

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra kybernetiky

DIPLOMOVÁ PRÁCE

PLZEŇ, 2014

Bc. MARTIN MATURA

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra kybernetiky

Fonetická segmentace řeči a možnosti
její automatické korekce

Plzeň, 2014

Bc. Martin Matura

Podklad pro zadání DIPLOMOVÉ práce studenta

PŘEDKLÁDÁ:	ADRESA	OSOBNÍ ČÍSLO
Bc. MATURA Martin	Nábřeží 1. máje 1848, Písek - Budějovické Předměstí	A10N0144P

TÉMA ČESKY:

Fonetická segmentace řeči a možnosti její automatické korekce

NÁZEV ANGLICKY:

Phonetic Segmentation of Speech and Possibilities of its Automatic Correction

VEDOUČÍ PRÁCE:

Doc. Ing. Jindřich Matoušek, Ph.D. - KKY

ZÁSADY PRO VYPRACOVÁNÍ:

1. Seznamte se s algoritmy automatické fonetické segmentace řeči a s možnostmi automatické korekce fonetické segmentace v kontextu syntézy řeči z textu.
2. Algoritmus automatické korekce fonetické segmentace vyvinutý v rámci bakalářské práce zobecněte na všechny typy fonetických hranic. Proveďte experimenty s různými typy hranic.
3. Nadefinujte úlohu korekce fonetické segmentace jako klasifikační úlohu. Vyberte vhodné příznaky a porovnejte různé typy klasifikátorů.
4. Nadefinujte metodiku značení velkých segmentačních chyb a odhalte, jak jsou navržené algoritmy úspěšné v odhalování velkých chyb segmentace.
5. Všechny navržené algoritmy realizujte formou počítačového programu a získané výsledky vyhodnoťte.

SEZNAM DOPORUČENÉ LITERATURY:

- PSUTKA, J., MÜLLER, L., MATOUŠEK, J., RADOVÁ, V.: Mluvíme s počítačem česky. Academia, Praha, 2006.
TAYLOR, P. Text-to-Speech Synthesis. Cambridge: Cambridge University Press, 2009.
LIN, C-Y., JANG, J.-S.: Automatic Phonetic Segmentation by Score Predictive Model for the Corpora of Mandarin Singing Voices. IEEE Transactions on Audio, Speech, and Language Processing, 15, 7, 2007, s. 2151-2159.

Podpis studenta:

Datum:

Podpis vedoucího práce:

Datum:

PROHLÁŠENÍ

Předkládám tímto k posouzení a obhajobě diplomovou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne

.....

Poděkování

Děkuji doc. Ing. Jindřichu Matouškovi, Ph.D. za trpělivé vedení mé diplomové práce a za důležité informace a podnětné návrhy, které ji obohatily.

Přístup k výpočetním a úložným zařízením ve vlastnictví stran a projektů, které přispívají k národní síti infrastruktury MetaCentra, poskytnutých v rámci programu „Projects of Large Infrastructure for Research, Development, and Innovations“ (LM2010005), je velmi ceněn.

Bc. Martin Matura

Název práce v ČJ

Fonetická segmentace řeči a možnosti její automatické korekce

Abstrakt v ČJ

Tato práce navazuje na mou bakalářskou práci s názvem Automatická korekce fonetické segmentace řeči. Je zde podrobněji rozebrán regresní přístup, kterého využívám při opravě hranic automaticky segmentovaných řečových jednotek a bylo provedeno množství experimentů, které měly za cíl zvýšit úspěšnost a přesnost korekce. Dále se práce zabývá srovnáním regresního a klasifikačního přístupu pro zadanou úlohu korekce a navržením postupu, jak odhalit velké chyby segmentace.

Klíčová slova v ČJ

automatická fonetická segmentace, korekce segmentace řečových jednotek, regresní model, klasifikační metody, hranice řečových jednotek

Název práce v AJ

Phonetic Segmentation of Speech and Possibilities of its Automatic Correction

Abstrakt v AJ

This thesis is continuation of my bachelor's thesis named Automatic Correction of Phonetic Segmentation of Speech. It in detail describes regression approach which I used to correct boundaries of automatically segmented phonemes. There is going to be a lot of experiments in order to increase the chance for successful and precise correction. Furthermore a comparison between regression and classification approach has been done and a procedure how to detect large errors of automatic phonetic segmentation was suggested.

Klíčová slova v AJ

Automatic phonetic segmentation, correction of segmentation of speech units, regression model, classification methods, boundaries of speech units

Obsah

1	Úvod	7
2	Fonetická segmentace řeči	8
2.1	Algoritmy pro fonetickou segmentaci	8
2.2	Hranice řečových jednotek	10
2.3	Konkatenační metoda	11
2.4	Korekce fonetické segmentace	12
2.4.1	Problematika regresní úlohy	14
2.4.2	Problematika klasifikační úlohy	16
3	Experimenty v oblasti automatické korekce	18
3.1	Korekce jako regresní úloha	18
3.1.1	Zavedení dynamiky	20
3.1.2	Omezení maximální odchylky	23
3.1.3	Zobecnění na všechny typy hranic	25
3.1.4	Dvoufázové vyhledávání parametrů pro regresní model	29
3.1.5	Cross-validace	31
3.1.6	Oprava hranic mezi konkrétními hláskami	34
3.1.7	Další experimenty na konkrétních hranicích	36
3.1.8	Finální korekce se zavedením všech experimentů	47
3.2	Korekce jako klasifikační úloha	49
3.2.1	Klasifikace pomocí SVM	51
3.2.2	Algoritmus k-nejbližších sousedů	53
3.2.3	Extremely Randomized Trees	55
3.2.4	Srovnání s regresním přístupem	58
4	Velké chyby segmentace	60
4.1	Analýza průměrných a největších chyb	61
4.1.1	Průměrné chyby	62
4.1.2	Největší chyby	64
4.2	Závislost velkých chyb a skóre	66
4.2.1	Vlastnosti skóre u velkých a malých chyb	66
4.2.2	Hledání prahové hodnoty statistických veličin	71
4.3	Detekce velkých chyb pro jeden typ hranice	74
5	Závěr	79
	Citovaná literatura	81
	Přílohy	82
A	Tabulky	83
B	Obrázky	87
C	Obsah příloženého CD	91

1 Úvod

Tato práce navazuje na mou bakalářskou práci, ve které jsem se zabýval návrhem programu pro automatickou korekci fonetické segmentace řeči. Fonetická segmentace řeči je proces, při kterém dochází v souvislém řečovém signálu k určení hranic mezi jednotlivými řečovými jednotkami – nejčastěji hláskami (fonémy). Řečové jednotky jsou pak ukládány do databáze a využívají se při zpětném syntetizování řeči. V dnešní době je nejpoužívanějším přístupem pro syntézu řeči tzv. konkatenací syntéza, která pro tvorbu řeči využívá řetězení (konkatenaci) jednotlivých řečových jednotek. Z toho důvodu je tedy velmi důležitá kvalita řečových jednotek, protože od ní se přímo odvíjí kvalita syntetizované promluvy. Je obecnou snahou, aby výsledná promluva byla co nejkvalitnější a nejpřirozenější.

Fonetická segmentace je časově velmi náročný proces, pokud ho dělá člověk-expert a při množství dat, které je potřeba segmentovat není možné, aby tato činnost byla prováděna ručně i přesto, že je tak dosaženo nejpřesnější segmentace. Byly proto vyvinuty algoritmy, které automaticky naleznou hranice řečových jednotek, avšak tyto algoritmy nepracují se stoprocentní přesností a někdy se stane, že nalezená hranice mezi řečovými jednotkami neodpovídá skutečné hranici. Proto jsou vyvíjeny korekční programy, které výslednou segmentaci ještě opravují. Takový program jsem vytvářel v rámci mé bakalářské práce a testoval jsem ho na dvou modelových typech hranic.

Má diplomová práce má za cíl korekční program rozvinout tak, aby opravoval všechny typy hranic a především pak vyvinout metody, které dopomohou ke kvalitnější a přesnější korekci. Bude také provedeno srovnání regresního přístupu, který používám, s přístupem klasifikačním, kde bude předvedena úspěšnost tří různých typů klasifikátorů. Nakonec se pak zaměříme na velké chyby segmentace, které jsou velkým zdrojem problémů z několika důvodů, které budou vysvětleny.

2 Fonetická segmentace řeči

V úvodu bylo naznačeno, že správná fonetická segmentace je nezbytným předpokladem pro kvalitní řeč vytvořenou pomocí konkatenční metody (viz část 2.3). Jejím úkolem je v souvislém řečovém signálu nalézt časové okamžiky, kdy dochází ke změně jedné řečové jednotky v druhou řečovou jednotku a tyto okamžiky zaznamenat. Řečovou jednotkou přitom může být různá realizace hlásek, z nichž se v současných systémech používají difony, trifony, slabiky, subfonémové jednotky a nejčastěji pak samotné hlásky (fóny).

Bohužel díky koartikulaci, tedy jevu, kdy se během výslovnosti vzájemně ovlivňují artikulační pohyby, bývá nalezení přesné hranice někdy velmi obtížné, protože řečové jednotky se částečně vzájemně prolínají. V takových situacích má pak bezpochyby člověk-expert výhodu oproti algoritmům pro automatickou fonetickou segmentaci. Jak už ale bylo řečeno, ruční segmentace je pracná a časově neúnosně náročná činnost, proto se využívá především automatické segmentace.

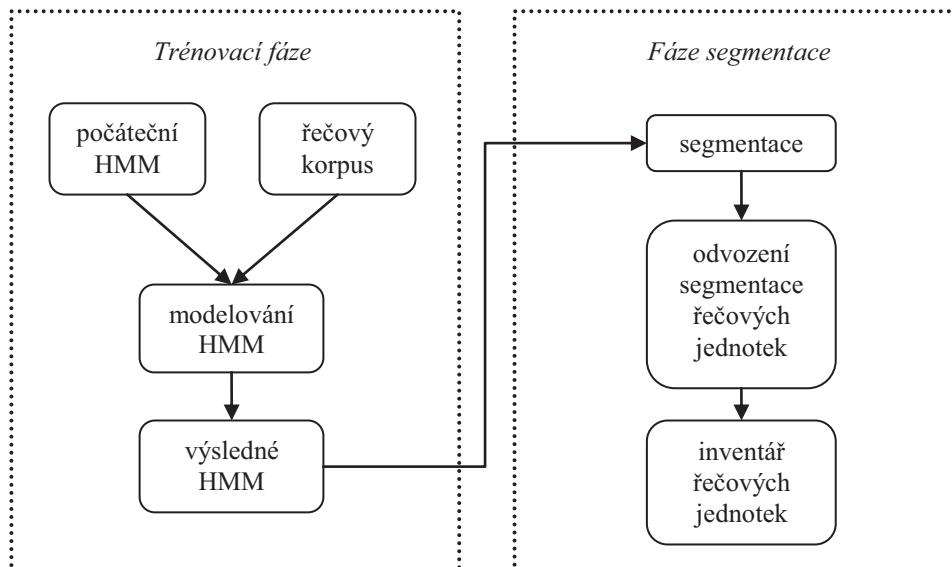
2.1 Algoritmy pro fonetickou segmentaci

Pro potřeby automatické segmentace se nejčastěji používají dva přístupy. Prvním přístupem je segmentace s využitím skrytých Markovových modelů (HMM z angl. *hidden Markov Model*), druhým přístupem je pak porovnávání se syntetickou řečí s použitím techniky dynamického borcení času (DTW z angl. *dynamic time warping*).

Každý má své klady i zápory a nelze obecně říci, že jeden z nich je výrazně lepší než ten druhý. Segmentace pomocí HMM má sice větší průměrnou chybu, ale je stabilnější, a proto bývá používána častěji, jelikož při použití techniky DTW občas dochází k velkým segmentačním chybám, které se považují za více závažné. Nyní stručně popíšeme, jak fungují oba algoritmy, jejichž podrobnější popis lze najít v [1], a poté se podíváme na problematiku spojenou s určováním hranic mezi řečovými jednotkami.

HMM

Nejvíce rozšířené je využívání tzv. mnohočetných HMM (angl. *multiple HMM*), které se používají pro modelování řečových jednotek. Segmentace pomocí tohoto algoritmu probíhá ve dvou fázích – *trénovací fáze* a *fáze segmentace*.



Obrázek 2.1: Automatická segmentace pomocí mnohočetných HMM.

Pro trénovací fázi je potřeba mít velké množství řečových promluv, které slouží jako trénovací data. Nejprve se inicializují HMM modely pomocí nějaké dodatečné informace, kterou máme o promluvách v řečovém korpusu, a poté se provádí odhad parametrů jednotlivých HMM. Výsledkem jsou modely řečových jednotek s odhadnutými parametry. V segmentační fázi se pak rozeberou trénovací promluvy a jednotlivé HMM se přiřadí daným úsekům, kterým odpovídají. Takto se určí úseky, které jsou modelované jednotlivými HMM, a dojde k nastavení hranic mezi nimi.

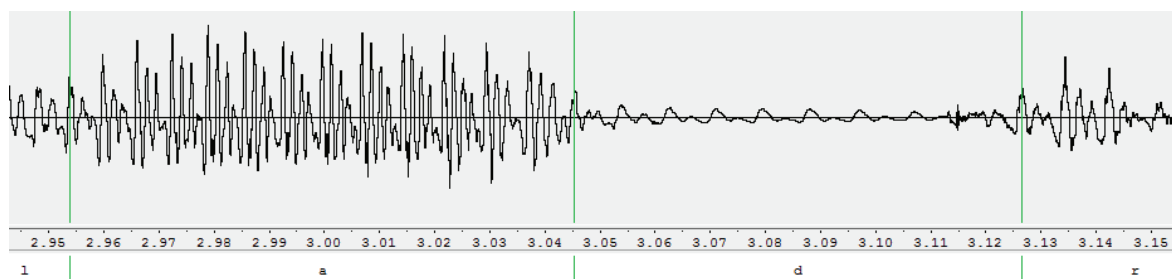
DTW

V této metodě využíváme syntetizovaný signál, ve kterém známe hranice řečových jednotek, a s tímto signálem porovnáváme přirozený řečový signál, ve kterém chceme hranice určit. Aby bylo možné signály porovnat, jsou oba popsány sadou parametrů. K porovnávání se pak využívá právě algoritmus dynamického borcení časové osy [2].

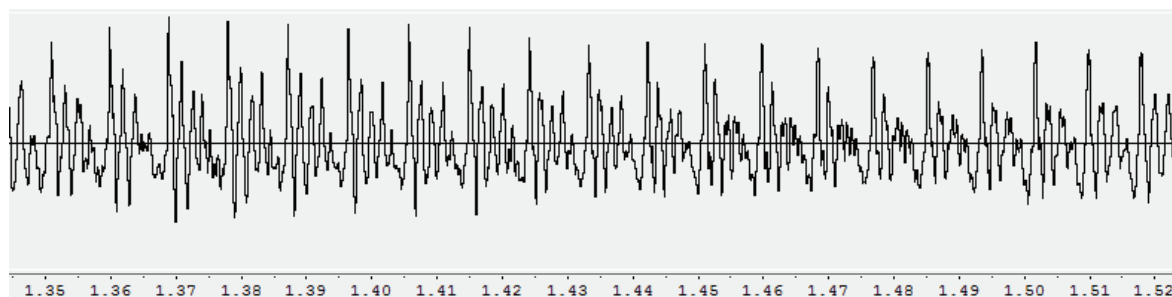
2.2 Hranice řečových jednotek

Řečové jednotky jsou základním stavebním kamenem pro syntézu řeči pomocí konkatenční metody a jsou uloženy v inventáři řečových jednotek, který vzniká segmentací řečových korpusů (soubory řečových záznamů). Záleží pouze na algoritmech, které zajišťují fonetickou segmentaci, jak dobře budou hranice mezi řečovými jednotkami určené, a tím pádem, jak kvalitní bude vzniklý inventář. Už jsme poodhalili některé výhody a nevýhody segmentačních algoritmů, ale je potřeba říci, že v některých případech není jejich úkol tak snadný, jak by se na první pohled zdálo.

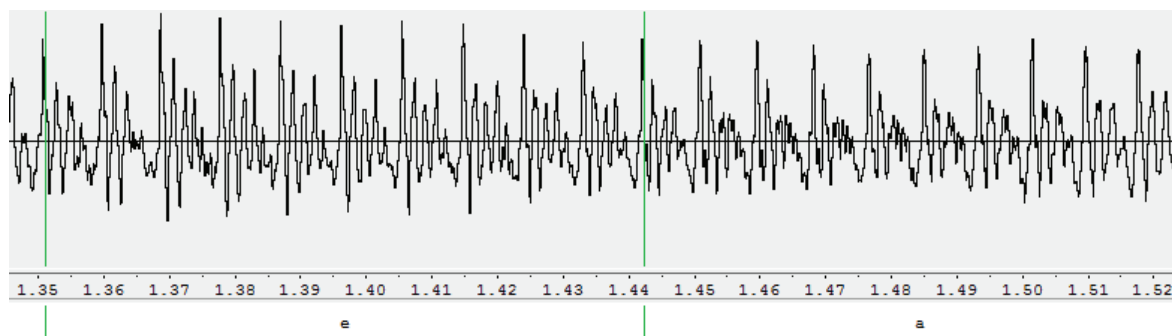
Na obrázku 2.2 je řečový signál s vymezenými hranicemi mezi jednotlivými řečovými jednotkami – v tomto případě hláskami *a-d*. Přejít z jedné hlásky na druhou je zde více než zřejmé a v takovémto případě algoritmy většinou nemívají problém hranici správně určit. Podívejme se ale na obrázek 2.3, kde je zobrazen signál pro hlásky *e-a*. Díky koartikulaci a podobnému postavení mluvidel při vytváření obou hlásek je zde velmi těžké určit místo, kde jedna hláska končí a druhá začíná. V takovýchto případech často mívá problém i člověk-expert, který zde musí využít svých znalostí a zkušeností, aby hranici dobře určil, viz obrázek 2.4.



Obrázek 2.2: Dobře rozlišitelná hranice mezi hláskami a-d.



Obrázek 2.3: Signál s pro hlásky e-a.



Obrázek 2.4: Správně určená hranice mezi hláskami e-a.

Jak bylo řečeno, řečovou jednotkou může být různá realizace hlásek. V našem programu pracujeme s řečovými jednotkami přímo na úrovni hlásek, avšak s jednou úpravou. Jelikož program musí pro každý typ hranice vytvořit specifický model, podle kterého bude oprava probíhat, byly hlásky s podobnými akustickými vlastnostmi rozděleny do pěti skupin – *frikativy* (FRI), *nosovky* (NLQ), *pauzy* (PAU), *plosivy* (PLO) a *samohlásky* (VOW). Typy fonetických hranic jsou pak tvořeny vzájemnými kombinacemi hláskových skupin:

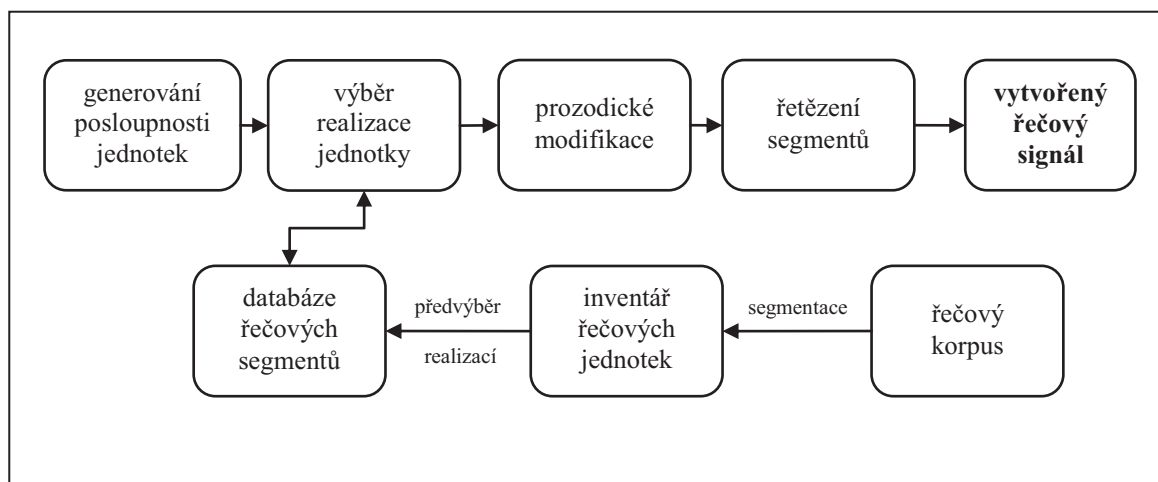
FRI-FRI	NLQ-FRI	PAU-FRI	PLO-FRI	VOW-FRI
FRI-NLQ	NLQ-NLQ	PAU-NLQ	PLO-NLQ	VOW-NLQ
FRI-PAU	NLQ-PAU	PAU-PAU	PLO-PAU	VOW-PAU
FRI-VOW	NLQ-VOW	PAU-VOW	PLO-VOW	VOW-VOW

Takto rozdělená data byla dodána vedoucím práce a program tedy bude opravovat všechny výše uvedené kombinace hranic. Za povšimnutí stojí, že v uvedených kombinacích se nikde nevyskytuje typ hranice *-PLO. Je to z toho důvodu, že začátek signálu *plosiv* má stejný průběh jako pauza a tudíž jsou tyto přechody zahrnuty pod typem *-PAU.

2.3 Konkatenční metoda

Z výše uvedeného už tedy víme, proč určení správných hranic mezi řečovými jednotkami není triviální záležitostí. Snaha o stoprocentní přesnost při určování hranic vychází z podstaty jejich následného využití. Segmentované řečové jednotky se totiž dále využívají v metodě konkatenční syntézy řeči. Tento typ syntézy pracuje na principu skládání řečových jednotek v požadované posloupnosti a tím vzniká promluva neboli nově vytvořený řečový signál. Tato metoda je v posledních letech nejčastěji používanou

technikou pro umělé vytváření řeči, protože využívá přirozené segmenty řeči a výsledná řeč obecně dosahuje vysoké kvality. Na následujícím obrázku je zobrazen proces konkatenační syntézy. Podrobnější popis lze nalézt v [1] str. 547 - 553.



Obrázek 2.5: Blokové schéma konkatenační syntézy.

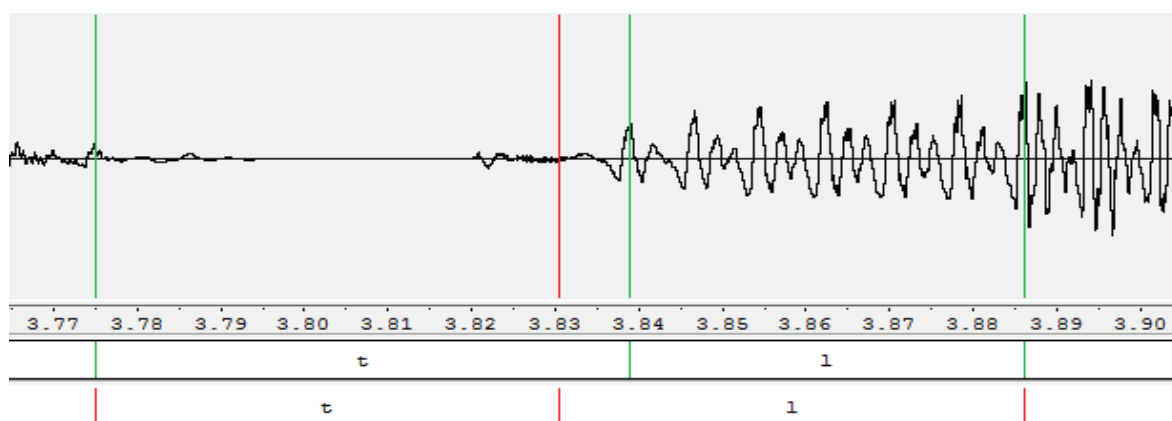
I tato metoda má však svá úskalí. Problémem je potřeba velkého množství dat, závislost na řečníkovi a především nebezpečí špatné kvality. Z principu metody totiž vyplývá, že pokud budou data, která se využívají při vytváření řeči, špatně segmentována, budou se v místech řetězení jednotek vyskytovat řečové artefakty. Ty mohou být v případě malých chyb téměř neznatelné, ale v případě velkých segmentačních chyb může v promluvě dojít i k „výskytu“ jiné řečové jednotky, která nebyla vůbec požadována. Z toho důvodu je žádoucí, abychom měli před samotnou syntézou řeči co nejkvalitněji připravená data, a proto se po vlastní automatické fonetické segmentaci ještě provádí její korekce.

2.4 Korekce fonetické segmentace

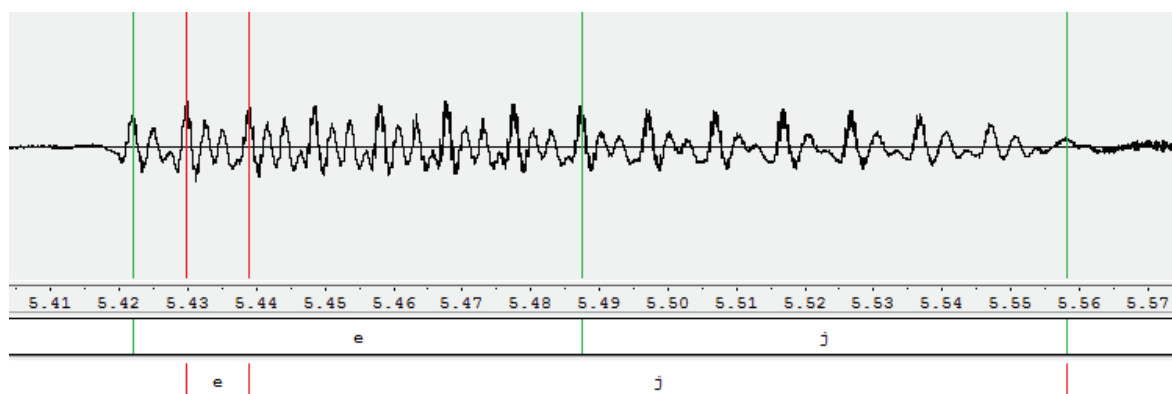
Nyní se dostáváme k samotnému jádru věci. V předchozích odstavcích byl rozebrán postup při vytváření uměle syntetizované řeči a je už tedy zřejmé, že jedním z kritických bodů, které ovlivňují kvalitu výsledné řeči, je kvalita fonetické segmentace, jež je na počátku celého procesu (nezabýváme-li se samotným vznikem řečového korpusu, ze kterého je segmentace prováděna). I přesto, že jsou segmentační algoritmy při určování hranic řečových jednotek poměrně úspěšné, dochází k určitým nepřesnostem, které se následně opravují.

Proces opravy hranic se nějakou dobu prováděl pouze manuálně. Experti, kteří měli potřebné znalosti především z fonetiky a ze zpracování signálů, ručně procházeli segmentované promluvy a jednotlivé hranice, pokud byly špatně určené, opravovali. Takováto práce je ovšem časově velmi náročná a do jisté míry také zanesená nepřesnostmi. Jak je známo, člověk není neomylný a může udělat chybu a navíc ani dva experti se nemusí shodnout a každý může hranici určit jinak. Právě tedy z časové úspory a z potřeby zavedení určité jednotnosti se začaly vyvíjet algoritmy pro automatickou korekci.

