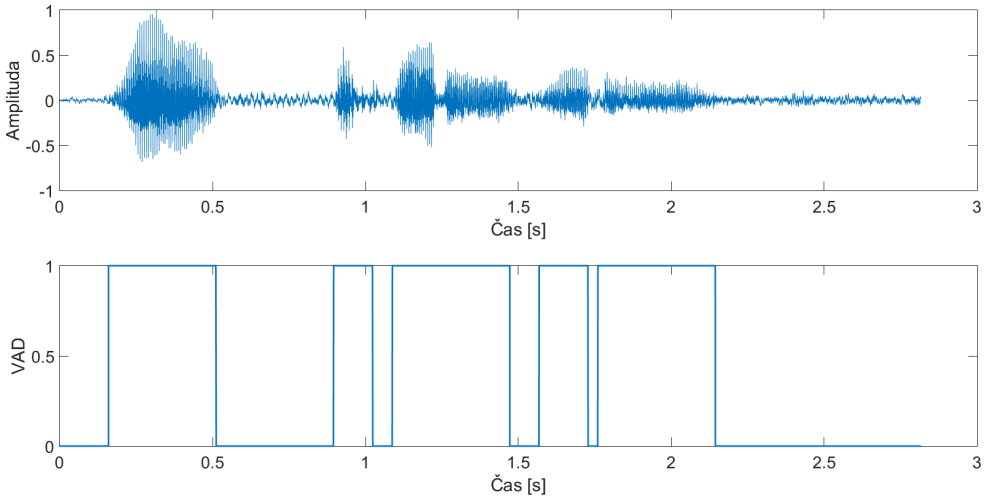
Jelikož se jedná o dosud neviděný typ šumu, tak je efektivita sítě o poznání horší, ale i tak lze vidět podobné chování sítě přes jednotlivé GSNR úrovně. Tedy například, že Přesnost se téměř nemění a s nižším GSNR klesá pouze Sensitivita (a s tím spojený pokles Míry shody).

Je ovšem zajímavé, že šum typu Ulice je stacionárního charakteru a v přesnosti sítě se tato vlastnost projevila, jelikož má dokonce lepší přesnost, než testovací sada Autobus. Avšak Sensitivita sítě je podstatně nižší, což znamená, že síť nebyla schopná správně rozeznat všechny řečové segmenty, ale i tak se Sensitivita pohybovala v rozmezí 60-85%.

Souhrn Síť je schopná rozpoznávat zašuměné nahrávky jak se stacionárním šumem, tak i nestacionárním. Ačkoli v případě nestacionárního šumu jsou výsledky poněkud horší, i tak se zdají být přijatelné. Z grafů lze usoudit, že čím má šum stacionárnější charakter, tím větší Přesnost síť má, ačkoli není schopná správně označit všechny řečové segmenty, o čemž vypovídá Sensitivita. Díky tomu, že síť byla schopná rozpoznat neviděný typ šumu s dobrými výsledky, bych tuto síť označil za robustní.

3.3.3 Ukázka výstupu VAD algoritmu

Na Obrázku č. 6 je zobrazena ukázka výstupu VAD sítě pro soubor z testovací množiny s aditivním šumem typu Autobus a s GSNR 10 dB.



Obrázek 6: Ukázka výstupu VAD sítě vygenerovaná prostředím Matlab

Porovnáním amplitudy ze vstupního signálu s VAD výstupem lze usoudit, že pro tento soubor je detekce řeči vcelku přesná.

Tuto skutečnost potvrzují i výsledky klasifikačních kritérií, kdy Míra shody je 97,1%, Sensitivita je 94,7% a Přesnost je 100%.

4 Experimenty s odhadem GSNR

Jakmile jsem vybral nejlepší epochu sítě z hlediska VAD, mohl jsem přejít k druhé části mé úlohy, což je odhad SNR úrovně nahrávky. Výstup algoritmu je GSNR, což v mém případě znamená poměr naakumulovaných energií řeči a šumu v nahrávce o průměrné délce 3 sekundy. Pro odhad šumové energie pomocí energie řečových segmentů v nahrávce jsem zvolil adaptivní algoritmus.

