VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

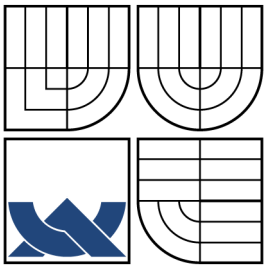
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

DETEKCE CHYBNÉ VÝSLOVNOSTI V MLUVENÉ ŘEČI

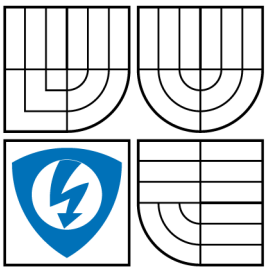
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE BC. MICHAL STRUHAŘ
AUTHOR

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

DETEKCE CHYBNÉ VÝSLOVNOSTI V MLUVENÉ ŘEČI
DETECTION OF SPEECH DISORDERS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — BC. MICHAL STRUHAŘ
AUTHOR

VEDOUCÍ PRÁCE — ING. PETR SYSEL, PHD.
SUPERVISOR

BRNO 2008

ZDE VLOŽIT LIST ZADÁNÍ

Z důvodu správného číslování stránek

ZDE VLOŽIT PRVNÍ LIST LICENČNÍ SMOUVY

Z důvodu správného číslování stránek

ZDE VLOŽIT DRUHÝ LIST LICENČNÍ SMOUVY

Z důvodu správného číslování stránek

ABSTRAKT

Práce se zabývá problematikou detekce chybné výslovnosti v mluvené řeči. Jedním z cílů této práce je výběr vhodných parametrizací. Jedná se o krátkodobou energii, funkci středního počtu průchodu signálu nulou, lineární prediktivní analýzu, perceptivní lineární prediktivní analýzu, metodu RASTA, kepstrální analýzu a melovské kepstrální koeficienty. Dalším cílem je konstrukce detektoru chybné výslovnosti na bázi DTW (dynamické borcení času) a umělé neuronové sítě. Samotná detekce probíhá na základě získaných příznaků z vybraných analýz a fonetického přepisu promluvy. Parametrizace, detektor i fonetická transkripce českého jazyka jsou implemetovány v simulačním prostředí MATLAB.

KLÍČOVÁ SLOVA

Detekce chybné výslovnosti, rozpoznávání řeči, analýza řeči, fonetická transkripce, neuronová síť, DTW, dyslalia.

ABSTRACT

This thesis deals with detection of speech disorders. One of the aims of this thesis is choosing suitable parameterization: short-time energy, zero-crossing rate, linear predictive analysis, perceptual linear predictive analysis, RASTA method, cepstral analysis and mel-frequency cepstral coefficient can be chosed for detections. Next aim is construction of detector of speech disorders based on DTW (Dynamic Time Warping) and artificial neuron network. Single detection proceeds on the base of collected tokens from chosen analysis and phonetic transcription of speech. Analyses, detector and phonetic transcription of Czech language are implemented in simulation environment of MATLAB.

KEYWORDS

Detection of Speech Disorders, Speech Recognition, Speech Analysis, Phonetic Transcription, Neural Network, DTW, Dyslalia.

STRUHAŘ M. *Detekce chybné výslovnosti v mluvené řeči*. Brno: Vysoké Učení Technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav telekomunikací, 2008. Počet stran 74, Počet stran příloh 4. Diplomová práce. Vedoucí práce byl Ing. Petr Sysel, PhD.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Detekce chybné výslovnosti v mluvené řeči“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ....

(podpis autora)

PODĚKOVÁNÍ

Tímto bych rád poděkoval panu Ing. Petrovi Syslovi, PhD. za metodickou pomoc, cenné rady a čas, který věnoval mé diplomové práci.

V Brně dne ....

(podpis autora)

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

ÚVOD

Schopnost komunikace prostřednictvím mluvené řeči je základním a nejdůležitějším prostředkem výměny informací mezi inteligentními bytostmi. Jestliže je tato schopnost jedince narušena, je snaha nalézt její příčinu, míru narušení a vhodnou terapii.

Aby byl stroj schopen rozpoznat v mluveném projevu chybnou výslovnost, je potřeba technicky i algoritmicky vyřešit dílčí úlohy analýzy a rozpoznavání řeči.

Vlastnosti řečového signálu v průběhu času mění pomalu, proto můžeme provést segmentaci souvislého zvukového signálu. Přičemž analýzy budou prováděny na jednotlivých segmentech. Segmentace se provádí násobením signálu časově posouvaným oknem.

Cílem této práce je vytvořit sadu skriptů, které umožní parametrizovat řečový signál pomocí těchto analýz:

- Krátkodobá energie,
- krátkodobá funkce středního počtu průchodu signálu nulou,
- krátkodobá kepstrální analýza,
- krátkodobá lineární prediktivní analýza,
- perceptivní lineární prediktivní analýza,
- melovské kepstrální koeficienty.
- analýza RASTA,

Následně zvolit ty parametrizace, které budou vhodné pro detekci chybné výslovnosti.

Od PaedDr. Lenky Němcové byly získány nahrávky promluv dětí s vadami výslovnosti. Tyto řečové signály budou nejprve zpracovány vybranými analýzami.

Bude vytvořena sada skriptů, která umožní detekci chybné výslovnosti (konkrétně dyslálii). Jádrem detektoru bude hybridní rozpoznávač na bázi DTW a dopředné umělé neuronové sítě.

Detekce bude probíhat na základě znalosti fonetického přepisu promluvy. Proto je nutné také vytvořit funkce a skripty, které umožní fonetickou transkripci českého jazyka.

Pro vyhodnocení úspěšnosti detekce chybné výslovnosti bude nutné zjistit účinnost rozpoznávače.

Veškeré skripty budou vytvořeny v simulačním prostředí MATLAB verze 7.5.

1 MLUVENÁ ŘEČ

Za nejmenší jednotku řeči, která může rozlišovat jednotlivá slova, lze považovat **foném**. Fonémy lze od sebe odlišit například podle způsobu a místa tvoření, podle artikulujícího orgánu nebo podle sluchového vjemu. Jejich počet se ve světových jazycích pohybuje od 12 do 60, český jazyk jich obsahuje 36.

Fonémy se spojují do posloupností promluvených celků, které jsou pravidelným opakováním různých posloupností **slabik**. Určitá kombinace slabik se nazývá **slovo**.

Zásadním jevem, který lze při procesu vytváření řeči pozorovat, je, že foném se může značně měnit vyslovením v různých kontextech. Jeho akustická realizace závisí na předchozím a následujícím zvuku, tempu a intonaci. Tuto závislost nazýváme **koartikulace**. Tento jev dal podnět pro zavedení **fónu** jako minimální fonetické jednotky identifikující odlišné primitivní zvuky řeči. Všechny odlišné fóny určitého fonému se nazývají **alofóny**.

1.1 Vytváření řeči člověkem

Zdrojem řečového signálu jsou lidské řečové orgány, které se skládají z hlasivek, dutiny hrdelní, ústní a nosní, měkkého a tvrdého patra, zubů, jazyka, plic a spjatými dýchacími svaly. Vytvořené zvuky lze podle jejich charakteru rozdělit na **znělé** a **neznělé**. Zdrojem všech znělých zvuků, které vykazují periodicitu, jsou kmitající hlasivky. Kmitočet kmitů závisí na tlaku vzduchu a svalovém napětí hlasivek a nazývá se **kmitočet základního tónu** F_0 a pohybuje se v rozmezí 50 až 500 Hz. Základní tón lidského hlasu $T_0 = 1/F_0$ je přítomen při tvoření všech znělých zvuků, tj. samohlásek a znělých souhlásek. Neznělé zvuky jsou vytvářeny třením výdechového proudu vzduchu o překážku. Všechny **hlásky** můžeme rozdělit na:

- Samohlásky (vokály)

  Při artikulaci samohlásek je snahou udržet průchod vzduchu hlasovým traktem co nejvolnější. V akustickém spektru každé samohlásky se objevuje kromě základního hlasivkového tónu řada vyšších zesílených tónů, které vznikají rezonancí v dutinách hlasového traktu. Nazývají se **formanty**, a označují se F_1,F_2,...,F_n, kde formantem s nejnižším kmitočtem je F_1. Pro české samohlásky jsou nejdůležitější dva formanty F_1 a F_2 (tab. 1.1)

- Souhlásky (konsonanty)(tab. 1.2)

  V případě souhlásek je, na rozdíl od samohlásek, v akustickém spektru přítomen charakteristický šum. Souhlásky jsou vytvářeny vzduchovou turbulencí,

Tab. 1.1: Samohlásky a jejich formanty F_1, F_2 a F_3 [5]

samohláska	F_1 [Hz]	F_2 [Hz]	F_3 [Hz]
i	300 - 500	2000 - 2800	2300 - 3500
e	480 - 700	1560 - 2100	2000 - 3000
a	700 - 1100	1100 - 1500	1500 - 3000
o	500 - 700	850 - 1200	1500 - 3000
u	300 - 500	600 - 1000	1900 - 2900

která vzniká třením výdechového proudu vzduchu o přepážku vytvořenou artikulačními orgány (špička jazyka, zuby nebo rty). Podle typu vytvoření překážky můžeme souhlásky rozdělit na:

- Závěrové (okluzívy)

  Překážka je úplná, v okamžiku její zrušení vzniká charakteristický krátký šum, který se podobá explozi. Lze je dále rozdělit na souhlásky:

  a) obouretné (labiály) – `p`, `b`, `m`,
  b) dásňové (alveoláry) – `t`, `d`, `n`,
  c) tvrdopatrové (palatály) – `ť`, `ď`, `ň`,
  d) měkkopatrové (veláry) – `k`, `g`.

- Úžinové (frikativy)

  V některém místě artikulačního ústrojí dojde k zúžení cesty výdechového proudu. Při tření v úžině vzniká charakteristický třecí šum. Lze je dále rozdělit na souhlásky:

  a) retozubné (labiodentály) – `f`, `v`,
  b) sykavé (sibilanty) – `s`, `z`, `š`, `ž`,
  c) tvrdopatrové (palatální frikativa) – `j`,
  d) měkkopatrové (velární) – `ch`,
  e) hrtanové (laryngální) – `h`,
  f) bokové (laterální) – `l`,
  g) kmitavé (vibranty) – `r`, `ř`.

- Polozávěrové (semiokluzívy)

  U těchto souhlásek se vyskytují oba typy překážek. Do této skupiny patří: `c`, `č`.

Dále můžeme rozdělit souhlásky podle znělosti na:

- Neznělé

  Při vyslovování těchto souhlásek hlasivky nekmitají, ale mohou být od sebe oddáleny, přivřeny, mohou se prudce rozevřít nebo zavřít podle toho, jakou souhlásku mají generovat.

- Znělé

  Jestliže jsou vyslovovány znělé souhlásky, dochází ke kmitání hlasivek a výdechový vzduch má potom periodický charakter.

Většinu šumových souhlásek lze seřadit do dvojic, v nichž se obě souhlásky shodují ve způsobu artikulace, ale liší se znělostí. Nazýváme je **párové**. Existují také souhlásky **nepárové**, které jsou vždy znělé a nemají neznělý protějšek.

Tab. 1.2: Dělení českých souhlásek [2]

souhlásky		závěrové				úžinové				polozávěrové	
párové	neznělé	p	t	ť	k	s	š	f	ch	c	č
	znělé	b	d	ď	g	z	ž	v	h	dz	dž
nepárové	znělé	m, n, ň				l, j, r, ř					

1.2 Nejběžnější vada výslovnosti - dyslalia

Nejběžnější poruchou komunikační schopnosti je dyslalie. **Dyslalia** (patlavost) spočívá v neschopnosti nebo poruše používání zvukových vzorů řeči v procesu komunikace podle řečových zvyklostí a norem příslušného jazyka.

K poklesu nesprávné výslovnosti dochází se zvyšujícím se věkem. Příčinou tohoto poklesu je proces dozrávání jedince, osvojení si dovednosti čtení a psaní, kdy dochází k uvědomění si různorodosti jednotlivých hlásek, vytvoření souvislostí mezi fonémy a **grafémy** (grafickými symboly hlásek).

1.2.1 Fonologické příznaky

Fonologický příznak je vlastnost zvukových složek jazyka, které tvoří základ pro popis fonémů nebo melodických schémat. Zde je jejich přehled podle pramenu [9].

Fonologické příznaky ve struktuře slova a slabik

- Delece (vynechání) koncového konsonantu (pes–pe),

- delece tzv. slabé slabiky na začátku nebo uprostřed slova (motyka–tika, telefon–tefon),
- shluk konsonantů se nějakým způsobem zjednoduší (park – pak, stůl–tůl),
- reduplikace – slabika nebo její část se opakují (voda–vovo),
- epenteze – doprostřed slova se vloží obvykle neznělý vokál (vlak–v.lak),
- metateze – transpozice (prohození) dvou zvuků ve slově (Karel–Kaler, revolver–levorver).

Harmonologie

Mechanismus harmonie spočívá v ovlivnění jednoho zvuku druhým, podobným. Proto se hovoří též o **asimilaci**. Předchází-li zvuk, který změnu zapříčinil, zvuku ovlivněnému, označuje se asimilace jako **progresivní**, následuje-li za ovlivněným zvukem, jde o asimilaci **regresivní**.

Podle místa vzniku se rozlišuje asimilace:

- Velární – nonvelární zvuk se asimiluje na velární v důsledku vlivu veláry (tak–kak: regresivní asimilace, kout–kouk: progresivní asimilace),
- nazální – nonnazální zvuk se asimiluje v důsledku vlivu nazálního[1] konsonantu (komín–momín: regresivní asimilace),
- labiální – nonlabiální zvuk se asimiluje s labiálním konsonantem pod vlivem labiálního konsonantu (buk–bub:progresivní asimilace, tabule–babule: regresivní asimilace).

Podle akustického dojmu a způsobu tvoření se znělý konsonant mění na neznělý a naopak:

- Prevokalická znělost – neznělé konsonanty se před vokálem mění na znělé (tam–dam),
- ztráta znělosti koncového konsonantu – jde o chybu např. anglickém jazyce, v českém se tato spodoba normou (dub–dup).

[1] nazály–nosní souhlásky (m, n, ň)

Proces substituce

Zvuk jedné hlásky se nahradí jiným, náhrada zvuku si vyžádá změnu místa či způsobu artikulace.

Podle místa artikulace:

- frontalizace – substituce se vyskytne vpředu nebo před standardní produkcí (káva–tráva), velární hláska je nahlazena alveorální,
- finalizace – substituce zvuky produkovanými vzadu nebo vzadu dole v dutině ústní (táta–káka, dům–gům).

Podle způsobu artikulace:

- Okluze – frikativy nebo afrikáty se nahrazují okluzivou (sám–tam, bič–bit),
- afrikace – frikativy se nahrazují afrikáty (sype–cipe, šije–čije),
- skluz likvid – prevokalické likvidy[2] se nahrazují skluzem (lampa–iampa),
- vokalizace – likvidy nebo nazály se nahrazují vokálem (vlk–vuk),
- denazalizace – nazály se nahrazují harmonickým závěrem (máma–bába),
- deafrikace – afrikáty se nahrazují friktivami (cop–sop, čípky–šípky),
- glotální náhrada – glotální okluze[3] nahrazuje zvuky obvykle uvnitř nebo na konci slova(prst–pe.).

1.2.2 Příčiny dyslalie

Příčiny dyslalie můžeme rozdělit podle toho, zda je funkční nebo organicky podmíněná. [10]

Dyslalie funkční (funkcionální) označuje stav, kdy mluvidla jsou bez poruchy a přesto dochází k nesprávné výslovnosti. Rozeznáváme dva typy funkční dyslalie:

- motorický – vzniká jako důsledek celkové neobratnosti i neobratnosti mluvidel,
- senzorický – vzniká jako důsledek nesprávného vnímání i diferenciace mluvních zvuků. Je to vývojový nedostatek pohybové a sluchové diferenciace.

[2] likvidy–plynné souhlásky (l, r)

[3] glotální okluze–v hlasivkové štěrbině dojde k závěru

Dyslalie organická je způsobena nedostatky a změnami na mluvních orgánech nebo též jako následek porušení sluchových drah. Mezi nejčastější příčiny patří vlivy:

- dědičnost,
- vliv prostředí – především vliv nesprávného řečového vzoru, chyby ve výchovném přístupu, bilingvální prostředí apod.,
- poruchy zrakového a sluchového vnímání,
- poškození dostředivých a odstředivých drah – v literatuře se ukazuje na úzkou souvislost mezi motorickým vývojem dítěte a výslovností, která vyžaduje přesnou koordinaci pohybu mluvidel,
- poškození centrální části – závažné postižení centrální nervové soustavy, v jejichž symptomatologii může být i porucha řeči,
- anatomické úchylky mluvidel – např. chybný skus, přirostlá podjazyčná uzdička, obrny jazyka nebo rtů.