Tento zdánlivě lehký úkol automatické korekce je však zatížen řadou problémů, které by člověk většinou relativně snadno vyřešil. Stačilo by si jen přehrát segmentovanou promluvu, podívat se na rozmístění hranic a pak už jen porovnávat a dle svých zkušeností opravit dané hranice. Chceme-li ale tento proces provádět automaticky bez účasti člověka, nemáme možnost takovéto kontroly a tak je třeba vzít v potaz několik základních věcí.



Obrázek 2.6: Ukázka malé segmentační chyby (zeleně správné hranice, červeně automatická segmentace).



Obrázek 2.7: Ukázka velké segmentační chyby (zeleně správné hranice, červeně automatická segmentace).

Především nemáme žádnou znalost o kvalitě segmentace. Segmentovaná promluva může mít spoustu chyb nebo dokonce může být celá špatně, ale také může být celá správně. Nemáme žádnou apriorní znalost o velikosti chyb. Nevíme, jestli jsou chyby v řádech jednotek, desítek nebo stovek milisekund anebo dokonce v řádech sekund. Nevíme ani, jestli jsou chyby nějak vychýlené (pravostranné nebo levostranné) a nemůžeme dokonce ani zaručit, že byly opravdu nalezeny hranice mezi veškerými řečovými jednotkami.

Dostaneme-li tedy do rukou segmentovanou promluvu, tak o ní předem nemůžeme říci prakticky nic. Program pro automatickou korekci, který v našem případě pracuje na principu regrese, se musí s těmito všemi aspekty sám vypořádat. Abychom porozuměli, jak je toho schopen, přiblížíme si nyní problematiku regresní úlohy.

2.4.1 Problematika regresní úlohy

Regresní úlohu můžeme definovat jako odhadování (estimaci) nějaké proměnné na základě znalosti jiných veličin. Bavíme-li se o našem programu na opravu hranic, bude estimační složkou regresní *skóre předpovídající model* (SPM), který bude rozhodovat o správnosti hranic, a s jehož pomocí bude docházet k opravě jednotlivých segmentovaných hranic. Tento model je ale potřeba nejprve natrénovat a k tomu se využívají MLF soubory.

MLF soubory

V každém ze souborů MLF (*Master Label File*) jsou uloženy informace o segmentovaných datech viz obrázek 2.8.

automaticky nasegmentovaná data	ručně nasegmentovaná data
#!MLF!#	#!MLF!#
"*/oznam00168_00.rec"	"*/oznam00002_00.lab"
0 2785000 PAU-PAU -64.769394 SIL	0 2647783 PAU-FRI
2785000 3161250 PLO-VOW -70.944191 test	2647783 3658120 FRI-VOW
3161250 3918750 VOW-FRI -63.469074	3658120 4373120 VOW-FRI
3918750 4545000 FRI-PAU -73.298912	4373120 5158750 FRI-VOW
4545000 4665000 PLO-PAU -94.274910	.
"*/oznam00249_00.rec"	"*/oznam00013_00.lab"
0 2465000 PAU-PAU -63.467907 SIL	0 3024021 PAU-FRI
2465000 3170000 PLO-VOW -68.994781 aby	3024021 3669370 FRI-FRI
3170000 3711250 VOW-PAU -66.702927	3669370 4103750 FRI-VOW
3711250 4335000 PLO-VOW -66.425774	4103750 5593120 VOW-FRI
4335000 5004370 VOW-NLQ -60.367809	5593120 5920000 FRI-NLQ
	5920000 6348120 ?NLQ-VOW

Obrázek 2.8: Struktura MLF souborů

V trénovací fázi jsou používána data uložená v referenčním MLF souboru, který obsahuje ruční segmentaci a ve fázi korekce používáme MLF soubor s automaticky provedenou segmentací. Z každého MLF souboru využíváme informaci o typu hranice a o tom, v jakém časovém okamžiku se v promluvě vyskytuje, a dále pak využíváme název zvukové stopy, ke které se segmentace vztahuje.

SPM

Regresní skóre předpovídající model je hlavním prvkem celého programu, protože na jeho činnosti závisí konečná kvalita provedené opravy. V první fázi je model natrénován pomocí předem připravených trénovacích dat z MLF souborů, tj. pomocí metody *grid search* (část 3.1.4) je hledána nejlepší kombinace tří parametrů c (cost), g (gamma), p (epsilon), pro které je model optimálně nastaven. Ve druhé fázi model odhaduje skóre hranic v opravovaných promluvách, podle kterého se pak určují výsledné správné hranice.

Model byl vytvořen pomocí *metody podpůrných vektorů* (SVM z angl. *support vector machine*) a pro jeho vytvoření bylo využito nástrojů z *LIBSVM* [3]. Metodu SVM jsme vybrali, protože dosahuje lepších výsledků než jiné metody jako například lineární regrese nebo neuronové sítě [4]. Vytvoření modelu, resp. nalezení jeho parametrů pomocí metody *grid search*, je výpočetně velmi náročné, a proto jsme při výpočtech využívali služeb *MetaCentra* [5]. Více o regresním modelu lze nalézt v [6] odst. 3.2.

Skóre

Skóre slouží k ohodnocení hranic. V první fázi, kdy dochází k natrénování modelu, je pro všechny hranice z trénovacího souboru vypočteno skóre podle vztahu (2.1). Toto skóre je závislé na vzdálenosti (d) zpracovávané hranice od správné hranice a na maximální odchylce (σ_{max}) mezi ruční a automatickou segmentací:

$$s = 100 * e^{-\left(\frac{20*d^2}{\sigma_{max}^2}\right)} \quad (2.1)$$

Ve druhé fázi, kdy dochází k opravě hranic, jsou hodnoty skóre predikovány pomocí regresního SPM. Tyto hodnoty jsou přiřazovány jednotlivým hranicím a na jejich základě pak dochází k samotné opravě.

2.4.2 Problematika klasifikační úlohy

Jedním z cílů práce je porovnání klasifikačního a regresního přístupu, a proto se v části 3.2 podíváme, jak úspěšná je korekce pomocí klasifikace. Korekční program pracuje na estimačním principu – podle příznaků¹ rozhoduje, která hranice je tou pravou. U regresní úlohy byl estimační složkou (estimátorem) regresní SPM, který pracuje se spojitými hodnotami. Dalším estimátorem je klasifikátor, který na rozdíl od SPM pracuje s diskrétními hodnotami. Program jako takový zůstane v zásadě stejný, bude v něm změněna pouze ona rozhodovací složka.

V trénovací fázi tedy využijeme stejná data, ale nebudeme trénovat regresní model, nýbrž klasifikátor, a ve fázi korekce nebudeme hranici vybírat podle nejvyššího skóre, ale pouze na základě informace od klasifikátoru, který bude vždy provádět klasifikaci do dvou tříd – správná hranice a špatná hranice. Budou otestovány a porovnány výsledky tří typů klasifikátorů, o kterých si teď povíme několik faktů.

Podíváme se na:

- *klasifikaci za pomoci SVM*
- *klasifikaci podle k-nejbližších sousedů*
- *klasifikaci pomocí metody Extremely Randomized Trees*

Klasifikace pomocí SVM

Pro použití metody podpůrných vektorů bylo využito nástrojů z LIBSVM. Klasifikátor se nastavuje pomocí dvou základních parametrů – c (cost) a g (gamma). Parametr c představuje cenu a g je parametrem pro RBF kernel (z angl. *radial basis function kernel*), který slouží jako rozhodovací funkce a řídí se předpisem:

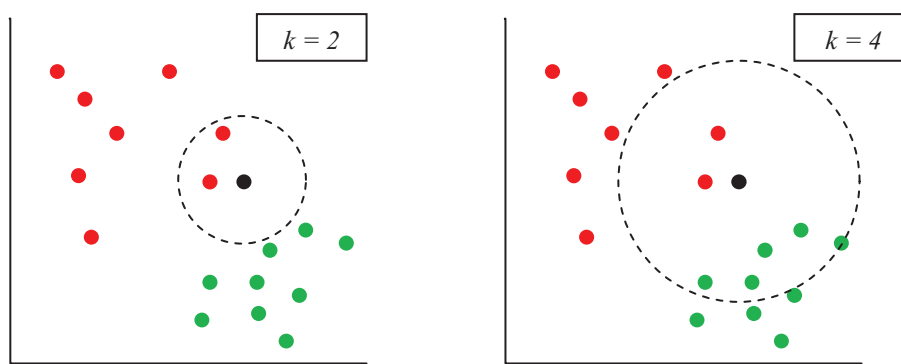
$$\text{RBF: } e^{-g*|u-v|^2}, \quad (2.2)$$

kde u, v jsou dva vzorky, v našem případě příznakové vektory. Oba parametry c i g jsou hledány pomocí metody *grid search*, která byla také použita pro vyhledávání parametrů regresního SPM. Tento klasifikátor byl testován hlavně pro porovnání výsledků regresního a klasifikačního přístupu, když oba přístupy využívají nástrojů support vector machine.

¹ Příznaky jsou parametry, které slouží k popsání hranic mezi řečovými jednotkami.

Klasifikace podle k-nejbližších sousedů

Další klasifikační přístupy, které jsme vybrali, jsou z balíku nástrojů pro strojové učení scikit-learn [7]. Prvním je klasifikátor natrénovaný pomocí algoritmu k-nejbližších sousedů, který patří, stejně jako ostatní použité klasifikátory, do třídy učení s učitelem. Rozhodnutí o tom, do které třídy bude neznámá hranice zařazena, se provádí na základě toho, do jaké třídy patří k-vektorů příznaků, které jsou od vektoru příznaků naší neznámé hranice nejbliže (viz obrázek 2.9 – pro k rovno dvěma by byla neznámá hranice tedy černá tečka určena jako špatná, ale pro k rovno čtyřem už jako správná).



Obrázek 2.9: Klasifikace podle k-nejbližších sousedů.

Klasifikace pomocí metody Extremely randomized trees

Poslední klasifikátor, který byl podroben testování, je založen na algoritmu Extremely randomized trees. Tento algoritmus patří do skupiny *ensemble* metod, které kombinují předpověď více modelů za účelem zobecnění a vylepšení robustnosti modelu, konkrétně pak do skupiny průměrových metod (*averaging methods*).

Tyto metody se zakládají na vytvoření jednoho modelu pomocí více nezávislých modelů a následném zprůměrování jejich výsledků, čímž se snižuje variance celkového modelu a dochází se tak k lepším výsledkům. Algoritmus se vyznačuje tím, že z trénovacích dat je natrénováno několik rozhodovacích stromů, jejichž uzly jsou vytvářeny náhodným dělením, respektive dělením podle náhodného rozhodovacího prahu. Hlavními parametry pro nastavování klasifikátoru jsou počet rozhodovacích stromů (*estimators*) a počet příznaků (*features*) využívaných při dělení uzlů.

3 Experimenty v oblasti automatické korekce

Hlavním cílem práce je, aby program pro automatickou korekci fonetické segmentace řeči opravoval všechny typy fonetických hranic a jeho činnost byla co nejpřesnější. Za tímto účelem bylo provedeno množství postupů a experimentů, které budou v následující kapitole detailně rozebrány, a bude vyhodnoceno, zda byly či nebyly přínosem. Kapitola je rozdělena do tří hlavních částí. V první části se podíváme na hlavní zaměření této práce, kdy je korekce hranic definována jako regresní úloha a ve druhé části uvedeme pokusy s druhým možným přístupem, kdy je korekce definována jako klasifikační úloha.

3.1 Korekce jako regresní úloha

Před samotným rozбором jednotlivých prací stručně popíšeme, jak program pracuje. Detailní popis včetně popisu příznaků lze nalézt v [6] kapitola 3.

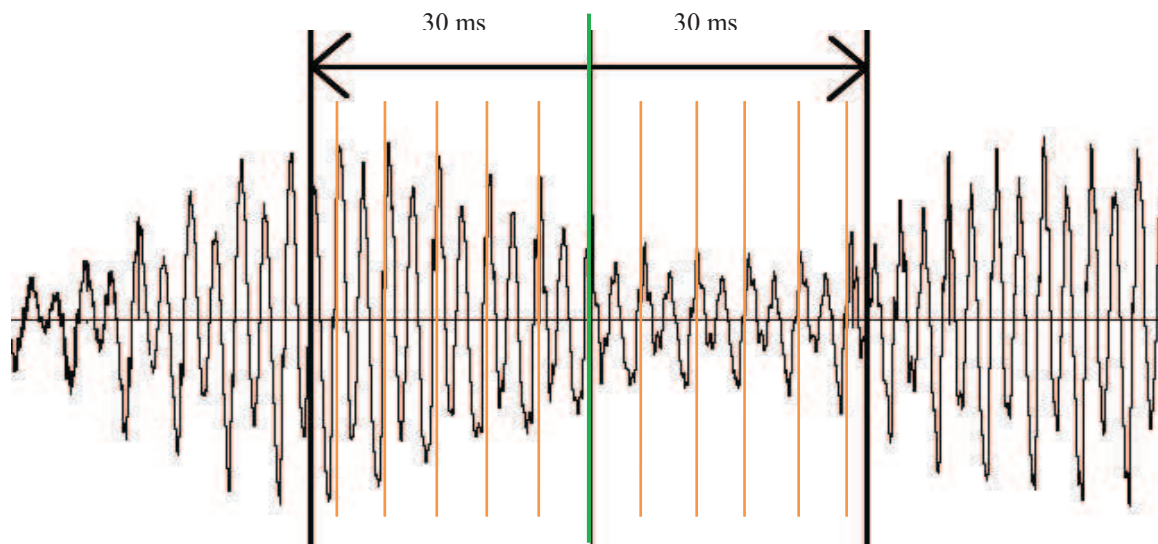
Činnost programu pro jeden konkrétní typ hranice:

Skládá se ze dvou fází – trénovací a korekční. V trénovací fázi je procházen referenční MLF soubor a okolo každé nalezené hranice je vlevo i vpravo vymezena oblast o velikosti 30 milisekund (obrázek 3.1), ve které jsou v místech pitchmarků¹ vybráni kandidáti na správnou hranici. Pro každého kandidáta (včetně správné hranice) je pak spočtena sada příznaků, která daného kandidáta definuje. Mezi počítané příznaky patří:

1. velikost okénka – oblast okolo kandidáta, ze které se mu počítají příznaky
2. znělost – jestli se kandidát pohybuje ve znělém nebo neznělém okolí
3. frekvence F0 – frekvence základního hlasivkového tónu
4. ZCR – počet průchodů nulou
5. energie – krátkodobá energie signálu
6. LSF koeficienty – dvanáct čárových spektrálních frekvencí
7. MFCC – dvanáct Melovských frekvenčních keprálních koeficientů

¹ Značka hlasivkové periody. Je to časový okamžik, kdy dojde v rámci jedné periody hlasivkových kmitů k největšímu přiblížení hlasivek.

Z příznaků všech kandidátů je vytvořen trénovací soubor, všechny hodnoty v něm jsou převedeny na interval $<-1; 1>$ a jsou uloženy škálovací parametry potřebné k tomuto převedení. Trénovací soubor se následně využívá k nalezení vhodných parametrů pomocí metody grid search a nejlepší nalezené parametry jsou pak použity k natrénování SPM.



Obrázek 3.1: Výběr kandidátů v trénovací fázi (zeleně správná hranice, oranžově vybraní kandidáti).

Ve fázi korekce je v souboru, který chceme opravit, tj. v automaticky segmentovaném souboru, pro každou hranici vymezena oblast o určité velikosti pro výběr kandidátů. Z této oblasti jsou pak v místech pitchmarků vybráni kandidáti na správnou hranici a jsou jim spočteny stejné příznaky jako v trénovací fázi. Z těchto příznaků je vytvořen testovací soubor a všechny hodnoty v něm jsou převedeny na interval $<-1; 1>$ s využitím škálovacích parametrů získaných v trénovací fázi. Data kandidátů v takto upraveném testovacím souboru jsou vyhodnocena pomocí SPM, který pro každého z nich vypočítá hodnotu skóre. Nakonec je každé skóre přiřazeno k odpovídajícímu kandidátu a ten kandidát z oblasti okolo opravované hranice, který má největší skóre, je považován za správnou hranici a jeho čas nahradí čas původní hranice v opravovaném souboru.

Program byl v [6] testován na opravu dvou typů hranic VOW-NLQ a VOW-PAU. Nejlepší dosažené výsledky ukazuje tabulka 3.1.

Typ hranice	Kolik hranic bylo						Celkem hranic
	zlepšeno		nezměněno		zhoršeno		
VOW-NLQ	59	34%	75	43%	40	23%	174
VOW-PAU	86	35%	90	37%	67	28%	243

Tabulka 3.1: Nejlepší dosažené výsledky pro hranice VOW-NLQ a VOW-PAU.

Při vyhodnocení výsledků jednotlivých úprav si budeme všimát především rozdílů v počtu správně a špatně opravených a v počtu nezměněných hranic. Později potom i celkové segmentační chyby v celém opravovaném souboru.

A nyní tedy navážeme jednotlivými úpravami postupně tak, jak byly provedeny.

3.1.1 Zavedení dynamiky

Program ve své činnosti využívá převážně statických prvků – pevná velikost okna pro výběr kandidátů, statické příznaky – a zavedení dynamiky by mohlo umožnit sledovat například vývoj signálu v čase či potenciálně špatnou hranici a podle těchto okolností by pak bylo možné se lépe přizpůsobit. Byly tedy provedeny dvě úpravy:

- byla zavedena *dynamická velikost oblasti*
- mezi příznaky jsme přidali *dynamické koeficienty*.

Dynamická velikost oblasti pro výběr kandidátů během korekční fáze

Myšlenka, která je za zavedením této úpravy, vychází z výsledků pokusů provedených v [6], kde byla v korekční fázi testována různá velikost oblastí pro výběr kandidátů. Pro obě testované hranice docházelo k různě dobrým výsledkům v rámci různě velkých oblastí, a proto jsme se rozhodli, že oblast nebudeme nastavovat staticky na pevnou hodnotu, ale že její velikost bude dynamická. Pro určení velikosti jsme vycházeli z postupu uvedeného v [4], kde je hodnota velikosti oblasti dána předpovědí o kvalitě právě zpracovávané hranice.

V korekční fázi, ještě před vybráním kandidátů, je pro automaticky segmentovanou hranici, která je právě zpracovávána, vypočtena sada příznaků, vytvořen testovací soubor a pomocí SPM určeno skórové ohodnocení. Na jeho základě je pak podle vzorce (3.1) spočtena velikost oblasti (VO), ze které se vyberou kandidáti na opravu dané hranice,

$$VO = \sqrt{\frac{\sigma_{max}}{20} * \log\left(\frac{100}{s}\right)}, \quad (3.1)$$

kde σ_{max} je maximální odchylka pro daný typ hranice, která byla nalezena v trénovací fázi. Skóre (s) by mělo vypovídat o kvalitě automaticky segmentované hranice (rozebráno v odstavci 4.2), tudíž čím menší skóre hranice obdrží, tím větší bude velikost oblasti pro výběr kandidátů. U méně ohodnocené hranice se totiž předpokládá, že se bude nacházet dále od správné hranice a je tak potřeba rozšířit oblast, ze které se kandidáti vybírají, abychom měli větší šanci, že správná hranice bude mezi vybranými kandidáty. Naopak u dobře ohodnocené hranice se předpokládá, že je kvalitní a není potřeba, aby bylo vybíráno tolik kandidátů, protože SPM má pak větší šanci nesprávně ohodnotit některého kandidáta a jeho následným výběrem se chyba segmentace po opravě zhorší.

Dynamické koeficienty

Každý kandidát, resp. hranice, je popsán sadou 28 statických příznaků, které poměrně důkladně popisují vlastnosti kandidáta, resp. hranice v daném okamžiku. Dynamické koeficienty odrážejí chování příznaků v čase a měly by nám tedy pomoci ještě lépe charakterizovat každou hranici, protože zohledňují, jak se daný příznak v průběhu času chová. To by nám mohlo velmi pomoci právě proto, že hledáme místo přechodu v průběhu času, kde se jedna hláska mění v druhou.

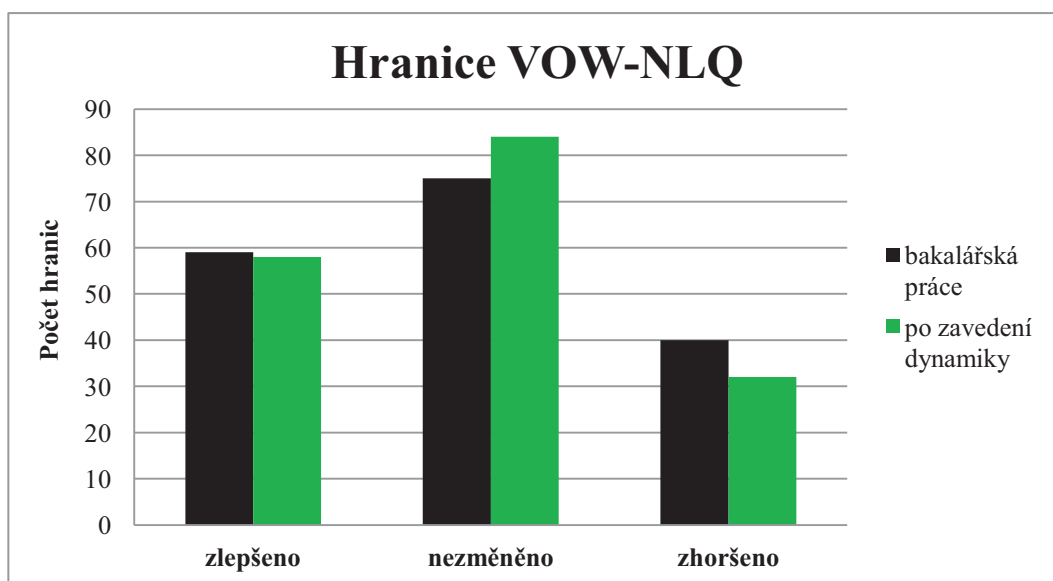
Každému statickému příznaku bude příslušet jeden dynamický koeficient, takže počet příznaků se nám v konečném výsledku zdvojnásobí, tj. vektor příznaků bude nyní obsahovat 56 hodnot. Koeficienty se určují lineární regresí a pro jejich výpočet bylo použito následujícího vzorce ([4] str. 2154):

$$\Delta F(t) = \frac{\sum_{r=-M}^M F(t+r)r}{\sum_{r=-M}^M r^2}, \quad (3.2)$$

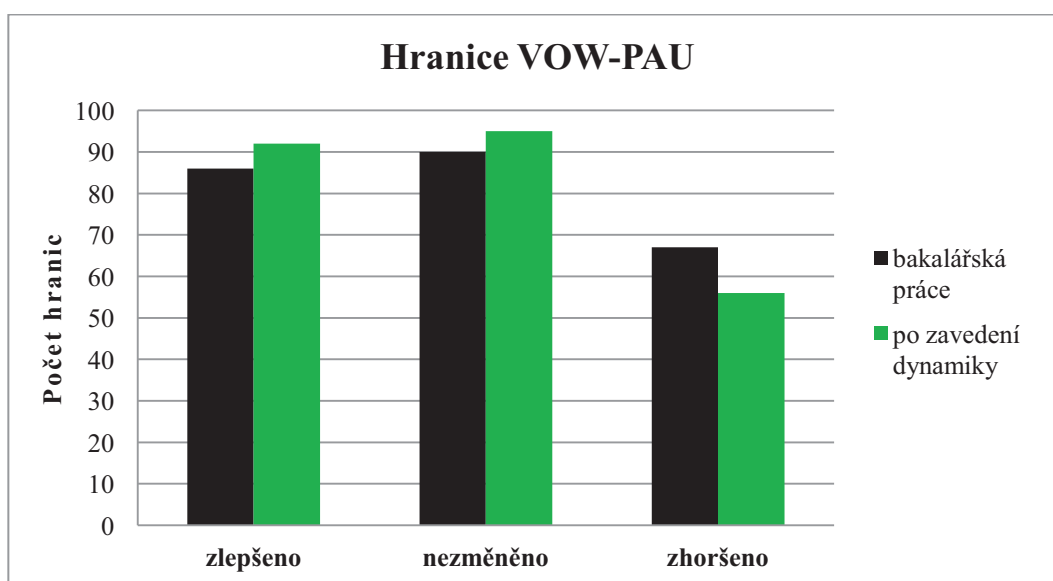
kde F jsou statické příznaky, M rovná se dvěma a je to počet předchůdců a následníků, ze kterých se počítá dynamika příznaku, a t je index zpracovávaného příznaku. Tabulka 3.2 shrnuje výsledky před a po zavedení dynamiky.

Typ hranice	Kolik hranic bylo						Celkem hranic
	zlepšeno		nezměněno		zhoršeno		
VOW-NLQ	58	33%	84	48%	32	18%	174
VOW-NLQ	59	34%	75	43%	40	23%	174
VOW-PAU	92	38%	95	39%	56	23%	243
VOW-PAU	86	35%	90	37%	67	28%	243

Tabulka 3.2: Výsledky korekce před (černě) a po (zeleně) zavedení dynamiky



Obrázek 3.2: Výsledky pro hranici VOW-NLQ po zavedení dynamiky



Obrázek 3.3: Výsledky pro hranici VOW-PAU po zavedení dynamiky

Obrázky 3.2 a 3.3 názorně ukazují, že zavedení dynamiky mělo vliv především na chybně opravované hranice. I když se počet dobře opravených hranic víceméně nezměnil, což byl sice náš prvotní cíl, počet zhoršených hranic pro opravu obou typů hranic mírně klesl. To je pro nás také úspěch, protože by bylo více než žádoucí, aby program svou činností nezhoršoval žádný nebo jen minimální počet hranic. Provedené změny jsme tak ponechali.