4.1 Adaptivní odhad šumu

Tento algoritmus využívá VAD informace a pohyblivého okénka s délkou 30 vzorků a koeficientem zapomínání 0,98 (viz. kapitola 4.4).

Algoritmus pracuje tak, že na počátku má prázdný vektor, do kterého postupně zprava přidává okamžitý výkon vzorků označených neuronovou sítí jako neřečové, dokud takto nedojde k celkem 30 vzorkům. Poté se již všechny vzorky posunou ve vektoru o jednu pozici doleva, přičemž nejnovější vzorek se přidá zprava a nejstarší vzorek vypadne z okénka.

Samotný odhad lokální energie šumu je pak realizován pomocí zmíněného koeficientu, kde každý vzorek v okénku je vynásoben mocninou tohoto koeficientu a následně zprůměrován, čímž získáme odhad výkonu šumu $\hat{E}_v$ v každém vzorku nahrávky. Následující rovnice (21) ukazuje tento výpočet pro i-tý vzorek signálu.

$$\hat{E}_v[i] = \frac{\sum_{j=0}^{L-1} \alpha^j \cdot v((L-1)-j)}{L} \tag{21}$$

kde L je velikost okénka (30 vzorků), α je koeficient zapomínání (0,98) a $v((L-1)-j)$ je příslušná hodnota výkonu v okénku

Největší váhu má tedy výkon nejaktuálnějšího vzorku a naopak prvek na první pozici má svoji hodnotu výrazně sníženou.

Zvolil jsem počítání průměrné hodnoty výkonu namísto okamžité kvůli tomu, že je třeba získat robustní odhad výkonu šumu v krátkém intervale kvůli nestacionaritě šumu.

4.2 Odhad globálního SNR

Jakmile máme vektor odhadu energie šumu v každém vzorku, zbývá jen vypočíst průměrnou energii šumu $\hat{\sigma}_n^v$ a průměrnou energii řeči $\hat{\sigma}_s^2$ v signálu [8].

$$\hat{\sigma}_v^2 = \frac{1}{l_v} \sum_{n=0}^{l_v-1} x^2[n] \cdot |1 - vad[n]| \tag{22}$$

$$\hat{\sigma}_s^2 = \frac{1}{l_s} \sum_{n=0}^{l_s-1} (x^2[n] - \hat{\sigma}_v^2[n]) \cdot vad[n] \tag{23}$$

kde $\hat{\sigma}_n^2$ je tedy vektor odhadů energie šumu v každém vzorku nahrávky, l_v je počet neřečových segmentů v nahrávce, l_s je počet řečových segmentů v nahrávce a vad[n] je vektor s informací o výskytu řeči v segmentu.

Globální odhad SNR se pak vypočítá podle rovnice (24)

$$GSNR = 10 \cdot log \frac{\hat{\sigma}_s^2}{\hat{\sigma}_v^2} \tag{24}$$

Tedy čím přesněji odhadneme vektor σ_{n}, tím přesnější pak bude odhad globálního SNR.

4.3 Vliv hranice VAD na odhad GSNR

Ovšem správný odhad šumu ve vzorcích není jediný důležitý prvek v tomto algoritmu. Odhad také závisí na množství detekovaných řečových segmentů. Čím více řečových segmentů, tím větší šance správného odhadu.

To ovlivňuje hranice VAD, neboli práh LSNR pro cílová data, kdy je segment označen jako řečový.

Po několika experimentech jsem zjistil, že optimální hranice, kdy je vhodné segment označit za řečový, je -5 dB lokálního SNR. Pomocí této hranice jsem vytvořil ideální VAD, který byl použit jako cílový vektor při trénování VAD sítě.