Výslovnost je vážně narušena i tehdy, má-li jedinec narušené sluchové nebo zrakové vnímání. Při narušeném sluchovém vnímání je narušena dostředivá složka řečově–komunikačního procesu a jedinec postrádá zpětnou kontrolu správné výslovnosti. Vliv poruchy zrakového vnímání na poruchy výslovnosti je dán nemožností vnímat artikulační pohyby u druhých lidí a tak je není možné napodobovat. Vliv se projevuje především v raném věku, s vyšším věkem vliv poruchy zrakového vnímání klesá. [10]

2 FONETICKÁ TRANSKRIPCE ČEŠŤINY

Pro přesný a jednoznačný zápis zvuků různých jazyků slouží **fonetická transkripce**. K tomu využívá množinu symbolů definovaných **fonetickou abecedou**. Existuje více těchto fonetických abeced, např. :

- **IPA** (Internacional Phonetic Alphabet) - obsahuje všechny fonetické značky, tj. je nezávislá na jazyce. Umožňuje tak fonetikům porovnávat jazyky navzájem. Nevýhodou je obtížná reprezentace v počítači.
- **SAMPA** (Speech Assesment Methods Phonetic Alphabet) - kódováním většiny symbolů IPA na 7-bitové ASCII znaky.
- **X-SAMPA** (eXtended SAMPA) - kódováním všech symbolů IPA na ASCII znaky včetně diakritiky.
- **ČFA** (Česká Fonetická Abeceda) - jedná se o českou verzi SAMPA.

Fonémy se v mluvené řeči nevyslovují jednotlivě, ale spojitě ve slabikách či slovech. Tím dochází nejen k vzájemnému ovlivňování těchto fonémů, ale i vypoušťení stávajících nebo vytváření nových. Z tohoto důvodu je nutné, aby fonetická transkripce popisovala i tyto změny.

2.1 Automatická fonetická transkripce

Důvodem zavedení fonetické transkripce je potřeba pracovat s mluvenou reprezentací psané podoby řeči. Tato potřeba vyvstává zejména v oblastech syntézy řeči z psaného textu (Text-To-Speech systémy) a rozpoznávání řeči. Z tohoto důvodu je výhodné provádět automatický přepis **ortografické** podoby řeči na fonetickou.

Protože čeština patří mezi **flexivní jazyky**[1], je velmi obtížné vytvořit tzv. **fonetický slovník**, který by obsahoval všechny fonetické reprezentace psaného textu. Proto lze s výhodou použít **fonetická pravidla**. Toto pravidlo může být například zapsáno ve tvaru[1]:

$$A \rightarrow B/C\_D, \tag{2.1}$$

slovní interpretace je následující: Řetězec A se zamění za řetězec B, jestliže bezprostředně po sobě následují řetězce CAD.

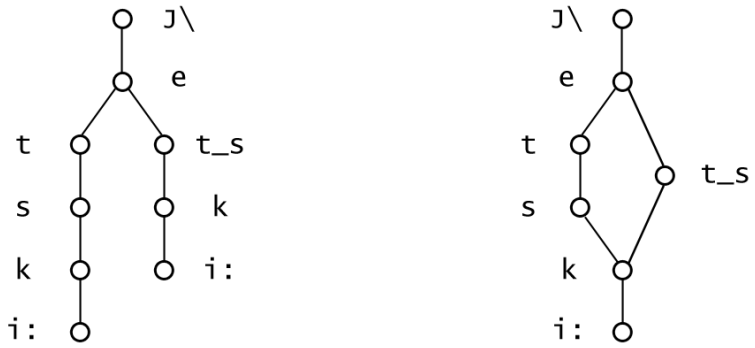
V případě fonetické transkripce slov cizích a přejatých, by fonetická pravidla byla příliš komplikovaná nebo jen stěží popsatelná, proto se s výhodou využívá **fonetický slovník výjimek**, ve kterém jsou takovéto výrazy uloženy.

[1]Flexivní jazyk je takový, kdy od jednoho slova lze odvodit velké množství tvarů.

Algoritmus automatické fonetické transkripce lze popsat následujícími kroky:

- Text je výhodné zpracovávat zprava doleva kvůli problému vícenásobné **asimilaci znělosti**[2].
- Na každý znak se uplatní výjimka z fonetického slovníku výjimek. V případě slovníku sestaveného z celých slov, se přítomnost výjimky zjišťuje na začátku slova. Výjimka má vyšší prioritu uplatnění před fonetickým pravidlem.
- Aplikace fonetických pravidel (mimo text, který byl převeden pomocí fonetického slovníku výjimek). Během aplikování fonetických pravidel může docházet k různým variantám přepisu, například slovo `špatně` lze zapsat jako `/shpatJe/` nebo `/shpacJe/`[3].
- Pokud nelze na zpracovávaný znak aplikovat žádnou výjimku nebo pravidlo, je tento znak pouze přepsán do dané fonetické abecedy.

Takto převedený text lze prezentovat pomocí **fonetického stromu** nebo **fonetického grafu**.



Obr. 2.1: Prezentace slova `dětský` pomocí a)fonetického stromu b)fonetického grafu

Fonetické pravidla lze nalézt např. v [1].

[2]Asimilace (spodoba) znělosti - změna znělosti souhlásek uvnitř hláskové skupiny.
[3]Použitá fonetická abeceda: SAMPA.

3 PARAMETRIZACE ŘEČI

Základem většiny metod analýzy akustického signálu řeči je předpoklad, že se jeho vlastnosti v průběhu času mění pomalu [2]. Tento předpoklad vede na aplikaci metod krátkodobé analýzy, při nichž se úseky signálu zpracovávají odděleně. Tyto úseky nazýváme segmenty a jsou reprezentovány časovým úsekem většinou o délce 10 až 30 ms. Výsledkem analýzy segmentu je pak číslo nebo vektor. Protože segmenty na sebe navazují nebo se částečně překrývají, dostáváme časové posloupnosti čísel, které popisují promluvený celek.

3.1 Zpracování v časové oblasti

Většinu metod krátkodobé analýzy v časové oblasti lze vyjádřit vztahem [2]

$$Q_n = \sum_{k=-\infty}^{\infty} \tau(s[k])w[n-k], \tag{3.1}$$

kde Q_n je krátkodobá charakteristika v čase n, $s[k]$ je vzorek akustického signálu získaný PCM v čase k, $\tau(.)$ vyjadřuje příslušnou transformační funkci a $w[n]$ je váhová posloupnost neboli tzv. **okénko**, kterým se vybírají, resp. váží, vzorky $s[k]$. V systémech analýzy řečového signálu se volí obvykle $n = Ni - 1$ při zpracování, kdy segmenty na sebe navazují, nebo $n = [N(i+1)/2]-1$ pro segmenty, které se překrývají o polovinu. Přitom i je pořadí analyzovaného segmentu ($i = ..., -1, 0, 1, ...$) a N je počet vzorků segmentu.

Úkolem okénka je vybrat příslušné vzorky signálu a přidělit jim při zpracování určitou váhu. Uvážíme-li důsledky působení okénka na signál, tj. že všechny vzorky mimo segment jsou váženy nulou, můžeme vztah (3.1) pro výpočet charakteristiky jednoho izolovaného segmentu přepsat do tvaru

$$Q_n = \sum_{k=0}^{N-1} \tau(s[k])w[N-1-k]. \tag{3.2}$$

Vzorky $s[k]$ se pak vždy vztahují ke konkrétnímu segmentu (první vzorek je vždy $s[0]$).

3.1.1 Krátkodobá energie

Funkci krátkodobé energie signálu lze definovat vztahem [2]

$$E_n = \sum_{k=-\infty}^{\infty} (s[k]w[n-k])^2, \tag{3.3}$$

kde $s[k]$ je vzorek signálu v čase k a $w[n]$ je příslušný typ okénka. Při měření krátkodobé energie lze doporučit délku segmentů 10-20 ms při kmitočtu vzorkovaní 8-10 kHz. Hodnoty funkce krátkodobé energie poskytují pro každý segment informaci o celkové energii v segmentu. Jedním z nedostatků této charakteristiky je její značná citlivost na velké změny úrovně signálu.

3.1.2 Krátkodobá funkce středního počtu průchodů nulou

Kmitočet průchodů signálu nulovou úrovní můžeme chápat jako jednoduchou charakteristiku popisující spektrální vlastnosti signálu a lze ji definovat podle [3] jako

$$Z_n = \sum_{k=-\infty}^{\infty} \left(\frac{|\mathrm{sgn}(s[n]) - \mathrm{sgn}(s[n-1])|}{2} \right) w[n-k], \tag{3.4}$$

kde

$$\mathrm{sgn}(s[n]) = \begin{cases} +1, & s[n] \geq 0 \\ -1, & s[n] < 0 \end{cases}. \tag{3.5}$$

3.1.3 Krátkodobá autokorelační funkce

Krátkodobá autokorelační funkce je definována vztahem [2]

$$R_n[m] = \sum_{k=-\infty}^{\infty} s[k]w[n-k]s[k+m]w[n-k-m], \tag{3.6}$$

kde $w[n]$ je opět okénko. Autokorelační funkce vykazuje některé vlastnosti, pro které se jí využívá zejména při indikaci periodicity signálu. Jestliže je totiž zpracovávaný signál periodický s periodou T, pak autokorelační funkce nabývá maximálních hodnot právě pro $m = 0, T, 2T, \ldots$. Z uvedeného důvodu je charakteristika velmi vhodná například pro určování periody základního hlasivkového tónu. Při určování této fonetické charakteristiky musí být však segment dostatečně dlouhý, aby obsahoval alespoň dvě periody signálu. Autokorelační koeficienty jsou důležitým základem též pro výpočet koeficientů lineární prediktivní analýzy.

3.2 Zpracování ve spektrální oblasti

Podobně jako u metod zpracování v časové oblasti, tak i ve spektrální oblasti se pracuje s představou přibližné stacionarity signálu, a proto je účelné mluvit v tomto případě o krátkodobé spektrální analýze. Nejčastěji používané postupy jsou zde přitom založeny na aplikaci krátkodobé Fourierovy transformace.

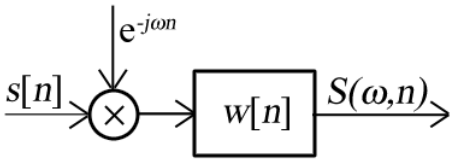
3.2.1 Krátkodobá Fourierova transformace

Předpokládejme, že metodou PCM byly získány vzorky $s[k]$ řečového signálu. Krátkodobá Fourierova transformace $S(\omega, n)$ je pak definována vztahem [2]

$$S(\omega, n) = \sum_{k=-\infty}^{\infty} s[k]w[n-k]\mathrm{e}^{-\mathrm{j}\omega k}, \tag{3.7}$$

kde $w[n]$ je funkce okénka, která vybírá pro zpracování určený úsek signálu. Je zřejmé, že takto vyjádřený Fourierův obraz je funkcí jak spojitě proměnné kmitočtu ω, tak i diskrétně proměnného času n a odpovídá konvoluci okénka $w[n]$ a vzorku $s[n]$ modulovaného $\mathrm{e}^{-\mathrm{j}\omega n}$. Lze tedy psát

$$S(\omega, n) = (s[n]\mathrm{e}^{-\mathrm{j}\omega n}) * h[n]. \tag{3.8}$$



Obr. 3.1: Systém pro určování krátkodobé Fourierovy transformace

Systém reprezentovaný touto rovnicí je na obr. 3.1. Koeficienty získané krátkodobou diskrétní Fourierovou transformací se využívají zejména ve spektrálních analyzátorech řeči nebo v řečových syntetizérech.

3.3 Časově-kmitočtová analýza

Při analýze signálů, jejichž charakter se v čase rychle mění, tedy signálů přechodového charakteru, je účelné uvažovat o kmitočtovém obsahu krátkých signálových úseků, což znamená rozvinout koncept tzv. **krátkodobých spekter**. Tento koncept umožňuje formulovat obecněji spektrum jako dvojrozměrnou funkci, závislou na kmitočtu a pozici v čase [4].

Praktická analýza vychází z konečných úseků signálu, vymezených použitým okénkem. Pokud má tedy okénko vhodnou délku a je formulováno jako klouzavé na časové ose, může být tento přístup použit při časově-kmitočtové analýze.

Zatímco v klasické spektrální analýze jde o určení velikosti a fáze harmonických složek různých kmitočtů a neomezeného trvání, nyní jde o co nejpřesnější lokalizaci

výskytu složek signálu jak ve spektrální, tak i v časové oblasti. Proto je pozorovací interval určen kompromisem mezi požadavkem na dostatečnou rozlišovací schopnost ve spektrální oblasti a současně snahou o velké rozlišení v čase (rozlišitelná diference kmitočtů je nepřímo úměrná délce okénka).

Krátkodobá spektra se zpravidla pořizují v celých sériích na základě signálových dat z delšího úseku signálu. V nejjednodušším případě lze použít dělení posloupnosti vzorků na úseky o délce N vzorků z celkového úseku signálu o M vzorcích, potom dosáhneme kmitočtové rozlišovací schopnosti odpovídající délce okna NT, a tento časový úsek znamená současně nejmenší rozlišitelný rozdíl v čase. Větší časové rozlišovací schopnosti můžeme dosáhnout, jestliže budou mít dílčí okna přesah. Takovýto soubor spekter se nazývá **spektrogram** [4], a bývá zobrazen jako dvojrozměrný obraz, v němž jedna souřadnice odpovídá kmitočtu, druhá času a barva odpovídá modulu.

3.4 Lineární prediktivní analýza

3.4.1 Výpočet LPC analýzy autokorelační metodou

Lineární prediktivní kódování (LPC) je jednou z nejefektivnějších metod analýzy akustického signálu. Je to metoda, která se snaží na krátkodobém základu odhadnout přímo z řečového signálu parametry modelu vytváření řeči. Je schopna zabezpečit velmi přesné odhady uvedených parametrů při přijatelné výpočetní náročnosti.

Princip metody LPC je založen na předpokladu, že k-tý vzorek signálu $s[k]$ lze popsat lineární kombinací Q předchozích vzorků a buzení $u[k]$.

$$s[k] = -\sum_{i=1}^{Q} a_i s[k-i] + Gu[k], \tag{3.9}$$

kde G je koeficient zesílení a Q je řád modelu.

V [2] je definován vztah, který musí splňovat koeficienty modelu:

$$\sum_{i=1}^{Q} a_i R_n[|j-i|] = -R_n[j] \quad , 1 \leq j \leq Q, \quad i = 1, 2, \ldots, Q, \tag{3.10}$$

kde R_n je krátkodobá autokorelační fukce a Q je řád prediktoru.

Po převedení vztahu (3.10) do maticového tvaru, lze aplikovat Levinson-Durbinův algoritmus. Hledané koeficienty pak určíme pomocí rekurzivních vztahů:

$$E_n^{(0)} = R_n[0], \tag{3.11}$$

$$k_i = -(R_n[i] + \sum_{j=1}^{i-1} a_j^{(i-1)} R_n[i-j])/E_n^{(i-1)}, \tag{3.12}$$

$$a_i^{(i)} = k_i, \tag{3.13}$$

$$a_j^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)} \quad , 1 \leq j \leq i-1, \tag{3.14}$$

$$E_n^{(i)} = (1 - k_i^2) E_n^{(i-1)}, \tag{3.15}$$

kde a_j^i je j-tý parametr prediktoru řádu i. E_n je chyba predikce.

3.4.2 Perceptivní lineární prediktivní analýza

Lineární prediktivní analýza velmi dobře popisuje spektrální vlastnosti řečového signálu. Bohužel však tento popis neodpovídá způsobu, jakým vnímá člověk řečové signály. Účinný postup, jakým lze tento nedostatek odstranit se nazývá **perceptivní lineární prediktivní analýza** (PLP), která využívá tří prvků psychoakustiky a to:

- Kritické pásmo spektrální citlivosti,
- křivky stejné hlasitosti,
- závislost mezi intenzitou zvuku a jeho vnímanou hlasitostí.

Výkonové spektrum je pak transformováno do sluchového spektra, které je následně aproximováno autoregresním celopólovým modelem, jako je tomu u lineární prediktivní analýzy. Výpočet PLP se skládá z následujících kroků [1]:

Výpočet výkonového spektra

Segment řečového signálu $s[k]$ je vážen Hammingovým okénkem a jsou vypočteny vzorky signálového spektra $S(\omega)$ (viz odstavec 3.2.1). Krátkodobé výkonové spektrum řečového signálu $P(\omega)$ je definováno vztahem

$$P(\omega) = |S(\omega)|^2. \tag{3.16}$$

Nelineární transformace kmitočtů a kritická pásma spektrální citlivosti slyšení

Člověk vnímá změny ve výšce zvuku přibližně logaritmicky. Toto vnímání je taktéž ovlivněno maskováním zvuků. Šířka pásma, ve které je zvuk maskován se nazývá **šířka kritického pásma**, přičemž její velikost se s kmitočtem mění. Tento jev lze

popsat nelineární transformací originální osy kmitočtů ω[rad/s] na osu kmitočtů $\Omega(\omega)$[bark] podle vztahu

$$\Omega(\omega) = 6\ln\left(\frac{\omega}{1200\pi} + \sqrt{\left(\frac{\omega}{1200\pi}\right)^2 + 1}\right), \tag{3.17}$$

kde $\omega = 2\pi f$[rad/s]. Dále je nutné zkonstruovat pásmové propusti, které popisují maskující křivky simulující kritická pásma slyšení. Prototyp takovéto pásmové propusti lze popsat vztahem

$$\Psi(z) = \begin{cases} 0 & \text{pro} \quad z < -2.5 \\ 10^{z+0,5} & \text{pro} \quad -2,5 \leq z \ \leq -0,5 \\ 1 & \text{pro} \quad -0,5 < z \ < 0,5 \\ 10^{-2,5(z-0,5)} & \text{pro} \quad 0,5 \leq z \ \leq 1,3 \\ 0 & \text{pro} \quad z \ > 1,3 \end{cases} \tag{3.18}$$

Přizpůsobení kritických pásmových filtrů křivkám stejné hlasitosti

Intenzitu zvuku v závislosti na kmitočtu vnímá člověk jako **hlasitost zvuku**. Aby bylo možné přizpůsobit výkonové spektrum $P(\omega)$ této vlastnosti, je potřeba provést preemfázi kritických pásmových propustí pomocí aproximující křivky $E(\omega)$.

$$\Phi_m(\Omega) = E(\Omega)\Psi(\Omega - \Omega_m), \tag{3.19}$$

kde Ω_m [bark] je střední kmitočet m-té kritické pásmové propusti, $m = 0, \ldots, M-1$.