3.1.2 Omezení maximální odchylky

Po zavedení dynamického určování velikosti oblasti pro výběr kandidátů v korekční fázi, došlo k navýšení počtu kandidátů pro oba typy hranic a tím i k prodloužení výpočetního času. Další úprava spočívá v omezení maximální odchylky² (σ_{max}), která ovlivňuje velikost oblasti, a tím pádem i počet kandidátů. Pokud hodnotu maximální odchylky snížíme, zmenší se tak i velikost oblasti pro výběr kandidátů, viz vzorec (3.1). Pro omezení maximální odchylky jsme využili směrodatnou odchylku (3.3) automatické a ruční segmentace, spočtenou z trénovacích dat:

$$\sigma = \sqrt{\frac{1}{N} * \sum_{i=1}^N (x_i - \bar{x})^2}, \quad (3.3)$$

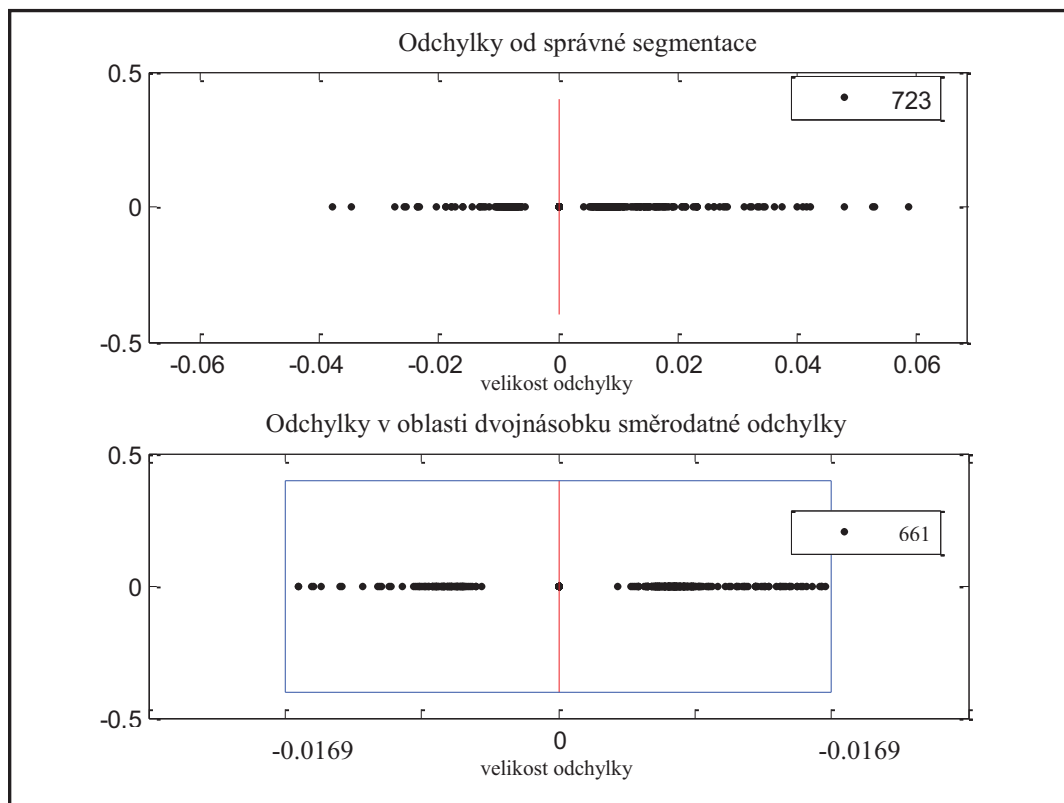
kde N je celkový počet vzorků, x_i je i -tý vzorek a \bar{x} je průměr. Jako omezující hodnotu jsme vybrali dvojnásobnou velikost σ , protože automatická segmentace obecně obsahuje převážně menší chyby a z trénovacích dat jsme zjistili, že v oblasti vymezené dvojnásobkem směrodatné odchylky se vyskytuje převážná většina hranic:

$$\sigma_{max} = 2 * \sigma \quad (3.4)$$

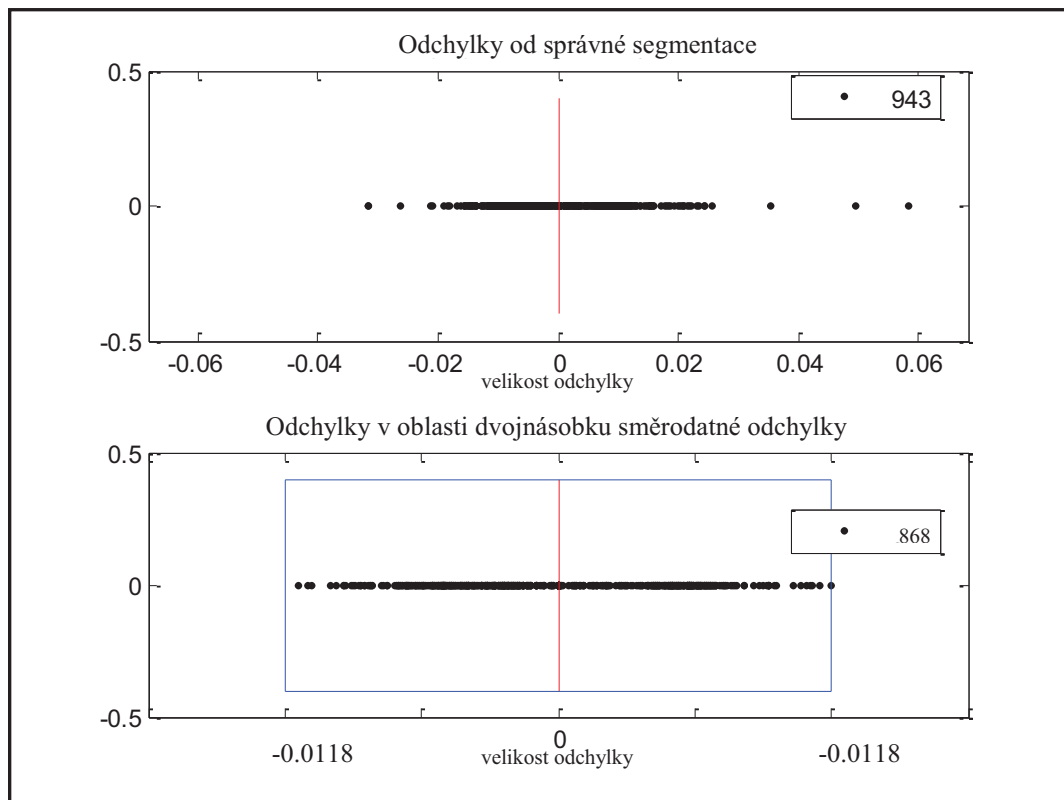
Jak ukazují obrázky 3.4 a 3.5 pro typ VOW-NLQ je to 661 (91,4 %) hranic a pro VOW-PAU pak 868 (92,0 %) hranic. Výhodou tohoto omezení je tedy i potlačení vlivu odlehlých hodnot tzv. outlierů³, vzniklých velkou chybou segmentace, kteří zanáší do procesu výběru velikosti oblasti svou velkou odchylku. Na následujících obrázcích jsou pro zkoumané typy hranic vyobrazeny všechny odchylky nalezené v trénovacích datech a následně počet odchylek, které spadají do oblasti vymezené dvojnásobnou hodnotou jejich směrodatné odchylky.

² Největší rozdíl mezi ruční a automatickou segmentací zjištěný v trénovacích datech.

³ Hranice určená automatickou segmentací, která se nachází velmi daleko od správné hranice.



Obrázek 3.4: Odchylky pro hranici VOW-NLQ



Obrázek 3.5: Odchylky pro hranici VOW-PAU

Pro každou hranici byla tedy během trénovací fáze vypočtena dvojnásobná hodnota směrodatné odchylky a ta byla použita jako maximální odchylka ve vzorci (3.1) pro výpočet velikosti oblasti. To nám přineslo následující výsledky.

Typ hranice	Kolik hranic bylo						Celkem hranic	Kandidátů
	zlepšeno		nezměněno		zhoršeno			
VOW-NLQ	62	36%	76	44%	36	21%	174	1665
VOW-NLQ	58	33%	84	48%	32	18%	174	1859
VOW-PAU	92	38%	98	40%	53	22%	243	2396
VOW-PAU	92	38%	95	39%	56	23%	243	2725

Tabulka 3.3: Omezení maximální odchylky na dvojnásobek směrodatné odchylky (zeleně)

Tabulka 3.3 srovnává výsledky po zavedení omezení maximální odchylky s výsledky po zavedení dynamiky. Pro první uvedenou hranici došlo sice ke zvýšení počtu opravených hranic, ale zároveň došlo ke stejnému zvýšení počtu zhoršených hranic. Pro druhý typ hranice, byly výsledky o něco lepší, protože došlo k poklesu zhoršených hranic při zachování počtu zlepšených hranic. V obou případech pak došlo k poklesu počtu kandidátů a to zhruba o 11 až 12 %. Protože získané výsledky byly převážně pozitivní, provedenou změnu jsme ponechali.

3.1.3 Zobecnění na všechny typy hranic

Po předchozích dvou úpravách bylo třeba rozšířit program i na ostatní typy hranic, abychom měli větší množství dat pro porovnávání v dalších experimentech a mohli jsme tak testovat, jak se oprava daří i na jiných kombinacích hranic. Z experimentování s předchozími dvěma typy hranic totiž vyšlo najevo, že každý typ reaguje na různé úpravy jiným způsobem. V průběhu adaptování programu na všechny typy hranic byl ještě korigován výpočet dvou příznaků. První změna se týkala *počtu průchodů nulou* (ZCR z angl. *zero crossing rate*), jejichž výslednou hodnotu jsme znormovali na pevnou velikost okna (20 ms), stejně jako je normován příznak krátkodobé energie signálu, abychom odstranili vliv rozdílně velkých oken, ze kterých se příznaky k hranicím počítají. Druhá úprava spočívala v nahrazení diskrétních hodnot {0; 0.5; 1} příznaku *znělost* spojitými hodnotami z intervalu [0; 1], které kandidátovu znělost reprezentují přesněji. A nyní se tedy podíváme, jak program funguje na ostatních typech hranic.

Tabulka 3.4 ukazuje výsledky korekce všech typů hranic automaticky segmentovaného souboru. Ke každému *typu hranice* je zobrazena chyba automatické segmentace (*ref-auto*), tj. součet všech odchylek mezi ruční a automatickou segmentací a chyba opravené segmentace (*ref-fix*), tj. součet všech odchylek mezi ruční a opravenou segmentací. Dále pak *průměrné zlepšení*, resp. *zhoršení*, což je průměrná změna mezi hranicí v automatické a opravené segmentaci, pokud dojde při opravě ke zlepšení, resp. zhoršení hranice. Nakonec je pak uveden počet *zlepšených*, *nezměněných* a *zhoršených* hranic společně s jejich procentuálním podílem na celkovém počtu a *celkový počet hranic* daného typu v opravovaném souboru.

Poslední řádek v tabulce je vyhodnocení, které shrnuje všechny výsledky do celkové segmentační chyby, celkového průměrného zlepšení/zhoršení a celkového počtu zlepšených, nezměněných, zhoršených a všech hranic.

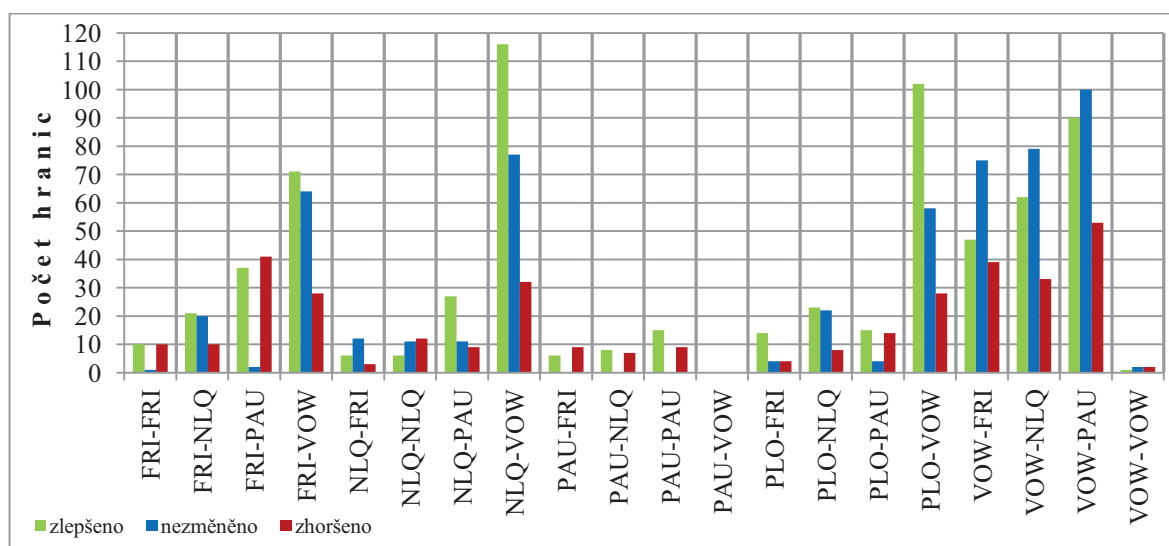
Typ Hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2228	6,149	5,942	10	48%	1	5%	10	48%	21
FRI-NLQ	0,4870	0,3628	10,306	9,230	21	41%	20	39%	10	20%	51
FRI-PAU	0,4619	0,4333	4,092	2,996	37	46%	2	3%	41	51%	80
FRI-VOW	0,6668	0,3575	6,309	4,951	71	44%	64	39%	28	17%	163
NLQ-FRI	0,1255	0,0911	9,085	6,709	6	29%	12	57%	3	14%	21
NLQ-NLQ	0,2144	0,2932	9,115	11,125	6	21%	11	38%	12	41%	29
NLQ-PAU	0,3617	0,2401	7,358	8,564	27	57%	11	23%	9	19%	47
NLQ-VOW	1,9411	1,1262	9,688	9,656	116	52%	77	34%	32	14%	225
PAU-FRI	0,0438	0,0821	1,674	5,368	6	40%	0	0%	9	60%	15
PAU-NLQ	0,1119	0,1079	2,673	2,481	8	53%	0	0%	7	47%	15
PAU-PAU	0,4933	0,5873	7,608	23,123	15	63%	0	0%	9	38%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2128	4,643	5,929	14	64%	4	18%	4	18%	22
PLO-NLQ	0,1874	0,1029	6,777	8,923	23	43%	22	42%	8	15%	53
PLO-PAU	0,1746	0,1559	5,609	4,671	15	45%	4	12%	14	42%	33
PLO-VOW	0,8970	0,3909	6,745	6,495	102	54%	58	31%	28	15%	188
VOW-FRI	0,6728	0,5524	8,115	6,691	47	29%	75	47%	39	24%	161
VOW-NLQ	1,4056	1,1019	9,606	8,845	62	36%	79	45%	33	19%	174
VOW-PAU	1,1957	0,9455	7,175	7,462	90	37%	100	41%	53	22%	243
VOW-VOW	0,0386	0,0538	7,000	11,094	1	20%	2	40%	2	40%	5
	9,9580	7,4205	6,486	7,513	677	43,1%	542	34,5%	351	22,4%	1570

Tabulka 3.4: Výsledky korekce všech typů hranic

Jak bylo řečeno v teoretické části 2.2, pro naše potřeby jsme rozlišili dvacet kombinací hranic, takže jsme museli natrénovat dvacet různých SPM, což bylo časově velmi náročné. Z těchto 20 opravovaných typů hranic došlo u 4 typů ke zhoršení a u 15 typů ke zlepšení celkové chyby segmentace. Jeden typ hranice (PAU-VOW) nemohl být vyhodnocen, protože referenční a automaticky nalezené hranice si neodpovídaly⁴. Celková chyba segmentace v celém souboru se tak snížila o 25,5 % (2,54 s). Pro názornost je na obrázku 3.6 graficky znázorněno, jak dopadla oprava pro jednotlivé typy hranic.

Podíváme-li se na výsledky z pohledu počtu zlepšených, nezměněných a zhoršených hranic zjistíme, že zhruba třetina hranic (34,5%) se po opravě nezměnila a počet zlepšených hranic (43,1 %) skoro dvojnásobně přesáhl počet zhoršených hranic (22,4 %). Z celkového pohledu jsou to tedy lepší výsledky, než jsme měli u prvních dvou testovaných hranic.

Při pohledu na průměrné zlepšení a zhoršení si můžeme všimnout, že rozdíl mezi nimi je u naprosté většiny hranic jen kolem 1 až 2 ms. Zaměříme-li se pak na čtyři typy, u kterých došlo ke zhoršení segmentace, je patrné, že u nich toto neplatí a průměrné zhoršení je daleko větší než průměrné zlepšení. To by mohlo být způsobeno typem hranice, protože tři ze čtyř zhoršených typů jsou kombinace téže skupiny hlásek, takže by na SPM mohla mít vliv určitá podobnost hlásek anebo by mohl být problém v nedostatečném množství dat pro natrénování modelu (všechny chybové typy neměly příliš mnoho trénovacích dat).



Obrázek 3.6: Zobrazení počtu zlepšených, nezměněných a zhoršených hranic

⁴ Tento případ nastává, pokud segmentační algoritmus označí hranici za jiný typ než je v referenčních datech.

I přesto, že segmentace většiny hranic dosáhla po korekci zlepšení, je možné, avšak nepravděpodobné, že některé SPM nebyly dobře natrénované. Funkčnost SPM jsme ale stejně raději ověřili tak, že jsme programem nechali opravit automaticky segmentovaný soubor, který byl použit pro natrénování modelů. Pokud jsou modely dobře natrénované, měly by hranice v trénovacím souboru opravit daleko lépe než nějaké neznámé hranice, protože už se s nimi jednou setkaly v trénovací fázi. Výsledky korekce shrnuje tabulka 3.5.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,5409	0,0428	7,561	0,434	66	78%	17	20%	2	2%	85
FRI-NLQ	0,8296	0,0404	8,164	0,536	97	57%	68	40%	5	3%	170
FRI-PAU	1,1728	0,2755	4,406	1,002	213	76%	26	9%	41	15%	280
FRI-VOW	2,6773	0,1804	7,102	2,846	360	57%	252	40%	21	3%	633
NLQ-FRI	0,3776	0,0011	9,655	0,000	39	41%	56	59%	0	0%	95
NLQ-NLQ	0,9127	0,2552	11,451	18,125	59	61%	36	38%	1	1%	96
NLQ-PAU	0,6666	0,1014	7,266	2,601	81	54%	60	40%	9	6%	150
NLQ-VOW	6,4671	1,5450	10,111	15,583	513	58%	350	40%	17	2%	880
PAU-FRI	0,2782	0,0928	4,225	0,587	45	83%	1	2%	8	15%	54
PAU-NLQ	0,1094	0,0626	2,343	1,321	29	59%	4	8%	16	33%	49
PAU-PAU	2,8796	2,2787	14,822	19,253	86	70%	1	1%	35	29%	122
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2906	0,1211	3,993	1,425	51	57%	15	17%	24	27%	90
PLO-NLQ	0,7137	0,0466	7,711	3,843	89	49%	87	48%	5	3%	181
PLO-PAU	0,4677	0,1168	4,918	0,814	73	75%	14	14%	10	10%	97
PLO-VOW	3,3480	0,2249	7,322	6,021	429	56%	330	43%	3	0%	762
VOW-FRI	2,0595	0,2000	8,064	4,587	234	38%	370	61%	6	1%	610
VOW-NLQ	4,2978	0,8521	11,029	18,616	331	46%	381	53%	11	2%	723
VOW-PAU	4,4153	0,7311	7,744	3,714	493	52%	414	44%	36	4%	943
VOW-VOW	0,1157	0,0437	18,000	0,000	4	67%	2	33%	0	0%	6
	32,620	7,2123	7,794	5,065	3292	55%	2484	41%	250	4%	6026

Tabulka 3.5: Výsledky korekce pro trénovací soubor

Výsledky ukazují, že regresní SPM fungují dobře a na známých datech podávají velice dobré výsledky – došlo ke snížení celkové segmentační chyby o 77,9 %. Pro neznámá data však výsledky nejsou tak přesvědčivé, jak jsme se mohli přesvědčit výše, a proto se pokusíme vylepšit činnost SPM přidáním druhé fáze do metody vyhledávání parametrů, sloužících k jeho natrénování.

3.1.4 Dvofázové vyhledávání parametrů pro regresní model

Natrénování regresního skóre předpovídajícího modelu spočívá v jeho správném nastavení pomocí tří následujících parametrů:

- c (cost) – parametr ohodnocení, který hledá rovnováhu mezi jednoduchostí a korektností rozhodování
- g (gamma) – parametr ve funkci RBF kernelu, který ovlivňuje, jaký vliv budou mít jednotlivé trénovací vzorky
- p (epsilon) – parametr pro ztrátovou funkci, který definuje její oblast

Pro hledání parametrů byl využit skript *gridregression.py* z nástrojů LIBSVM [3], který provádí natrénování SPM pro různé nastavení parametrů a nakonec vybere takové nastavení, pro které model dosahoval nejlepších výsledků. Toto nastavení parametrů je pak použito ve finálním SPM. Primárně tento skript pracuje jen v jedné fázi s deseti složkovou křížovou-validací (cross-validace viz část 3.1.5), kdy je parametrický prostor prohledáván v hrubém měřítku. My jsme přidali ještě druhou fázi, s jemnějším prohledáváním.

V první fázi jsou s krokem jedna z níže uvedených intervalů vybírány kombinace hodnot exponentů c_{1exp} , g_{1exp} , p_{1exp} a pomocí nich jsou spočteny příslušné parametry c_1 , g_1 , p_1 :

$$c_1 = 2^{c_{1exp}}, \quad c_{1exp} \in \langle -1, 9 \rangle \quad (3.5)$$

$$g_1 = 2^{g_{1exp}}, \quad g_{1exp} \in \langle -8, 0 \rangle \quad (3.6)$$

$$p_1 = 2^{p_{1exp}}, \quad p_{1exp} \in \langle -12, 1 \rangle \quad (3.7)$$

Pro každou kombinaci parametrů je natrénován SPM a exponenty (c'_{1exp} , g'_{1exp} , p'_{1exp}), které sloužily k vypočtení parametrů, pro které model dosáhl nejlepších výsledků, pokračují do druhé „jemnější“ fáze. V této fázi jsou nové exponenty vybírány pouze z intervalu $-0,75$ až $+0,75$ od nejlepších exponentů z první fáze a to právě s jemnějším krokem o hodnotě $0,25$:

$$c_2 = 2^{c_{2exp}}, \quad c_{2exp} \in \langle (c'_{1exp} - 0.75), (c'_{1exp} + 0.75) \rangle \quad (3.8)$$

$$g_2 = 2^{g_{2exp}}, \quad g_{2exp} \in \langle (g'_{1exp} - 0.75), (g'_{1exp} + 0.75) \rangle \quad (3.9)$$

$$p_2 = 2^{p_{2exp}}, \quad p_{2exp} \in \langle (p'_{1exp} - 0.75), (p'_{1exp} + 0.75) \rangle \quad (3.10)$$

Tento postup by nám měl zajistit nalezení ještě lepších parametrů než v případě jednofázového hledání parametrů. Tabulka 3.6 ukazuje výsledky korekce po zavedení jemnějšího prohledávání. Jako poslední řádek byly pro srovnání přidány výsledky z jednofázového vyhledávání parametrů.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2129	7,136	5,942	10	48%	1	5%	10	48%	21
FRI-NLQ	0,4870	0,4041	9,622	8,432	20	39%	18	35%	13	25%	51
FRI-PAU	0,4619	0,4274	3,855	2,970	39	49%	2	3%	39	49%	80
FRI-VOW	0,6668	0,3729	6,102	5,225	73	45%	61	37%	29	18%	163
NLQ-FRI	0,1255	0,0812	9,136	9,844	7	33%	12	57%	2	10%	21
NLQ-NLQ	0,2144	0,2987	8,831	13,290	7	24%	11	38%	11	38%	29
NLQ-PAU	0,3617	0,2199	7,524	8,604	28	60%	11	23%	8	17%	47
NLQ-VOW	1,9411	1,2021	9,762	9,841	113	50%	75	33%	37	16%	225
PAU-FRI	0,0438	0,0761	1,674	4,707	6	40%	0	0%	9	60%	15
PAU-NLQ	0,1119	0,1079	2,673	2,481	8	53%	0	0%	7	47%	15
PAU-PAU	0,4933	0,5850	6,316	20,707	15	63%	0	0%	9	38%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2181	5,379	6,807	13	59%	4	18%	5	23%	22
PLO-NLQ	0,1874	0,1069	6,777	8,376	23	43%	21	40%	9	17%	53
PLO-PAU	0,1746	0,1600	5,192	4,517	15	45%	4	12%	14	42%	33
PLO-VOW	0,8970	0,3779	6,480	6,447	104	55%	60	32%	24	13%	188
VOW-FRI	0,6728	0,5582	7,991	6,447	45	28%	78	48%	38	24%	161
VOW-NLQ	1,4056	1,1393	9,448	9,393	59	34%	84	48%	31	18%	174
VOW-PAU	1,1957	0,9215	7,231	7,241	91	37%	99	41%	53	22%	243
VOW-VOW	0,0386	0,0480	6,375	11,063	2	40%	1	20%	2	40%	5
Dvoufázové	9,9580	7,5182	6,375	7,617	678	43%	542	35%	350	22%	1570
Jednofázové	9,9580	7,4205	6,486	7,513	677	43%	542	35%	351	22%	1570

Tabulka 3.6: Výsledky korekce s jemnějším vyhledáváním parametrů SPM (krok 0.25)

Můžeme si všimnout, že pomocí dvoufázového vyhledávání parametrů jsme ve výsledku měli pouze o jednu zlepšenou hranici více a o jednu zhoršenou hranici méně oproti jednofázovému vyhledávání. Taktéž průměrné hodnoty zlepšení a zhoršení byly velice podobné a lišily se jen o desetinu milisekund. I přesto by se ale dalo říci, že výsledky získané pomocí jednofázového hledání byly lepší, protože celková segmentační chyba byla o 98 ms menší. To je způsobeno právě nepatrně větší průměrnou hodnotou zlepšení a nepatrně menší průměrnou hodnotou zhoršení oproti dvoufázovému vyhledávání.

Jelikož jsme předpokládali, že výsledky budou znatelně lepší, vyzkoušeli jsme na třech vybraných typech hranic ještě jemnější prohledávání prostoru parametrů s krokem 0,1. V tabulce 3.7 jsou srovnány výsledky tohoto pokusu s předchozími výsledky.

Krok	Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
		ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
0.10	FRI-FRI	0,2249	0,2188	7,116	5,797	9	43%	2	10%	10	48%	21
0.25		0,2249	0,2129	7,136	5,942	10	48%	1	5%	10	48%	21
0.10	NLQ-FRI	0,1255	0,0911	9,085	6,709	6	29%	12	57%	3	14%	21
0.25		0,1255	0,0812	9,136	9,844	7	33%	12	57%	2	10%	21
0.10	NLQ-NLQ	0,2144	0,3014	8,027	14,325	7	24%	12	41%	10	34%	29
0.25		0,2144	0,2987	8,831	13,290	7	24%	11	38%	11	38%	29

Tabulka 3.7: Výsledky korekce s jemnějším vyhledáváním parametrů SPM (krok 0.10)

Zjistili jsme, že i po zavedení ještě jemnějšího kroku nedošlo k žádnému zlepšení ve výsledné korekci. K tomu by tedy s největší pravděpodobností došlo jen v případě, že by krok zvolený v první fázi prohledávání prostoru parametrů měl velkou hodnotu, například 5 nebo 10. Z výše uvedeného tak vyplývá, že původně zvolený krok o velikosti jedna je dostačující k nalezení optimálních parametrů modelu a jeho dalším zjemněním už výsledek korekce nezlepšíme. Z toho důvodu budeme i nadále vyhledávat parametry v jedné fázi.

3.1.5 Cross-validace

Předchozím krokem se nám tedy nepodařilo výsledek korekce vylepšit, proto jsme se rozhodli otestovat, zda výsledek nějakým způsobem nezávisí na samotných datech. K tomu jsme použili metodu křížové validace (*cross-validace*), která zkoumá, jak dobře bude model – v našem případě SPM – pracovat s neznámými daty, která má predikovat. Tato metoda se hojně využívá právě při predikci, aby se zamezilo tzv. přetrénování modelu. My už jsme se s ní setkali a využíváme ji v metodě grid search, při hledání parametrů SPM. Cross-validaci tedy využijeme k získání robustnějších výsledků tím, že systematictěji vybereme trénovací a testovací data. Experiment provedeme na dvou nejpočetnějších typech hranic – VOW-NLQ a VOW-PAU.