V případě vyššího limitu síť nerozpoznávala tolik řečových segmentů, aby byl odhad přesný (viz. Tabulka č.1). A v případě menšího limitu začínala mít síť problém správně rozeznat šum od řeči, což vedlo k označení prakticky všech segmentů jako řečové, čímž SNR odhad vycházel nekonečno.

Tabulky 1,2 a 3 zobrazují odhady na testovací sadě s 256 zvukovými stopami, u kterých byl použit typ šumu Autobus.

Tabulka 1: Odhad GSNR pomocí VAD sítě s limitem 10 dB lokálního SNR

SNR	Bias	Variance	MSE	Špatné odhady
0	-2,7	32,3	7376,2	34
5	-1,7	8,4	211,5	2
10	-1,1	0,9	0,9	0

Mimo obvyklých kritérií přibylo nové kritérium Špatné odhady. Pod tímto pojmem zahrnuji počet takových nahrávek, u kterých algoritmus pro odhad GSNR vypočetl zápornou či nulovou

Tabulka 2: Odhad GSNR pomocí VAD sítě s limitem 0 dB lokálního SNR

SNR	Bias	Variance	MSE	Špatné odhady
0	-4,3	1,4	36,0	0
5	-2,4	0,9	4,7	0
10	-1,1	0,9	0,9	0

Tabulka 3: Odhad GSNR pomocí VAD sítě s limitem -5 dB lokálního SNR

SNR	Bias	Variance	MSE	Špatné odhady
0	-3,7	0,9	10,7	0
5	-2,0	0,8	2,5	0
10	-1,2	0,8	0,8	0

energii řeči v signálu. V takovýchto případech nastavuji odhadovanou úroveň na -10 dB (tedy velmi zarušená nahrávka), což značně ovlivňuje výslednou Varianci.

Z experimentů jsem zjistil, že obecně platí, že čím menší je limit, tím více řečových segmentů je detekováno a tím získáme přesnější odhad. Ovšem s limitem menším jak -5 dB již síť začíná výrazně hůře rozeznávat řeč a šum (konkrétně např. u -20 dB již všechny segmenty označuje jako řečové), takže jsem vybral právě -5 dB jako optimální hranici.

Je také vhodné poznamenat, že s nižším limitem LSNR pro označení řečových segmentů se výrazně potlačily odhady se zápornou energií řeči (žádné „Špatné odhady“). Toto ovšem platí pouze pro nahrávky s GSNR 0 dB a výše. Jakmile se nahrávky pohybují pod touto úrovní, záporné odhady energie řeči se začínají vyskytovat ve větší míře.

To je dáno tím, že máme stanovenou hranici VAD -5 dB a v takovýchto případech se většina segmentů pohybuje pod touto hranicí, což způsobí rapidní snížení počtu detekovaných řečových segmentů. Toto ovšem není nutně negativní vlastnost, jelikož nahrávka, která má GSNR méně jak -5dB již není moc zřetelná.

4.4 Vliv volných parametrů na adaptivní odhad GSNR

Velmi klíčovou roli v odhadu SNR úrovně hraje právě koeficient zapomínání a délka okénka, které je třeba zvolit tak, aby pokud možno co nejlépe ovlivnili odhad algoritmu.

V Tabulce č. 4 je možné vidět vliv různých nastavení těchto parametrů na odhad GSNR úrovně. Pro získání těchto hodnot byla použita testovací sada 256 nahrávek s šumem typu Autobus o úrovni 0 dB GSNR.

Tabulka 4: Vliv změny parametrů na odhad GSNR

Koeficient zapomínání / Délka okna		0,97	0,975	0,98	0,985	0,99
35						
	Bias	-3,9	-3,8	-3,8	-3,7	-3,5
	Variance	0,7	0,7	0,8	0,8	1,0
	Špatné odhady	0	0	0	0	0
30						
	Bias	-3,8	-3,8	-3,7	-3,6	-3,4
	Variance	0,8	0,8	0,9	1,0	1,2
	Špatné odhady	0	0	0	0	0
25						
	Bias	-3,7	-3,6	-3,5	-3,4	-3,1
	Variance	0,9	1,0	1,1	1,3	1,9
	Špatné odhady	0	0	0	0	0
10						
	Bias	-2,6	-2,4	-2,3	-2,1	-1,9
	Variance	4,4	4,7	5,2	6,1	6,9
	Špatné odhady	4	4	4	6	6

Ze získaných hodnot je možné vyčíst, že se změnou parametrů je vždy jedno kritérium na úkor toho druhého.