Funkci $E(\Omega)$ získáme aplikací vztahu (3.17) na funkci $E(\omega)$, která vyjadřuje aproximaci závislosti citlivosti lidského sluchu na kmitočtu a je definována vztahem (pro úroveň 40 Ph)

$$E(\omega) = K\frac{\omega^4(\omega^2 + 56,9 \cdot 10^6)}{(\omega^2 + 6.3 \cdot 10^6)^2(\omega^2 + 379,4 \cdot 10^6)(\omega^6 + 9,6 \cdot 10^{26})}, \tag{3.20}$$

kde $\omega = 2\pi f$ a K je konstanta, jejíž velikost se může zvolit tak, aby kritický pásmový filtr dosáhl úrovně 0 dB pro nejvyšší hodnoty intenzity.

Vážená spektrální sumarizace vzorků výkonového spektra

Vliv m-tého kritického pásmového filtru, který byl přizpůsoben křivkám stejné hlasitosti, na vypočteného výkonové spektrum $P(\omega)$ lze definovat vztahem

$$\Xi(\Omega_m) = \sum_{\omega=\omega_{md}}^{\omega_{mh}} P(\omega)\Phi\Big(\Omega(\omega)\Big), \tag{3.21}$$

kde $m = 1, \ldots, M-2$. Sumační meze ω_{md} a ω_{mh} lze vypočíst z inverzního vztahu k rovnici (3.17).

Uplatnění vztahu mezi intenzitou zvuku a vnímanou hlasitostí

Na hodnoty $\Xi(\Omega_m)$ je dále potřeba uplatnit závislost mezi intenzitou zvuku a vnímanou hlasitostí

$$\xi(\Omega_m) = \sqrt[3]{\left(\Xi(\Omega_m)\right)}, \tag{3.22}$$

kde $m = 1, \ldots, M-2$.

Aproximace celopólového modelu

Podle [1] lze získat autokorelační funkci $R(i)$ ze vztahu

$$\begin{aligned} R(i) = \; & \tfrac{1}{2(M-1)} \Bigg\{ \xi(\Omega_0)\cos(i\omega_0) + 2\left(\sum_{m=1}^{M-2} \xi(\Omega_m)\cos(i\omega_m) \right) + \\ & + \xi(\Omega_{M-1})\cos(i\omega_{M-1}) \Bigg\} \end{aligned} \tag{3.23}$$

kde $i = 0, \ldots, Q$, $\omega_{M-1} = \pi$, Q je řád modelu.

Z takto vypočtené autokorelační funkce $R(i)$ lze pomocí Levinson-Durbinova algoritmu určit koeficienty lineární predikce.

3.4.3 Metoda RASTA

Protože **metoda RASTA** (RelAtive SpecTRal) byla navržena jako jakési rozšíření PLP analýzy, byla zařazena do kapitoly o lineární predikci.

Tato metoda je založena na vlastnostech lidského sluchu, a to konkrétně na citlivosti na pomalu se měnící podněty. V důsledku to znamená, že tato metoda potlačuje ty spektrální složky řečového signálu, které se mění rychleji nebo pomaleji, než je rychlost změn řeči.

Metodou RASTA lze modifikovat parametrizaci PLP, takto upravená PLP analýza se nazývá **RASTA-PLP**. Kroky výpočtu této parametrizace jsou shodné s PLP analýzou uvedenou v části 3.4.2, pouze s tím rozdílem, že mezi kroky **vážená spektrální sumarizace vzorků výkonového spektra** a **uplatnění vztahu mezi intenzitou zvuku a vnímanou hlasitostí** jsou začleněny následující body[1]:

Komprese jednotlivých komponent spektrálních amplitud vhodnou statickou nelineární transformací

Komprese se provede pomocí vztahu:

$$y(x) = \ln(1 + Jx), \tag{3.24}$$

kde J je kladná konstanta, závislá na řečovém signálu. Je vhodné ji zvolit nepřímo úměrnou velikosti energie šumu z úseku signálu bez přítomnosti řeči.

Filtrace časové trajektorie komponent spektrálních amplitud

K filtraci se využívá **filtr RASTA** s přenosovou funkcí

$$H(z) = \frac{0.2 + 0.1z^{-1} - 0.1z^{-3} - 0.2z^{-4}}{1 - 0.98z^{-1}}. \tag{3.25}$$

Vyjádření pomocí diferenční rovnice

$$y[k] = 0.98y[k-1] + 0.2x[k] + 0.1x[k-1] - 0.1x[k-3] - 0.2x[k-4], \tag{3.26}$$

kde $x[k]$ je vstupní složka filtru, a $y[k]$ je výstupní složka filtru.

Transformace filtrovaných komponent expandující statickou nelineární transformací

Jedná se o aplikaci inverzního vztahu k výrazu (3.24), tedy

$$y = \frac{\mathrm{e}^y - 1}{J}, \tag{3.27}$$

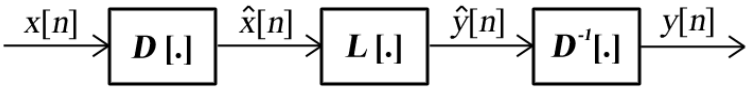
kde e je základ přirozeného logaritmu.

Protože jmenovatel nebo čitatel může nabývat i záporných hodnot, je výhodnější použít aproximativní transformaci

$$y = \frac{\mathrm{e}^y}{J}. \tag{3.28}$$

3.5 Homomorfní zpracování řeči

Homomorfní analýza patří ke skupině postupů nelineárního zpracování signálů, které jsou založeny na využití zobecněného principu superpozice. Tyto postupy se hodí pro oddělení signálů, které vznikly konvolucí či násobením dvou nebo více složek. Cílem analýzy je určit parametry systému [2].



Obr. 3.2: Obecné schéma homomorfního systému.

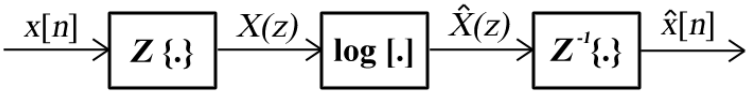
Obecné schéma homomorfního systému je na obr. 3.2, kde D je charakteristický systém. Jeho úkolem je převést konvolutorní součin vstupních signálů na součet modifikovaných vstupních signálů. Příkladem takového systému může být systém znázorněný na obr. 3.3, přičemž

$$x[n] = x_1[n] * x_2[n], \tag{3.29}$$

$$X(z) = X_1(z) \cdot X_2(z), \tag{3.30}$$

$$\hat{X}(z) = \hat{X}_1(z) + \hat{X}_2(z), \tag{3.31}$$

$$\hat{x}[n] = \hat{x}_1[n] + \hat{x}_2[n], \tag{3.32}$$



Obr. 3.3: Blokové schéma charakteristického systému D.

L je lineární systém, který realizuje lineární filtraci součtu vstupních signálů. A systém D^{-1} je inverzní k systému D, tedy převádí součet na konvoluci.

3.5.1 Krátkodobá kepstrální analýza

Jestliže použijeme substituci:

$$z = \mathrm{e}^{\mathrm{j}\omega}, \tag{3.33}$$

a popíšeme $X[k]$ a $\hat{X}[k]$ takto [2]:

$$X[k] = \sum_{n=0}^{N-1} x[n]\mathrm{e}^{\mathrm{j}\frac{2\pi kn}{N}} \quad , k = 0, 1, \ldots, N-1, \tag{3.34}$$

$$\hat{X}[k] = \log(X[k]) \quad , k = 0, 1, \ldots, N-1, \tag{3.35}$$

pak podle [2] lze vypočíst **krátkodobé komplexní kepstrum** podle vztahu:

$$\hat{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \hat{X}[k]\mathrm{e}^{\mathrm{j}\frac{2\pi kn}{N}} \quad , n = 0, 1, \ldots, N-1, \tag{3.36}$$

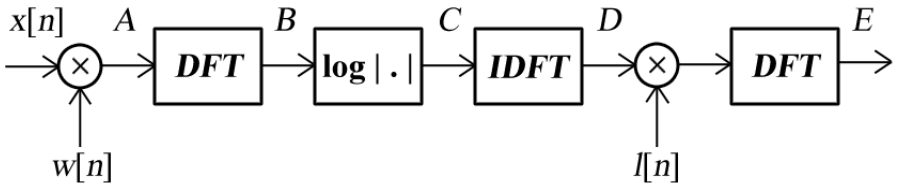
a **krátkodobé kepstrum** podle vztahu :

$$c[n] = \frac{1}{N} \sum_{k=0}^{N-1} \log |X[k]| \mathrm{e}^{\mathrm{j}\frac{2\pi kn}{N}} \quad , n = 0, 1, \ldots, N-1. \tag{3.37}$$

Proces krátkodobé kepstrální analýzy je znázorněn na obr. 3.4. Předpokládejme, že signál A vznikl diskrétní konvolucí $x[n]$ a okénkové funkce $w[n]$. Tento signál je přiveden na vstup bloku DFT a výstupní signál B, který je Fourierovou transformací konvoluce buzení a impulzní odezvy hlasového ústrojí přichází na blok log|.|. Jeho výstup C je sumou logaritmů transformace buzení a transformace impulzní odezvy hlasového ústrojí a je složen z pomalu se měnících složek způsobených přenosy hlasového ústrojí a rychle se měnících periodických složek způsobených periodickým buzením. Abychom získali vyhlazenou spektrální obálku E, je potřeba odstranit rychle se měnící složky ze signálu C. A to tak, že vynásobíme kepstrum **kepstrálním okénkem** $l[n]$, který vybere ze signálu kepstra D odpovídající část:

$$l[n] = \begin{cases} 1, & |n| < n_0, \\ 0, & |n| \geq n_0, \end{cases} \tag{3.38}$$

kde n_0 je vybráno tak, aby bylo menší než perioda základního hlasivkového tónu.



Obr. 3.4: Blokové schéma krátkodobá kepstrální analýzy.

3.5.2 Kepstrální koeficienty LPC

Hlasový trakt lze modelovat lineárním systémem, který může být popsán také pomocí kepstrálních koeficientů. Je možné je získat z koeficientů LPC pomocí následujících vztahů [1]

$$\begin{aligned} c(1) &= -a(1), \\ c(k) &= -a_k - \sum_{i=1}^{k-1} \left(\tfrac{i}{k}\right) c(i) a_{k-1}, \qquad \text{pro} \quad 2 \leq k \leq Q, \\ c(k) &= -\sum_{i=1}^{Q} \left(\tfrac{k-i}{k}\right) c(k-i) a_i, \qquad \text{pro} \quad k = Q+1, Q+2, \ldots. \end{aligned} \tag{3.39}$$

kde $a(i)$ jsou LPC koeficienty, Q je řád modelu, $k = 1, \ldots, Q^*$, $Q^* \geq Q$, přičemž se často volí $Q^* = Q$.

Tyto kepstrální koeficienty jsou však obecně odlišné od kepstrálních koeficientů popsaných v odstavci 3.5.1, protože jsou vztaženy k vyhlazené spektrální obálce získané pomocí LPC analýzy.

Koeficienty s vyššími indexy nabývají nižších hodnot, proto je výhodné provést **liftering** podle vztahu

$$c_{lift}(n) = \left(1 + \frac{L}{2} \sin\left(\pi \frac{n}{L}\right)\right) c(n), \tag{3.40}$$

kde L je váha lifteringu, typicky $L = 22$.

3.5.3 Melovské kepstrální koeficienty

Zpracování pomocí melovských kepstrálních koeficientů je navrženo, podobně jako perceptivní lineární prediktivní analýza, s ohledem na nelineární vnímání kmitočtů lidským sluchem. Oproti PLP analýze k tomu využívá trojúhelníkových filtrů, které jsou lineárně rozloženy v **melovské škále**

$$f_{mel} = 2595 \log_{10}\left(1 + \frac{f}{700}\right), \tag{3.41}$$

kde f je kmitočet v normální škále [Hz], a f_{mel} je kmitočet v nelineární melovské škále [mel].

Postup získání melovských kepstrálních koeficientů je následující:

Preemfáze řečového signálu

Jedná se o filtraci signálu s přenosovou funkcí

$$H(z) = 1 - \alpha \mathrm{z}^{-1}, \tag{3.42}$$

kde α nabývá hodnot od 0,93 do 0,98.

Segmentace a váhování řečového signálu

Násobení segmentů řečového signálu s okénkovou funkcí. Zpravidla se jedná o Hammingovo okénko, segmenty mají délku 10 - 30 ms.

Filtrace melovskou bankou filtrů

Střední kmitočty jednotlivých filtrů se dají vypočíst podle vztahu

$$b_{m,i} = b_{m,i-1} + \Delta_m, \tag{3.43}$$

kde $b_{m,0} = 0$ mel, $i = 1, 2, \ldots, M'$, $\Delta_m = B_{mel}/(M'+1)$, M' je počet trojúhelníkových filtrů v bance filtrů a B_{mel} [mel] je celková šířka přenášeného pásma.

Odezvy jednotlivých filtrů pak lze vyjádřit

$$y_m(i) = \sum_{f=b_{i-1}}^{b_{i+1}} |S(f)|u(f,i), \tag{3.44}$$

$u(f,i)$ vyjadřuje trojúhelníkové filtry

$$u(f,i) = \begin{cases} (f - b_{i-1})/(b_i - b_{i-1}) & \text{pro} \quad b_{i-1} \leq f < b_i \\ (f - b_{i+1})/(b_i - b_{i+1}) & \text{pro} \quad b_i \leq f < b_{i+1} \\ 0 & \text{pro} \end{cases} \tag{3.45}$$

Výpočet logaritmů výstupů z jednotlivých filtrů.

DCT

Posledním krokem výpočtu je provedení inverzní diskrétní Fourierovy transformace. Vzhledem k tomu, že výkonové spektrum je reálné a symetrické, lze IDFT redukovat na disktrétní kosínovou transformaci

$$c_m(j) = \sum_{i=i}^{M'} \log y_m(i) \cos\Big(\frac{\pi j}{M'}(i - 0,5)\Big), \tag{3.46}$$

kde M' je počet pásem melovské banky filtrů, M je počet melovských kepstrálních koeficientů. $c_m(0)$ je často nahrazován logaritmem krátkodobé energie.

3.6 Vektorová kvantizace

Pojem **kvantizace** označuje proces, pomocí něhož lze převést analogovou hodnotu na hodnotu z konečného počtu číselných hodnot. Jestliže se tento proces uplatní na jednotlivé hodnoty, jedná se tzv. **skalární kvantizaci**. V případě, že je nutné provádět kvantizaci spojeného bloku (například jeden vektor příznaků segmentovaného řečového signálu), jedná se o **kvantizaci vektorovou**.

3.6.1 Kódová kniha

Nechť existuje Q dimenzionální množina X, ve které existuje množina vektorů $\boldsymbol{x} = [x_1, x_2, \ldots, x_Q]^T$. L úrovňový Q dimenzionální vektorový kvanzitér převede každý vstupní vektor $\boldsymbol{x}$ na kvantovaný (reprodukční) vektor $\boldsymbol{v} = q(\boldsymbol{x})$, tím, že jej vybere z konečné reprodukční abecedy $\boldsymbol{V} = \{\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v}_L\}$, přičemž v praxi bývají vektory $\boldsymbol{v}$ opět Q dimenzionální. Tento vektorový kvanzizér lze plně popsat pomocí tzv. **kódové knihy**.

Při kvantování vzniká kvantizační chyba, kterou lze vyjádřit pomocí míry odlišnoti $d(\boldsymbol{x}, \boldsymbol{v})$ mezi vstupním vektorm $\boldsymbol{x}$ a reprodukčním vektorem $\boldsymbol{v}$. Aby byl vektorový kvantizér optimální, musí se zajistit, aby celkové kvantizační zkreslení J bylo minimální. Toto celkové zkreslení lze také nazvat **kriteriální funkcí**.

$$J = \sum_{i=1}^{L} \min_{v_i} \sum_{x \in T_i} d(\boldsymbol{x}, \boldsymbol{v}_i), \tag{3.47}$$

kde L je počet položek kódové knihy, centroid $\boldsymbol{v}_i$ je takový vektor, který minimalizuje průměrné zkreslení v oblasti X_i a T_i jsou tzv. **shluky**. Shluky jsou podmnožinou **trénovací množiny** T, kterou tvoří konečný počet vstupních vektorů $\boldsymbol{x}$. Odvození kriteriální funkce J lze nalézt v [1].

Rozklad trénovací množiny T na shluky T_i a vyhledávání centroidů lze realizovat pomocí **k-means algoritmu**.

3.6.2 K-means algoritmus

K-means algoritmus je iteračním algoritmem, který je vhodné implementovat rekurzivně. Probíhá v následujících krocích:

1. Výběr L počátečních centroidů $\boldsymbol{v}_1(1), \boldsymbol{v}_2(1), \ldots, \boldsymbol{v}_L(1)$, přičemž $\boldsymbol{v}_i(k)$ značí centroid i-tého shluku v k-té iteraci. V první iteraci se centroidy mohou volit libovolně.