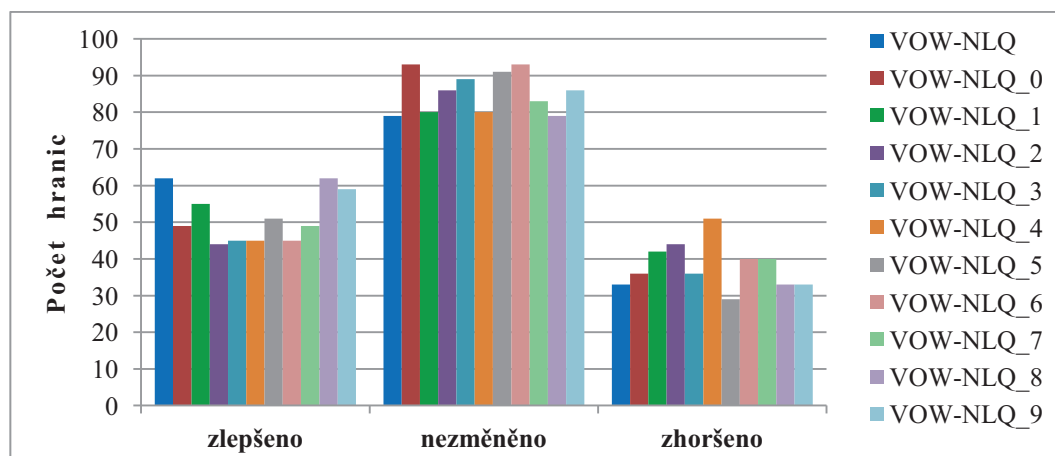
Typ hranice	Počet hranic v souboru		Celkem hranic	Poměr rozdělení dat	
	trénovacím	pro korekci		pro trénink	pro korekci
VOW-NLQ	723	174	897	80%	20%
VOW-PAU	943	243	1186	80%	20%

Tabulka 3.8: Informace o hranicích VOW-NLQ a VOW-PAU

Postup je takový, že vezmeme všechna data, která máme k dispozici, tj. trénovací soubor i soubor pro korekci, a sloučíme je dohromady. Data v takto vzniklém souboru následně náhodně rozdělíme na trénovací data a data pro korekci v poměru, který odpovídal původnímu poměru před sloučením obou souborů (viz tabulka 3.8). Poté provedeme natrénování SPM a korekci a celý tento proces desetkrát zopakujeme. Všechny výsledky následně zprůměrujeme a tím dostaneme deseti složkovou cross-validaci trénovacích dat.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno	nezměněno	zhoršeno	zlepšeno	nezměněno	zhoršeno	
<i>VOW-NLQ</i>	<i>1,4056</i>	<i>1,1019</i>	<i>9,606</i>	<i>8,845</i>	62	36%	79	45%	33	19%	174
VOW-NLQ_0	1,2116	1,1203	10,111	11,226	49	28%	93	52%	36	20%	178
VOW-NLQ_1	1,2812	1,1707	10,338	10,904	55	31%	80	45%	42	24%	177
VOW-NLQ_2	0,9237	0,9983	8,916	10,613	44	25%	86	49%	44	25%	174
VOW-NLQ_3	1,1596	1,1971	9,181	12,519	45	26%	89	52%	36	21%	170
VOW-NLQ_4	1,0287	1,1968	9,325	11,524	45	26%	80	45%	51	29%	176
VOW-NLQ_5	1,1551	0,9587	9,647	10,193	51	30%	91	53%	29	17%	171
VOW-NLQ_6	1,0907	1,0491	10,215	10,452	45	25%	93	52%	40	22%	178
VOW-NLQ_7	1,1709	1,1454	9,480	10,975	49	28%	83	48%	40	23%	172
VOW-NLQ_8	1,2212	1,0196	9,047	10,888	62	36%	79	45%	33	19%	174
VOW-NLQ_9	1,3675	1,2078	10,042	13,114	59	33%	86	48%	33	19%	178
PRŮMĚR:	1,1610	1,1064	9,630	11,241	50	29%	86	49%	38	22%	175

Tabulka 3.9: Výsledky cross-validace pro VOW-NLQ

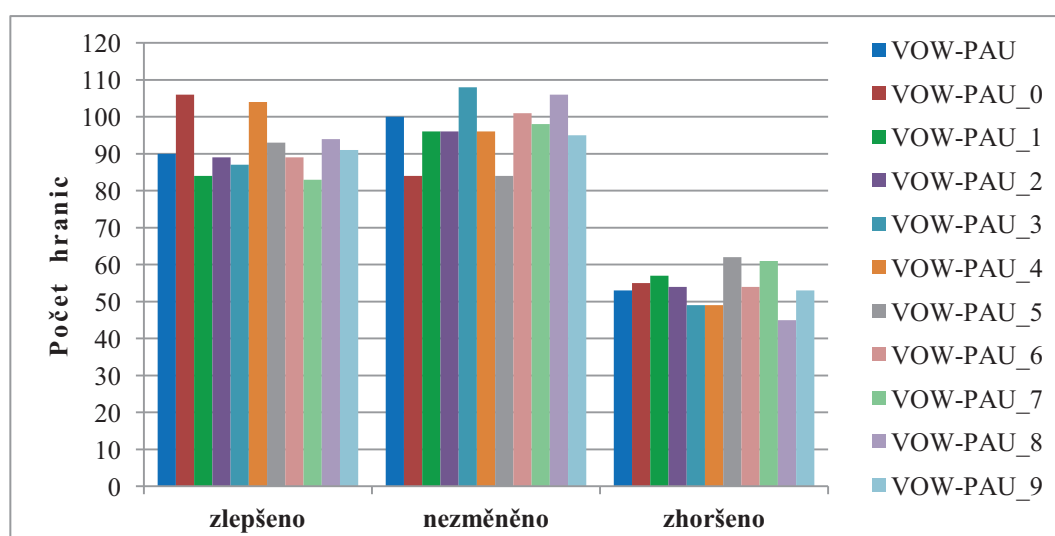


Obrázek 3.7: Výsledky cross-validace pro VOW-NLQ

Pokud provedeme srovnání získaných výsledků mezi hranicemi VOW-NLQ (tabulka 3.9) a VOW-PAU (tabulka 3.10), dojdeme k závěru, že výsledky druhé zmiňované hranice jsou stabilnější a původní vzorek vcelku odpovídá zprůměrovanému výsledku všech deseti oprav. U hranice VOW-NLQ byly výsledky více rozkolísané a setkali jsme se dokonce i se třemi případy, kdy opravený soubor měl větší segmentační chybu než původní soubor. Zdá se také, že jsme u této hranice měli původně trochu štěstí v datech, jelikož výsledky původní opravy jsou nejlepší ze všech. Pro srovnání je pro oba případy vždy v prvním řádku ještě uveden výsledek původní korekce bez cross-validace.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
VOW-PAU	1,1957	0,9455	7,175	7,462	90	37%	100	41%	53	22%	243
VOW-PAU_0	1,2554	0,8509	7,475	7,052	106	43%	84	34%	55	22%	245
VOW-PAU_1	1,0346	0,8690	6,769	7,070	84	35%	96	41%	57	24%	237
VOW-PAU_2	1,1952	0,8853	7,388	6,438	89	37%	96	40%	54	23%	239
VOW-PAU_3	1,1245	0,8720	6,863	7,032	87	36%	108	44%	49	20%	244
VOW-PAU_4	1,2786	0,8806	7,023	6,784	104	42%	96	39%	49	20%	249
VOW-PAU_5	1,2681	1,0617	7,258	7,559	93	39%	84	35%	62	26%	239
VOW-PAU_6	1,1427	0,9130	6,881	7,087	89	36%	101	41%	54	22%	244
VOW-PAU_7	1,0743	0,9107	6,871	6,668	83	34%	98	40%	61	25%	242
VOW-PAU_8	1,3137	0,9164	7,558	6,960	94	38%	106	43%	45	18%	245
VOW-PAU_9	1,1661	0,9156	6,566	6,548	91	38%	95	40%	53	22%	239
PRŮMĚR:	1,185	0,908	7,065	6,920	92	38%	96	40%	54	22%	242

Tabulka 3.10: Výsledky cross-validace pro hranici VOW-PAU



Obrázek 3.8: Výsledky cross-validace pro hranici VOW-PAU

Cross-validací jsme tedy ověřili robustnost získaných výsledků pro hranice VOW-PAU a VOW-NLQ. Cross-validace prvního zmiňovaného typu víceméně odpovídá původně získanému výsledku, ale jak se zdá, regresní modely některých typů hranic budou více ovlivňovány neznámými daty, jak jsme mohli vidět v případě hranice VOW-NLQ. Tato ovlivnitelnost by mohla být zapříčiněna velkými chybami segmentace, se kterými si SPM zatím neumí příliš poradit (bude rozebráno v části 4.3) nebo některými akustickými odlišnostmi mezi hláskami v našich hláskových skupinách.

3.1.6 Oprava hranic mezi konkrétními hláskami

Jedním z možných problémů, kvůli kterému korekce nedosahuje lepších výsledků, je rozdělení hlásek do skupin podle jejich akustických vlastností. Nyní pracujeme s pěti skupinami hlásek, do kterých byly jednotlivé hlásky rozděleny podle toho, jak jsou si akusticky podobné. Sdružením do skupin jsme jednak částečně zobecnili problém korekce tím, že jsme snížili počet typů hranic, pro které je třeba natrénovat SPM, a zároveň jsme se tím i vyhnuli potřebě daleko většího množství dat. Rozhodli jsme se otestovat, zda by náš regresní model pracoval lépe, pokud bychom ho natrénovali přímo na opravu hranic mezi dvěma konkrétními typy hlásek. Abychom mohli model natrénovat, potřebovali jsme soubor automaticky segmentovaný přímo na úrovni jednotlivých hlásek a k němu i ručně segmentovaná data. Oba soubory byly dodány vedoucím práce.

Prvním krokem bylo zjištění počtu všech typů hranic (přechodů) mezi jednotlivými hláskami. To z toho důvodu, abychom vybrali nejčastěji zastoupený přechod a měli tak co nejvíce dat pro natrénování modelu. Nejpočetnějšími přechody byly $J-I^5$ (104 výskytů) a $n-a$ (88 výskytů) – oba tyto přechody spadají do skupiny NLQ-VOW. Potom jsme soubor segmentovaný na úrovni hlásek rozdělili ve stejném poměru jako byla rozdělena data pro NLQ-VOW, aby zůstal zachován počet hranic použitých pro trénink. Pro natrénování SPM jsme tak použili 80 % dat a na zbylých 20 % jsme prováděli korekci. Pro robustnější výsledky jsme opět použili metodu deseti složkové cross-validace. Vše je zobrazeno v tabulkách 3.11 a 3.12.

⁵ Písmenkem „J“ jsou ve fonetické transkripci označeny fonémy, které znějí jako „ň“ a „I“ pak označuje „í“.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I_0	0,1267	0,1919	8,265	12,274	4	21%	7	37%	8	42%	19
J-I_1	0,4494	0,5551	8,424	20,578	7	30%	8	35%	8	35%	23
J-I_2	0,2238	0,2191	12,272	19,677	10	45%	6	27%	6	27%	22
J-I_3	0,2786	0,4390	14,312	16,569	5	19%	7	27%	14	54%	26
J-I_4	0,1976	0,2464	11,036	15,753	7	35%	5	25%	8	40%	20
J-I_5	0,2654	0,2794	13,141	29,780	8	42%	7	37%	4	21%	19
J-I_6	0,1870	0,2730	9,312	14,725	5	26%	5	26%	9	47%	19
J-I_7	0,2424	0,2562	12,571	14,545	7	39%	4	22%	7	39%	18
J-I_8	0,3394	0,4528	12,055	19,068	8	38%	2	10%	11	52%	21
J-I_9	0,2274	0,1578	15,266	10,524	8	44%	5	28%	5	28%	18
PRŮMĚR:	0,2538	0,3071	11,665	17,349	6,9	34%	5,6	27%	8,0	39%	20,5
NLQ-VOW	1,9411	1,1262	9,688	9,656	116	52%	77	34%	32	14%	225

Tabulka 3.11: Výsledky korekce konkrétní hranice (J-I)

Výsledky jasně dokazují, jak se kvalita korekce v rámci přechodů liší a to jak vzájemně, tak především pak od celé skupiny, do které byly tyto přechody zahrnuty. U prvního přechodu jsme u osmi z deseti složek cross-validace zaznamenali zhoršení segmentační chyby a celkový zprůměrovaný výsledek je tedy v porovnání s výsledkem korekce celé skupiny NLQ-VOW podstatně horší.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
n-a_0	0,1147	0,0524	9,667	8,250	9	53%	5	29%	3	18%	17
n-a_1	0,0945	0,0413	7,983	9,314	9	64%	3	21%	2	14%	14
n-a_2	0,1514	0,0162	11,262	0,000	12	86%	2	14%	0	0%	14
n-a_3	0,1630	0,0561	12,114	14,312	10	63%	5	31%	1	6%	16
n-a_4	0,1372	0,0397	8,865	0,000	11	69%	5	31%	0	0%	16
n-a_5	0,1280	0,0386	8,947	9,062	11	73%	3	20%	1	7%	15
n-a_6	0,0885	0,0137	9,119	7,188	9	75%	2	17%	1	8%	12
n-a_7	0,1040	0,0398	8,672	5,188	8	53%	6	40%	1	7%	15
n-a_8	0,1048	0,0394	8,070	7,188	9	53%	7	41%	1	6%	17
n-a_9	0,0958	0,0322	8,599	5,188	8	67%	3	25%	1	8%	12
PRŮMĚR:	0,1182	0,0369	9,330	6,569	9,6	65%	4,1	28%	1,1	7%	14,8
NLQ-VOW	1,9411	1,1262	9,688	9,656	116	52%	77	34%	32	14%	225

Tabulka 3.12: Výsledky korekce konkrétní hranice (n-a)

Naproti tomu korekce u druhého přechodu dopadla znatelně lépe. Nejenže došlo ke snížení segmentační chyby všech jednotlivých složek cross-validace a tím pádem i celkové zprůměrované segmentační chyby, ale navíc je tento zprůměrovaný výsledek lepší než výsledek korekce celé skupiny NLQ-VOW.

Ze získaných poznatků by se dalo vyvodit následující. Pokud z hláskové skupiny, se kterou pracujeme, vybereme nějaké konkrétní hlásky a natrénujeme SPM přímo na opravu tohoto konkrétního přechodu, nemusí to nezbytně vést ke zlepšení výsledků, jak jsme se původně domnívali. Některé typy přechodů (viz *J-I*) jsou totiž samy o sobě problémové a i pro člověka-experta je obtížné je správně určit vzhledem k velké podobnosti signálů obou hlásek. Na druhou stranu některé přechody, jako zkoumaný přechod *n-a*, se dají poměrně dobře rozlišit a pokud natrénujeme SPM pouze na tento typ přechodu, dosahujeme pak znatelně lepších výsledků než v případě, že bychom tento přechod opravovali v rámci celé skupiny. Je ovšem nutno přihlídnout k tomu, že zde získané výsledky by mohly být zkreslené vzhledem k nižšímu počtu trénovacích dat.

Nabízelo by se tedy další možný postup, který je nad rámec této práce. Nejprve by se natrénovaly SPM pro všechny konkrétní typy přechodů a provedla by se jejich korekce, pak by se vyhledaly výsledky korekce pro všechny konkrétní přechody v původně opraveném souboru, kde probíhala oprava pomocí SPM natrénovaného na celé hláskové skupiny, a výsledky obou korekcí by se vzájemně porovnály. Pokud by pomocí obecného SPM došlo u některých přechodů (jako typ *n-a*) k horším výsledkům než při opravě vlastním SPM, vyplatilo by se tyto přechody opravovat zvlášť, a naopak, pokud by korekce pomocí vlastního SPM dopadla hůře, vyplatilo by se opravovat tyto přechody pomocí obecného SPM, který by těžil i z informací o dobře identifikovatelných přechodech.

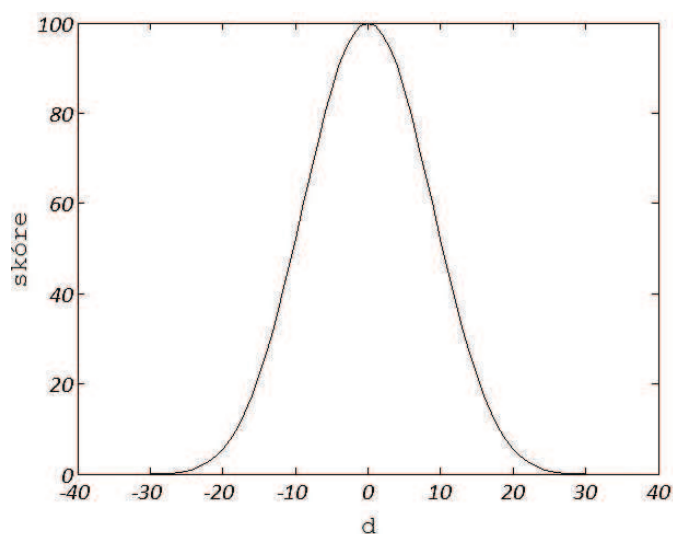
3.1.7 Další experimenty na konkrétních hranicích

Jelikož chování SPM na specifických přechodech byla věc, kterou jsme chtěli více prozkoumat jednak z důvodů naší domněnky o dosažení lepších výsledků a jednak z důvodu nižší výpočetní a časové náročnosti, vyzkoušeli jsme ještě množství dalších experimentů, z nichž některé přinesly zajímavé výsledky. V experimentech je potřeba počítat s tím, že počet trénovacích dat byl menší než při trénování celé skupiny.

Oprava dle velikosti skóre

Myšlenka, která nás napadla v rámci vyhodnocování výsledků předchozích konkrétních přechodů, se týkala kvality opravených hranic. Viděli jsme, že SPM umí podat dobré výsledky – viz předchozí část přechod $n-a$, kde jsme se počtem zlepšených hranic pohybovali okolo 65 % a počet zhoršených hranic byl přitom jen něco okolo 10 %. Bohužel někdy je korekce více na škodu než k užtku, jako tomu bylo u přechodu $J-I$. U těchto obtížně identifikovatelných přechodů by dle výsledků bylo lepší korekci vůbec neprovádět. Co kdybychom ale opravili jen některou část.

Regresní model pracuje na principu přiřazování skóre jednotlivým kandidátům a ten, který je ohodnocený nejvyšším skóre, se nakonec stává novou hranicí. Při trénování byly správné (ručně určené) hranice ohodnoceny skóre 100 a ostatní kandidáti okolo nich dostávali ohodnocení podle vztahu (2.1) v závislosti na maximální odchylce σ_{max} a na vzdálenosti d od správné hranice – čím větší vzdálenost tím nižší skóre (obrázek 3.9):



Obrázek 3.9: Závislost velikosti skóre na vzdálenosti

To znamená, že pokud model předpovídá skóre pro nějakou neznámou hranici, měla by se hodnota skóre blížit hodnotě 100, pokud ona neznámá hranice bude tou správnou a naopak, pokud půjde o špatnou hranici, resp. čím méně se modelu bude „zdát“, že se jedná o správnou hranici, tím by skóre mělo být menší. Vezmeme-li v potaz tento předpoklad, můžeme navrhnout důmyslnější opravování založené na velikosti skóre vybrané hranice.

Pokud bude při opravě na správnou hranici vybrán některý kandidát, který bude mít sice nejvyšší skóre v dané skupině kandidátů, ale jeho skóre bude obecně malé, přihlédneme k tomu, že model „si myslí“ (vzhledem k nízkému ohodnocení), že tato hranice není příliš dobrá a opravu neprovedeme. Důležitým faktorem pro tento proces bude zvolení nějaké prahové hodnoty skóre, podle které budeme rozhodovat, zda opravu ještě provést nebo už ponechat původní hranici. Jelikož víme pouze to, že výsledné skóre by se mělo pohybovat zhruba v intervalu [0; 100], rozhodli jsme se začít na prahové hodnotě padesát a postupně jsme ji snižovali až na prahovou hodnotu dvacet. Výsledky počáteční opravy a výsledky opravy s hodnotou skóre, pro kterou oprava vyšla nejlépe, budou uvedeny v tabulkách dále. Ostatní výsledky lze najít v příloze A (Tab. 1 – 6).

Testování bylo opět provedeno na konkrétních přechodech *J-I* a *n-a* s tím rozdílem, že tentokrát jsme měli k dispozici soubor pouze s těmito daty a byli jsme tedy schopni přesně určit odpovídající si ručně určené a automaticky segmentované hranice. Ve výsledcích jsou tak nyní zahrnuty i hranice s velkými chybami, které dříve nebyly zahrnuty ve vyhodnocení, protože jim nebylo možné správně přiřadit ekvivalentní hranici z ručně segmentovaných dat. Výsledky v tabulkách 3.13 a 3.15 se tedy mírně liší od předchozích (tab. 3.11 a 3.12) z toho důvodu, že nyní jsme do vyhodnocení zahrnuli i velké chyby segmentace, což se projevilo převážně nárůstem celkové segmentační chyby.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I_0	1,5916	1,6871	8,265	12,863	4	19%	7	33%	10	48%	21
J-I_1	0,4494	0,5551	8,424	20,578	7	30%	8	35%	8	35%	23
J-I_2	0,2238	0,2191	12,272	19,677	10	45%	6	27%	6	27%	22
J-I_3	0,2786	0,4390	14,312	16,569	5	19%	7	27%	14	54%	26
J-I_4	1,5490	1,5921	12,539	15,926	8	36%	5	23%	9	41%	22
J-I_5	1,7302	1,7977	13,141	28,760	8	38%	7	33%	6	29%	21
J-I_6	1,3493	1,4583	9,312	14,139	5	24%	5	24%	11	52%	21
J-I_7	1,5757	1,6125	12,571	13,868	7	33%	5	24%	9	43%	21
J-I_8	1,9930	2,0999	11,437	19,068	9	39%	3	13%	11	48%	23
J-I_9	2,4054	2,3214	13,662	10,524	10	48%	6	29%	5	24%	21
PRŮMĚR:	1,3146	1,3782	11,594	17,197	7,3	33%	5,9	27%	8,9	40%	22,1

Tabulka 3.13: Výsledky korekce konkrétní hranice (J-I) se zahrnutím velkých segmentačních chyb

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I_0	1,5916	1,5748	8,265	5,416	4	19%	14	67%	3	14%	21
J-I_1	0,4494	0,4455	8,571	15,156	4	17%	17	74%	2	9%	23
J-I_2	0,2238	0,1673	13,035	15,937	8	36%	11	50%	3	14%	22
J-I_3	0,2786	0,2815	14,312	10,634	5	19%	14	54%	7	27%	26
J-I_4	1,5490	1,4869	11,167	2,421	6	27%	14	64%	2	9%	22
J-I_5	1,7302	1,6317	14,071	0,000	7	33%	14	67%	0	0%	21
J-I_6	1,3493	1,3184	9,312	5,229	5	24%	13	62%	3	14%	21
J-I_7	1,5757	1,5137	13,187	8,531	6	29%	13	62%	2	10%	21
J-I_8	1,9930	2,0001	14,475	15,875	5	22%	13	57%	5	22%	23
J-I_9	2,4054	2,3248	14,553	7,104	7	33%	11	52%	3	14%	21
PRŮMĚR:	1,3146	1,2745	12,095	8,630	5,7	26%	13,4	61%	3,0	14%	22,1

Tabulka 3.14: Konkrétní hranice J-I s opravou podle skóre většího než 30

Postupným snižováním prahu jsme prozkoumali čtyři jeho hodnoty – 50, 40, 30 a 20. Jak se zdá, skóre kandidátů, kteří jsou nakonec vybráni jako správná hranice, není tak vysoké, jak jsme předpokládali. Už totiž pro prahovou hodnotu padesát zůstane valná většina hranic neopravena, což tedy dokazuje, že je jejich skóre nižší než tato hodnota. Nejlepších výsledků jsme pak u přechodu *J-I* dosáhli pro práh třicet (tabulka 3.14), kdy celková průměrná segmentační chyba klesla o 104 ms a o více než polovinu se snížila hodnota průměrného zhoršení. Počet zlepšených hranic se pak pohyboval blízko původního počtu zlepšených hranic a zároveň se počet zhoršených hranic snížil na třetinu.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem Hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
n-a_0	0,8594	0,7887	9,538	8,250	10	56%	5	28%	3	17%	18
n-a_1	2,1194	2,1182	7,983	14,145	9	53%	3	18%	5	29%	17
n-a_2	1,0340	0,8814	12,014	15,563	14	82%	2	12%	1	6%	17
n-a_3	0,9133	0,8209	12,114	14,375	10	59%	5	29%	2	12%	17
n-a_4	0,8114	0,7072	8,689	0,000	12	71%	5	29%	0	0%	17
n-a_5	2,3273	2,2632	8,899	14,201	12	67%	3	17%	3	17%	18
n-a_6	3,0832	2,9848	11,168	11,876	12	71%	2	12%	3	18%	17
n-a_7	2,0192	1,9807	9,611	15,998	9	50%	6	33%	3	17%	18
n-a_8	0,5251	0,4769	8,070	12,251	9	50%	7	39%	2	11%	18
n-a_9	2,7576	2,7174	9,079	12,661	10	59%	3	18%	4	24%	17
PRŮMĚR:	1,6450	1,5739	9,716	11,932	10,7	61%	4,1	24%	2,6	15%	17,4

Tabulka 3.15: Výsledky korekce konkrétní hranice (n-a) se zahrnutím velkých segmentačních chyb

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
n-a_0	0,8594	0,7887	9,538	8,250	10	56%	5	24%	3	17%	18
n-a_1	2,1194	2,0721	7,983	8,209	9	53%	5	29%	3	18%	17
n-a_2	1,0340	0,8989	11,262	0,000	12	71%	5	29%	0	0%	17
n-a_3	0,9133	0,8065	12,114	14,312	10	59%	6	35%	1	6%	17
n-a_4	0,8114	0,7139	8,865	0,000	11	65%	6	35%	0	0%	17
n-a_5	2,3273	2,2384	8,899	8,938	12	67%	4	22%	2	11%	18
n-a_6	3,0832	3,0076	9,570	10,032	10	59%	5	29%	2	12%	17
n-a_7	2,0192	1,9550	8,672	5,188	8	44%	9	50%	1	6%	18
n-a_8	0,5251	0,4596	8,070	7,188	9	50%	8	44%	1	6%	18
n-a_9	2,7576	2,6892	9,158	7,001	9	53%	6	35%	2	12%	17
PRŮMĚR:	1,6450	1,5630	9,413	6,912	10,0	57%	5,9	34%	1,5	9%	17,4

Tabulka 3.16: Konkrétní hranice n-a s opravou podle skóre většího než 20

U druhého zkoumaného přechodu k tak velkým změnám nedošlo. Je to nejspíše díky tomu, že výsledky korekce byly už tak poměrně slušné. Je třeba ale poukázat na to, že zavedení prahu nám opět přineslo snížení celkové průměrné segmentační chyby tentokrát o 82 ms, o necelou polovinu poklesl čas průměrného zhoršení hranic a skoro o polovinu klesl počet špatně opravených hranic.