Čím větší velikost okénka, tím stabilnější odhad energie šumu získáme (v případě stacionárního šumu toto znamená i přesnější odhad), což nám dá malou varianci, ale za cenu většího biasu, jelikož průměrujeme přes větší množství vzorků, čímž se samotný odhad stává nepřesným.

Krátké okénko sice znamená větší flexibilitu odhadu (schopnost správně odhadnout i dynamický šum), ale energie šumu může být odhadnuta nepřesně. A s tím v krajních případech může dojít k záporným odhadům energie řeči, což zvětší Varianci, protože v takových případech algoritmus dosazuje konstantu. Podobné chování má i změna koeficientu zapomínání.

Vybral jsem tedy vhodný kompromis s délkou okna 30 vzorků a koeficientem zapomínání 0,980.

4.5 Evaluace

Zde se dostáváme k nejdůležitějším statistickým údajům této práce. Jedná se o celkový přehled úspěšnosti odhadu globální úrovně odstupu řeči od šumu na testovacích sadách.

Pro srovnání jsem použil ideální VAD (tuto informaci mám díky tomu, že jsem zarušené nahrávky programově slučoval) ve spojení s algoritmem odhadu SNR. Kromě ideálního VAD jsem přidal i porovnání s již existujícím nástrojem WADA (viz. kapitola 2.8) pro odhad globálního SNR, pomocí kterého srovnávám efektivitu mého algoritmu.

4.5.1 Testovací sada se známými daty - Autobus

Všechny zobrazené výsledky nemají žádné Špatné odhady, proto se tato hodnota již v tabulce nevyskytuje.

Tabulka 5: Srovnání odhadů pro šum typu Autobus

Autobus			
VAD síť + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-3,7	0,9	10,7
5	-2,0	0,8	2,5
10	-1,2	0,8	0,8
Ideální VAD + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-3,8	0,9	10,7
5	-2,1	0,9	3,5
10	-1,2	0,7	0,7
WADA			
SNR	**Bias**	**Variance**	**MSE**
0	-1,0	2,0	4,4
5	-0,7	1,0	0,5
10	-0,6	1,3	0,6

Z Tabulky č. 5 je možné vidět chování celého algoritmu odhadu. Jelikož je odhad SNR založen na počítání s energií řečových segmentů, tak čím méně těchto segmentů se v nahrávce vyskytuje, tím méně přesný je samotný odhad.

To lze zpozorovat z hodnot Biasu jednotlivých úrovní, kdy při nahrávkách s globálním SNR 0 dB se hodnoty odhadu průměrně liší od cílové hodnoty (zde tedy 0 dB) zhruba o 3,5 dB (±1dB) a oproti tomu při 10 dB globálního SNR nahrávek se odhad liší pouze o 1,2 dB (±1dB).

Z tabulky je možné vidět, že celková přesnost odhadu ideálního VAD se příliš neliší od natrénované VAD sítě. Po bližším zkoumání výsledků VAD sítě jsem zjistil, že většina špatně označených rámců se pohybuje okolo -3 dB úrovně lokálního SNR (tedy relativně malá úroveň energie). Znamená to, že tato nepřesnost nemá moc velký vliv na výsledný odhad, což je důvod, proč jsou výsledky VAD sítě a ideálního VAD srovnatelné.

WADA dává lepší výsledky než náš odhad, ale tento druh šumu (stacionární bus) odpovídá jejímu modelu šumu (modeluje šum jako Gaussovu náhodnou veličinu).