2. Všechny vektory $\boldsymbol{x}$ trénovací množiny T se v každé iteraci rozdělí do shluků podle vztahu

$$\boldsymbol{x} \in T_j(k), \quad \text{jestliže} \quad d(\boldsymbol{x}, \boldsymbol{v}_j) < d(\boldsymbol{x}, \boldsymbol{v}_i), \tag{3.48}$$

3. Z takto vytvořených nových shluků $T_i(k)$ lze vypočíst nové centroidy

$$\boldsymbol{v}_j(k+1) = \frac{1}{n_j(k)} \sum_{\boldsymbol{x} \in T_j(k)} \boldsymbol{x}, \tag{3.49}$$

   kde $n_j(k)$ je počet vektorů $\boldsymbol{x}$ j-tého shluku v k-té iteraci. Při dodržení dílčí kriteriální funkce J_j

$$J_j(k+1) = \sum_{\boldsymbol{x} \in T_j(k)} d^2(\boldsymbol{x}, \boldsymbol{v}_j(k+1)), \tag{3.50}$$

4. Jestliže nedošlo v žádném ze shluků T_i ke změně centroidu, nebo jestliže dojde k dosažení definovaného prahu kriteriální funkce J

$$J(k) = \sum_{i=1}^{L} J_i(k), \tag{3.51}$$

pak lze algoritmus ukončit, v opačném případě se pokračuje opakováním bodu 2 až 4.

4 ROZPOZNÁVÁNÍ ŘEČI

Rozpoznávání řeči pomocí stroje je obtížným úkolem. Důvody souvisí s variabilitou řečníka (např. emoční stav, situace pronášené promluvy, různorodost řečníků), prostředím, ve kterém je rozpoznávaná promluva pronášena, ale také na typu a složitosti úlohy (např. rozpoznávání izolovaných slov, čtené promluvy nebo souvislé řeči).

Existují dvě skupiny pro přístup k rozpoznávání řeči:

- **Porovnávání se vzory** – principem je vypočtení vzdálenosti vzorového obrazu od testovaného na základě metody **dynamického programovaní**
- **statistické metody** – promluvy jsou modelovány pomocí tzv. **skrytých Markovových modelů**

4.1 Dynamické borcení času

Při zkoumání promluv promlouvané týmž řečníkem se zjistilo, že základními odlišnostmi v těchto signálech jsou nejen v nestejné délce slov, ale také zejména v nepoměru mezi délkami fonémů uvnitř slova. Toto nelze vyřešit pomocí lineární časové normalizace, proto se využívá nelineární časové normalizace, která je modelovaná nelineárním borcením časové osy. Metoda, která využívá tohoto efektu a dynamického programování (coby mechanismu určování vzdáleností mezi obrazy) se nazývá **dynamické borcení času** (dynamic time warping) – DTW.

4.1.1 Princip

Jestliže existuje obraz testovaného slova $\boldsymbol{A}$ a obraz referenčního slova $\boldsymbol{B}$

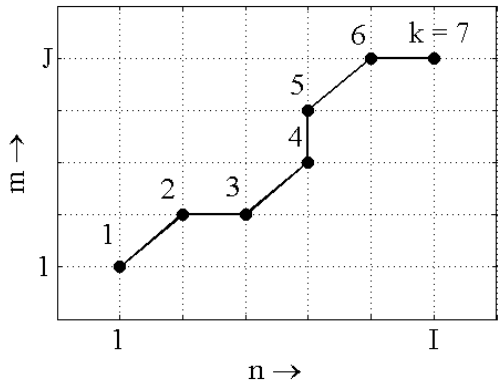
$$\begin{aligned} \boldsymbol{A} &= \{\boldsymbol{a}(1), \boldsymbol{a}(2), \ldots, \boldsymbol{a}(n), \ldots, \boldsymbol{a}(I)\}, \\ \boldsymbol{B} &= \{\boldsymbol{b}(1), \boldsymbol{b}(2), \ldots, \boldsymbol{b}(m), \ldots, \boldsymbol{b}(J)\}, \end{aligned} \tag{4.1}$$

přičemž $\boldsymbol{a}(n)$ je n-tý vektor příznaků, $\boldsymbol{b}(m)$ je m-tý vektor příznaků, I, J je počet vektorů příznaků.

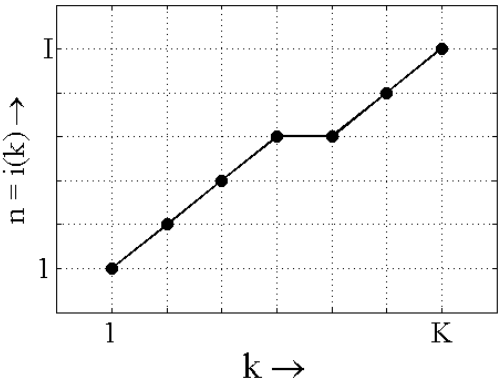
Algoritmus DTW vyhledá v rovině (m, n) optimální cestu (viz obr. 4.1). Pak je možné vyjádřit časově proměnné m a n pomocí funkce k (viz obr. 4.2 a obr. 4.3)

$$\begin{aligned} n &= i(k), \qquad k = 1, \ldots, K, \\ m &= j(k), \qquad k = 1, \ldots, K, \end{aligned} \tag{4.2}$$

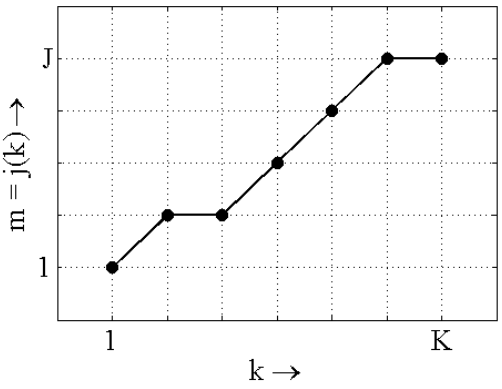
kde K je délka obecné časové osy.



Obr. 4.1: Optimální cesta nalezená algoritmem DTW.



Obr. 4.2: Funkce $i(k)$ pro krokování testovaným obrazem.



Obr. 4.3: Funkce $j(k)$ pro krokování referenčním obrazem.

Omezení cesty

Je potřeba vymezit hraniční body

$$\begin{aligned} i(1) &= 1, \quad i(K) = I, \\ j(1) &= 1, \quad j(K) = J. \end{aligned} \tag{4.3}$$

Aby nedocházelo k nadměrné kompresi nebo expanzi aplikují se na funkci DTW omezení na lokální souvislost a lokální strmost

$$\begin{aligned} 0 \leq i(k) - i(k-1) \leq I^*, \\ 0 \leq j(k) - j(k-1) \leq J^*, \end{aligned} \tag{4.4}$$

kde I^*, $J^* = 1, 2, 3$. Pokud se I^* nebo J^* zvolí větší než 1, může funkce DTW při porovnávání některé segmenty přeskočit.

Zobecní-li se podmínky lokálního omezení strmosti (4.4), při splnění počáteční a koncové podmínky (4.3) lze vymezit oblast průchodu funkce DTW

$$\begin{aligned} 1 + \alpha[i(k) - 1] \leq j(k) \leq 1 + \beta[i(k) - 1], \\ J + \beta[i(k) - I] \leq j(k) \leq J + \alpha[i(k) - I], \end{aligned} \tag{4.5}$$

kde α je minimální a β maximální směrnice vymezující přípustnou oblast.

Měření vzdálenosti

Skutečnou minimální celkovou vzdálenost mezi obrazy $\boldsymbol{A}$ a $\boldsymbol{B}$ lze vyjádřit vztahem

$$D(\boldsymbol{A}, \boldsymbol{B}) = \min_{i(k), j(k), K} \left(\frac{\sum_{k=1}^{K} d[i(k), j(k)] W(k)}{N(W)} \right) \tag{4.6}$$

kde $d[i(k), j(k)]$ je lokální vzdálenost mezi n-tým segmentem obrazu $\boldsymbol{A}$ a m-tým segmentem obrazu $\boldsymbol{B}$, $W(k)$ je hodnota váhové funkce odpovídající k-tému kroku cesty, $N(W)$ je normalizační faktor závislý na vahách.

Váhová funkce a normalizační faktor

Váhová funkce $W(k)$ závisí pouze na lokální cestě. Normalizační faktor $N(W)$ je funkcí váhové funkce, a zavádí se z důvodu kompenzace délky (počtu kroků) funkce DTW.

Existují čtyři typy váhových funkcí:

typ a) symetrická

$$W(k) = [i(k) - i(k-1)] + [j(k) - j(k-1)], \tag{4.7}$$

$$N(W) = \sum_{k=1}^{K} W(k) = I + J, \tag{4.8}$$

typ b) asymetrická

1)

$$W(k) = i(k) - i(k-1), \tag{4.9}$$

$$N(W) = I, \tag{4.10}$$

2)

$$W(k) = j(k) - j(k-1), \tag{4.11}$$

$$N(W) = J. \tag{4.12}$$

typ c)

$$W(k) = \min[i(k) - i(k-1), j(k) - j(k-1)], \tag{4.13}$$

typ d)

$$W(k) = \max[i(k) - i(k-1), j(k) - j(k-1)]. \tag{4.14}$$

Hodnota normalizačního faktoru pro typ c) a d) je silně závislá na průběhu cesty, v praxi se nejlépe osvědčilo použít konstantu.

$$N(W) = I. \tag{4.15}$$

5 UMĚLÉ NEURONOVÉ SÍTĚ

Problematika umělých neuronových sítí je značně rozsáhlá, a omezený prostor této práce neumožňuje se jí dopodrobna zabývat. Proto se bude tato kapitola snažit ukázat pouze možnosti a princip dopředné sítě typu backpropagation.

Umělá neuronová síť vychází do jisté míry z biologické neuronové síťě, od které převzala velmi zjednodušený matematický model. Z hlediska realizace nemají tyto síťě spolu nic společného.

Pod pojmem umělé neuronové síťě si lze představit obecně nelineární, adaptabilní stroj se schopností učit se[1].

5.1 Neuron a jeho matematický popis

Základním procesním prvkem umělé neuronové síťě je **neuron**, tento název je používán z historických důvodů, jeho podobnost s biologickým modelem neuronu je značně vzdálená.

Každý neuron má více vstupů, ale jen jeden výstup, tento výstup však může být přiveden na vstup téhož neuronu nebo vstup jiných neuronů. Každý neuron na základě kombinace vstupů a vlastních parametrů - **vah** a **prahu**, vypočte hodnotu výstupu, kterou transformuje na základě **aktivační funkce**.

Výpočet výstupní hodnoty neuronu je následující:

$$y = f\Big(\sum_{i=1}^{N} w_i x_i - \theta\Big), \tag{5.1}$$

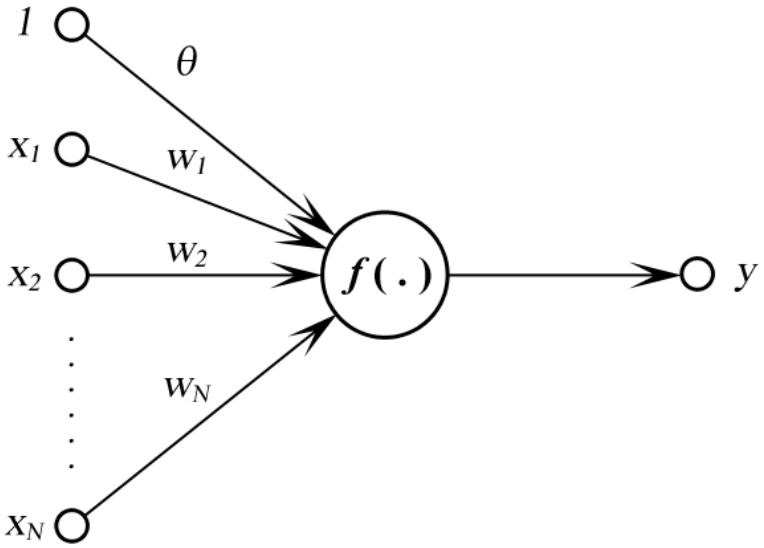
přičemž $\boldsymbol{x} = [x_1, x_2, \ldots, x_N]^T$ je vstupní vektor, $\boldsymbol{w} = [w_1, w_2, \ldots, w_N]^T$ je vektor vah a θ udává hodnotu prahu neuronu. Funkce $f(\alpha)$ je aktivační funkci neuronu a je zpravidla nelineární.

Práh neuronu si lze představit jako nultý vstup neuronu x_0, jehož hodnota je rovna jedné, váha w_0 toho vstupu pak bude odpovídat právě prahu neuronu θ, na základě této úvahy a úpravě předchozího vztahu, lze získat rovnici

$$y = f\Big(\boldsymbol{w}^T \boldsymbol{x}\Big). \tag{5.2}$$

přičemž $\boldsymbol{x} = [1, x_1, \ldots, x_N]^T$ je vstupní vektor, $\boldsymbol{w} = [\theta, w_1, \ldots, w_N]^T$ je vektor vah.

[1] Pozor, zde nelze mluvit o umělé inteligenci.



Obr. 5.1: Schéma neuronu

5.2 Vícevrstvé umělé neuronové síťě

Na základě zavádění vazeb (tj. propojení) mezi neurony lze uvažovat o síťích **dopředných** a **rekurentních**. Dopředné sítě se vyznačují vlastností šířením vstupních dat na výstup pouze jedním - dopředným směrem, kdežto u rekurentních sítí jsou zaváděny zpětné vazby nejen mezi neurony, ale také mezi vrstvami. **Vrstvou** se rozumí M paralelně pracujících neuronů, které na sobě nezávisle mohou transformovat vstupní vektor na výstupní hodnotu. Pokud se zapojí více těchto vrstev do kaskády, hovoří se o tzv. **vícevrstvých umělých neuronových síťích**.

5.2.1 Backpropagation

Aby bylo možné využívat síť ke konkrétním účelům, je nezbytné nalézt takové váhy w ve všech vrstvách dané sítě tak, aby bylo možné výstupní data nějakým způsobem prezentovat. Tento problém se dá vyřešit postupným předkládáním **učební množiny**, tj. dvojic vstupně-výstupních vektorů, které spolu souvisí. Algoritmus modifikace vah neuronů během překládání učebních množin se nazývá **učení**.

Jednou z vícevrstvých dopředných umělých neuronových sítí je síť typu **backpropagation**. Tato síť je charakteristická svým učebním procesem, který pracuje na principu **zpětného šíření chyby**. Odvození toho učebního algoritmu lze nalézt v [4], zde je uveden pouze výsledný vztah

$$ {}_i^{n-1}\boldsymbol{w}(t+1) = {}_i^{n-1}\boldsymbol{w}(t) + 2\mu_t f'\left({}_i^{n-1}\alpha_p\right){}_i^{n-1}e_p\boldsymbol{y}_p, \tag{5.3} $$

kde t je krok iterace, koeficient μ_t určuje rychlost učení, ${}_i^{n-1}\boldsymbol{w}$ značí sobor vah $\boldsymbol{w}$ i-tého neuronu v $n-1$ vrstvě, $f^{'}(.)$ je derivace aktivační funkce a $\boldsymbol{y}_p$ je vektor výstupů předchozí vrstvy. Chybu ${}_i^{n-1}e_p$ lze vypočíst podle vztahu

$$ {}_i^{n-1}e_p = \sum_{j=1}^{M} \left[{}_j^{n-2}e_p f^{'}\left({}_j^{n-2}\alpha_p\right) {}_j^{n-2}w_i \right], \tag{5.4} $$

Epochou učení se rozumí provádění učebních kroků se všemi dvojicemi učební množiny. Přičemž **učebním krokem** se nazývá postupná modifikace všech vah neuronové sítě. Každý krok učení probíhá takto:

1. Zjištění odezvy neuronové sítě $\boldsymbol{y}_p$ na předložený vstupního vektoru $\boldsymbol{x}_p$.
2. Výpočet chybového vektoru $\boldsymbol{e}_p = \boldsymbol{d}_p - \boldsymbol{e}_p$, kde $\boldsymbol{d}_p$ je požadovaný výstupní vektor.
3. Výpočet rozložení chyby podle vztahu (5.4), přičemž pro výstupní vrstvu platí ${}^{n-2}\boldsymbol{e}_p = \boldsymbol{e}_p$.
4. Modifikace vektoru vah podle vztahu (5.3).

Body 3 a 4 se opakují do doby, kdy jsou upraveny všechny váhy neuronové sítě.

6 VÝBĚR VHODNÝCH PARAMETRIZACÍ

Cílem se této kapitoly je vhodně zobrazit schopnost klasifikace jednotlivých parametrizací a následně vybrat ty parametrizace nebo jejich kombinace, které budou vhodné pro detekci chybné výslovnosti řeči.

Od PaedDr. Lenky Němcové byly získány promluvy s chybnou výslovností řeči. Způsob vzniku nahrávek je popsán v [10].

Nejprve bylo vybráno více jak 1200 úseků řečových signálů charakterizující třicet hlásek abecedy, které byly vysloveny správně:

A, B, C, Č, D, Ď, E, F, G, H, Ch, I, J, K, L, M, N, Ň, O, R, P, Ř, S, Š, T, Ť, U, V, Z, Ž,

a čtyři hlásky, které byly vysloveny chybně:

R, Ř, S, Š,

z důvodu rozlišení budou v dalším textu označeny:

vR, vŘ, vS, vŠ.

Tyto signály byly nejprve segmentovány pomocí Hammingova okénka s délkou 1024 vzorků (to odpovídá časovému úseku $21,\overline{3}$ms při $f_{vz} = 48$kHz) a posunem mezi segmenty o 256 vzorků (překryv jednotlivých segmentů je tedy 75%). Každý segment byl podroben analýzám, jejichž výčet je uveden v kapitole 6.1.