U přechodu *n-a* jsme také lepší výsledky dostali pro prahovou hodnotu rovnou dvaceti (viz tabulka 3.16), což je nižší hodnota než v předchozím případě. To by tedy mohlo poukazovat na rozdílnou distribuci skóre pro různé typy hranic. Co je také zajímavé, je fakt, že porovnáme-li oba výsledky korekce pro práh padesát (tabulka 3.17), zjistíme, že počet nezměněných hranic u přechodu *n-a* není ani zdaleka tak vysoký jako tomu bylo u přechodu *J-I*. Tímto tedy můžeme potvrdit náš předpoklad, že dobře identifikovatelné hranice mají obecně vyšší skórové ohodnocení a naopak.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I původní	1,3146	1,3782	11,594	17,197	7,3	33%	5,9	27%	8,9	40%	22,1
J-I skóre>50	1,3146	1,3005	7,314	2,047	1,7	8%	19,6	89%	0,8	4%	22,1
n-a původní	1,6450	1,5739	9,716	11,932	10,7	61%	4,1	24%	2,6	15%	17,4
n-a skóre>50	1,6450	1,5847	8,824	6,700	7,9	45%	8,4	48%	1,1	6%	17,4

Tabulka 3.17: Srovnání počtu nezměněných hranic pro práh 50

Přínos této úpravy vyhlíží velmi pozitivně, protože se zdá, že bychom tímto způsobem mohli částečně zamezit chybným opravám.

I když už víme, že pro různé přechody by byla vhodná různá velikost prahové hodnoty, vyzkoušeli jsme opravu podle skóre (opět pro prahy 50, 40, 30, 20) otestovat na celém našem souboru a opravit tímto způsobem všechny kombinace hláskových skupin. Výsledky nejvydařenější korekce ukazuje tabulka 3.18. Ostatní výsledky pro opravu podle skóre pro všechny hláskové skupiny jsou pak v příloze A (Tab. 7 – 9).

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2228	6,149	5,942	10	48%	1	5%	10	48%	21
FRI-NLQ	0,4870	0,3745	9,717	8,019	19	37%	23	45%	9	18%	51
FRI-PAU	0,4619	0,4325	4,092	3,052	37	46%	3	4%	40	50%	80
FRI-VOW	0,6668	0,3566	6,241	4,632	69	42%	68	42%	26	16%	163
NLQ-FRI	0,1255	0,0714	9,085	0,438	6	29%	14	67%	1	5%	21
NLQ-NLQ	0,2144	0,2855	9,115	11,437	6	21%	12	41%	11	38%	29
NLQ-PAU	0,3617	0,2287	7,358	8,213	27	57%	12	26%	8	17%	47
NLQ-VOW	1,9411	1,0534	9,626	7,537	111	49%	90	40%	24	11%	225
PAU-FRI	0,0438	0,0821	1,674	5,368	6	40%	0	0%	9	60%	15
PAU-NLQ	0,1119	0,1079	2,673	2,481	8	53%	0	0%	7	47%	15
PAU-PAU	0,4933	0,5873	7,608	23,123	15	63%	0	0%	9	38%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2128	4,643	5,929	14	64%	4	18%	4	18%	22
PLO-NLQ	0,1874	0,1040	7,037	8,923	22	42%	23	43%	8	15%	53
PLO-PAU	0,1746	0,1637	5,452	4,671	14	42%	5	15%	14	42%	33
PLO-VOW	0,8970	0,4058	6,664	6,495	101	54%	59	31%	28	15%	188
VOW-FRI	0,6728	0,5672	7,938	6,766	44	27%	81	50%	36	22%	161
VOW-NLQ	1,4056	1,0804	9,799	7,919	55	32%	92	53%	27	16%	174
VOW-PAU	1,1957	0,9548	7,151	7,462	89	37%	101	42%	53	22%	243
VOW-VOW	0,0386	0,0386	0,000	0,000	0	0%	5	100%	0	0%	5
Skóre >20	9,9581	7,3302	6,101	6,420	653	42%	593	38%	324	21%	1570
<i>Tab. 3.4</i>	<i>9,9580</i>	<i>7,4205</i>	<i>6,486</i>	<i>7,513</i>	<i>677</i>	<i>43%</i>	<i>542</i>	<i>35%</i>	<i>351</i>	<i>22%</i>	<i>1570</i>

Tabulka 3.18: Výsledky korekce všech kombinací skupin podle skóre většího než 20

Jak ukazuje tabulka, společná prahová hodnota pro opravu všech hláskových skupin nepřináší dříve pozorované změny. Sice poklesne celková segmentační chyba o 90 ms, ale to je tentokrát v porovnání s celkovým počtem opravovaných hranic zanedbatelná hodnota. Dle mého názoru by možná šlo pozorovaných změn dosáhnout, pokud bychom hranice nebrali jako celek, ale vzali bychom postupně jednotlivé typy a pro každý z nich bychom našli vhodnou prahovou hodnotu. Rozsah takových pokusů je už však nad rámec této práce.

Úprava maximální odchylky

V předchozím případě jsme se zabývali velikostí skóre u vybíraných hranic, u kterého jsme si řekli, že jedním z parametrů pro jeho výpočet je i maximální odchylka mezi ruční a automatickou segmentací. V části 3.1.2 jsme zjišťovali, jak se změní výsledky, pokud budeme ignorovat outliery neboli velké chyby segmentace, a maximální odchylku jsme omezili na dvojnásobek směrodatné odchylky počítané v trénovacích datech.

V tomto experimentu jsme se k velikosti maximální odchylky vrátili a pokusili jsme se její velikost ještě vhodněji přizpůsobit. Ze vzorců (2.1) a (3.1) vidíme, že maximální odchylka je využívána jak pro výpočet skóre, tak pro určení velikosti oblasti pro vybírání kandidátů v korekční fázi a její velikost hraje v každém vzorci specifickou roli.

U výpočtu velikosti skóre udává, jak velkou budou mít hranice penalizaci za určitou vzdálenost od správné hranice – čím vyšší bude maximální odchylka, tím obecně vyšší skóre budou hranice dostávat. To tedy znamená, že pro vysokou maximální odchylku budou i vzdálenější hranice dostávat vcelku vysoké skóre a nebude tak moc patrný rozdíl mezi správnou a vzdálenou hranicí. Abychom tomuto zamezili, vyzkoušíme hodnotu maximální odchylky ve vzorci pro skóre nastavit pevně na 30 ms. To je totiž vzdálenost, ze které jsou vybírání kandidáti v trénovací fázi, ze kterých je pak trénován SPM.

U výpočtu velikosti oblasti jsme již maximální odchylku rozebírali a víme, že čím vyšší odchylka bude, tím větší bude oblast pro výběr kandidátů. Tím získáme více možností pro opravu, ale vzhledem k tomu, že se v segmentaci vyskytují převážně menší chyby, jsme v části 3.1.2 odchylku omezovali na dvojnásobnou hodnotu směrodatné odchylky, aby oblast pro výběr nebyla tak rozsáhlá a SPM měl menší šanci na chybnou korekci. A nyní vyzkoušíme ještě větší omezení a to přímo na hodnotu směrodatné odchylky.

V následujících tabulkách 3.19 a 3.20 budou zobrazeny výsledky korekce přechodů $J-I$ a $n-a$ při maximální odchylce, která je rovna:

- 30 milisekundám – ve vzorci pro výpočet skóre
- směrodatné odchylce – ve vzorci pro výpočet velikosti oblasti

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I_0	1,5916	1,6610	8,350	12,354	5	24%	7	33%	9	43%	21
J-I_1	0,4494	0,5138	8,110	14,133	6	26%	9	39%	8	35%	23
J-I_2	0,2238	0,1934	11,503	14,115	10	45%	6	27%	6	27%	22
J-I_3	0,2786	0,3777	10,891	13,700	6	23%	8	31%	12	46%	26
J-I_4	1,5490	1,6104	11,036	13,859	7	32%	5	23%	10	45%	22
J-I_5	1,7302	1,7360	13,141	18,489	8	38%	7	33%	6	29%	21
J-I_6	1,3493	1,4251	9,312	11,125	5	24%	5	24%	11	52%	21
J-I_7	1,5757	1,5935	11,660	11,042	7	33%	5	24%	9	43%	21
J-I_8	1,9930	2,1066	8,891	16,790	8	35%	4	17%	11	48%	23
J-I_9	2,4054	2,3427	11,537	10,524	10	48%	6	29%	5	24%	21
PRŮMĚR:	1,3146	1,3560	10,443	13,613	7,2	33%	6,2	28%	8,7	39%	22,1
Tab. 3.13	1,3146	1,3782	11,594	17,197	7,3	33%	5,9	27%	8,9	40%	22,1

 Tabulka 3.19: Výsledek korekce přechodu *J-I* s upravenými hodnotami max. odchylky

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
n-a_0	0,8594	0,7987	10,089	10,000	7	39%	10	56%	1	6%	18
n-a_1	2,1194	2,0797	8,854	10,016	9	53%	4	24%	4	24%	17
n-a_2	1,0340	0,9275	9,532	7,875	12	71%	4	24%	1	6%	17
n-a_3	0,9133	0,8211	9,951	7,312	10	59%	6	35%	1	6%	17
n-a_4	0,8114	0,7147	8,797	0,000	11	65%	6	35%	0	0%	17
n-a_5	2,3273	2,2557	8,856	8,437	10	56%	6	33%	2	11%	18
n-a_6	3,0832	3,0075	8,636	9,313	12	71%	2	12%	3	18%	17
n-a_7	2,0192	1,9523	9,611	9,813	7	39%	9	50%	2	11%	18
n-a_8	0,5251	0,4596	8,070	7,188	9	50%	8	44%	1	6%	18
n-a_9	2,7576	2,7214	7,438	11,656	8	47%	7	41%	2	12%	17
PRŮMĚR:	1,6450	1,5738	8,983	8,161	9,5	55%	6,2	36%	1,7	10%	17,4
Tab. 3.15	1,6450	1,5739	9,716	11,932	10,7	61%	4,1	24%	2,6	15%	17,4

 Tabulka 3.20: : Výsledek korekce přechodu *n-a* s upravenými hodnotami max. odchylky

Tyto změny nepřinesly do výsledku příliš mnoho nového. V případě přechodu *J-I* se poměr zlepšených, nezměněných a zhoršených hranic prakticky nezměnil, avšak klesla celková průměrná segmentační chyba, což bylo způsobeno sníženou hodnotou průměrného zhoršení. U přechodu *n-a* jsme pak zaznamenali jednak pokles průměrné hodnoty zhoršení hranic a také snížení počtu zhoršených, ale i zlepšených hranic a to zhruba o 5 %. Redukce maximální odchylky měla tedy za následek především snížení průměrné chyby, které se SPM dopouští, a to proto, že se snížil počet vzdálenějších kandidátů a pozměnilo se jejich ohodnocení.

Další konkrétní přechody

V předchozích částech byly pro testování vybrány dvě hranice mezi konkrétními hláskami, které měly nejpočetnější zastoupení. Oba přechody spadaly do skupiny NLQ-VOW, kde se oprava vcelku vydařila – 50 % zlepšených, 37 % nezměněných a 13 % zhoršených hranic. Pro větší variabilitu jsme se tak rozhodli ještě vyzkoušet další přechody. Z původních skupin jsme vybrali čtyři skupiny: *FRI-VOW*, *VOW-PAU*, *VOW-NLQ* a *VOW-FRI*, které patřily mezi největší a které měly největší podíl špatně opravených hranic. Z těchto vybraných skupin jsme vyčlenili nejvíce zastoupené typy přechodů: *v-i*, *i-pau*, *e-n*, *o-v*, pro každou z nich jsme opět za pomoci cross-validace natrénovali SPM a provedli korekci.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
<i>v-i_0</i>	0,0560	0,0176	9,413	8,688	5	45%	5	45%	1	9%	11
<i>v-i_1</i>	0,0319	0,0150	11,593	6,249	2	22%	6	66%	1	11%	9
<i>v-i_2</i>	0,0760	0,0527	8,157	9,303	4	36%	6	54%	1	9%	11
<i>v-i_3</i>	0,0648	0,0373	9,147	0,000	3	27%	8	72%	0	0%	11
<i>v-i_4</i>	0,0760	0,0523	8,125	8,813	4	36%	6	54%	1	9%	11
<i>v-i_5</i>	0,0531	0,0241	8,828	6,249	4	44%	4	44%	1	11%	9
<i>v-i_6</i>	0,0331	0,0426	7,937	8,319	3	30%	3	30%	4	40%	10
<i>v-i_7</i>	0,0703	0,0516	9,335	9,303	3	30%	6	60%	1	10%	10
<i>v-i_8</i>	0,0391	0,0333	7,729	8,688	3	27%	6	54%	2	18%	11
<i>v-i_9</i>	0,0586	0,0237	10,375	3,312	4	36%	5	45%	2	18%	11
PRŮMĚR:	0,0559	0,0350	9,064	6,892	3,5	33,7%	5,5	52,9%	1,4	13,5%	10,4
<i>FRI-VOW</i>	0,6668	0,3575	6,309	4,951	71	44%	64	39%	28	17%	163

Tabulka 3.21: Výsledky korekce přechodu *v-i*

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
<i>i-pau_0</i>	0,2699	0,1958	8,639	4,172	11	68%	0	0%	5	31%	16
<i>i-pau_1</i>	0,2407	0,2168	7,783	10,797	10	66%	0	0%	5	33%	15
<i>i-pau_2</i>	0,2606	0,2541	9,202	9,585	8	53%	0	0%	7	46%	15
<i>i-pau_3</i>	0,3255	0,3049	7,580	7,924	9	60%	0	0%	6	40%	15
<i>i-pau_4</i>	0,2508	0,2101	6,290	7,117	11	73%	0	0%	4	26%	15
<i>i-pau_5</i>	0,1549	0,1116	10,833	4,059	7	46%	0	0%	8	53%	15
<i>i-pau_6</i>	0,1597	0,1704	5,693	6,404	6	42%	1	7%	7	50%	14
<i>i-pau_7</i>	0,1632	0,1858	3,499	7,215	8	53%	0	0%	7	46%	15
<i>i-pau_8</i>	0,3151	0,2337	9,287	3,833	10	71%	1	7%	3	21%	14
<i>i-pau_9</i>	0,1975	0,1844	10,129	8,254	7	50%	0	0%	7	50%	14
PRŮMĚR:	0,2338	0,2068	7,893	6,936	8,7	58,8%	0,2	1,4%	5,9	39,9%	14,8
<i>VOW-PAU</i>	1,1957	0,9455	7,175	7,462	90	37%	100	41%	53	22%	243

Tabulka 3.22: Výsledky korekce přechodu *i-pau*

Všechny čtyři přechody v tabulkách 3.21 až 3.24 mají ve výsledku sníženou celkovou průměrnou segmentační chybu, ale znatelnější rozdíl byl pouze u prvních dvou přechodů. Ve druhých dvou přechodech jsme zaznamenali velké procento nezměněných hranic, což by mohlo být tím, že hranice jsou již správně určeny a model je tak ponechal. To, že by u těchto dvou přechodů mohla být většina hranic korektních, usuzujeme z nízké celkové segmentační chyby. Pro srovnání je v posledním řádku každé tabulky uveden výsledek korekce příslušné hláskové skupiny, ze které přechod pochází.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
e-n_0	0,0409	0,0361	10,844	8,428	2	11%	13	76%	2	11%	17
e-n_1	0,0597	0,0480	10,282	8,812	2	11%	14	82%	1	5%	17
e-n_2	0,0751	0,0490	8,687	0,000	3	17%	14	82%	0	0%	17
e-n_3	0,0542	0,0469	8,208	8,687	3	17%	12	70%	2	11%	17
e-n_4	0,0460	0,0466	8,626	8,937	2	11%	13	76%	2	11%	17
e-n_5	0,0262	0,0272	7,062	8,045	1	6%	14	87%	1	6%	16
e-n_6	0,0507	0,0419	8,438	8,045	2	10%	16	84%	1	5%	19
e-n_7	0,0365	0,0551	0,000	9,312	0	0%	16	88%	2	11%	18
e-n_8	0,0474	0,0448	8,458	11,407	3	17%	12	70%	2	11%	17
e-n_9	0,0305	0,0476	0,000	8,563	0	0%	15	88%	2	11%	17
PRŮMĚR:	0,0467	0,0443	7,060	8,024	1,8	10,5%	13,9	80,8%	1,5	8,7%	17,2
VOW-NLQ	1,4056	1,1019	9,606	8,845	62	36%	79	45%	33	19%	174

Tabulka 3.23: Výsledky korekce přechodu e-n

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
o-v_0	0,0393	0,0475	0,000	8,250	0	0%	11	91%	1	8%	12
o-v_1	0,0412	0,0541	6,937	9,938	1	7%	11	78%	2	14%	14
o-v_2	0,0485	0,0487	10,906	7,354	2	14%	9	64%	3	21%	14
o-v_3	0,0244	0,0258	6,937	8,312	1	7%	12	85%	1	7%	14
o-v_4	0,0603	0,0401	10,094	0,000	2	14%	12	85%	0	0%	14
o-v_5	0,0273	0,0154	8,750	5,625	2	12%	13	81%	1	6%	16
o-v_6	0,0326	0,0543	11,125	8,788	2	13%	8	53%	5	33%	15
o-v_7	0,0338	0,0285	8,938	6,281	2	15%	9	69%	2	15%	13
o-v_8	0,0607	0,0313	11,422	8,125	4	28%	8	57%	2	14%	14
o-v_9	0,0457	0,0649	8,843	7,375	2	13%	8	53%	5	33%	15
PRŮMĚR:	0,0414	0,0411	8,395	7,005	1,8	12,8%	10,1	71,6%	2,2	15,6%	14,1
VOW-FRI	0,6728	0,5524	8,115	6,691	47	29%	75	47%	39	24%	161

Tabulka 3.24: Výsledky korekce přechodu o-v

Ze všech těchto výsledků není žádný podobný výsledku své skupiny. Zdá se tedy, že pokud bychom měli dostatek dat a provedli korekci všech přechodů v hláskových skupinách za pomoci specifických SPM, jednotlivé výsledky by se pravděpodobně značně lišily. Díky tomu a dalším znalostem bychom mohli odhadnout, které hranice půjdou dobře automaticky opravovat.

Ve zkoumaných přechodech jsme měli přechod *o-v*, přičemž obě jeho hlásky jsou znělé. Ze všech předchozích znalostí, které jsme doposud získali – *rozdílná oprava u různých typů hranic, lepší oprava dobře rozlišitelných hranic* a to, že *výsledky jednotlivých přechodů neodpovídají výsledkům celé skupiny* – můžeme vyvodit, že pokud v přechodu *o-v* zaměníme znělé „v“ za neznělé „s“, dostaneme lépe rozlišitelnou hranici a její korekci pomocí specifického přechodu dostaneme lepší výsledky, o čemž se můžeme přesvědčit v následující tabulce.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno	nezměněno	zhoršeno				
o-s_0	0,0679	0,0456	3,189	0,000	7	46%	8	53%	0	0%	15
o-s_1	0,0443	0,0439	2,363	6,750	3	18%	12	75%	1	6%	16
o-s_2	0,0789	0,0733	2,178	3,157	4	23%	12	70%	1	5%	17
o-s_3	0,0680	0,0721	3,478	7,167	5	31%	8	50%	3	18%	16
o-s_4	0,0733	0,0550	3,803	8,250	7	43%	8	50%	1	6%	16
o-s_5	0,0928	0,0656	4,982	2,551	7	38%	8	44%	3	16%	18
o-s_6	0,0809	0,0522	3,579	0,000	8	50%	8	50%	0	0%	16
o-s_7	0,0731	0,0453	6,466	4,533	5	33%	9	60%	1	6%	15
o-s_8	0,0673	0,0586	4,014	5,706	5	29%	10	58%	2	11%	17
o-s_9	0,0349	0,0306	2,795	3,413	4	26%	9	60%	2	13%	15
PRŮMĚR:	0,0681	0,0542	3,685	4,153	5,5	34,2%	9,2	57,1%	1,4	8,7%	16,1
<i>VOW-FRI</i>	<i>0,6728</i>	<i>0,4646</i>	<i>8,210</i>	<i>6,974</i>	<i>50</i>	<i>31,1%</i>	<i>82</i>	<i>50,9%</i>	<i>29</i>	<i>18,0%</i>	<i>161</i>

Tabulka 3.25: Výsledky korekce přechodu *o-s*

Snížení počtu složek cross-validace

Nejnáročnější částí při automatické korekci je nalezení vhodných parametrů pro SPM, které zabere značnou dávku času i počítačového výkonu. Parametry se hledají pomocí metody grid search, která využívá deseti složkovou cross-validaci, takže pro každou kombinaci parametrů z parametrického prostoru se desetkrát natrénuje SPM a následně se zprůměrovaný výsledek porovná s výsledkem ostatních kombinací.

Jednou možností jak tento proces urychlit je omezení prostoru parametrů, což by se ovšem mohlo negativně projevit na výsledku korekce, protože by nemusely být nalezeny optimální parametry. Druhá možnost spočívá ve zredukování počtu složek cross-validace. Počet složek cross-validace zodpovídá za robustnější interpretaci výsledků a při omezení jejich počtu z deseti na pět bychom ve výsledcích neměli zaznamenat velkou změnu a přitom bychom dosáhli znatelného zrychlení celého procesu, protože počet natrénování jednotlivých modelů při hledání parametrů by se snížil na polovinu. Snížení počtu složek jsme testovali na přechodech *J-I* a *n-a* (tabulka 3.26).

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I 5-složek	1,3146	1,3779	11,594	17,142	7,3	33%	5,9	27%	8,9	40%	22,1
J-I 10-složek	1,3146	1,3782	11,594	17,197	7,3	33%	5,9	27%	8,9	40%	22,1
n-a 5-složek	1,6450	1,5763	9,611	11,857	10,7	61%	4,1	24%	2,6	15%	17,4
n-a 10-složek	1,6450	1,5739	9,716	11,932	10,7	61%	4,1	24%	2,6	15%	17,4

Tabulka 3.26: Srovnání výsledků při použití pěti a deseti složkové cross-validace

Po redukci počtu složek zůstaly výsledky korekce pro oba typy hranic prakticky nezměněny – pozorované odchylky se pohybují v řádech tisícín a desítitísícín sekundy. To nám umožňuje při trénování dalších SPM využívat pouze pěti složkové cross-validace a snížit tak potřebný výpočetní čas.

3.1.8 Finální korekce se zavedením všech experimentů

Provedli jsme značné množství experimentů a úprav, jejichž dílčí výsledky jsme shrnuli v předchozích částech, a nyní se podíváme, jaký vliv budou mít na celý automaticky segmentovaný soubor. Do korekčního programu, který pracuje na bázi regrese, jsme tedy od části 3.1.3 (Tabulka 3.4: Výsledky korekce všech typů hranic) zahrnuli následující:

- *opravu podle skóre*
- *úpravu maximální odchylky*
- *snížení počtu složek cross-validace*

V tabulce 3.27 je finální podoba korekce jako regresní úlohy.

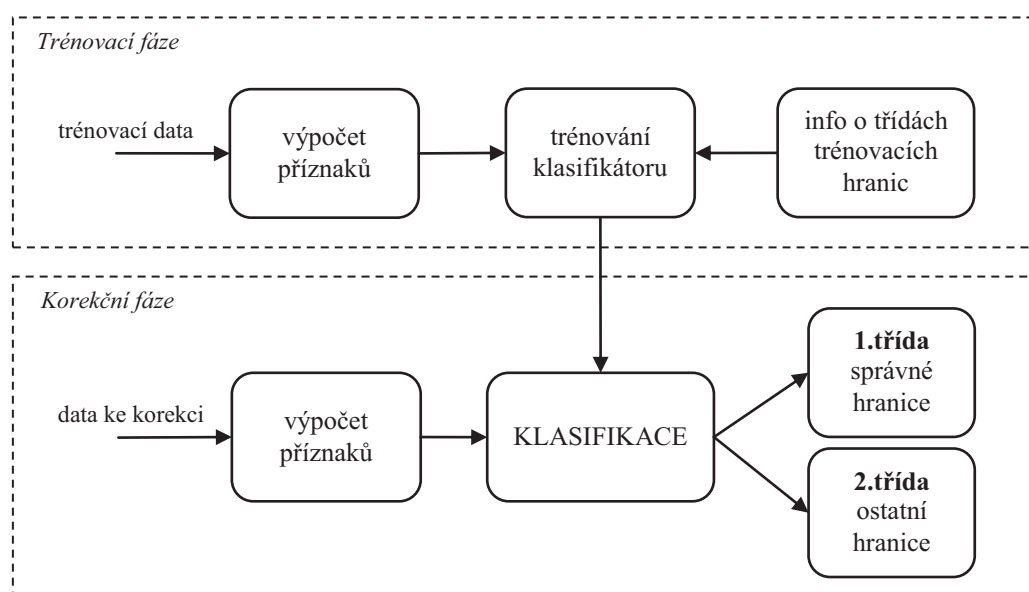
Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2227	6,149	6,588	10	47%	2	9%	9	42%	21
FRI-NLQ	0,4870	0,4498	6,835	8,019	16	31%	26	50%	9	17%	51
FRI-PAU	0,4619	0,4544	3,508	3,057	37	46%	3	3%	40	50%	80
FRI-VOW	0,6668	0,3711	5,995	4,667	68	41%	71	43%	24	14%	163
NLQ-FRI	0,1255	0,0841	6,898	0,000	6	28%	15	71%	0	0%	21
NLQ-NLQ	0,2144	0,2346	9,063	7,277	5	17%	15	51%	9	31%	29
NLQ-PAU	0,3617	0,2129	7,114	6,177	27	57%	13	27%	7	14%	47
NLQ-VOW	1,9411	1,1901	8,638	6,699	104	46%	99	44%	22	9%	225
PAU-FRI	0,0438	0,0832	1,829	4,851	5	33%	0	0%	10	66%	15
PAU-NLQ	0,1119	0,1119	2,673	3,046	8	53%	0	0%	7	46%	15
PAU-PAU	0,4933	0,5602	5,999	15,081	14	58%	0	0%	10	41%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2128	4,643	5,929	14	63%	4	18%	4	18%	22
PLO-NLQ	0,1874	0,0720	6,866	7,191	21	39%	28	52%	4	7%	53
PLO-PAU	0,1746	0,1817	3,684	4,191	14	42%	5	15%	14	42%	33
PLO-VOW	0,8970	0,3646	6,683	6,782	102	54%	64	34%	22	11%	188
VOW-FRI	0,6728	0,5084	7,939	7,111	44	27%	91	56%	26	16%	161
VOW-NLQ	1,4056	1,0974	9,545	7,678	54	31%	93	53%	27	15%	174
VOW-PAU	1,1957	0,8872	7,120	6,490	88	36%	106	43%	49	20%	243
VOW-VOW	0,0386	0,0386	0,000	0,000	0	0%	5	100%	0	0%	5
	9,9580	7,3377	5,559	5,542	637	40,6%	640	40,8%	293	18,7%	1570

Tabulka 3.27: Finální korekce všech typů hranic pomocí regresních modelů

Z celkového počtu dvaceti typů hranic byl výsledek segmentace u jedenácti z nich zlepšen, u sedmi zhoršen a u dvou se nezměnil. Celkovou segmentační chybu to ovlivnilo tak, že se mírně zlepšila, ale znatelnější výsledky můžeme pozorovat spíše na ostatních hodnotách. Průměrné zlepšení a zhoršení hranic se vyrovnalo oproti předchozímu, kdy byla průměrná chyba zhoršení větší. Nejznatelnější změny pak přinesl počet hranic. Byl snížen počet zlepšených i zhoršených hranic a přímo úměrně tomu vzrostl počet nezměněných hranic. Zavedením úprav z experimentů jsme sice snížili počet zlepšených hranic o 40 (2,5 %), ale zároveň jsme tak omezili i počet zhoršených hranic a to o 58 (3,7 %). Tím jsme se dostali z celkových 1570 hranic na 637 zlepšených (40,6 %), 293 zhoršených (18,7 %) a 640 nezměněných hranic (40,8 %). To se nám jeví jako lepší výsledek než v původním případě především vzhledem k většímu poklesu hranic, u kterých došlo ke zhoršení.