4.5.2 Testovací sada se známými daty - Kafeterie

Tabulka 6: Srovnání odhadů pro šum typu Kafeterie

	Kafeterie		
VAD síť + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-3,4	2,0	45,6
5	-2,0	1,0	3,9
10	-1,3	1,0	1,6
Ideální VAD + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-3,7	0,8	9,4
5	-2,0	0,9	3,3
10	-1,2	0,9	1,2
WADA			
SNR	**Bias**	**Variance**	**MSE**
0	-2,2	2,1	21,4
5	-1,4	1,4	4,0
10	-1,0	1,7	3,3

Další údaj, který lze zpozorovat, je nezvyklý skok variance VAD sítě při úrovni GSNR 0 dB (oproti tendenci ostatních šumů, kdy variance nepřesáhne hodnoty 1). Jak bylo řečeno, všechny zobrazené hodnoty nemají žádné Špatné odhady. Nejedná se tedy o problém s odhadem záporné energie řeči (kde by se aktivovala dolní mez odhadu na statických -10 dB). Bližší kontrolou výsledků jsem zjistil, že se v testovací sadě vyskytla jedna nahrávka, která

byla špatně odhadnuta na -13 dB SNR. Tato nahrávka v sobě obsahovala znatelně výraznější dynamický šum (cinknutí příboru), který algoritmus špatně zpracoval.

Jedná se o signál, který se svými vlastnostmi silně odlišuje od ostatních v dané skupině (tzv. outlier). V tomto případě to znamená, že algoritmus nebyl schopen danou nahrávku správně odhadnout a nahrávka dostala velmi vzdálenou hodnotu od cíle.

Zároveň lze vidět u tabulky Kafeterie, že lidská řeč v pozadí a dynamické šumy také negativně ovlivňují WADA z hlediska odhadu SNR úrovně. V porovnání s ostatními typy šumu má o poznání horší přesnost odhadu.

4.5.3 Testovací sada se známými daty - Chodník

Tabulka 7: Srovnání odhadů pro šum typu Chodník

Chodník			
VAD síť + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-3,8	0,9	11,1
5	-2,2	0,8	3,4
10	-1,3	0,8	1,0
Ideální VAD + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-3,8	0,8	9,8
5	-2,2	0,9	3,5
10	-1,2	0,9	1,2
WADA			
SNR	**Bias**	**Variance**	**MSE**
0	-1,6	1,1	3,5
5	-1,0	0,9	0,8
10	-0,8	1,3	1,0

Obecně lze říct, že WADA má značně lepší Bias odhadů přes všechny úrovně cíleného GSNR, ale má o něco horší Varianci. To prakticky znamená, že má přesnější průměrný odhad, ale má více odhadů, které se liší od cílové úrovně ve větší míře.

4.5.4 Testovací sada se neznámými daty - Ulice

Nejzajímavější část výsledků je právě testovací sada s neznámými daty, jelikož se zde objevila dosud neviděná data. Jedná se tedy o skutečný test robustnosti sítě.

Tabulka 8: Srovnání odhadů pro šum typu Ulice

Ulice			
VAD síť + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-4,5	2,3	101,1
5	-2,5	0,8	4,4
10	-1,2	0,8	0,8
Ideální VAD + SNR odhad			
SNR	**Bias**	**Variance**	**MSE**
0	-3,9	0,7	8,3
5	-2,2	0,6	1,6
10	-1,2	0,7	0,7
WADA			
SNR	**Bias**	**Variance**	**MSE**
0	-0,7	3,1	4,1
5	-0,5	1,0	0,3
10	-0,4	1,2	0,2

U cílové úrovně 0 dB pro VAD síť se variance zdá být podstatně vyšší, než by naznačovaly hodnoty pro 10 a 5 dB. V tomto případě nenastal případ, kdy by algoritmus odhadl zápornou energii řeči a nastavil staticky -10 dB (tedy to, co označuji jako Špatný odhad), ale stejně tak jako u Kafeterie se našla jedna nahrávka, kterou algoritmus natolik špatně odhadl, že dostala podstatně menší hodnotu a tím pádem větší vzdálenost od cílové hodnoty (což se projeví zvětšenou Variancí).