Pro každý takto segmentovaný a analyzovaný úsek řečového signálu byly pomocí těchto příznaků sestaveny obrazy[1] pro jednotlivé parametrizace.

Aby bylo možné porovnat parametrizace mezi sebou, je nutné nejprve zavést **koeficient parametrizace**, který udává schopnost klasifikace danou parametrizací.

Protože počty vektorů jednotlivých obrazů se liší, lze s výhodou využít DTW algoritmus. Pro výpočet lokálních vzdáleností byla zvolena euklidovská vzdálenost. Výstupem DTW algoritmu bude skalár, který lze také chápat jako vyjádření míry odlišnosti mezi dvěma obrazy.

Po porovnání všech obrazů (každý s každým) v rámci jedné hlásky (třídy), je vypočtena průměrná míra odlišnosti třídy $\boldsymbol{v}_{mean}$.

Dále je nutné určit jednoho reprezantanta každé třídy. K tomu je výhodné využít algoritmu k-means, který dokáže rozdělit danou třídu do shluků tak, aby součet všech euklidovských vzdáleností mezi vzory a centroidy byl co nejmenší.

Problémem tohoto algoritmu je výpočet centroidů. K-means algoritmus totiž vyžaduje, aby centroid (a také i ostatní obrazy) byly pouze vektory shodné délky, kdežto obrazy jednotlivých hlásek jsou matice sestavené z různého počtu vektorů shodné délky. Je tedy vhodné nahradit míru zkreslení tohoto algoritmu výpočtem DTW. Druhým závažným problémem je možné zkreslení hodnot příznakových matic centroidů, proto by bylo nanejvýš vhodné, aby byly tyto centroidy kvantovány

[1] Zde je výraz „obraz“ myšlen ve významu reprezentace daného úseku řečového signálu pomocí několika vektorů získaných příznaků. Počet těchto vektorů odpovídá počtu segmentů.

původními vzory dané třídy, tzn. centroidem se může stát jen reálný vzor z dané třídy.

Takto modifikovaný k-means algoritmus [2] je schopen nalézt centroid pro každou třídu. Dále jsou opět pomocí DTW algoritmu srovnávány obrazy všech reprezentantů (centroidů) mezi sebou. Sestaví se vektor $\boldsymbol{v}_{min}$, jehož prvky jsou minima míry odlišnosti každého reprezentanta od ostatních.

Koeficient k-té parametrizace $P_v^{(k)}$ pak lze vypočíst jako poměr průměrných hodnot $\mu(\boldsymbol{v})$ vektorů $\boldsymbol{v}_{min}$ a $\boldsymbol{v}_{mean}$

$$P_v^{(k)} = \frac{\mu(\boldsymbol{v}_{min}^{(k)})}{\mu(\boldsymbol{v}_{mean}^{(k)})}. \tag{6.1}$$

6.1 Koeficienty parametrizací

Zkoumány byly tyto parametrizace:

- *ceps* – Kepstrální koeficiety krátkodobé kepstrální analýzy,
  1. $ceps_1$ – odpovídá prvnímu koeficientu *ceps*,
  2. $ceps_{1-10}$ – odpovídá $1 \div 10$ koeficientu *ceps*,
  3. $ceps_{1-16}$ – odpovídá $1 \div 16$ koeficientu *ceps*,
  4. $ceps_{1-2}$ – odpovídá prvnímu a druhému koeficientu *ceps*,
  5. $ceps_{1-24}$ – odpovídá $1 \div 24$ koeficientu *ceps*,
  6. $ceps_{1-40}$ – odpovídá $1 \div 40$ koeficientu *ceps*,
  7. $ceps_{1-56}$ – odpovídá $1 \div 56$ koeficientu *ceps*,
  8. $ceps_{1-84}$ – odpovídá $1 \div 84$ koeficientu *ceps*,
- 9. E – krátkodobá energie,
- *lpc* – koeficienty krátkodobé lineární prediktivní analýzy
  10. lpc_{10} – odpovídá *lpc* pro model řádu 10,
  11. lpc_{12} – odpovídá *lpc* pro model řádu 12,
  12. lpc_{16} – odpovídá *lpc* pro model řádu 16,
  13. lpc_{24} – odpovídá *lpc* pro model řádu 24,
- *lpcceps* – kepstrální koeficienty krátkodobé lineární prediktivní analýzy

[2] Ukázka funkčnosti je demonstrována skriptem `ukazka_kmeans.m`.

14. $lpcceps_{10}$ – odpovídá *lpcceps* pro model řádu 10,
15. $lpcceps_{12}$ – odpovídá *lpcceps* pro model řádu 12,
16. $lpcceps_{16}$ – odpovídá *lpcceps* pro model řádu 16,
17. $lpcceps_{24}$ – odpovídá *lpcceps* pro model řádu 24,

- $mfcc$ – melovské kepstrální koeficienty

18. $mfcc_{28}$ – odpovídá $mfcc$ s počtem pásmový filtrů 28,
19. $mfcc_{40}$ – odpovídá $mfcc$ s počtem pásmový filtrů 40,
20. $mfcc_{56}$ – odpovídá $mfcc$ s počtem pásmový filtrů 56,
21. $mfcc_{84}$ – odpovídá $mfcc$ s počtem pásmový filtrů 84,

- *plp* – koeficienty perceptivní lineární prediktivní analýzy

22. $plp_{10,28}$ – odpovídá *plp* pro model řádu 10 a počtu pásmových filtrů 28,
23. $plp_{10,40}$ – odpovídá *plp* pro model řádu 10 a počtu pásmových filtrů 40,
24. $plp_{10,56}$ – odpovídá *plp* pro model řádu 10 a počtu pásmových filtrů 56,
25. $plp_{10,84}$ – odpovídá *plp* pro model řádu 10 a počtu pásmových filtrů 84,
26. $plp_{12,28}$ – odpovídá *plp* pro model řádu 12 a počtu pásmových filtrů 28,
27. $plp_{12,40}$ – odpovídá *plp* pro model řádu 12 a počtu pásmových filtrů 40,
28. $plp_{12,56}$ – odpovídá *plp* pro model řádu 12 a počtu pásmových filtrů 56,
29. $plp_{12,84}$ – odpovídá *plp* pro model řádu 12 a počtu pásmových filtrů 84,
30. $plp_{16,28}$ – odpovídá *plp* pro model řádu 16 a počtu pásmových filtrů 28,
31. $plp_{16,40}$ – odpovídá *plp* pro model řádu 16 a počtu pásmových filtrů 40,
32. $plp_{16,56}$ – odpovídá *plp* pro model řádu 16 a počtu pásmových filtrů 56,
33. $plp_{16,84}$ – odpovídá *plp* pro model řádu 16 a počtu pásmových filtrů 84,
34. $plp_{24,28}$ – odpovídá *plp* pro model řádu 24 a počtu pásmových filtrů 28,
35. $plp_{24,40}$ – odpovídá *plp* pro model řádu 24 a počtu pásmových filtrů 40,
36. $plp_{24,56}$ – odpovídá *plp* pro model řádu 24 a počtu pásmových filtrů 56,
37. $plp_{24,84}$ – odpovídá *plp* pro model řádu 24 a počtu pásmových filtrů 84,

- *plpceps* – kepstrální koeficienty perceptivní lineární prediktivní analýzy

38. $plpceps_{10,28}$ – odpovídá *plpceps* pro model řádu 10 a počtu pásmových filtrů 28,

39. $plpceps_{10,40}$ – odpovídá *plpceps* pro model řádu 10 a počtu pásmových filtrů 40,
40. $plpceps_{10,56}$ – odpovídá *plpceps* pro model řádu 10 a počtu pásmových filtrů 56,
41. $plpceps_{10,84}$ – odpovídá *plpceps* pro model řádu 10 a počtu pásmových filtrů 84,
42. $plpceps_{12,28}$ – odpovídá *plpceps* pro model řádu 12 a počtu pásmových filtrů 28,
43. $plpceps_{12,40}$ – odpovídá *plpceps* pro model řádu 12 a počtu pásmových filtrů 40,
44. $plpceps_{12,56}$ – odpovídá *plpceps* pro model řádu 12 a počtu pásmových filtrů 56,
45. $plpceps_{12,84}$ – odpovídá *plpceps* pro model řádu 12 a počtu pásmových filtrů 84,
46. $plpceps_{16,28}$ – odpovídá *plpceps* pro model řádu 16 a počtu pásmových filtrů 28,
47. $plpceps_{16,40}$ – odpovídá *plpceps* pro model řádu 16 a počtu pásmových filtrů 40,
48. $plpceps_{16,56}$ – odpovídá *plpceps* pro model řádu 16 a počtu pásmových filtrů 56,
49. $plpceps_{16,84}$ – odpovídá *plpceps* pro model řádu 16 a počtu pásmových filtrů 84,
50. $plpceps_{24,28}$ – odpovídá *plpceps* pro model řádu 24 a počtu pásmových filtrů 28,
51. $plpceps_{24,40}$ – odpovídá *plpceps* pro model řádu 24 a počtu pásmových filtrů 40,
52. $plpceps_{24,56}$ – odpovídá *plpceps* pro model řádu 24 a počtu pásmových filtrů 56,
53. $plpceps_{24,84}$ – odpovídá *plpceps* pro model řádu 24 a počtu pásmových filtrů 84,

- *rastaplp* – koeficienty RASTA-PLP analýzy

54. $rastaplp_{10,28}$ – odpovídá *rastaplp* pro model řádu 10 a počtu pásmových filtrů 28,

55. $rastaplp_{10,40}$ – odpovídá *rastaplp* pro model řádu 10 a počtu pásmových filtrů 40,
56. $rastaplp_{10,56}$ – odpovídá *rastaplp* pro model řádu 10 a počtu pásmových filtrů 56,
57. $rastaplp_{10,84}$ – odpovídá *rastaplp* pro model řádu 10 a počtu pásmových filtrů 84,
58. $rastaplp_{12,28}$ – odpovídá *rastaplp* pro model řádu 12 a počtu pásmových filtrů 28,
59. $rastaplp_{12,40}$ – odpovídá *rastaplp* pro model řádu 12 a počtu pásmových filtrů 40,
60. $rastaplp_{12,56}$ – odpovídá *rastaplp* pro model řádu 12 a počtu pásmových filtrů 56,
61. $rastaplp_{12,84}$ – odpovídá *rastaplp* pro model řádu 12 a počtu pásmových filtrů 84,
62. $rastaplp_{16,28}$ – odpovídá *rastaplp* pro model řádu 16 a počtu pásmových filtrů 28,
63. $rastaplp_{16,40}$ – odpovídá *rastaplp* pro model řádu 16 a počtu pásmových filtrů 40,
64. $rastaplp_{16,56}$ – odpovídá *rastaplp* pro model řádu 16 a počtu pásmových filtrů 56,
65. $rastaplp_{16,84}$ – odpovídá *rastaplp* pro model řádu 16 a počtu pásmových filtrů 84,
66. $rastaplp_{24,28}$ – odpovídá *rastaplp* pro model řádu 24 a počtu pásmových filtrů 28,
67. $rastaplp_{24,40}$ – odpovídá *rastaplp* pro model řádu 24 a počtu pásmových filtrů 40,
68. $rastaplp_{24,56}$ – odpovídá *rastaplp* pro model řádu 24 a počtu pásmových filtrů 56,
69. $rastaplp_{24,84}$ – odpovídá *rastaplp* pro model řádu 24 a počtu pásmových filtrů 84,

- *rastaplpceps* – kepstrální koeficienty RASTA-PLP analýzy

70. $rastaplpceps_{10,28}$ – odpovídá *rastaplpceps* pro model řádu 10 a počtu pásmových filtrů 28,

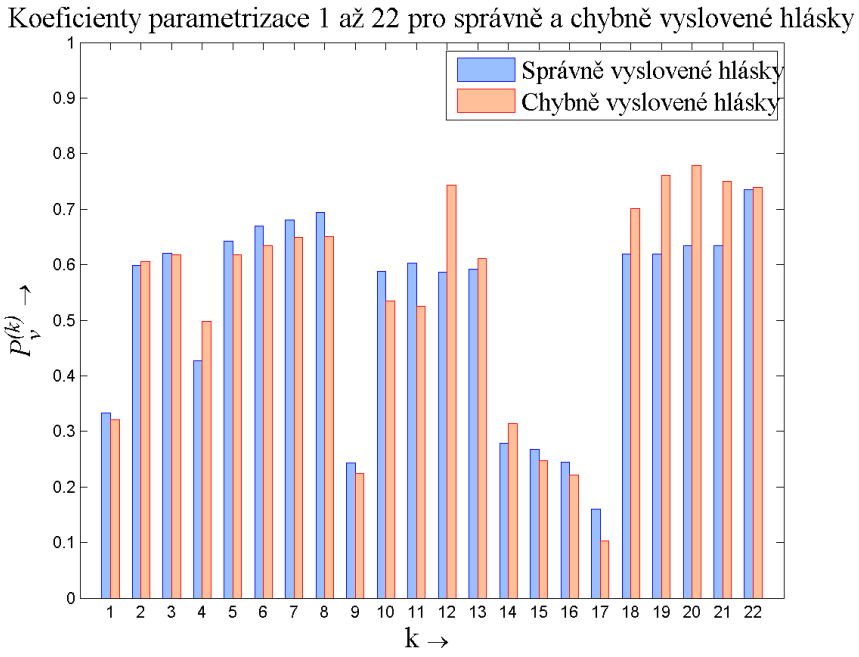
71. $rastaplpceps_{10,40}$ – odpovídá *rastaplpceps* pro model řádu 10 a počtu pásmových filtrů 40,
72. $rastaplpceps_{10,56}$ – odpovídá *rastaplpceps* pro model řádu 10 a počtu pásmových filtrů 56,
73. $rastaplpceps_{10,84}$ – odpovídá *rastaplpceps* pro model řádu 10 a počtu pásmových filtrů 84,
74. $rastaplpceps_{12,28}$ – odpovídá *rastaplpceps* pro model řádu 12 a počtu pásmových filtrů 28,
75. $rastaplpceps_{12,40}$ – odpovídá *rastaplpceps* pro model řádu 12 a počtu pásmových filtrů 40,
76. $rastaplpceps_{12,56}$ – odpovídá *rastaplpceps* pro model řádu 12 a počtu pásmových filtrů 56,
77. $rastaplpceps_{12,84}$ – odpovídá *rastaplpceps* pro model řádu 12 a počtu pásmových filtrů 84,
78. $rastaplpceps_{16,28}$ – odpovídá *rastaplpceps* pro model řádu 16 a počtu pásmových filtrů 28,
79. $rastaplpceps_{16,40}$ – odpovídá *rastaplpceps* pro model řádu 16 a počtu pásmových filtrů 40,
80. $rastaplpceps_{16,56}$ – odpovídá *rastaplpceps* pro model řádu 16 a počtu pásmových filtrů 56,
81. $rastaplpceps_{16,84}$ – odpovídá *rastaplpceps* pro model řádu 16 a počtu pásmových filtrů 84,
82. $rastaplpceps_{24,28}$ – odpovídá *rastaplpceps* pro model řádu 24 a počtu pásmových filtrů 28,
83. $rastaplpceps_{24,40}$ – odpovídá *rastaplpceps* pro model řádu 24 a počtu pásmových filtrů 40,
84. $rastaplpceps_{24,56}$ – odpovídá *rastaplpceps* pro model řádu 24 a počtu pásmových filtrů 56,
85. $rastaplpceps_{24,84}$ – odpovídá *rastaplpceps* pro model řádu 24 a počtu pásmových filtrů 84,

- 86. ZCR – krátkodobá funkce středního počtu průchodu signálu nulou.

6.1.1 Výběr parametrizací pro detekci chybné výslovnosti

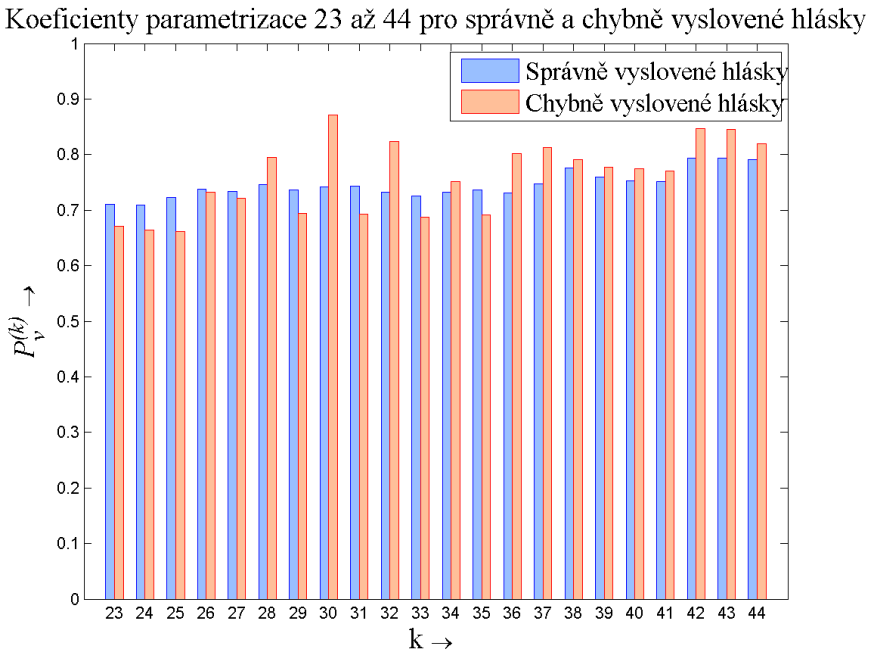
Porovnání koeficientů parametrizací je zobrazen na obr. 6.1 až 6.4. Čím je koeficient parametrizace větší, tím je schopnost klasifikace danou parametrizací vyšší. Koeficient parametrizace byl vypočten zvlášť pro správně a chybně vyslovené hlásky. Jestliže je koeficient parametrizace pro chybně vyslovené hlásky mnohem vyšší než pro hlásky vyslovené správně, pak lze o dané parametrizace prohlásit, že je vhodná pro detekci funkčních vad dyslalie.