3.2 Korekce jako klasifikační úloha

Druhým možným přístupem ke korekci, který nyní rozebereme, je rozhodování pomocí klasifikátorů. Na obrázku 3.10 je schéma klasifikační úlohy, kterou budeme řešit. Princip se velmi podobá regresní úloze, ale na rozdíl od regrese, kde každý kandidát dostal ohodnocující skóre, podle kterého bylo rozhodnuto, zda se jedná o správnou hranici či nikoliv, zde budeme hranice a jejich kandidáty řadit (klasifikovat) do dvou tříd, to jest na správné a na všechny ostatní hranice.



Obrázek 3.10: Schéma klasifikační úlohy

Úkol všech klasifikátorů, které budeme testovat, je tedy zřejmý. Opět budeme pracovat v trénovací a korekční fázi. Nejprve provedeme natrénování klasifikátorů, to znamená, že pro každý klasifikátor budou hledány takové parametry, aby přesnost klasifikace na trénovacích datech byla co největší. Poté natrénovaný klasifikátor přejde do druhé fáze, kde mu budou předkládána data, která chceme opravit, a on je bude na základě svého natrénování rozdělovat na správné a všechny ostatní (špatné) hranice.

Problémem klasifikace je, že klasifikátor ne vždy zařadí hranici do té třídy, do které patří. Může se stát, že v oblasti okolo hranice určené automatem, ze které se vybírají kandidáti na správnou hranici, se objeví dva a více kandidátů, které klasifikátor zařadí do třídy „správná hranice“ anebo naopak, že žádná hranice nebude označena jako správná. S oběma těmito případy jsme se v našich úlohách setkali a řešili jsme je následujícím způsobem.

Nenalezení správné hranice

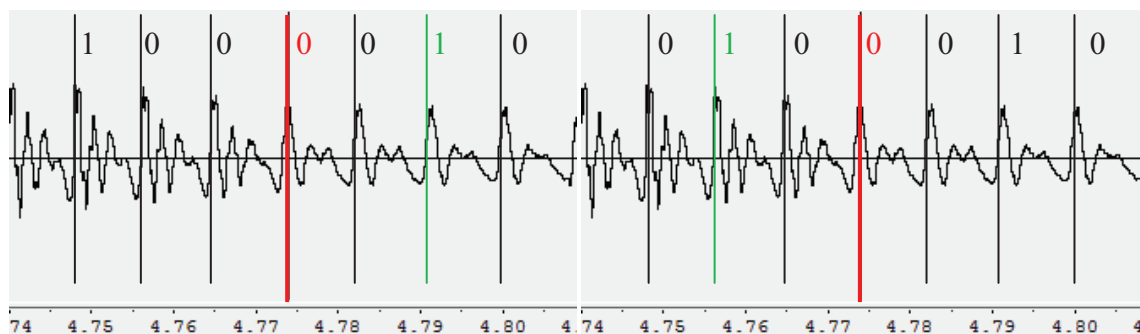
Začneme s tím jednodušším a to je případ, kdy klasifikátor neoznačí žádnou hranici v dané oblasti jako tu správnou. Tento případ nastává, pokud se příznaky žádného kandidáta nepodobají příznakům správných hranic. Klasifikátor pak nemá jinou možnost, než zařadit všechny kandidáty do třídy „ostatní hranice“. Pokud se tak stane, nemáme jak určit správnou hranici, protože všichni kandidáti jsou ve stejné třídě a nám nezbude nic jiného než ponechat stávající hranici.

Nalezení dvou a více správných hranic

Složitější případ nastává, když je více hranic zařazeno mezi „správné hranice“. Klasifikátor značí správné hranice hodnotou 1 a ostatní hranice hodnotou 0. Pokud se nám tedy vyskytne v určité oblasti okolo opravované hranice více kandidátů, kteří byli klasifikátorem označeni jako 1, musíme se rozhodnout, kterého vybereme jako náhradu původní hranice, tj. hranice nalezené automatickou segmentací řečového signálu, kterou chceme opravit. Při vybírání nové hranice postupujeme od původní hranice *souměrně na obě strany* a zjišťujeme, jakou hodnotu dostali okolní kandidáti.

Pokud je původní hranice označena jako 1, nemusíme nic zkoumat. I když se v oblasti nacházejí jiní kandidáti označeni jako 1, vycházíme z předpokladu, že algoritmus, který dělal automatickou segmentaci, ji udělal správně a ponecháme tedy původní hranici.

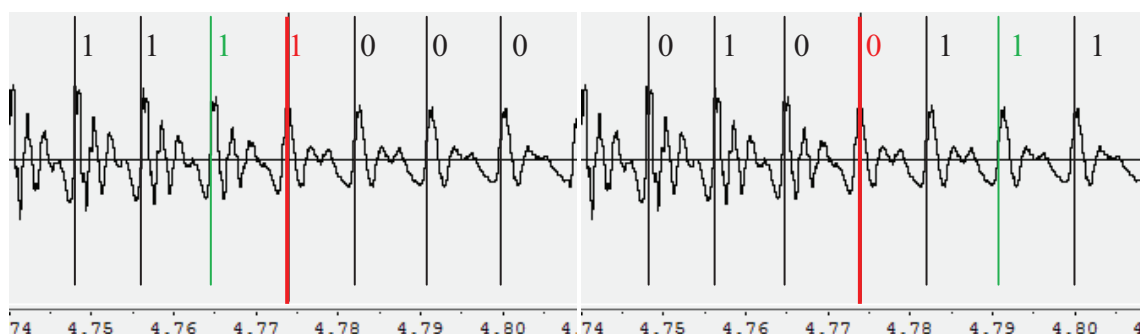
Pokud je původní hranice označena jako 0 a nalezneme více kandidátů s označením 1, bude za novou hranici dosazen kandidát, který je blíže k té původní. V případě, že jsou dva kandidáti stejně vzdáleni, vybere se ten, který předchází původní hranici (obrázek 3.11).



Obrázek 3.11: Způsob vybírání nových hranic (červeně původní, zeleně vybraná)

Nalezení jednoho a více shluků správných hranic

Nejkomplikovanější rozhodování máme tehdy, vyskytne-li se v jedné oblasti více jedniček za sebou a vytvoří se nám tak jeden či více shluků. V tu chvíli postupujeme tak, že pokud je původní hranice označena jako 1 a je součástí shluku, vybere se jako nová hranice jeho střed; je-li ve shluku sudý počet hranic, vybere se ta středová, která je blíže k původní. Pokud je původní hranice označena jako 0, pak opět hledáme jedničky souměrně na obě strany a první jednička, na kterou narazíme, je vybrána jako nová hranice. V případě, že je tato jednička součástí shluku, vybere se opět jeho střed, resp. bližší hranice k té původní, jak je tomu na obrázku 3.12.



Obrázek 3.12: Způsob vybírání nových hranic u shluků (červeně původní, zeleně vybraná)

K vytvoření shluku nejspíše dochází, když má více kandidátů za sebou příznaky podobné příznakům správných hranic. Ze získaných výsledků však bohužel nemůžeme říci, že by se tento jev vytváření shluků vyskytoval pouze okolo správných hranic, jejichž okolní hranice by právě také dostaly ohodnocení 1.

3.2.1 Klasifikace pomocí SVM

První testovaný klasifikátor, který jsme vytvořili pomocí nástrojů support vector machine, budeme označovat jako SVC (z angl. *Support vector classifier*). Chceme zjistit, jak si povede v porovnání s regresním přístupem, který využil metod postavených na stejném základu. Jako u regresního přístupu bylo i u tohoto klasifikátoru použito k natrénování parametrů c a g , které jsme opět hledali pomocí metody grid search.

Dosažené výsledky z korekce všech hláskových skupin jsou v tabulce 3.28. Již na první pohled vidíme výraznou změnu oproti regresnímu přístupu, která je nejvíce patrná u počtu zlepšených, nezměněných a zhoršených hranic. SVC opravil pouze necelou třetinu hranic, přesněji 27,1 %, z toho 20,9 % hranic posunul blíže ke správné hranici a 6,2 % zhoršil. Průměrné zhoršení pak bylo s hodnotou 5,46 ms o cca jednu milisekundu horší než průměrné zlepšení. Celková chyba segmentace tak oproti regresnímu přístupu vzrostla zhruba o jednu vteřinu.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2132	5,845	0,000	2	9%	19	90%	0	0%	21
FRI-NLQ	0,4870	0,4798	10,088	21,625	5	9%	44	86%	2	3%	51
FRI-PAU	0,4619	0,4619	0,000	0,000	0	0%	80	100%	0	0%	80
FRI-VOW	0,6668	0,5115	4,788	4,629	45	27%	105	64%	13	7%	163
NLQ-FRI	0,1255	0,1282	3,531	9,750	2	9%	18	85%	1	4%	21
NLQ-NLQ	0,2144	0,2144	0,000	0,000	0	0%	29	100%	0	0%	29
NLQ-PAU	0,3617	0,2886	8,607	9,474	14	29%	28	59%	5	10%	47
NLQ-VOW	1,9411	1,4940	9,515	8,179	53	23%	165	73%	7	3%	225
PAU-FRI	0,0438	0,0523	0,352	2,221	1	6%	10	66%	4	26%	15
PAU-NLQ	0,1119	0,1137	1,976	1,882	1	6%	12	80%	2	13%	15
PAU-PAU	0,4933	0,4893	3,988	0,000	1	4%	23	95%	0	0%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2556	0,000	0,788	0	0%	20	90%	2	9%	22
PLO-NLQ	0,1874	0,0763	7,410	7,437	16	30%	36	67%	1	1%	53
PLO-PAU	0,1746	0,1844	0,000	9,813	0	0%	32	96%	1	3%	33
PLO-VOW	0,8970	0,4077	6,490	6,255	86	45%	91	48%	11	5%	188
VOW-FRI	0,6728	0,6287	8,717	7,444	23	14%	117	72%	21	13%	161
VOW-NLQ	1,4056	1,3589	8,797	9,813	12	6%	156	89%	6	3%	174
VOW-PAU	1,1957	0,9145	7,445	9,889	67	27%	154	63%	22	9%	243
VOW-VOW	0,0386	0,0386	0,000	0,000	0	0%	5	100%	0	0%	5
	9,9580	8,3116	4,377	5,460	328	20,9%	1144	72,9%	98	6,2%	1570

Tabulka 3.28: Výsledky korekce pomocí SVC

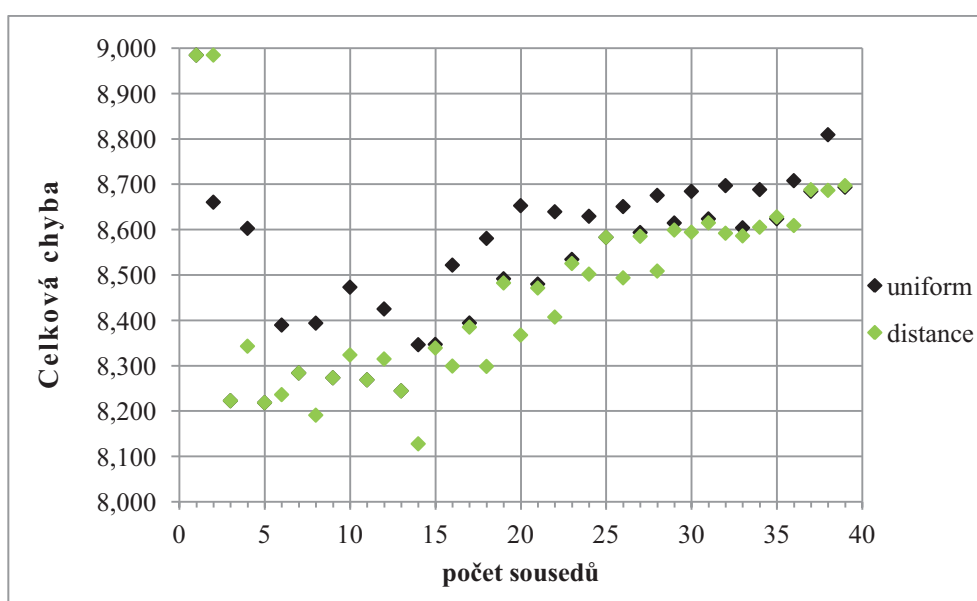
Je dobré si povšimnout, že zde více než u regresního přístupu dochází k případům, kdy se nějaký typ hranic jenlepší nebo naopak jen zhorší. To nejspíše souvisí se samotnou funkcí SVC. Jak si můžeme povšimnout, většina hranic (72,9 %) zůstala po opravě beze změny. Bylo vyzorováno, že se tak děje z důvodu nenalezení správné hranice. Ve většině případů totiž SVC nedokázal zařadit žádnou hranici z dané oblasti do třídy správných hranic, a proto nedošlo k žádné opravě. Je možné, že tento typ klasifikátoru není

pro tento typ úlohy vhodným kandidátem, protože nedokáže příliš dobře rozlišit správné hranice. Výhodou však je, že téměř 80 % všech opravených hranic bylo opraveno správně.

3.2.2 Algoritmus k-nejbližších sousedů

Druhý testovaný klasifikátor pracuje na principu porovnávání neznámých dat s určitým počtem již známých vzorů (sousedů), které jsou neznámému vzorku nejbližší. Tyto známé vzorky byly klasifikátoru předloženy již v trénovací fázi společně s informací o tom, do které třídy tyto vzory náleží.

Základními parametry pro nastavení klasifikátoru je počet sousedů (k), se kterými budeme neznámý vzorek porovnávat, a váha (*weights*) určující, jestli bude mít všech k -sousedů stejný vliv (*uniform*) na výsledek klasifikace anebo jestli bližší sousedé budou mít větší vliv (*distance*) než ti vzdálenější. Výběr parametrů byl proveden postupným otestováním jejich kombinací. Pro každé nastavení parametrů byl opraven soubor pro korekci a byla zjištěna celková chyba segmentace. Závislost této chyby na počtu sousedů a váze je znázorněna na obrázku 3.13 a výsledky korekce, která dopadla nejlépe, jsou v tabulce 3.29.



Obrázek 3.13: Testování parametrů pro algoritmus k-nejbližších sousedů

Nejmenší celková chyba korekce byla dosažena pro k rovno 14. Celkově korekce dopadla lépe než v případě SVC. Zvedla se hodnota průměrného zlepšení a na polovinu klesla

hodnota průměrného zhoršení. Opět vidíme velký počet nezměněných hranic, který ale v tomto případě není způsoben nenalezením žádné správné hranice v dané oblasti. U tohoto klasifikátoru jsme se naopak potýkali s problémem vybrat tu správnou hranici z většího množství hranic. Stávalo se totiž, že v jedné oblasti bylo do třídy „správná hranice“ zařazeno více hranic a proto jsme při rozhodování o tom, která z nich je ta správná hranice, postupovali podle algoritmu uvedeného v začátku části 3.2.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2041	6,938	0,000	3	14%	18	85%	0	0%	21
FRI-NLQ	0,4870	0,4255	10,250	0,000	6	11%	45	88%	0	0%	51
FRI-PAU	0,4619	0,4497	9,592	1,399	2	2%	73	91%	5	6%	80
FRI-VOW	0,6668	0,4695	5,768	4,497	42	25%	111	68%	10	6%	163
NLQ-FRI	0,1255	0,0980	9,208	0,126	3	14%	17	80%	1	4%	21
NLQ-NLQ	0,2144	0,2144	0,000	0,000	0	0%	29	100%	0	0%	29
NLQ-PAU	0,3617	0,2815	9,287	10,403	12	25%	32	68%	3	6%	47
NLQ-VOW	1,9411	1,4673	9,826	3,845	49	21%	174	77%	2	0%	225
PAU-FRI	0,0438	0,0436	0,352	0,102	1	6%	13	86%	1	6%	15
PAU-NLQ	0,1119	0,1077	2,992	1,802	2	13%	12	80%	1	6%	15
PAU-PAU	0,4933	0,4901	3,197	0,000	1	4%	23	95%	0	0%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2548	0,000	0,762	0	0%	21	95%	1	4%	22
PLO-NLQ	0,1874	0,0717	7,717	0,000	15	28%	38	71%	0	0%	53
PLO-PAU	0,1746	0,1746	0,000	0,000	0	0%	33	100%	0	0%	33
PLO-VOW	0,8970	0,4796	6,209	7,307	79	42%	99	52%	10	5%	188
VOW-FRI	0,6728	0,6459	8,361	7,721	18	11%	127	78%	16	9%	161
VOW-NLQ	1,4056	1,3399	8,819	6,875	9	5%	163	93%	2	1%	174
VOW-PAU	1,1957	0,8714	7,279	9,432	64	26%	164	67%	15	6%	243
VOW-VOW	0,0386	0,0386	0,000	0,000	0	0%	5	100%	0	0%	5
	9,9580	8,1279	5,290	2,714	306	19,5%	1197	76,2%	67	4,3%	1570

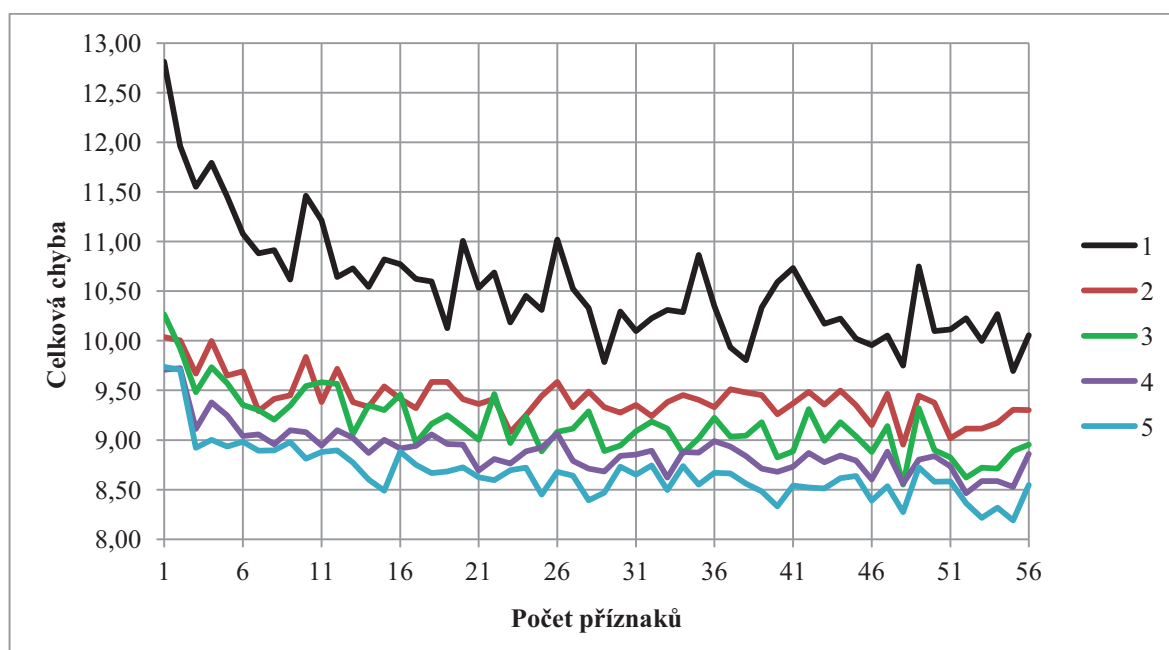
Tabulka 3.29: Výsledky korekce pomocí klasifikátoru nejbližších sousedů

Tento klasifikační přístup má velmi malou chybovost, která se zde pohybovala těsně nad hranicí čtyř procent, a díky klasifikaci většího množství hranic mezi správné hranice je zde také potenciál k lepšímu výsledku při nalezení jiného vhodného přístupu k vybírání hranic.

3.2.3 Extremely Randomized Trees

Třetím klasifikátorem bylo sjednocení více rozhodovacích stromů, které se dohromady podílely na konečném výsledku. Klasifikátor vytvořený metodou Extremely Randomized Trees (*ERT*) se nastavuje pomocí dvou základních parametrů, kterými jsou počet rozhodovacích stromů (*estimators*) a počet příznaků (*features*) využitých při dělení uzlů v rozhodovacích stromech. Pro optimální nastavení klasifikátoru jsme provedli ještě rozsáhlejší hledání parametrů než v případě klasifikace pomocí nejbližších sousedů, protože výsledky se pro rostoucí počet estimátorů neustále zlepšovaly.

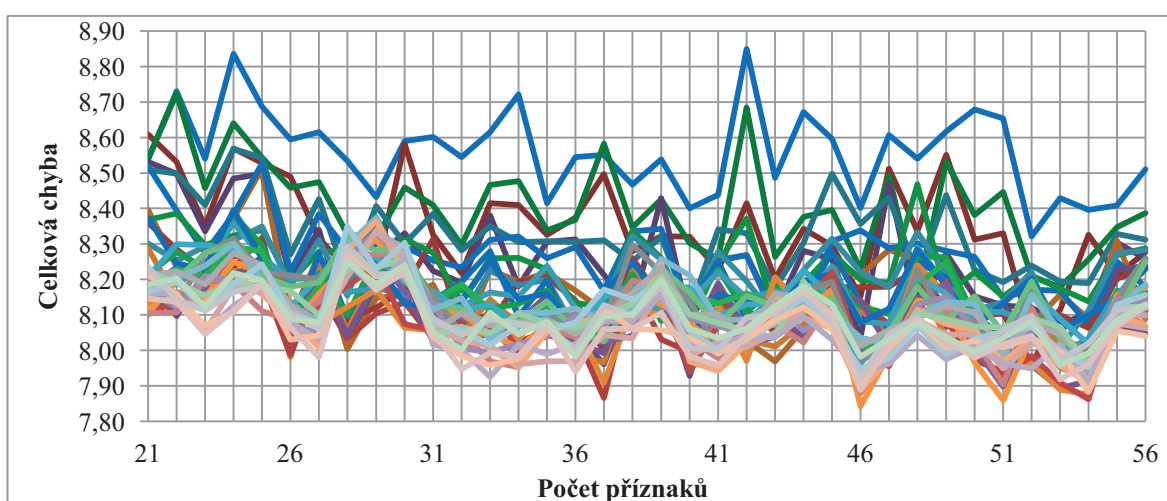
Začali jsme s testováním pro 1 až 56 features při počtu jednoho až pěti estimátorů a zjistili jsme, že pro malý počet features a estimátorů jsou výsledky korekce velmi špatné a celková chyba segmentace po opravě je dokonce ještě horší než byla před opravou. Jak ale počet features a estimátorů roste, snižuje se i celková chyba (viz obrázek 3.14).



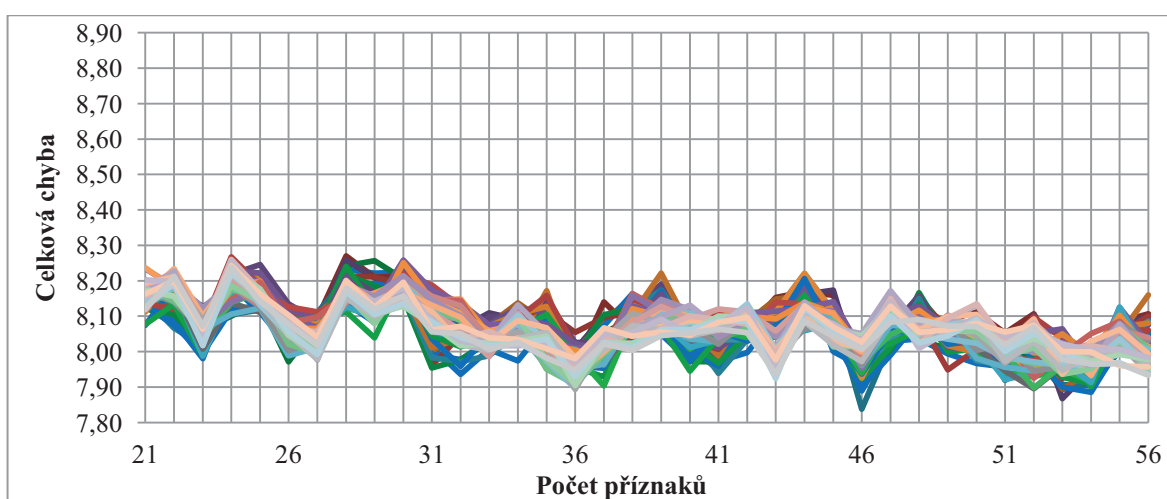
Obrázek 3.14: Závislost chyby na počtu příznaků (pro 1 až 5 estimátorů)

Z toho důvodu jsme se rozhodli dále zvyšovat počet estimátorů a parametr features testovat už jen od hodnoty 21 a výše. Testování pro jednotlivý počet features bylo provedeno do celkového počtu 149 estimátorů a postup chyby bychom mohli rozdělit na tři pomyslné části – klesání chyby, ustalování chyby a stav, kdy chyba dále neklesá.

V rozmezí 6 až 44 estimátorů (obrázek 3.15) jsme stále zaznamenávali pokles chyby, která se v průměru⁶ pohybovala okolo 8,2 s. V rozmezí 45 až 99 estimátorů (obrázek 3.16), se výsledná chyba segmentace více přiblížila hranici osmi vteřin (v průměru 8,07 s). V tomto rozmezí bylo také dosaženo nejnižší chyby a to pro 49 estimátorů a 46 features, kdy celková segmentační chyba klesla na 7,8376 s. V intervalu 100 až 149 estimátorů (obrázek 3.17), kde byla průměrná chyba 8,06 s, už k žádnému výraznému poklesu nedošlo. Jelikož průměrné chyby z posledních dvou rozmezí byly prakticky stejné, ukončili jsme hledání parametrů.