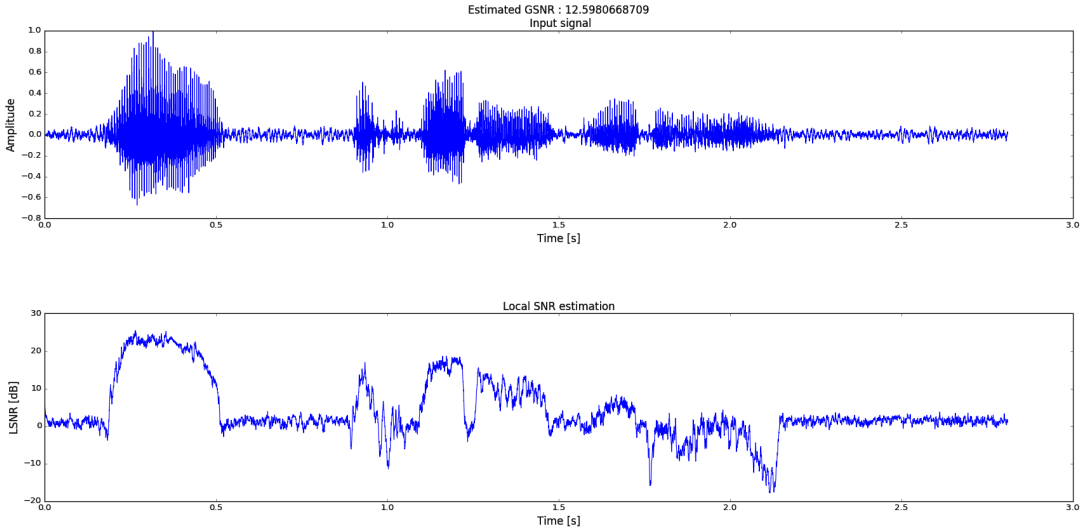
Stejný problém měl i WADA program s několika nahrávkami, jelikož jinak se i jeho Variance pohybuje okolo 1 dB. Ale zase byl mnohem přesnější s celkovým odhadem, jelikož se i u cílové hodnoty 0 dB lišil v průměru o méně než 1 dB.

Z celkového hodnocení MSE lze zpozorovat, že nejlepší odhady má WADA, následuje odhad SNR s ideálním VAD a těsně za nimi je odhad SNR pomocí VAD sítě. Ovšem WADA měl obecně větší množství hodnot, které byly více vzdáleny od cílového SNR, jelikož má o něco větší Varianci.

Průměrný Bias odhadu algoritmu pomocí VAD sítě se pohybuje okolo 3 dB, což je v přijatelném rozmezí tolerance vzhledem k charakteru úlohy.

4.6 Aplikace pro odhad globálního SNR

Jako závěrečný výstup mé práce jsem zvolil aplikaci psanou v programovacím jazyce Python, která pro specifikovaný vstup vypíše odhadnutý odstup řeči od šumu v celé nahrávce a vykreslí do grafu vstupní signál a časový průběh odhadu lokálního SNR v nahrávce. Jazyk pro aplikaci jsem zvolil anglický, jelikož tímto způsobem se aplikace stává univerzálnější z hlediska používání.



Obrázek 7: Ukázka výstupu aplikace pro signál s cílovou úrovní SNR 10 dB

V horní části grafu je zobrazena odhadovaná hodnota globálního SNR, v tomto případě se odhad od skutečnosti liší pouze o 2,5 dB, což souhlasí se statistickými údaji, kdy při cílovém SNR 10 dB má síť průměrný odhad 11 dB (±1dB). I od pohledu lze vidět, že průběh lokálního SNR indikuje přítomnost řeči, pohybuje-li se zhruba v oblasti nad 0 dB.

Aplikace přijímá jako vstup buď již vytvořený zvukový soubor (formátu .wav), nebo lze nastavit nahrávání z mikrofonu a program pak zpracuje nově vytvořenou nahrávku.