Parametrizace byly vybrány tak, aby jejich schopnost klasifikace byla co nejvyšší, nebo takové, jejichž schopnost detekovat chybnou výslovnost byla relativně vysoká. Z důvodu snížení výpočetní náročnosti procesu rozpoznávání byla posledním kritériem výběru délka příznakových vektorů. Přednost dostali ty parametrizace, jež produkují příznakové vektory kratších délek. Těmto kritériům nejlépe vyhověli následující parametrizace: $mfcc_{56}$, $plp_{16,28}$, $plprasta_{12,84}$ a ZCR.

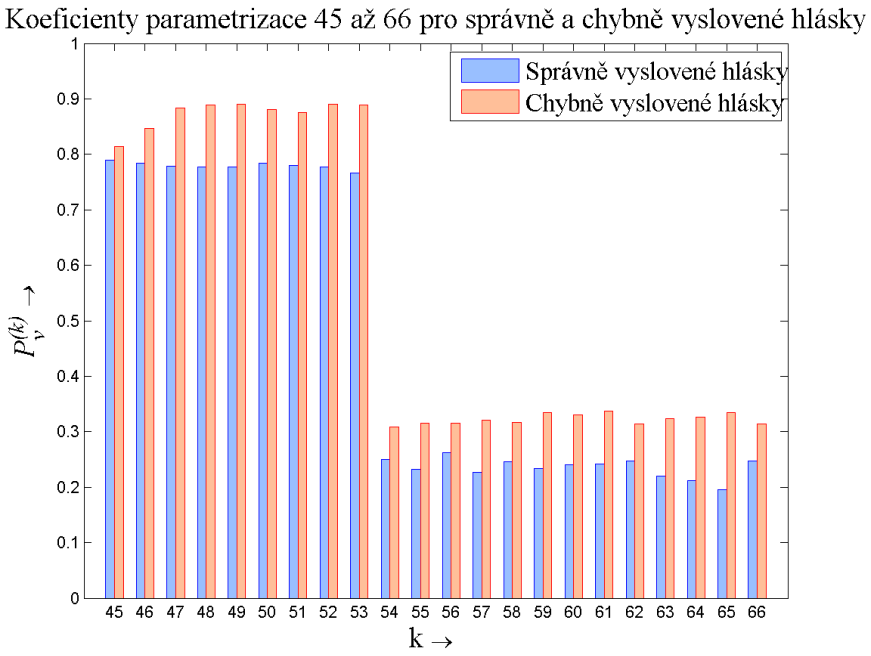


Obr. 6.1: Porovnání koeficientů parametrizace 1 až 22 pro správně a chybně vyslovené hlásky.

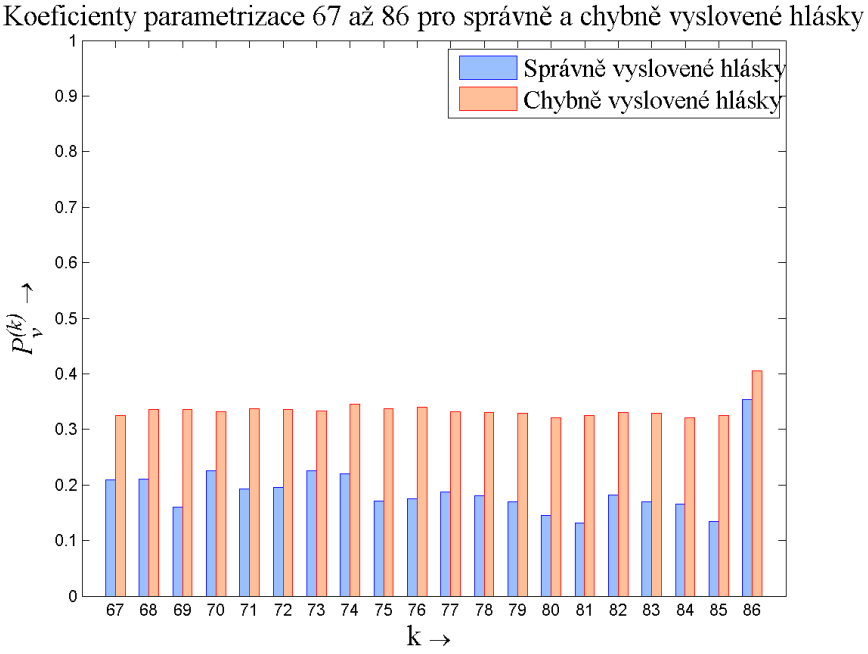
Z obr. 6.1 až 6.4 lze říci, že schopnost klasifikovat hlásky, které byly vysloveny správně a hláskami vyslovenými chybně je přibližně stejná, proto se úloha detekce chybné výslovnosti transformuje na úlohu rozpoznávání řeči. Pouze s tím rozdílem, že bude klasifikace rozšířena o vady výslovnosti.



Obr. 6.2: Porovnání koeficientů parametrizace 23 až 44 pro správně a chybně vyslovené hlásky.



Obr. 6.3: Porovnání koeficientů parametrizace 45 až 66 pro správně a chybně vyslovené hlásky.



Obr. 6.4: Porovnání koeficientů parametrizace 67 až 86 pro správně a chybně vyslovené hlásky.

6.2 Volba váhové funkce DTW algoritmu

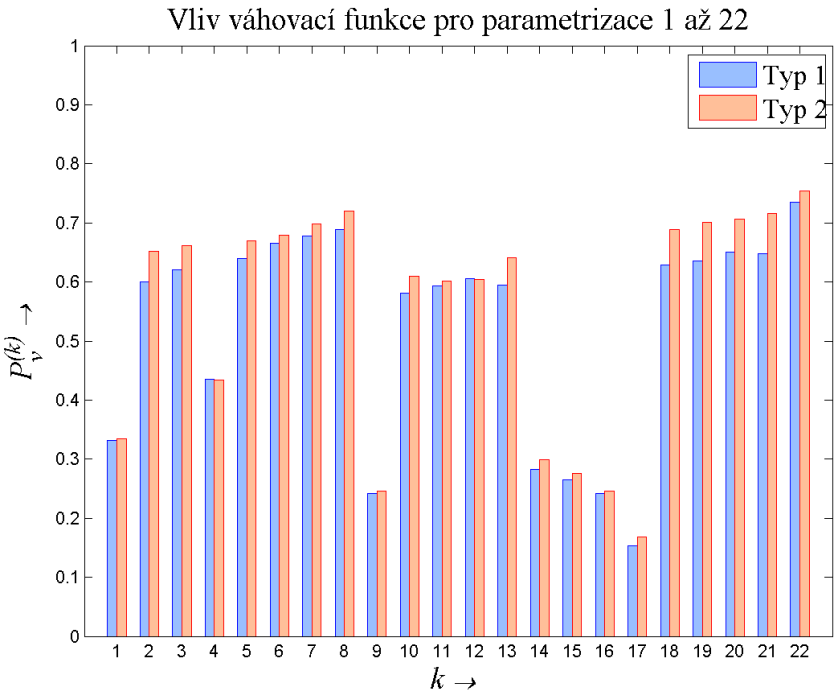
Dále je nutné zvolit váhovou funkci DTW algoritmu.

Obr. 6.5: Nastavení váhové funkce algoritmu DTW a) typ 1 b) typ 2

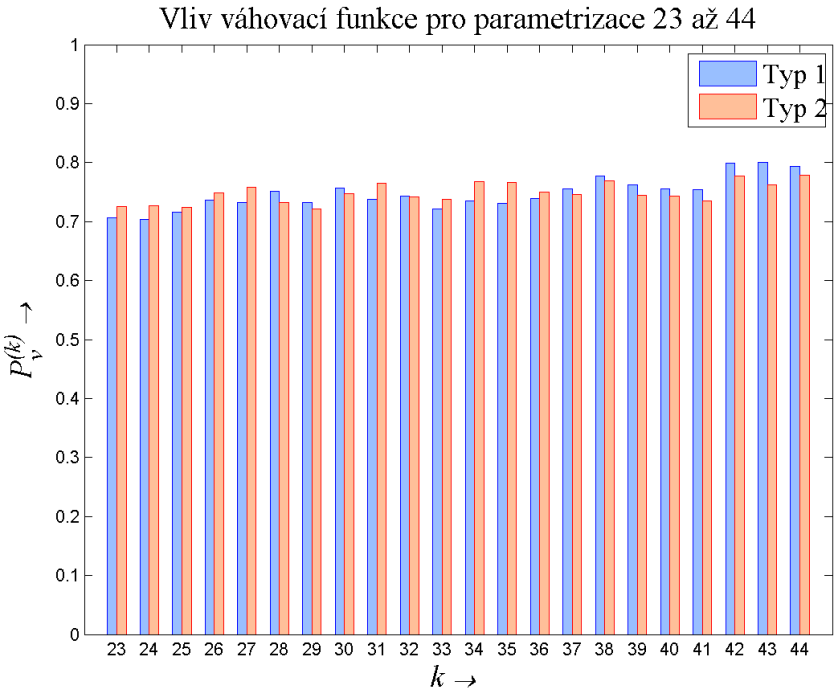
6.2.1 Vliv váhovací funkce na detekci hlásek

Z obr.6.6 až 6.9 je vidět, že použití různých váhových funkcí ovlivňuje výsledné koeficienty parametrizací. Z celkového pohledu není jasná vítězná váhová funkce, ale při zaměření se na vybrané parametrizace se zdá být použitelnější váhová funkce 1.typu.

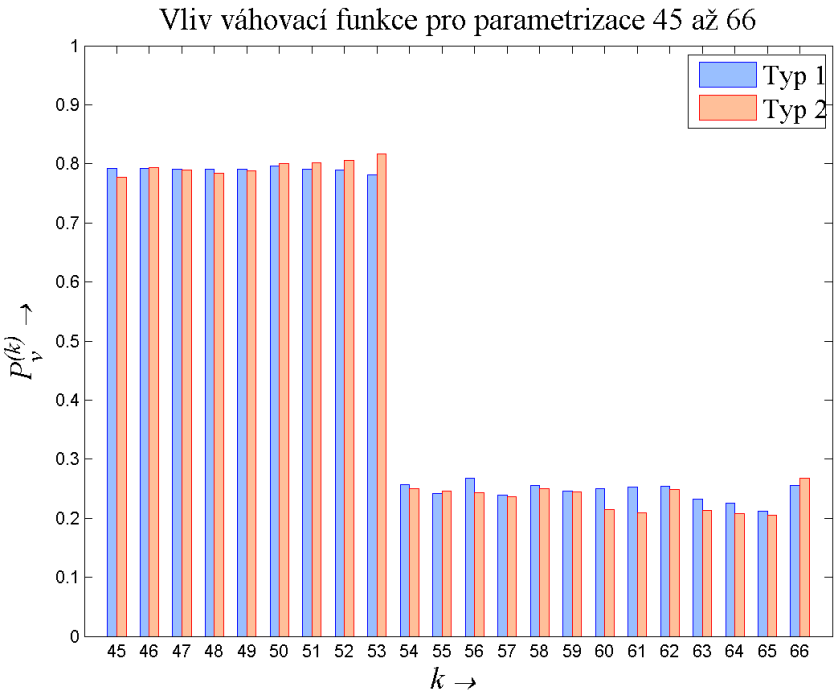
V další části práce proto nebude váhová funkce typu 2 použita.



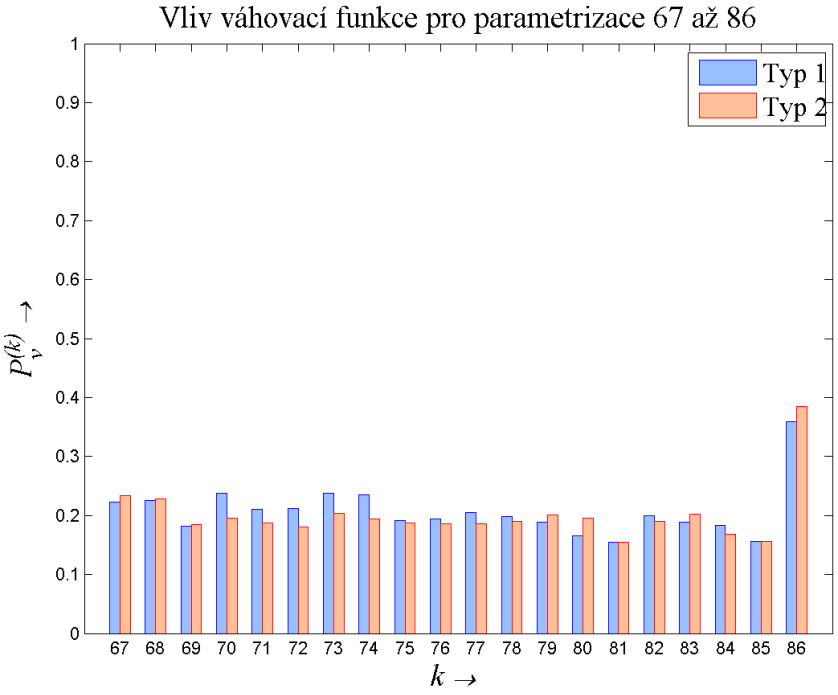
Obr. 6.6: Vliv váhovací funkce DTW na správnou detekci hlásek pro parametrizace 1 až 22



Obr. 6.7: Vliv váhovací funkce DTW na správnou detekci hlásek pro parametrizace 23 až 44



Obr. 6.8: Vliv váhovací funkce DTW na správnou detekci hlásek pro parametrizace 45 až 66



Obr. 6.9: Vliv váhovací funkce DTW na správnou detekci hlásek pro parametrizace 67 až 86

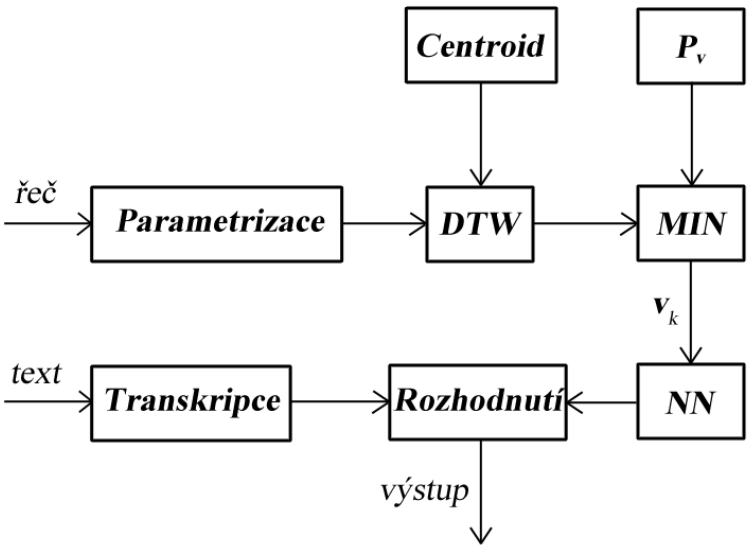
7 DETEKCE CHYBNÉ VÝSLOVNOSTI

V odstavci 6.1.1 bylo ukázáno, že úlohu detekce chybné výslovnosti lze převést na úlohu rozpoznávání řeči za předpokladu, že bude klasifikace rozšířena o vady výslovnosti.

Tato kapitola se zabývá návrhem hybridního rozpoznávače na bázi DTW a umělé neuronové sítě.

7.1 Konstrukce detektoru chybné výslovnosti

Na obr. 7.1 je znázorněno blokové schéma detektoru chybné výslovnosti.



Obr. 7.1: Blokové schéma detektoru

Popis jednotlivých bloků detektoru:

- **Parametrizace** - na základě předchozí kapitoly byly vybrány následující parametrizace: $mfcc_{56}$, $plp_{16,28}$, $plprasta_{12,84}$ a ZCR.

  Blok nejprve segmentuje řečový signál na úseky o délce 1024 vzorků (to odpovídá časovému úseku $21,\bar{3}$ ms při $f_{vz} = 48$ kHz) s posunem o 25% a dále pomocí těchto analýz parametrizuje segmentovaný řečový signál.

- **Centroid** - blok obsahuje příznaky pěti centroidů každé třídy hlásek, podrobnosti lze nalézt v odstavci 7.2.1.

- **DTW** - zde se příznaky segmentovaného řečového signálu porovnávají pomocí algoritmu DTW se všemi centroidy. Výsledkem je matice o rozměrech 4x34x5 (4 parametrizace, 34 tříd hlásek, 5 centroidů).

- **$\mathbf{P_v}$ + MIN** - nejprve je určen vektor $\boldsymbol{v}$ s hodnotami mír odlišnosti od nejpodobnějších centroidů. Jednotlivé prvky tohoto vektoru tvoří minima míry odlišnosti zkoumané promluvy a pěti centroidů jednotlivých tříd.

$$v[k] = \min_{\boldsymbol{C}^{(k)}} \left(D(\boldsymbol{R}, \boldsymbol{C}_i^{(k)}) \right), \tag{7.1}$$

kde $\boldsymbol{R}$ je obraz zkoumané promluvy, $\boldsymbol{C}_i^{(k)}$ je i-tý centroid k-té třídy hlásek, přičemž $i = 1, 2, \ldots, 5$ a $D(.)$ je míra odlišnosti získaná pomocí algoritmu DTW.