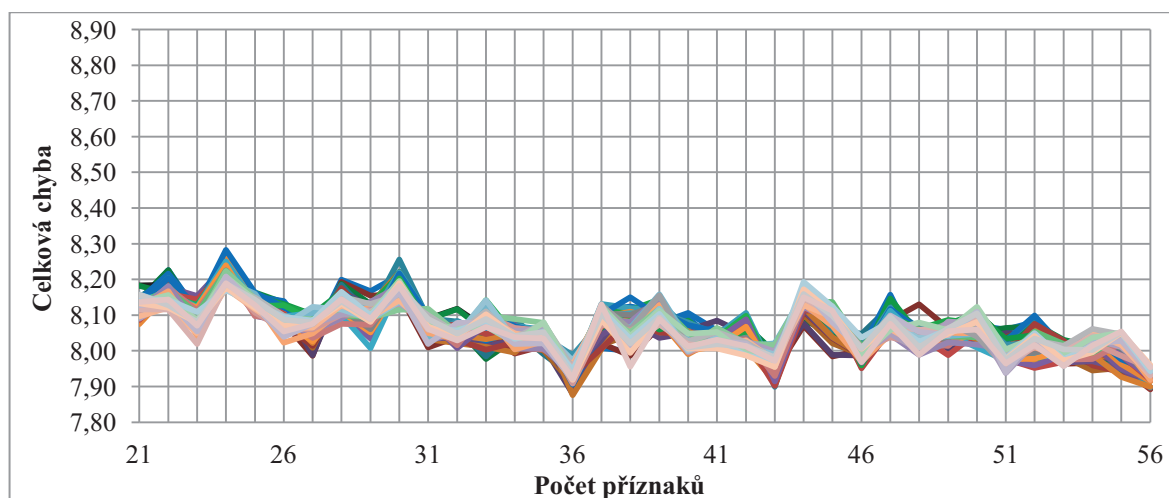


Obrázek 3.15: Závislost chyby na počtu příznaků (pro 6 až 44 estimátorů)



Obrázek 3.16: Závislost chyby na počtu příznaků pro 45 až 99 estimátorů

⁶ Průměrná chyba byla počítána z výsledných chyb všech features u všech estimátorů v daném intervalu.



Obrázek 3.17: Závislost chyby na počtu příznaků pro 100 až 149 estimátorů

Výsledky korekce s nejnižší chybou pomocí metody ERT zachycuje tabulka 3.30.

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno	nezměněno	zhoršeno	zlepšeno	nezměněno	zhoršeno	
FRI-FRI	0,2249	0,2183	6,563	0,000	1	4%	20	95%	0	0%	21
FRI-NLQ	0,4870	0,3866	13,842	12,968	11	21%	36	70%	4	7%	51
FRI-PAU	0,4619	0,4405	6,064	1,432	4	5%	74	92%	2	2%	80
FRI-VOW	0,6668	0,4148	5,524	3,705	53	32%	99	60%	11	6%	163
NLQ-FRI	0,1255	0,0901	7,100	0,126	5	23%	15	71%	1	4%	21
NLQ-NLQ	0,2144	0,2174	7,126	10,188	1	3%	27	93%	1	3%	29
NLQ-PAU	0,3617	0,2652	8,404	9,486	16	34%	27	57%	4	8%	47
NLQ-VOW	1,9411	1,5227	9,408	7,938	47	20%	175	77%	3	1%	225
PAU-FRI	0,0438	0,0442	0,570	0,498	2	13%	10	66%	3	20%	15
PAU-NLQ	0,1119	0,1057	4,007	1,802	2	13%	12	80%	1	6%	15
PAU-PAU	0,4933	0,4942	4,802	24,828	5	20%	18	75%	1	4%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2413	6,375	0,000	2	9%	20	90%	0	0%	22
PLO-NLQ	0,1874	0,0763	7,410	7,437	16	30%	36	67%	1	1%	53
PLO-PAU	0,1746	0,1705	4,101	0,000	1	3%	32	96%	0	0%	33
PLO-VOW	0,8970	0,3896	6,266	6,282	91	48%	87	46%	10	5%	188
VOW-FRI	0,6728	0,5622	8,048	6,038	22	13%	128	79%	11	6%	161
VOW-NLQ	1,4056	1,3448	8,659	8,625	11	6%	159	91%	4	2%	174
VOW-PAU	1,1957	0,8090	7,554	9,372	76	31%	147	60%	20	8%	243
VOW-VOW	0,0386	0,0442	0,000	5,625	0	0%	4	80%	1	20%	5
	9,9580	7,8376	6,091	5,817	366	23,3%	1126	71,7%	78	5,0%	1570

Tabulka 3.30: Výsledky korekce pomocí ERT (49 estimators, 46 features)

Jako předešlé dva klasifikační přístupy i tento třetí vykazuje velké množství nezměněných hranic, avšak oproti předešlým dvěma došlo k mírnému nárůstu (v průměru o 50) zlepšených hranic. To se odrazilo na nižší celkové segmentační chybě, která se tak nejvíce přiblížila k velikosti chyby u regresního přístupu.

3.2.4 Srovnání s regresním přístupem

Dokončili jsme rozbor jednotlivých klasifikačních přístupů a podrobněji si ukážeme, jak se liší jejich výsledky od regresního přístupu. Tabulka 3.31 shrnuje výsledky všech korekcí. Z tabulky je patrné, že každý přístup vykazuje lepší výsledky v jiném směru a že žádný z nich není stoprocentně lepší než ty ostatní.

Typ přístupu	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno	nezměněno	zhoršeno	zlepšeno	nezměněno	zhoršeno	
regrese	9,9580	7,3377	5,559	5,542	637	40,6%	640	40,8%	293	18,7%	1570
SVC	9,9580	8,3116	4,377	5,460	328	20,9%	1144	72,9%	98	6,2%	1570
k-sousedů	9,9580	8,1279	5,290	2,714	306	19,5%	1197	76,2%	67	4,3%	1570
ERT	9,9580	7,8376	6,091	5,817	366	23,3%	1126	71,7%	78	5,0%	1570

Tabulka 3.31: Srovnání jednotlivých přístupů

Regresní přístup jako jediný ze všech vykazuje velký podíl (téměř 60 %) hranic, u kterých došlo k nějaké změně, z čehož přibližně dvě třetiny jsou opraveny dobře a třetina špatně. Obě hodnoty, počet zlepšených i počet zhoršených hranic, jsou tak extrémny mezi všemi přístupy. Přestože je tedy počet zhoršených hranic nejvyšší ze všech, počet zlepšených hranic to více než dvojnásobně kompenzuje a celkově se tak dostáváme na nejmenší segmentační chybu ze všech.

Korekce pomocí SVC se zde jeví jako nejhorší ze všech. Je to díky nejvyšší segmentační chybě, nejnižšímu průměrnému zlepšení dobře opravených hranic a také díky tomu, že tato metoda v ničem nevykazovala žádné extrémně dobré výsledky.

Další v pořadí je klasifikace podle algoritmu k-nejbližších sousedů. Tento klasifikátor sice disponoval nejmenším počtem zlepšených hranic vůbec, ale na druhou stranu měl i nejmenší procento zhoršených hranic. Navíc se zhoršování neprojevovalo tak razantně jako u ostatních případů, protože v průměru docházelo ke zhoršení jen 2,7 ms, což je oproti

ostatním poloviční hodnota. Pokud bychom tedy měli opravu provést tak, abychom zhoršili co nejmenší počet hranic, nejspíše bychom použili tento klasifikátor.

Posledním, avšak neméně důležitým je klasifikátor navržený metodou ERT, který měl nejvyšší průměrné hodnoty zhoršení, ale i zlepšení, které se pohybovaly okolo 6 ms. Jako jediný ze všech klasifikátorů pokořil osmivteřinovou hranici celkové segmentační chyby, ale i přesto, že měl mezi klasifikátory nejmenší chybu a největší počet změněných hranic, byl pořád poměrně vzdálen od hodnot regresního přístupu.

4 Velké chyby segmentace

Chyby¹ v segmentaci řeči jsou důvodem, proč dochází k vývoji korekčních algoritmů. Mohou vznikat z různých důvodů a také mohou mít různý charakter v závislosti na umístění hranice. Mezi příčiny vzniku chyb při segmentaci řečového signálu patří například koartikulace, která ztěžuje práci segmentačních algoritmů tím, že jednotlivé hlásky na přechodu splývají. Další příčinou může být různá podobnost mezi hláskami, která způsobuje, že každou hranici je jinak obtížné určit, a nesmíme zapomenout na samotného řečníka, který namlouvá řečové promluvy. Ten může zanechat chybu do samotného řečového signálu nepřesným vyslovením, což v praxi znamená, že hranice mezi nepřesně vyslovenými hláskami bude mít odlišné akustické charakteristiky a model segmentačního algoritmu pak tuto hranici detekuje chybně.

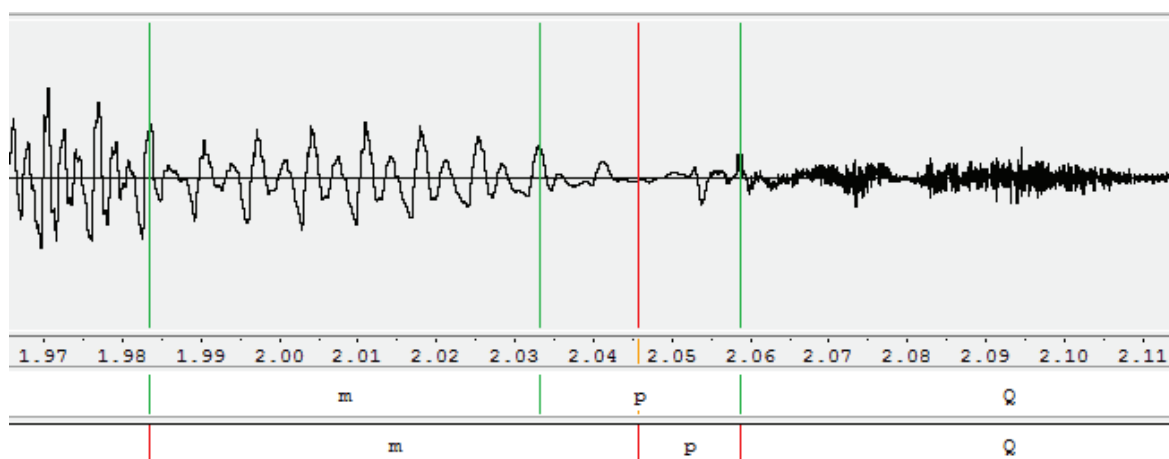
Mírně chybná detekce hranic, např. pár jednotek milisekund, však nemusí být až tak závažný problém jako je tomu v případě, že segmentační algoritmus udělá nějakou velkou chybu. Ty mohou začínat řádově na desítkách milisekund a mohou být teoreticky neomezené, respektive omezené jen délkou segmentované promluvy. Právě proto, že je velikost velkých chyb velmi proměnná, je značně obtížné tyto chyby automaticky detekovat. Přesto je nutné je opravovat, protože do dalšího procesu vytváření řeči vnášejí největší problémy.

V předchozí kapitole jsme se zabývali vylepšením korekce všech typů hranic bez ohledu na to, zda se jedná o hranice s velkými či malými chybami. Víme, že náš algoritmus pracuje tak, že segmentovanou hranici může posunout jen o určitou vzdálenost (v rámci velikosti oblasti pro výběr kandidátů při korekci) a pokud se tedy bude jednat o velkou chybu, nikdy ji nebude schopen posunout na správné místo. V této poslední kapitole se tedy podíváme a zhodnotíme možný postup, jak velké chyby automaticky odhalit. To by nám mohlo pomoci při jejich opravování, protože by bylo možné k nim přistupovat jinak než pomocí obecného algoritmu, který teď aplikujeme na všechny hranice bez výjimky.

¹ Chyba je odchylka automaticky segmentované hranice od místa, kde by měla být, tj. od správné hranice.

4.1 Analýza průměrných a největších chyb

Pro lepší představu o tom, jakou kvalitu má segmentovaný soubor, se kterým pracujeme, resp. jak velké chyby se v něm nacházejí, provedeme nyní jeho analýzu. Nejprve zjistíme, jaká je průměrná chybovost jednotlivých typů hranic a potom se pro každý typ zaměříme na největší chybu, která se v segmentaci vyskytuje. Kromě samotné velikosti chyb budeme ještě sledovat jejich zásah do okolních hlásek neboli to, jak chybně určená hranice přesahuje do svých sousedních hlásek. Může totiž nastat případ, že máme hranici mezi dvěma hláskami, z nichž jedna nebo obě mají krátký signál a potom i relativně menší chyba může způsobit velké škody, protože zabere například 50 % následující hlásky, jak je tomu na obrázku 4.1. Hláška [p] je dlouhá 26 ms a chyba hranice *m-p* je 13 ms (zeleně jsou zakresleny hranice ruční segmentace a červeně hranice automatické segmentace).



Obrázek 4.1: Ukázka chyby zasahující do poloviny následující hlásky

Vyhodnocování chyb a jejich zásahů do okolních hlásek je vždy realizováno pomocí tabulky, ve které jsou výsledky dvojice souborů oddělené dvojitou čarou. Jsou to dvojice trénovací plus korekční soubor a korekční soubor před opravou plus po opravě. Zprvė tedy budeme zkoumat, jestli se chybovost segmentace příliš neliší mezi trénovacími a korekčními daty a za druhé jak se chybovost změní po korekci. Tabulka má na konci souhrn (součet a průměr) jednotlivých hodnot, aby bylo možné udělat srovnání. Zvlášt' budeme také sledovat parametry chyb u hranic, které byly umístěny vlevo od správné hranice a zvlášt' u těch, které byly umístěny vpravo od správné hranice. To z toho důvodu,

abychom mohli potvrdit nebo vyvrátit tendenci segmentačního algoritmu dělat více levostranné či pravostranné chyby. K testování jsme využili informace z referenčních dat.

4.1.1 Průměrné chyby

V tabulce 4.1 srovnáváme průměrné chyby segmentace v souboru použitém k natrénování SPM a v souboru ke korekci. Z výsledků je patrné, že mezi soubory sice dochází k určitým odlišnostem u jednotlivých typů hranic, ale v celkovém průměru se výsledky liší pouze o 1 až 2 %. Průměrná chyba segmentace se pohybuje okolo 5 ms s procentuálním zásahem do okolních hlásek v rozmezí od čtyř do osmi procent a je zde také vidět mírný sklon k pravostranným chybám. Červeně jsou označeny průměrné chyby, které přesáhly hodnotu deseti milisekund a tak je u nich pravděpodobnější výskyt velkých chyb.

Typ hranice	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za
FRI-FRI	3%	-0,003	5%	8%	0,005	8%	7%	-0,007	13%	7%	0,005	7%
FRI-NLQ	4%	-0,004	8%	6%	0,004	9%	10%	-0,009	22%	6%	0,004	7%
FRI-PAU	1%	-0,001	1%	5%	0,004	5%	2%	-0,003	3%	4%	0,003	4%
FRI-VOW	1%	-0,001	1%	10%	0,007	8%	1%	-0,001	1%	9%	0,006	8%
NLQ-FRI	8%	-0,007	11%	4%	0,003	5%	8%	-0,006	7%	7%	0,005	7%
NLQ-NLQ	12%	-0,008	15%	6%	0,004	7%	18%	-0,010	23%	2%	0,002	2%
NLQ-PAU	5%	-0,004	5%	4%	0,003	4%	6%	-0,007	6%	6%	0,004	5%
NLQ-VOW	3%	-0,001	2%	24%	0,010	13%	5%	-0,002	3%	24%	0,010	12%
PAU-FRI	1%	-0,002	2%	2%	0,004	4%	1%	-0,002	2%	1%	0,001	1%
PAU-NLQ	0%	-0,001	1%	1%	0,002	3%	3%	-0,004	5%	1%	0,004	7%
PAU-PAU	1%	-0,002	4%	9%	0,020	23%	1%	-0,002	3%	12%	0,018	18%
PLO-FRI	3%	-0,002	3%	4%	0,002	3%	5%	-0,005	8%	14%	0,008	8%
PLO-NLQ	4%	-0,003	7%	8%	0,005	10%	6%	-0,004	10%	5%	0,003	7%
PLO-PAU	4%	-0,003	3%	3%	0,003	4%	5%	-0,005	4%	3%	0,002	3%
PLO-VOW	1%	-0,001	1%	11%	0,007	9%	2%	-0,001	2%	10%	0,006	8%
VOW-FRI	6%	-0,005	8%	5%	0,003	4%	5%	-0,005	6%	7%	0,005	7%
VOW-NLQ	12%	-0,009	19%	4%	0,003	5%	11%	-0,009	17%	6%	0,004	8%
VOW-PAU	5%	-0,004	6%	5%	0,004	4%	4%	-0,004	6%	6%	0,005	5%
VOW-VOW	10%	-0,010	16%	16%	0,013	17%	3%	-0,002	1%	15%	0,011	12%
Součet:	85%	-0,069	117%	134%	0,105	144%	105%	-0,086	142%	145%	0,105	138%
Průměr:	4%	-0,004	6%	7%	0,006	8%	6%	-0,005	7%	8%	0,006	7%

Tabulka 4.1: Průměrné hodnoty chyb v trénovacích (vlevo) a korekčních (vpravo) datech

Velké chyby segmentace

Druhou zkoumanou dvojicí byl korekční soubor před opravou a po opravě. Průměrné chyby jsou v tabulce 4.2 a můžeme vidět, že průměrné hodnoty zůstaly prakticky stejné. U chyb vlevo došlo k šesti zlepšením (zeleně) a sedmi zhoršením (červeně) a u chyb vpravo k sedmi zlepšením a devíti zhoršením, což jsou vcelku vyrovnané výsledky.

To pro nás znamená, že když po korekci celého souboru klesne celková segmentační chyba, není to způsobeno tím, že by se všechny hranice v průměru mírně zlepšily. Je to tím, že hranic, u kterých dojde k lepší opravě, je více, než hranic, u kterých dojde k horší opravě. I přesto, že počet zhoršení průměrných chyb je zde nepatrně větší, na výslednou průměrnou chybu to nemělo téměř žádný vliv.

Typ hranice	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za
FRI-FRI	7%	-0,007	13%	7%	0,005	7%	9%	-0,009	16%	4%	0,003	3%
FRI-NLQ	10%	-0,009	22%	6%	0,004	7%	10%	-0,009	22%	7%	0,006	8%
FRI-PAU	2%	-0,003	3%	4%	0,003	4%	4%	-0,004	4%	3%	0,002	3%
FRI-VOW	1%	-0,001	1%	9%	0,006	8%	3%	-0,003	3%	5%	0,003	4%
NLQ-FRI	8%	-0,006	7%	7%	0,005	7%	6%	-0,004	4%	11%	0,007	13%
NLQ-NLQ	18%	-0,010	23%	2%	0,002	2%	18%	-0,010	22%	6%	0,003	6%
NLQ-PAU	6%	-0,007	6%	6%	0,004	5%	5%	-0,005	3%	7%	0,004	6%
NLQ-VOW	5%	-0,002	3%	24%	0,010	12%	12%	-0,005	6%	19%	0,008	10%
PAU-FRI	1%	-0,002	2%	1%	0,001	1%	1%	-0,002	2%	2%	0,004	4%
PAU-NLQ	3%	-0,004	5%	1%	0,004	7%	3%	-0,004	6%	1%	0,003	5%
PAU-PAU	1%	-0,002	3%	12%	0,018	18%	1%	-0,004	6%	15%	0,019	18%
PLO-FRI	5%	-0,005	8%	14%	0,008	8%	6%	-0,006	10%	8%	0,005	5%
PLO-NLQ	6%	-0,004	10%	5%	0,003	7%	3%	-0,002	6%	11%	0,005	9%
PLO-PAU	5%	-0,005	4%	3%	0,002	3%	6%	-0,005	5%	2%	0,002	1%
PLO-VOW	2%	-0,001	2%	10%	0,006	8%	7%	-0,005	7%	5%	0,004	5%
VOW-FRI	5%	-0,005	6%	7%	0,005	7%	3%	-0,002	3%	9%	0,007	9%
VOW-NLQ	11%	-0,009	17%	6%	0,004	8%	10%	-0,008	16%	8%	0,006	12%
VOW-PAU	4%	-0,004	6%	6%	0,005	5%	3%	-0,004	5%	6%	0,005	6%
VOW-VOW	3%	-0,002	1%	15%	0,011	12%	2%	-0,001	1%	19%	0,015	15%
Součet:	105%	-0,086	142%	145%	0,105	138%	111%	-0,092	147%	148%	0,109	143%
Průměr:	6%	-0,005	7%	8%	0,006	7%	6%	-0,005	8%	8%	0,006	8%

Tabulka 4.2: Průměrné hodnoty chyb v korekčních datech před (vlevo) a po (vpravo) opravě

4.1.2 Největší chyby

Nyní se podíváme, jak na tom jsou jednotlivé typy hranic s maximálními segmentačními chybami. Tabulka 4.3 zachycuje ty největší levostranné a pravostranné chyby pro každý typ hranice a červeně jsou vyznačeny ty chyby, u kterých došlo k zásahu do okolních hlásek (ať už vlevo nebo vpravo) o více než 100 %. V tabulce je dobře vidět, že se maximální chyby pro různé typy hranic velmi liší.

Jak bylo naznačeno na začátku kapitoly, je to především různou obtížností detekce hranic. Některé hranice lze snadněji určit a tak v nich obecně algoritmus nedělá tak velké chyby, i když velká chyba se vlivem dalších okolností může vyskytnout prakticky u jakéhokoliv typu hranice.

Typ hranice	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za
FRI-FRI	20%	-0,020	44%	31%	0,029	30%	41%	-0,043	100%	43%	0,030	43%
FRI-NLQ	16%	-0,022	46%	15%	0,018	51%	56%	-0,039	59%	46%	0,040	27%
FRI-PAU	7%	-0,014	6%	15%	0,021	12%	29%	-0,039	46%	30%	0,025	35%
FRI-VOW	17%	-0,013	11%	24%	0,023	11%	36%	-0,023	19%	28%	0,019	33%
NLQ-FRI	26%	-0,027	47%	21%	0,028	44%	30%	-0,023	30%	29%	0,021	22%
NLQ-NLQ	32%	-0,038	43%	36%	0,032	67%	35%	-0,024	75%	12%	0,009	16%
NLQ-PAU	26%	-0,042	16%	32%	0,020	25%	17%	-0,024	8%	57%	0,021	33%
NLQ-VOW	206%	-0,088	109%	147%	0,081	70%	83%	-0,041	39%	123%	0,055	58%
PAU-FRI	23%	-0,043	38%	10%	0,023	27%	4%	-0,007	12%	3%	0,006	4%
PAU-NLQ	3%	-0,005	9%	4%	0,011	15%	12%	-0,034	40%	3%	0,010	19%
PAU-PAU	15%	-0,033	172%	23%	0,085	47%	8%	-0,036	55%	67%	0,087	46%
PLO-FRI	39%	-0,015	29%	57%	0,026	36%	44%	-0,055	79%	78%	0,059	44%
PLO-NLQ	22%	-0,023	91%	45%	0,020	37%	15%	-0,017	31%	11%	0,008	19%
PLO-PAU	38%	-0,027	49%	54%	0,030	53%	20%	-0,036	15%	11%	0,009	27%
PLO-VOW	43%	-0,034	75%	68%	0,025	34%	26%	-0,019	22%	47%	0,023	14%
VOW-FRI	26%	-0,025	44%	162%	0,052	100%	20%	-0,020	26%	53%	0,033	59%
VOW-NLQ	58%	-0,059	97%	32%	0,035	41%	72%	-0,048	119%	61%	0,044	50%
VOW-PAU	24%	-0,058	25%	10%	0,032	22%	34%	-0,037	78%	23%	0,028	9%
VOW-VOW	43%	-0,040	67%	58%	0,044	55%	8%	-0,006	4%	30%	0,021	26%
Součet:	682%	-0,623	1016%	843%	0,633	777%	590%	-0,571	857%	755%	0,547	585%
Průměr:	36%	-0,033	53%	44%	0,033	41%	31%	-0,030	45%	40%	0,029	31%

Tabulka 4.3: Velikosti největších chyb v trénovacích (vlevo) a korekčních (vpravo) datech

Největší chyby nemají příliš tendenci k pravostrannému či levostrannému chování a v průměru se pohybují od 29 do 33 ms s procentuálním zásahem do okolních hlásek

Velké chyby segmentace

v rozmezí 31 až 45 %. Nejzávažnější však je, že ze součtu jednotlivých chyb vyplývá, že v korekčním souboru, který má celkovou segmentační chybu 9,958 s, má pouze těchto 38 chyb (2,4 % hranic) zobrazených v tabulce na svědomí 11,2 % celkové chyby (1,118 s). Co se chyb přesahujících sousední hlásky o sto a více procent týče, mohou způsobit rozhození celé segmentace, protože pak může dojít k záměně řečových jednotek, jelikož některé hranice se mohou posunout na místa jiných hranic.

V tabulce 4.4 jsou největší chyby v korekčním souboru před opravou a po opravě. Z celkových třiceti osmi porovnávaných maxim se čtyři nezměnily, u čtrnácti došlo ke zlepšení a u dvaceti ke zhoršení. Jak je vidět, po korekci došlo spíše k prohloubení rozdílů a ke zvětšení chyb (hlavně levostranných) a snahou je, aby po korekci naopak docházelo k co největší redukci těchto velkých odchylek.

Typ hranice	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za	zásah před	chyba vlevo	zásah za	zásah před	chyba vpravo	zásah za
FRI-FRI	41%	-0,043	100%	43%	0,030	43%	41%	-0,047	119%	29%	0,020	29%
FRI-NLQ	56%	-0,039	59%	46%	0,040	27%	49%	-0,034	51%	56%	0,048	32%
FRI-PAU	29%	-0,039	46%	30%	0,025	35%	39%	-0,052	62%	9%	0,012	4%
FRI-VOW	36%	-0,023	19%	28%	0,019	33%	36%	-0,023	19%	31%	0,015	18%
NLQ-FRI	30%	-0,023	30%	29%	0,021	22%	6%	-0,009	9%	31%	0,021	66%
NLQ-NLQ	35%	-0,024	75%	12%	0,009	16%	85%	-0,045	91%	38%	0,015	41%
NLQ-PAU	17%	-0,024	8%	57%	0,021	33%	11%	-0,017	6%	57%	0,021	33%
NLQ-VOW	83%	-0,041	39%	123%	0,055	58%	100%	-0,050	47%	103%	0,046	48%
PAU-FRI	4%	-0,007	12%	3%	0,006	4%	9%	-0,015	23%	4%	0,020	18%
PAU-NLQ	12%	-0,034	40%	3%	0,010	19%	12%	-0,036	42%	5%	0,016	19%
PAU-PAU	8%	-0,036	55%	67%	0,087	46%	12%	-0,058	89%	65%	0,086	45%
PLO-FRI	44%	-0,055	79%	78%	0,059	44%	53%	-0,066	95%	60%	0,045	34%
PLO-NLQ	15%	-0,017	31%	11%	0,008	19%	11%	-0,008	20%	15%	0,009	17%
PLO-PAU	20%	-0,036	15%	11%	0,009	27%	22%	-0,040	17%	9%	0,013	4%
PLO-VOW	26%	-0,019	22%	47%	0,023	14%	28%	-0,022	38%	17%	0,018	28%
VOW-FRI	20%	-0,020	26%	53%	0,033	59%	12%	-0,010	23%	40%	0,031	30%
VOW-NLQ	72%	-0,048	119%	61%	0,044	50%	50%	-0,040	56%	74%	0,053	60%
VOW-PAU	34%	-0,037	78%	23%	0,028	9%	41%	-0,044	94%	25%	0,030	10%
VOW-VOW	8%	-0,006	4%	30%	0,021	26%	8%	-0,006	4%	47%	0,035	33%
Součet:	590%	-0,571	857%	755%	0,547	585%	625%	-0,623	906%	713%	0,552	570%
Průměr:	31%	-0,030	45%	40%	0,029	31%	33%	-0,033	48%	38%	0,029	30%

Tabulka 4.4: Velikosti největších chyb v korekčních datech před (vlevo) a po (vpravo) opravě

Je také nutno podotknout, že porovnáváme pouze maximální chyby a tudíž maximální chyba po opravě nemusí patřit té samé hranici, které patřila maximální chyba před opravou. V závislosti na úspěšnosti korekce totiž mohlo dojít k opravě původní maximální chyby a její místo tak nahradila jiná maximální chyba anebo mohlo dojít naopak ke zhoršení některé hranice natolik, že nahradila původní maximální chybu.