5 Závěr

V rámci diplomové práce jsem se seznámil s problematikou odhadu odstupu řeči od šumu, dostupnými metodami pro detekci výskytu řeči v nahrávce a konceptem neuronových sítí.

Neuronové sítě mi pomohly vyřešit problematiku odhadu odstupu řeči od šumu v řečové nahrávce. Zaměřil jsem se na použití neuronové sítě pro zjištění umístění řeči v nahrávce (tedy jako VAD prvek), což znamená, že síť měla za úkol zařadit segmenty vstupního signálu buď do kategorie šum nebo kategorie řeč.

Následně jsem pomocí VAD informace odhadoval globální úroveň SNR. Toho jsem docílil použitím adaptivního odhadu energie šumu v nahrávce, díky kterému jsem byl schopen získat odhad hodnoty energie šumu a řeči v daném segmentu nahrávky. Z těchto údajů bylo pak již vypočítání globální SNR velmi jednoduché.

Nakonec jsem vytvořil desktopovou aplikaci, která pro zadanou zvukovou nahrávku zobrazí průběh LSNR a odhad GSNR. Tímto byly splněny všechny body zadání.

Během experimentů jsem zjistil, že velmi záleží na hyperparametrech neuronové sítě, jelikož zásadně ovlivňují schopnost sítě správně kategorizovat vstupní data. Některé tyto hyperparametry bylo třeba vybrat dle uvážení k charakteru zpracovávaných dat a některé bylo třeba zjistit experimentálně.

Výsledné odhady algoritmu navrženého v práci jsou srovnatelné s metodou WADA. WADA je lepší na stacionárních datech a naše metoda je lepší na nestacionárních datech (až na Varianci s outlierem). WADA má výhodu, že je to statistický přístup, tedy nemá Neviděná data. Ovšem naše metoda je robustní i na těchto neviděných datech. K výsledným odhadům je třeba podotknout, že jelikož algoritmus odhadu globálního SNR je založen na vypočítání energie řečových segmentů, tak čím méně řečových segmentů se vyskytovalo v nahrávce, tím méně přesnější byl samotný odhad (viz. kapitola 4.5).

Jako případné další rozšíření práce by mohlo být trénování neuronové sítě přímo na rozpoznání úrovně lokálního SNR v rámcích nahrávky (tedy vynechat VAD prvek a odhadovací algoritmus na něm založený). Podle vhodně zvolených hyperparametrů sítě a počtu výstupních kategorií je možné, že výsledná síť bude mít i lepší výsledky z hlediska průběhu lokálního SNR, než má algoritmus navržený v této práci. Přesnost odhadu by záležela na velikosti rozlišení (počtu výstupních kategorií, např. kategorie po 1 dB z rozsahu od 10 dB do -5 dB). Výstupem takové sítě by bylo lokální SNR pro jednotlivé rámce, tedy průběh LSNR v rámci celé nahrávky. Ovšem odhadnout GSNR z průběhu LSNR je komplikované. Jelikož narozdíl od navržené metody, kde máme k dispozici průběh energie řeči a šumu, zde máme pouze informaci o logaritmu poměru těchto energií.

Použitá literatura

[1] SAKHNOV, Kirill, VERTELETSKAYA, Ekaterina a SIMAK, Boris. Approach for energy-based voice detector with adaptive scaling factor. In: *IAENG International Journal of Computer Science*. 2009, 36(4), 394. ISSN: 1819-9224.

[2] LOKHANDE, Nitin N., NEHE, Navnath S. a VIKHE, Pratap S. Voice activity detection algorithm for speech recognition applications. In: *IJCA Proceedings on International Conference in Computational Intelligence (ICCIA2012)*. Vol. 6. 2012, s.1–4.

[3] MORALES-CORDOVILLA, Juan A., MA, Ning, SÁNCHEZ, Victoria, CARMONA, José L., PEINADO, Antonio M. a BARKER, Jon. A pitch based noise estimation technique for robust speech recognition with missing data. In: *Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2011, s.4808–4811. ISSN: 1520-6149.