Vznikne matice o rozměrech 4x34. Aby bylo možné zvýhodnit ty parametrizace, u kterých byla zjištěna vyšší schopnost klasifikace, proběhne váhování získaných hodnot za pomoci koeficietů parametrizací podle vztahu

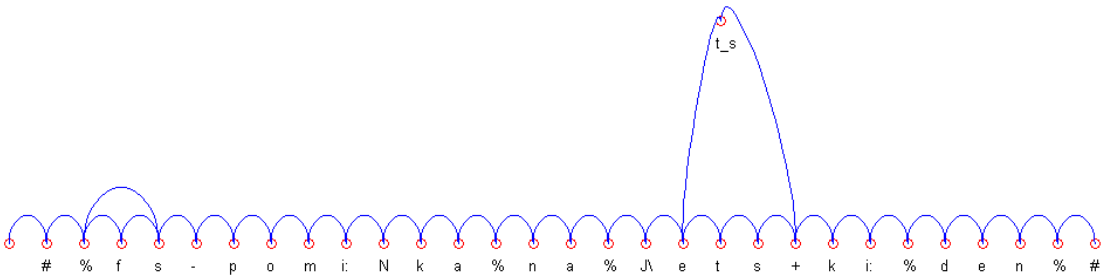
$$v_{NN}[k] = \frac{\sum_{k=1}^{K} P_v[k] v[k]}{\sum_{j=1}^{K} P_v[j]}, \tag{7.2}$$

kde P_v jsou koeficienty parametrizací, $\boldsymbol{v}$ je vektor s hodnotami mír odlišnosti od nejpodobějších centroidů, a $\boldsymbol{v}_{NN}$ je vektor o rozměrech 1x34, který je vstupem do umělé neuronové sítě a K je počet parametrizací.

- **NN** - blok s umělou neuronovou sítí, jejíž výstupem je vektor o rozměrech 1x34. Jedná se o dopřednou síť se strukturou 34-136-34. O sestavování trénovacích množin a použitém učebním algoritmu se lze více dozvědět v odstavci 7.2.2.

  Neurony ve výstupní vrstvě korespondují s jednotlivými hláskami: 1. `A`, 2. `B`, 3. `C`, 4. `Ch`, 5. `D`, 6. `E`, 7. `F`, 8. `G`, 9. `H`, 10. `I`, 11. `J`, 12. `K`, 13. `L`, 14. `M`, 15. `N`, 16. `O`, 17. `P`, 18. `R`, 19. `S`, 20. `T`, 21. `U`, 22. `V`, 23. `Z`, 24. `vR`, 25. `vS`, 26. `vŠ`, 27. `vŘ`, 28. `Š`, 29. `Ť`, 30. `Ž`, 31. `Č`, 32. `Ď`, 33. `Ň`, 34. `Ř`.

- **Transkripce** - blok provede automatickou transkripci českého jazyka. Na obrázku 7.2 je ukázka funkce tohoto bloku na výrazu `"vzpomínka na dětský den"` (použitá fonetická abeceda: SAMPA).



Obr. 7.2: Fonetický graf výrazu `"vzpomínka na dětský den"`.

Algoritmus automatické transkripce nevyužívá fonetický slovník výjimek, dále se očekává, že předkládaný výraz pro transkripci je rozšířen o předěly[1].

- **Rozhodnutí** - na základě fonetického přepisu očekávaného výrazu a výstupu umělé neuronové sítě blok vyhodnotí pomocí Levenshteinovy vzdálenosti[2], zda byla vyřčena správná promluva.

  Za nerozpoznanou se považuje taková hláska, pro níž je hodnota výstupního neuronu nižší než -0.75.

7.2 Výběr centroidů, tvorba trénovacích množin

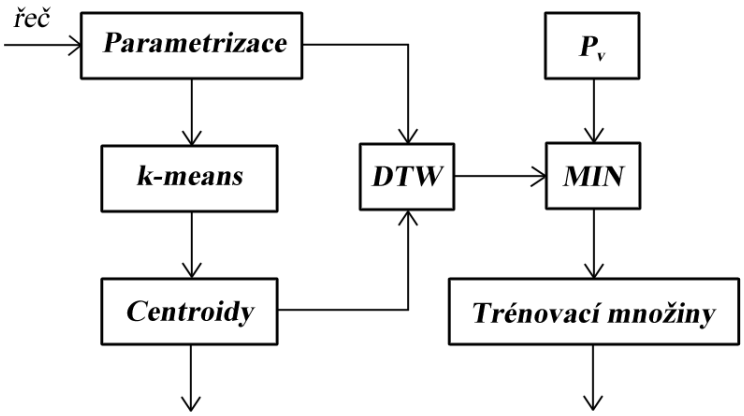
Následující blokové schéma reprezentuje proces výběru centroidů jednotlivých tříd a s jejich pomocí i vytváření trénovacích množin pro umělou neuronovou síť.

7.2.1 Výběr centroidů

V kapitole 6 bylo vybráno více jak 1200 úseků řečových signálu. Tyto úseky charakterizovali hlásky, které byly vysloveny správně anebo chybně. Tyto úseky je vhodné považovat za referenční vzory pro rozpoznávání, protože porovnávání těchto vzorů a promluvy pomocí DTW by bylo výpočetně náročné (musí se vzít v úvahu ještě počet parametrizací), je snaha zredukovat skupinu těchto vzorů na nezbytné minimum.

[1] Existuje několik druhů předělů - vnější předěl(hranice mezi slovy), vnitřní předěl(odděluje kořen slova), terminální předěl(pauza).

[2] Jedná se o algoritmus, který najde minimální počet operací potřebných k transformaci jednoho textového řetězce do druhého, přičemž operací je myšleno vložení, odstranění nebo nahrazení jednoho jediného znaku. Z praktického hlediska se jedná o modifikaci DTW algoritmu. Pomocí zpětné rekonstrukce optimální cesty lze určit, nad kterými znaky byla jaká operace provedena.



Obr. 7.3: Blokové schéma procesu tvorby trénovacích množin a výběru centroidů.

Proto bude výhodné vybrat z každé třídy hlásek pět takových vzorů, které budou co nejlépe vystihovat danou třídu. K tomu lze využít modifikovaný k-means algoritmus, popsaný v kapitole 6. Získané centroidy jsou využity nejen v rozpoznávači, ale také v procesu vytváření trénovacích množin.

7.2.2 Umělá neuronová síť a její učební množiny

K využití umělé neuronové sítě v rozpoznávači vedl předpoklad, že tato síť by měla být schopna (na základě předzpracovaných dat pomocí DTW) rozpoznávat fonémy i za situace, kdy promluva byla vyřčena jiným řečníkem, než těmi, kteří poskytli referenční vzory.

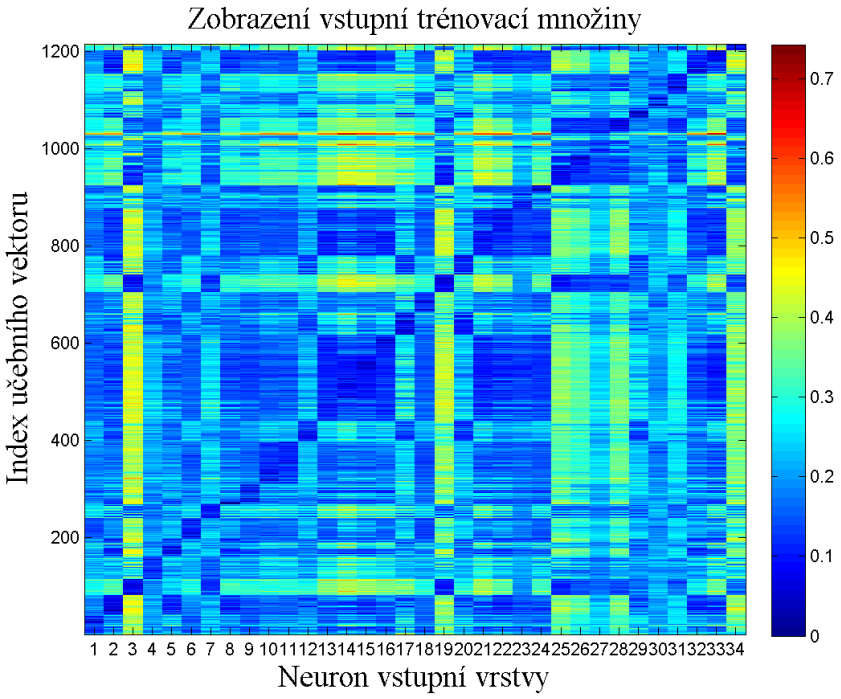
Sestavení učebních množin je velmi podobné získání vektoru $\boldsymbol{v}_k$ pomocí bloků **centroid, DTW, $\mathbf{P_v}$, MIN** ze schématu 7.1 pouze s rozdílem, že vstup do bloku **DTW** nejsou příznaky segmentované promluvy, ale příznaky všech referenčních vzorů. U každého referenčního vzoru je známa hláska, kterou daný vzor vyjadřuje, proto lze ke každému takovémuto vstupnímu učebnímu vektoru $\boldsymbol{v}_k$ přiřadit vektor výstupní. Prvky tohoto výstupní vektoru nabývají pouze hodnot -1 a 1.

Po různých experimentech se jako nejvhodnější zdá být síť typu backpropagation. Učícím algoritmem, který konvergoval nejrychleji, se ukázala být **metoda sdružených gradientů** (podrobnosti o této metodě lze nalézt v [12]). Aktivační funkce a počty neuronů jednotlivých vrstev byly zvoleny následovně:

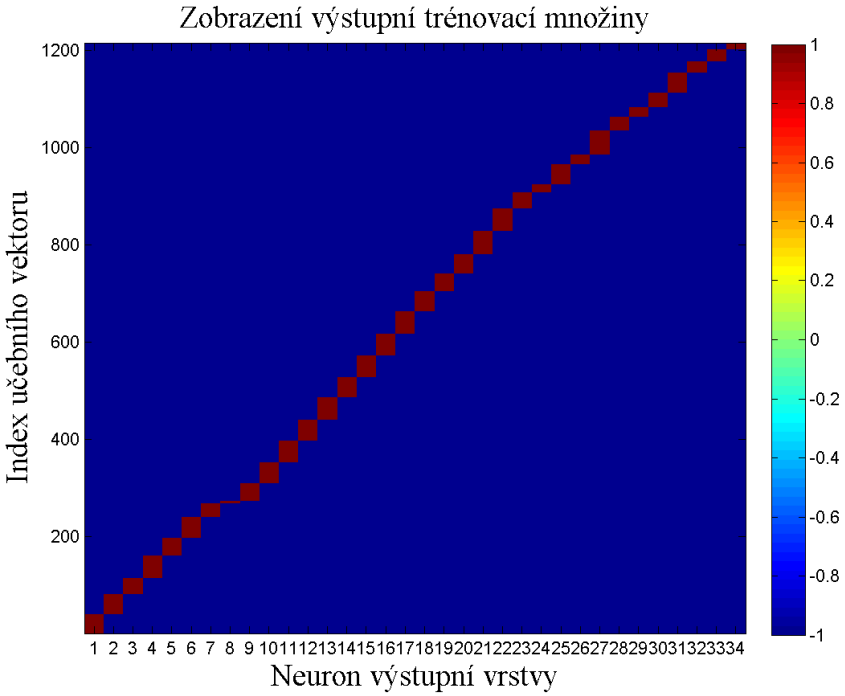
- 1. vrstva (vstupní)
  - aktivační funkce: tansigmoidální,
  - počet neuronů: 34,

- 2. vrstva (skrytá)
  - aktivační funkce: tansigmoidální,
  - počet neuronů: 136,
- 3. vrstva (výstupní)
  - aktivační funkce: lineární,
  - počet neuronů: 34.

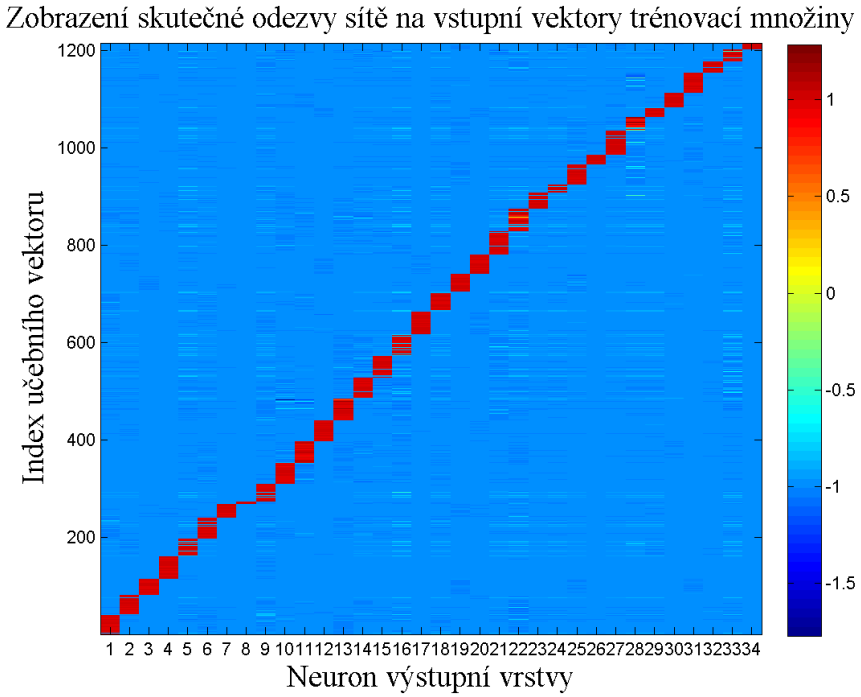
Pro trénování bylo náhodně vybráno 80% vektorů z trénovací množiny, zbytek vektorů byl použit pro testování neuronové sítě.



Obr. 7.4: Vstupní vektory trénovací množiny.

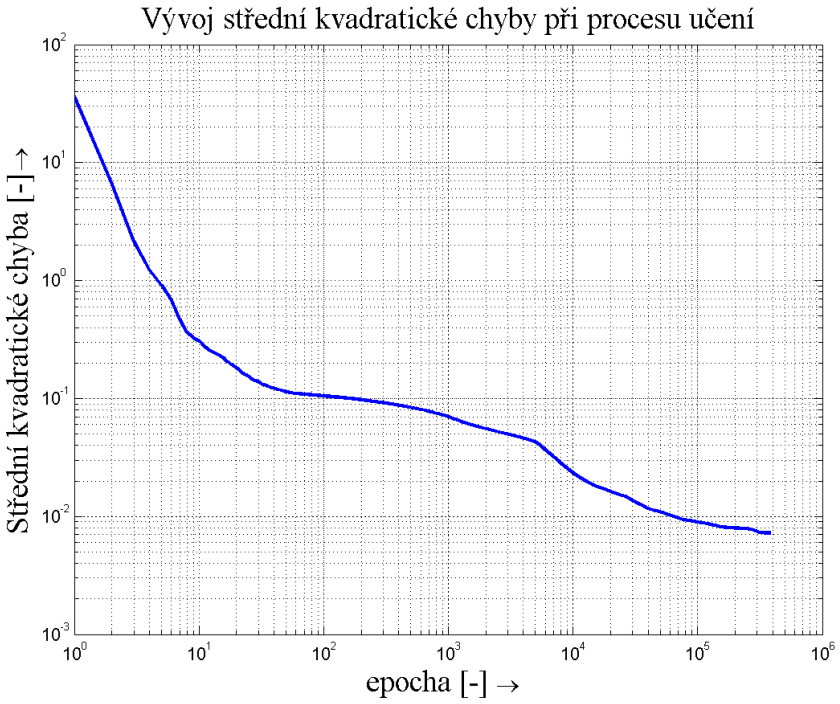


Obr. 7.5: Výstupní vektory trénovací množiny.



Obr. 7.6: Skutečná odezva neuronové sítě na vstupní vektory trénovací množiny.

Vývoj procesu učení je zachycen v grafu 7.7.



Obr. 7.7: Vývoj střední kvadratické chyby při procesu učení.

7.3 Účinnost rozpoznávače

Účinnost rozpoznávače lze určit porovnáním promluvy, kterou rozpoznávač rozpozná a promluvou, která byla skutečně vyřčena mluvčím. Porovnání je realizováno pomocí Levenshteinovy vzdálenosti se zpětnou rekonstrukcí optimální cesty.

Bylo prozkoumáno několik promluv celých slov, jejich přehled a vyhodnocení účinnosti je zobrazeno v tabulkách 7.1 až 7.10. Položky tabulek *vložené*, *vypuštěné*, *nahrazené* a *správné* udávají počet vložených, vypuštěných, nahrazených a správně klasifikovaných hlásek. Účinnost je poměrem správných hlásek ku součtu všech hlásek.

Pro hlásky, které jsou v položce *Skutečná promluva* tabulek 7.1 až 7.20 podtrženy, poskytly jednotlivý mluvčí referenční vzory pro sestavení inventáře.

Neidentifikovaná hláska je nahrazena znakem -.

Z tabulek 7.1 až 7.10 je vidět, byl splněn předpoklad z části 7.2.2, tedy že rozpoznávač není závislý na mluvčím.

Tab. 7.1: Účinnost rozpoznávače při vyřčení promluvy `banány`; mluvčí: Hrnčířová.

Skutečná promluva				
B A N A N I				
Rozpoznaná promluva				
B - Z A E -				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
2	4	0	0	33,33

Tab. 7.2: Účinnost rozpoznávače při vyřčení promluvy `kladivo`; mluvčí: Cydrichová.

Skutečná promluva				
T L A D I V O				
Rozpoznaná promluva				
T - - - I - O				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
3	4	0	0	42,86

Tab. 7.3: Účinnost rozpoznávače při vyřčení promluvy `čepice`; mluvčí: Ejeminghaze.

Skutečná promluva				
C E P C E				
Rozpoznaná promluva				
- E B - -				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
1	4	0	0	20,00

Tab. 7.4: Účinnost rozpoznávače při vyřčení promluvy `tak`; mluvčí: Nevolník.

Skutečná promluva				
T A K				
Rozpoznaná promluva				
T A -				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
2	1	0	0	66,66

Tab. 7.5: Účinnost rozpoznávače při vyřčení promluvy `řeka`; mluvčí: Hrnčířová.

Skutečná promluva				
Z E K A				
Rozpoznaná promluva				
- E B -				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
1	3	0	0	25,00

Tab. 7.6: Účinnost rozpoznávače při vyřčení promluvy `holenku`; mluvčí: Dolníčková.

Skutečná promluva				
H O L E N K U				
Rozpoznaná promluva				
- O - A M K U				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
2	5	0	0	28,57

Tab. 7.7: Účinnost rozpoznávače při vyřčení promluvy `kočičku`; mluvčí: Fadrhonc.

Skutečná promluva				
K O Č I Č K U				
Rozpoznaná promluva				
- - Č - Č K U				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
4	3	0	0	57,14

Tab. 7.8: Účinnost rozpoznávače při vyřčení promluvy `zuby`; mluvčí: Fadrhonc.

Skutečná promluva				
Z U B I				
Rozpoznaná promluva				
vS Ň B J				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
1	3	0	0	25,00

Tab. 7.9: Účinnost rozpoznávače při vyřčení promluvy `smolíček`; mluvčí: Polcar.

Skutečná promluva				
S M O L Í Č E K				
Rozpoznaná promluva				
- M U - Z - E K				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
3	5	0	0	37,50

Tab. 7.10: Účinnost rozpoznávače při vyřčení promluvy `prudce`; mluvčí: Nevolník.

Skutečná promluva				
P vR U T C E				
Rozpoznaná promluva				
P O O - C -				
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]	Účinnost [%]
2	4	0	0	33,33

7.4 Vyhodnocení rozpoznávače z hlediska detekce chybné promluvy

Jak bylo popsáno v části 1.2, do sféry dyslálie spadají nejen vady výslovnosti, které jsou vrozené, ale i takové případy, kdy dojde k nahrazení, vynechání nebo vložení jiných fonémů, než je standardem v daném jazyce.

Výstupem detektoru je proto informace o počtu nahrazených, vynechaných nebo vložených hláskách. Tato informace bude relavantní za předpokladu, že účinnost rozpoznávače bude 100%. Z výsledků v oddílu 7.3 je patrné, že se této účinnosti nepodařilo dosáhnout.

V tabulkách 7.11 až 7.20 jsou uvedeny výstupy detektoru. Porovnávány byly rozpoznané a očekávané promluvy.

Tab. 7.11: Detekce chybné výslovnosti v promluvě `banány`; mluvčí: Hrnčířová.

Očekáváná promluva
B A N A N I
Rozpoznaná promluva
B - Z A E -
Skutečná promluva
B A N A N I

Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
2	4	0	0

Tab. 7.12: Detekce chybné výslovnosti v promluvě `kladivo`; mluvčí: Cydrichová.

Očekáváná promluva
K L A D I V O
Rozpoznaná promluva
T - - - I - O
Skutečná promluva
T L A D I V O

Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
2	5	0	0

Tab. 7.13: Detekce chybné výslovnosti v promluvě `čepice`; mluvčí: Ejeminghaze.

Očekáváná promluva			
Č E P I C E			
Rozpoznaná promluva			
- E B - -			
Skutečná promluva			
C E P C E			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
1	4	0	1

Tab. 7.14: Detekce chybné výslovnosti v promluvě `tak`; mluvčí: Nevolník.

Očekáváná promluva			
T A K			
Rozpoznaná promluva			
T A -			
Skutečná promluva			
T A K			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
2	1	0	0

Tab. 7.15: Detekce chybné výslovnosti v promluvě `řeka`; mluvčí: Hrnčířová.

Očekáváná promluva			
Ř E K A			
Rozpoznaná promluva			
- E B -			
Skutečná promluva			
Z E K A			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
1	3	0	0

Tab. 7.16: Detekce chybné výslovnosti v promluvě `holenku`; mluvčí: Dolníčková.

Očekáváná promluva			
H O L E N K U			
Rozpoznaná promluva			
- O - A M K U			
Skutečná promluva			
H O L E N K U			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
2	5	0	0

Tab. 7.17: Detekce chybné výslovnosti v promluvě `kočičku`; mluvčí: Fadrhonc.

Očekáváná promluva			
K O Č I Č K U			
Rozpoznaná promluva			
- - Č - Č K U			
Skutečná promluva			
K O Č I Č K U			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
4	3	0	0

Tab. 7.18: Detekce chybné výslovnosti v promluvě `zuby`; mluvčí: Fadrhonc

Očekáváná promluva			
Z U B I			
Rozpoznaná promluva			
vS Ñ B J			
Skutečná promluva			
Z U B I			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
1	3	0	0

Tab. 7.19: Detekce chybné výslovnosti v promluvě `smolíček`; mluvčí: Polcar.

Očekáváná promluva			
S M O L Í Č E K			
Rozpoznaná promluva			
- M U - Z - E K			
Skutečná promluva			
S M O L Í Č E K			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
3	5	0	0

Tab. 7.20: Detekce chybné výslovnosti v promluvě `prudce`; mluvčí: Nevolník.

Očekáváná promluva			
P R U T C E			
Rozpoznaná promluva			
P O O - C -			
Skutečná promluva			
P vR U T C E			
Správné [-]	Nahrazené [-]	Vložené [-]	Vypuštěné [-]
2	4	0	0

8 ZÁVĚR

V simulačním prostředí MATLAB byly vytvořeny skripty s parametrizacemi uvedenými v části 6.1.

Dále byl vytvořen inventář, jež byl základem pro výpočet koeficientů parametrizací, které vyjadřují schopnost klasifikace danou analýzou.

Algoritmus k-means byl modifikován pro potřeby nalezení reprezentantů všech tříd hlásek, a v kombinaci s algorimem DTW byly určeny průměrné míry odlišnosti mezi všemi obrazy v rámci jedné hlásky (správně i chybně vyslovenými). A dále byly určeny minimální míry odlišnosti mezi všemi reprezentanty hlásek. Z těchto údajů bylo možno vypočíst již zmíněné koeficienty parametrizací.

Byl určen vliv váhové funkce algoritmu DTW na tuto schopnost. Nejvhodnější se ukázala váhová funkce typu 1.

Na základě zjištěných koeficientů parametrizace byly vybrány následují parametrizace (jejich přesný popis lze najít v části 6.1): $mfcc_{56}$, $plp_{16,28}$, $plprasta_{12,84}$ a ZCR.

V odstavci 6.1.1 bylo ukázáno, že úlohu detekce chybné výslovnosti lze převést na úlohu rozpoznávání řeči za předpokladu, že bude klasifikace rozšířena o vady výslovnosti.

Byl navržen hybridní rozpoznávač na bázi DTW a umělé neuronové sítě. Pro natrénování této sítě byl využit sestavený inventář, koeficienty parametrizací a pět centroidů každé třídy hlásek. Tyto centroidy byly vypočteny pomocí modifikovaného algoritmu k-means.

Byly vytvořeny skripty pro automatickou transkripci českého jazyka, která je nezbytná pro správnou detekci chybné výslovnosti.

Detekce chybné výslovnosti lze považovat za správnou v případě 100% účinnosti rozpoznávače. Bohužel se této účinnost nepodařilo dosáhnout, a pohybuje se v rozmezí 20,00% až 57,14%.

Detektor chybné výslovnosti nedokáže správně klasifikovat hlásky ani detekovat chybnou výslovnost bez znalosti hranic fonémů, proto by další práce měla být zaměřena právě na odstranění tohoto nedostatku. V úvahu připadá nalezení těchto hranic pomocí skrytých markovových modelů.

Pro zvýšení účinnosti by bylo vhodné prozkoumat možnost vytvoření inventáře s jemnějším dělením než jsou fonémy.

Po vyřešení výše zmíněných nedostatků, bude vhodné prozkoumat navržený detektor z hlediska šumové odolnosti.

LITERATURA

[1] PSUTKA, J. - MÜLLER, L. - MATOUŠEK, J. - RADOVÁ, V. *Mluvíme s počítačem česky*. 1.vydání. Praha: Academia, 2006. 752 s. ISBN 80-200-1309-1.

[2] PSUTKA, J. *Komunikace s počítačem mluvenou řečí*. 1.vydání. Praha: Academia, 1995. 287 s. ISBN 80-200-0203-0.

[3] DELLER, J. R. - ANSEN, J. H. L. - PROAKIS, J. G. *Discrete-Time Processing of Speech Signals*. Reprint edition. Wiley;IEEE, 2000. 963 s. ISBN 0780353862.

[4] JAN, J. *Číslicová filtace, analýza a restaurace signálů*. 2. opravené a rozšířené vydání. Brno: VUTIUM, 2002. 427 s. ISBN 80-214-1558-4.

[5] *http://fu.ff.cuni.cz/vyuka/akustika/vokaly.pdf* [online]

[6] ZLATNÍK, P. - ČMEJLA, R. *Hodnocení vývoje léčby u dětí s poruchami řeči http://noel.feld.cvut.cz/sbornik07/data/001_slajdy.pdf* [online]

[7] ZLATNÍK, P. - ČMEJLA, R. *Vyhodnocování vad řeči dětí s využitím algoritmu DTW http://noel.feld.cvut.cz/sbornik06/data/001_slajdy.pdf* [online]

[8] ČERNOCKÝ, J. *http://www.fit.vutbr.cz/černocky/speech/opora/zre_opora.pdf* [online]

[9] LECHTA, V. a kolektiv *Diagnostika narušené komunikační schopnosti*. 1.vydání. Praha: Portál, 2003. 360 s. ISBN 80-7178-801-5.

[10] KUREČKA, R. a kolektiv *Metody pro záznam a zpracování nahrávek s chybnou výslovností*. Výzkumná zpráva k řešení projektu MPO ČR, ev. č. FT-TA2/072, 2005. 58 s.

[11] KRČMOVÁ, M. *Fonetika http://is.muni.cz/elportal/estud/ff/js07/fonetika/materialy/index.html*[online]

[12] FILÍPEK, O. *Metoda sdružených gradientů pro optimalizace jednotek GAME https://dip.felk.cvut.cz/browse/pdfcache/filipo1_2007dipl.pdf*[online]

[13] THE MATHWORKS INC. *MATLAB HTML Documentation*. [online]. Součást simulačního prostředí MATLAB verze 7.5, 2007

[14] SYSEL, P. *segmentation.m*. [skript prostředí MATLAB]. Verze 1. Brno: VUT v Brně, 2005

A PŘÍLOHA

Obsah DVD

- `DP.pdf` – Tento dokument.
- `readme.txt` – Soubor obsahující informace o požadavcích na použitý software a úkony nezbytné pro použití skriptů.
- `metadata.pdf` – Dokument obsahující metadata diplomové práce.

+ `\LaTeX` – Obsahuje zdrojové kódy tohoto dokumentu.
+ `\Video` – Obsahuje soubor `tladivo.avi` pro demonstraci detekce chybné výslovnosti.
+ `\Matlab` – Obsahuje zdrojové kódy funkcí a skriptů prostředí MATLAB.
  - `start.m` – Zapíše cesty k funkcím, skriptům a datům do prostředí MATLAB, je nezbytné spustit tento skript na začátku práce.
+ `\Matlab\Algoritmy`
  - `dtw.m` – Algoritmus DTW, porovnávají se obrazy(matice) promluv.
  - `accdtw.m` – Algoritmus DTW, jiná implementace oproti `dtw.m`. Vstupem je celý referenční obraz, ale pouze jeden vektor testovaného obrazu. Výhodné v případě potřeby „odebírat“ průběžné hodnoty algoritmu. Využito ve funkci `rozpoznej.m`.
  - `kmeans.m` – Modifikovaný k-means algoritmus.
  - `levenstein.m` – Výpočet levensteinovi vzdálenosti se zpětnou rekonstrukcí optimální cesty.
+ `\Matlab\Analyzy` – Obsahuje zdrojové kódy použitých parametrizací.
  - `acorr.m` – Krátkodobá autokorelační funkce.
  - `ceps.m` – Krátkodobá kepstrální analýza.
  - `delta.m` – Dynamické (delta) koeficienty.
  - `energy.m` – Krátkodobá energie.
  - `lpc.m` – Lineární prediktivní analýza.
  - `mfcc.m` – Melovské kepstrální koeficienty.
  - `plp.m` – Perceptivní lineární prediktivní analýza.

  - `rastaplp.m` – RASTA-PLP analýza.
  - `zcr.m` – Funkce středního počtu průchodu signálu nulou.

+ `\Matlab\Analyzy\Podpora` – Obsahuje podpůrné funkce pro běh funkcí s analýzami.
  - `gram.m` – Zobrazí vyhlazený spektrogram na základě LPC, PLP nebo RASTA-PLP koeficientů.
  - `hz2bark.m` – Převod normální kmitočtové škály do barkové.
  - `hz2mel.m` – Převod normální kmitočtové škály do melovské.
  - `lpc2ceps.m` – Převod LPC, PLP nebo RASTA-PLP koeficientů na kepstrální.
  - `mel2hz.m` – Převod melovské kmitočtové škály do normální.
  - `normwindow.m` – Normování okénkové funkce tak, aby její energie byla rovna 1.
  - `preem.m` – Provede preemfázi signálu.
  - `segmentation.m` – Provede segmentaci signálu.[14]

+ `\Matlab\Skripty`
  - `ziskej_parametrizace.m` – Vypočte příznaky celého inventáře.
  - `rozclen_ceps.m` – Rozčlení kepstrální koeficienty na několik skupin.
  - `ziskej_koeficienty_parametrizaci.m` – Vypočte koeficienty parametrizací.
  - `vyber_centroidy.m` – Z parametrizovaného inventáře vybere pět centroidů.
  - `ziskej_trenovaci_mnoziny.m` – Sestaví trénovací množinu pro neuronovou síť.
  - `uprav_trenovaci_mnoziny.m` – Upraví tuto trénovací množinu do vhodné podoby.
  - `trenuj_BP.m` – Vytvoření neuronové sítě a její trénování.
  - `rozpoznej.m` – Umožní klasifikaci hlásek a detekci chybné výslovnosti.

+ `\Matlab\Skripty\Grafy`
  - `graf_pk.m` – Zobrazí koeficienty parametrizací (viz obr. 6.1 až 6.4).

  - `graf_pk_vahyDTW.m` – Zobrazí vliv váhové funkce DTW (viz obr. 6.6 až 6.9).
  - `zobraz_ucebni_mnoziny.m` – Zobrazí vstupní a výstupní vektory učební množiny (viz obr. 7.4 a 7.5).
  - `graf_chyby.m` – Zobrazí vývoj průměrné střední kvadratické chyby (MSE) při procesu učení (viz obr. 7.7).

+ `\Matlab\Skripty\Podpora` – Obsahuje podpůrné funkce pro běh hlavních skriptů.
  - `nacti_seznam.m` – Sestavuje potřebné seznamy inventáře, příznakových matic, koeficientů parametrizací, centroidů a dat pro učební množinu.
  - `getmatrix.m` – Vytvoří obraz (příznakovou matici) ze souboru `.wav`.
  - `param_coef.m` – Získá koeficient parametrizace jedné analýzy.
  - `param_graf.m` – Zobrazení grafu koeficientů parametrizací.
  - `tren.m` – Jádro sestavování trénovací množiny.

+ `\Matlab\Skripty\Ukazky`
  - `ukazka_kmeans.m` – Ukázka funkčnosti modifikovaného k-means algoritmu.
  - `ukazka_transkripce.m` – Ukázka transkripce českého jazyka.

+ `\Matlab\Transkripce` – Obsahuje skripty a funkce pro transkripci českého jazyka.
  - `readsubs.m` – Načte ze souboru `subs.rul` pravidla pro substituce výrazů při transkripci.
  - `readrules.m` – Načte uložená fonetická pravidla ze souborů `.rul`.
  - `applyrules.m` – Provede aplikaci načtených fonetických pravidel.
  - `inverze.m` – Převod z fonetické abecedy SAMPA na hlásky.
  - `transcribe.m` – Transkripce českého jazyka, použitá fonetická abeceda SAMPA.
  - `phoneticgraph.m` – Sestaví fonetický graf.
  - `plotphoneticgraph.m` – Zobrazí fonetický graf.
  - `*.rul` – Soubory obsahují pravidla pro automatickou fonetickou transkripci českého jazyka.

+ \Matlab\Data
    + NN – Obsahuje data neuronové sítě.
    + P_DTW111 – Obsahuje koeficienty parametrizací, použitá váhová funkce DTW byla typu 1.
    + P_DTW211 – Obsahuje koeficienty parametrizací, použitá váhová funkce DTW byla typu 2.
    + Priznaky – Obsahuje příznakové matice inventáře.
    + PriznakyACentroidy – Obsahuje příznakové matice inventáře a vybrané centroidy.
    + UcebniMnoziny – Obsahuje sestavenou trénovací množinu pro neuronovou síť.
+ \Matlab\Inventar – Obsahuje .wav soubory inventáře.
+ \Matlab\Wav – Obsahuje .wav soubory s testovacími promluvami (viz tab. 7.1 až 7.20).