[4] HAIGH, J.A. a MASON, J.S. Robust voice activity detection using cepstral features. In: *TENCON'93. Proceedings. Computer, Communication, Control and Power Engineering*. IEEE. Vol. 3. 1993, s.321–324. ISBN-10: 0780312333.

[5] SOHN, J., KIM, N.S. a SUNG, W. A statistical model-based voice activity detection. In: *Signal Processing Letters*. IEEE. 1999, 6(1), s.1–3. ISSN: 1070-9908.

[6] ELLIS, Dan. Objective measures of speech quality/SNR. Labrosa. [online]. 8.4.2011 [cit. 2016-02-15]. Dostupné z: `http://labrosa.ee.columbia.edu/projects/snreval/`

[7] PORAT, B. *A course in digital signal processing*. 1. vyd. New York: Wiley, 1997.

[8] VONDRASEK, Martin a POLLAK, Petr. Methods for speech SNR estimation: Evaluation tool and analysis of VAD dependency. In: *Radioengineering*. 2005, s.6–11. ISSN 1210-2512.

[9] ADAMEC, M. Moderní rozpoznávače řečové aktivity. VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ. [online]. 2008 [cit. 2016-02-20]. Dostupné z: `https://dspace.vutbr.cz/bitstream/handle/11012/16807/Diplomova_prace_Michal_Adamec.pdf`

[10] KRIESEL, D. A Brief Introduction to Neural Networks. [online]. 2007 [cit. 2016-01-12]. Dostupné z: `http://www.dkriesel.com/_media/science/neuronalenetze-en-zeta2-2col-dkrieselcom.pdf`

[11] TIWARI, V. MFCC and its applications in speaker recognition. In: *International Journal on Emerging Technologies*. 2010, 1(1), s.19–22.

[12] IVAKHNENKO, A.G. a LAPA, V.G. Cybernetic predicting devices. In: *CCM Information Corporation*. 1965.

[13] HOCHREITER, S., BENGIO, Y., FRASCONI, P. a SCHMIDHUBER, J. *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*. 2001.

[14] ZHANG, X.L. a WU, J. Deep belief networks based voice activity detection. In: *Audio, Speech, and Language Processing*. IEEE Transactions. 2013, 21(4), s.697–710.

[15] BROOKES, Mike. VOICEBOX: Speech Processing Toolbox for MATLAB. Imperial College London. [online]. 23.2.2016 [cit. 2016-02-23]. Dostupné z: `http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html`

[16] MEMISEVIC, R., ZACH, C., POLLEFEYS, M. a HINTON, G.E. Gated softmax classification. In: *Advances in neural information processing systems*. 2010, s. 1603–1611.

[17] RUBINSTEIN, R.Y. a KROESE, D.P. The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning. In: *Springer Science & Business Media*. 2013.

[18] FORTMANN-ROE, S. Understanding the Bias-Variance Tradeoff. [online]. 1.6.2012 [cit. 2016-02-25]. Dostupné z: `http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html`

[19] BARKER, John, MARXER, Ricard, VINCENT, Emmanuel a WATANABE, Shinji. The third 'CHiME' Speech Separation and Recognition Challenge: Dataset, task and baselines. *IEEE 2015 Automatic Speech Recognition and Understanding Workshop*. 2015.

[20] GAROFOLO, J.S., LAMEL, L.F., FISHER, W.M., FISCUS, J.G. a PALLETT, D.S. DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1. NASA STI/Recon Technical Report N, 93. 1993.

Obsah přiloženého CD

- text diplomové práce
  - diplomova_prace_2016_Michal_Muzicek.pdf
  - diplomova_prace_2016_Michal_Muzicek.tex
- zdrojový kód programu
  - desktopová aplikace SNREstimator (kód využívá jazyka Python)
- zkompilovaná verze programu
  - desktopová aplikace SNREstimator