Máme zdokumentovány velikosti velkých a průměrných chyb a přejdeme k části, ve které se budeme zabývat, čím se velké chyby odlišují od ostatních.

4.2 Závislost velkých chyb a skóre

V předchozí části jsme si ukázali, že velké chyby naší segmentace se pohybují v řádech několika desítek milisekund a tím se velmi odlišují od průměrné segmentační chyby a dále pak, že nedochází k jejich výraznému vychylování na pravou ani levou stranu. Abychom tak byli schopni velké chyby detekovat, musíme najít nějaký faktor, který je odlišuje od ostatních chyb. Prvním rozlišujícím faktorem, který by nás mohl napadnout, je jejich velikost, tj. vzdálenost chybné hranice od jejího správného místa. Problémem však je, že pokud dostaneme neznámý řečový signál, nevíme, kde se nacházejí správná místa pro každou hranici a nemůžeme proto udělat porovnání jako jsme ho dělali v předchozím případě.

Při experimentech v části 3.1.7 jsme navrhli metodu opravy, která brala v úvahu pouze hranice, které byly ohodnoceny dostatečně vysokým skóre. Při tomto pokusu nás napadlo, že by bylo možné velké chyby odlišit právě pomocí skóre. Pokud má totiž vysoké skóre označovat správnou hranici, mohlo by velmi nízké skóre označovat špatnou hranici, přesněji velkou chybu segmentace. Tuto myšlenku jsme tedy podrobili detailnějšímu prozkoumání, abychom zjistili, jestli je možné pomocí ní dále postupovat.

4.2.1 Vlastnosti skóre u velkých a malých chyb

Byly vybrány dvě hranice špatně určené automatickou segmentací, které představují velké chyby – jedna ze skupiny FRI-FRI a druhá ze skupiny NLQ-VOW. Pro obě chyby byly zjištěny důležité parametry, které shrnuje tabulka 4.5. Tyto parametry sledují jak je chyba velká, jak zasahuje do okolních hlásek a jak byla ohodnocena regresním modelem. Dále je

v tabulce zanesena informace o tom, jestli se opravou chyba zlepšila a také z jak velké oblasti (délka okna) okolo automaticky určené hranice byly vybíráni kandidáti na opravu, aby bylo zřejmé, jestli bylo pro regresní model možné nalézt správnou (referenční) hranici. Časy uvedené v tabulce jsou v sekundách.

typ hranice	čas			zásah	ref-	zásah	zásah	ref-	zásah	fix-	délka okna	skóre	
	ref	auto	fix	před	auto	za	před	fix	za	auto		auto	fix
FRI-FRI	8,077	8,034	8,034	41%	-0,043	100%	41%	-0,043	100%	0,000	0,016	6,54	6,54
NLQ-VOW	3,534	3,589	3,580	123%	0,055	58%	103%	0,046	48%	-0,009	0,019	8,72	12,37

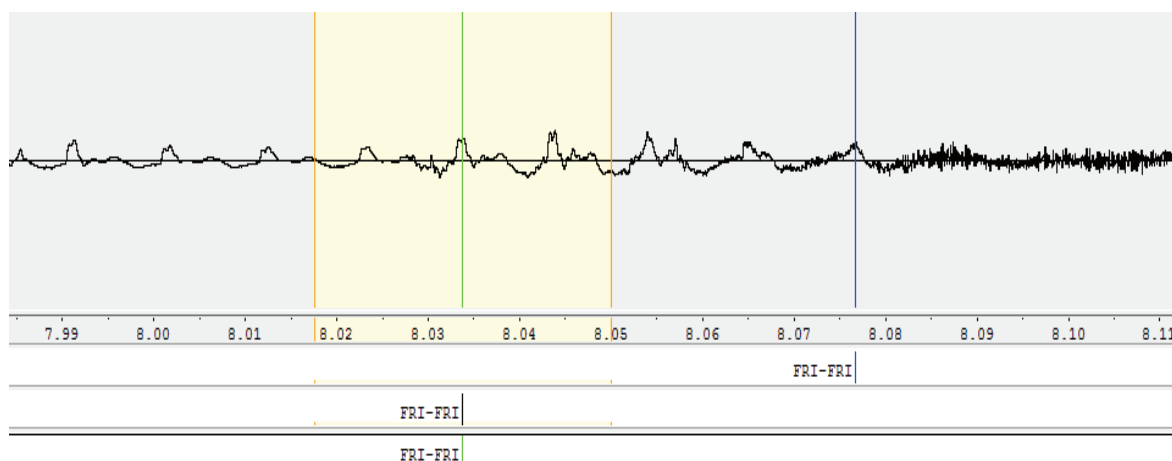
Tabulka 4.5: Velké chyby a jejich vlastnosti

Vysvětlení parametrů:

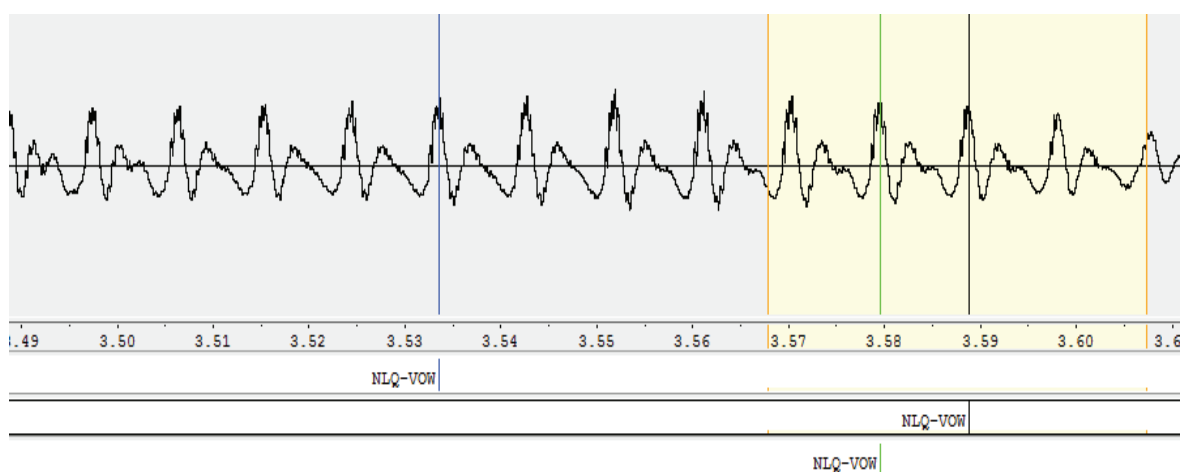
- čas ref – čas referenční hranice
- čas auto – čas automaticky určené hranice (automatická hranice)
- čas fix – čas opravené hranice
- ref-auto – odchylka automatické hranice od referenční hranice
- zásah před/za – jak moc hranice zasahuje do okolních hlásek
- ref-fix – odchylka opravené hranice od referenční hranice
- fix-auto – o kolik se liší opravená hranice od automatické hranice
- délka okna – do jaké vzdálenosti od automatické hranice byli vybíráni kandidáti
- skóre auto – skóre automatické hranice; dále uváděno jako SA
- skóre fix – skóre opravené hranice; dále uváděno jako SF

Z tabulky 4.5 je patrné, že tyto automatické hranice, které jsou velmi vzdálené od referenčních hranic, byly podle výše zmíněné myšlenky ohodnoceny velmi malým skóre, což je pro nás příznivá informace. Z tabulky lze vyčíst, že SF je také velmi malé a to i přesto, že došlo ke zlepšení skoro o 10 ms. To je zřejmě způsobeno tím, že kandidáti na opravu jsou vybíráni z okolí automaticky určené hranice, která je v těchto dvou konkrétních případech příliš daleko od správné hranice.

Jak ukazují obrázky 4.2 a 4.3 žlutě označené okolí, ze kterého jsou kandidáti vybíráni, neobsáhne danou správnou hranici. Barevné značení hranic v obrázcích bude vždy stejné a bude se vztahovat k typu hranice, tj. modrá bude referenční hranice, černá bude hranice určená automaticky segmentačním algoritmem a zelená bude opravená hranice.



Obrázek 4.2: Velká chyba pro hranici FRI-FRI (žlutě oblast pro výběr kandidátů)



Obrázek 4.3: Velká chyba pro hranici NLQ-VOW (žlutě oblast pro výběr kandidátů)

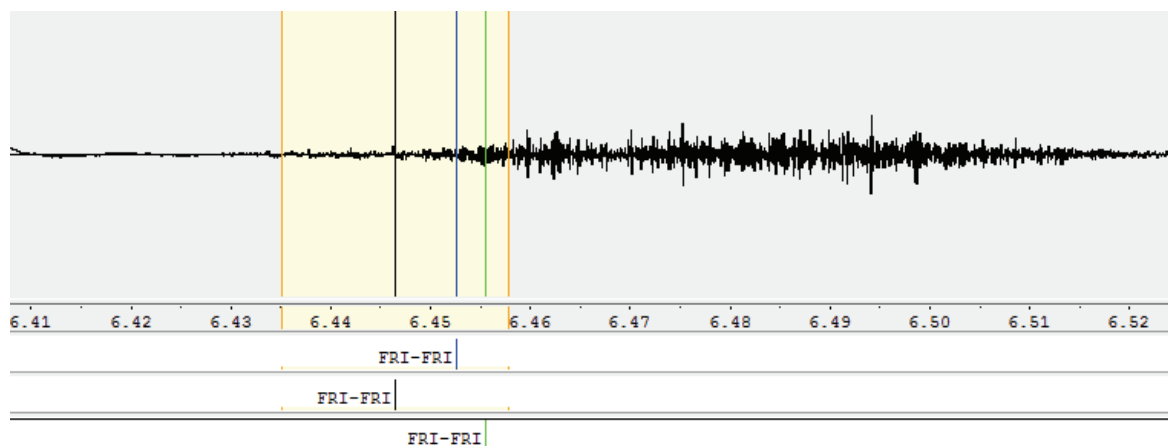
Model si tedy s opravou takto velkých chyb neporadí. Může sice takovouto hranici částečně zlepšit (přiblížit jí ke správné hranici), ale z principu jí nikdy nebude schopen úplně opravit.

Podíváme se teď na vlastnosti malých chyb a na to, zda opravdu platí, že malé chyby dostávají větší skóre. Ze skupin FRI-FRI a NLQ-VOW, ze kterých byly vybrány velké chyby, jsme vybrali zástupce malých chyb a jejich vlastnosti jsme zapsali do následující tabulky 4.6. Časy uvedené v tabulce jsou opět v sekundách.

typ hranice	čas			zásah před	ref- auto	zásah za	zásah před	ref- fix	zásah za	fix- auto	délka okna	skóre	
	ref	auto	fix									auto	fix
FRI-FRI	6,453	6,447	6,456	14%	-0,006	7%	7%	0,003	4%	0,009	0,012	24,5	45,7
FRI-FRI	1,046	1,051	1,046	6%	0,005	12%	0%	0,000	0%	-0,005	0,008	43,5	64,3
FRI-FRI	4,788	4,783	4,784	5%	-0,005	8%	4%	-0,004	7%	0,001	0,006	52,2	52,2
NLQ-VOW	1,774	1,780	1,774	10%	0,006	12%	0%	0,000	0%	-0,006	0,013	37,2	69,0
NLQ-VOW	0,717	0,723	0,717	10%	0,006	5%	0%	0,000	0%	-0,006	0,012	47,9	89,5
NLQ-VOW	6,922	6,928	6,928	13%	0,006	7%	13%	0,006	7%	0,000	0,005	89,8	89,8
NLQ-VOW	1,214	1,220	1,208	13%	0,006	8%	13%	-0,006	8%	-0,012	0,018	13,0	45,9

Tabulka 4.6: Malé chyby a jejich vlastnosti

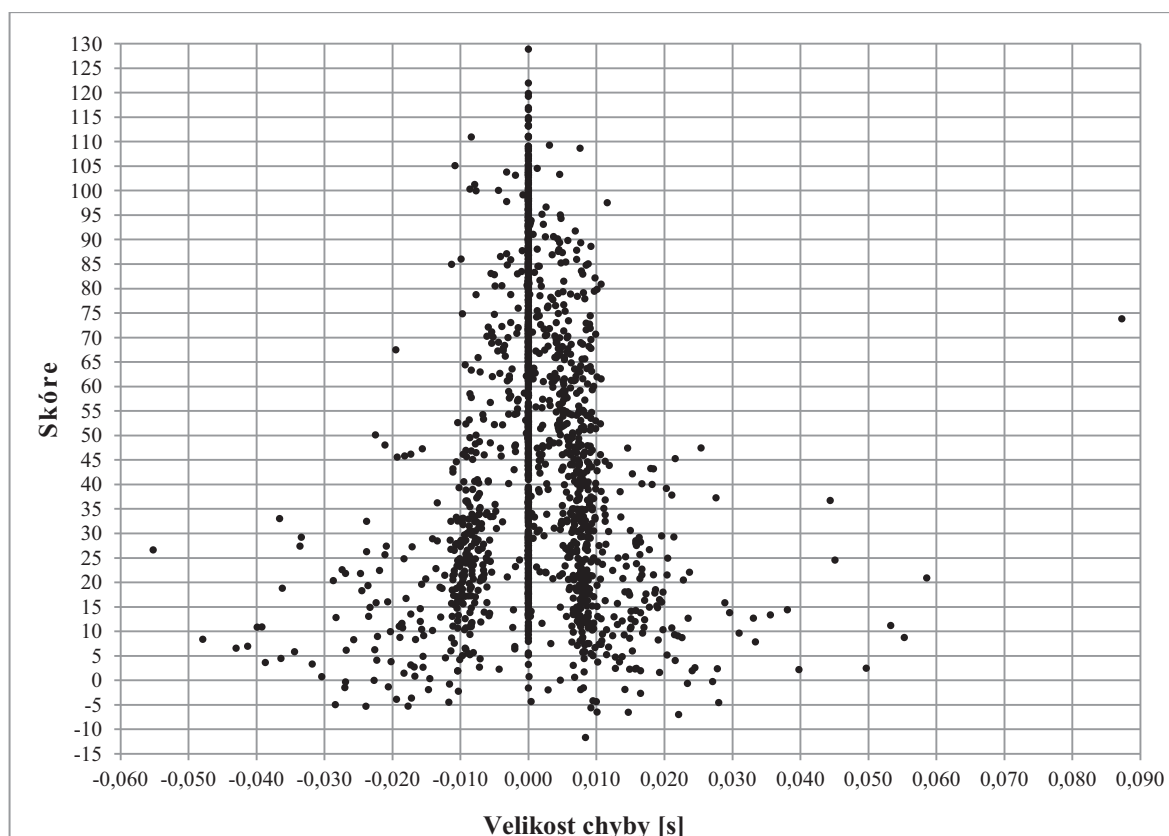
Ohodnocení automatických hranic v tomto případě vypadá o poznání lépe. Znatelně se zvětšilo SF a také SA je až na výjimku třikrát až sedmkrát větší než skóre u hranic reprezentující velké chyby. Následuje obrázek 4.4 s ukázkou první malé chyby a okolí, ze kterého byli vybíráni kandidáti pro opravu automatické hranice. Obrázky pro ostatní malé chyby je možné najít v příloze B (Obr. 1 až 6).



Obrázek 4.4: První malá chyba pro hranici FRI-FRI (žlutě oblast pro výběr kandidátů)

Na výše uvedených hranicích se nám tedy potvrdila domněnka, že čím dále je automatická hranice od správné hranice, tím nižší dostává skóre a naopak. Využijeme tohoto poznatku a zkusíme ve skóre automaticky určených hranic najít nějakou prahovou hodnotu, podle níž bychom mohli chyby dělit na malé a velké. Na následujícím obrázku 4.5, který znázorňuje závislosti mezi skórem automaticky určených hranic a odchylkou těchto hranic od referenčních hranic, můžeme vidět, že pokud chceme detekovat pouze velké chyby, nelze toho dosáhnout podle jednotné prahové hodnoty. Pokud bychom chtěli určit řekněme

všechny chyby větší než 30 ms (vyjímaje jednoho vysokého outliera, o kterém bude řeč později), museli bychom zvolit práh okolo hodnoty 40. V tom případě bychom ale detekovali i příliš velké množství malých chyb, které bohužel nedostaly dostatečně velké ohodnocení.



Obrázek 4.5: Závislost skóre automaticky určených hranic na velikosti chyby

Možná bychom však mohli nějakým způsobem zapojit také ohodnocení ostatních kandidátů, kteří se vyskytují v oblasti okolo automaticky určených hranic a ze kterých se vybírá hranice pro opravu. Obdobně by tu totiž mohlo platit, že okolo více vzdálené hranice by mohli mít kandidáti v průměru nižší skóre a naopak kandidáti v okolí bližší automatické hranice by mohly mít obecně vyšší skóre. Pokud by i toto platilo, mohli bychom toho využít a lépe tak rozpoznat velkou chybu od malé. Od všech předchozích hranic jsem tedy vybral skóre všech příslušných kandidátů z jejich okolí a spočítal jsem pro ně čtyři hodnoty. Jedná se o hodnoty průměru a mediánu ze všech skóre a o hodnoty průměru a mediánu jen ze tří nejvyšších hodnot skóre. Vše je v následující tabulce 4.7.

Typ hranice	Skóre kandidátů vlevo			SA	Skóre kandidátů vpravo			Ze všech		Ze tří nejvyšších		
								průměr	medián	průměr	medián	
FRI-FRI			0,3	6,5			-14,2	-2,47	0,29	-2,47	0,29	
NLQ-VOW		10,7	12,4	8,7			1,5	8,31	9,70	10,59	10,67	
FRI-FRI	18,3	3,6	32,5	24,5		13,9	45,7	23,08	21,41	34,22	32,51	
FRI-FRI			64,3	43,5			2,7	36,86	43,51	36,86	43,51	
FRI-FRI			48,2	52,2			42,5	47,64	48,20	47,64	48,20	
NLQ-VOW		30,7	69,0	37,2		19,2	-0,4	31,17	30,73	45,66	37,23	
NLQ-VOW			89,5	47,9			10,2	49,19	47,87	49,19	47,87	
NLQ-VOW				89,8				89,78	89,78	89,78	89,78	
NLQ-VOW	36,9	45,9	27,9	13,0		4,2	-3,2	1,4	18,02	13,02	36,91	36,88

Tabulka 4.7: Skóre v oblasti okolo automatické hranice a příslušné statistické veličiny

První dva řádky odpovídají velkým chybám a zbylé odpovídají malým chybám. Uprostřed kandidátů je vždy automaticky nalezená hranice s příslušným skóre. Každá tato hranice má v závislosti na své hodnotě skóre rozdílnou velikost okna pro výběr kandidátů a tudíž bude mít i rozdílný počet samotných kandidátů. Tučně je označeno nejvyšší skóre z každé skupiny kandidátů – tedy SF. U hranic NLQ-VOW je o jednoho zástupce více než u hranic FRI-FRI. To z toho důvodu, že třetí NLQ-VOW hranice dostala velmi vysoké skóre a nebyli vybráni žádní kandidáti na opravu, ze kterých bychom mohli počítat průměry a mediány – proto jsme přidali ještě jednu hranici tohoto typu.

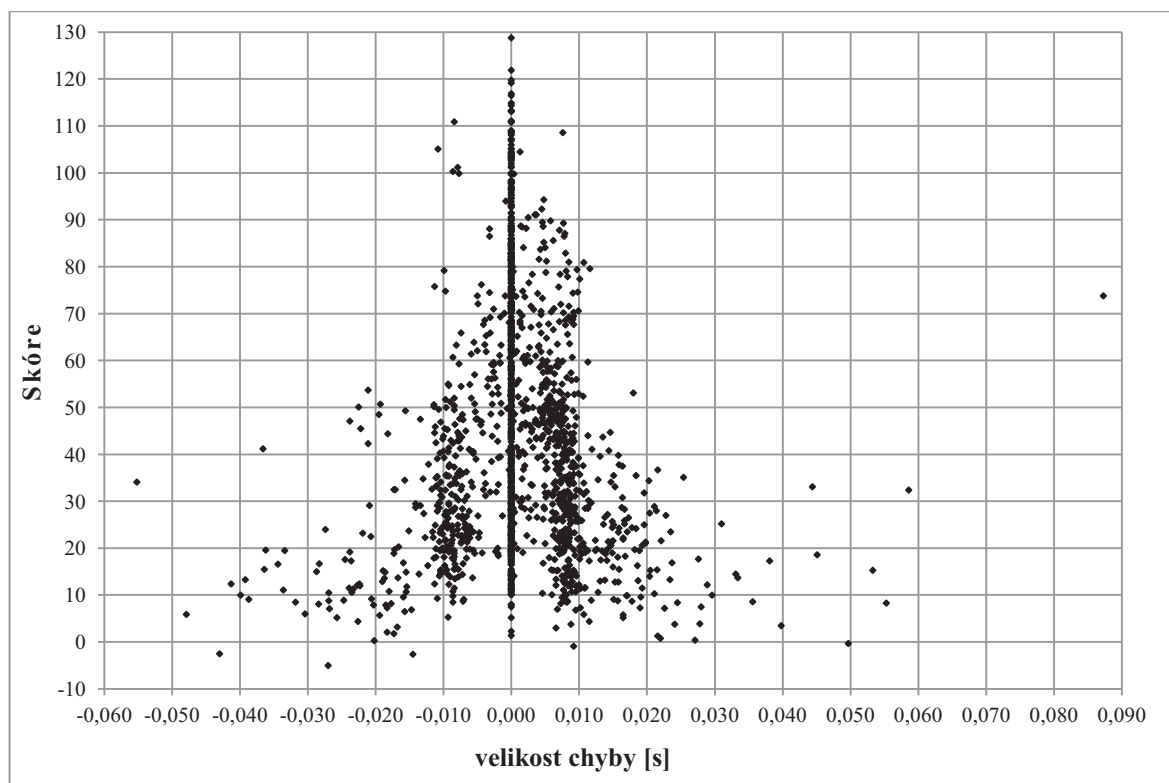
Ze získaných výsledků usuzujeme, že pro velké chyby většinou dostaneme i malé ohodnocení zbylých kandidátů. U malých chyb je skóre ostatních kandidátů většinou větší, ale lze nalézt i výjimky, kdy malá chyba dostane poměrně malé skóre, což by mohlo ztížit identifikaci velkých chyb pomocí průměrů a mediánů ze všech hodnot. Zdá se však, že pomocí hodnot průměrů a mediánů počítaných jen ze tří nejvyšších hodnot v dané oblasti by se dal tento problém obejít.

4.2.2 Hledání prahové hodnoty statistických veličin

Jak jsme zjistili, dalo by se hodnot průměrů a mediánů využít ke spolehlivější identifikaci velkých chyb než v případě, že použijeme pouze hodnoty skóre automatické hranice. K tomu bude ale třeba určit nějakou prahovou hodnotu, která by odlišila hodnoty těchto průměrů a mediánů patřící velkým chybám od hodnot patřící malým chybám. V ideálním

případě by všechny velké chyby měly například průměr či medián ze skóre kandidátů menší než dvacet a všechny malé chyby by byly nad touto hodnotou. Protože v části 4.1.2 bylo ukázáno, že v průměru se velké chyby pohybují od 28 ms, dáme si mírnou rezervu a za velkou chybu budeme považovat hranice s odchylkou větší než 25 ms.

K tomu, abychom zjistili, jestli lze nějaký takový práh stanovit, je zapotřebí určit odchylky automatických hranic od referenčních a k nim přiřadit hodnoty jednotlivých průměrů a mediánů. Ze souboru pro korekci jsme tedy vybrali všechny automaticky segmentované hranice, spočetli k nim odchylky, k nim dopočítali dané hodnoty průměrů a mediánů a vše zanesli do grafů. Dále je zobrazen jen obrázek s průměrem všech hodnot skóre, který vykazuje nejpraktičtější výsledky. Ostatní obrázky lze nalézt v příloze B (Obr. 7 až 9).



Obrázek 4.6: Závislost průměru hodnot skóre všech kandidátů na velikosti chyby

Grafy pro všechny čtyři statistické hodnoty (průměr všech, medián všech, průměr tří nejvyšších, medián tří nejvyšších) jsou tvarem velmi podobné a na první pohled v nich není vidět žádný velký rozdíl. Všechny mají směrem s větší chybou klesající tendenci, což jsme předpokládali, ale bohužel střed grafu v nízkých polohách, který odpovídá malým

chybám s nízkým ohodnocením, není až tak prázdný, jak jsme čekali. Kdybychom teď měli vybrat nějakou prahovou hodnotu skóre, podle které bychom určovali, zda se jedná o velkou či malou chybu, asi bychom žádnou příliš rozumnou hodnotu nenašli.

Jak bylo na obrázku 4.6 vidět, naneštěstí se poměrně hodně malých chyb vyskytuje v nízkých hodnotách skóre a naopak některé velké chyby nedostaly tak nízké ohodnocení, jak bylo požadováno. Pokud bychom si tedy vzali tento obrázek s průměrnými hodnotami skóre ze všech kandidátů a řekli bychom, že práh pro velké chyby bude například 10 a na všechny chyby, které mají průměrné skóre nižší než 10 bychom upozornili, dostaneme sice upozornění na cca 50 % velkých chyb, ale také upozorníme na malé chyby, které nejsou velkými chybami, a hlavně zbylých 50 % velkých chyb zůstane bez upozornění.

Pro lepší srovnání následuje tabulka 4.8 s počtem chyb, které mají skóre nižší než je určitá mez. To je poměrně důležitá informace, protože nám říká, kolik upozornění na velké chyby bychom dostali, pokud bychom zvolili určitý práh skóre. Jelikož velkých chyb není naštěstí příliš mnoho, bylo by ideální, kdyby i čísla v tabulce byla co nejnižší. Celkový počet všech chyb je 1570 a čísla v závorkách za jednotlivým počtem chyb udávají, kolik z nich je větších než 25 ms.

skóre	< 50	poměr	< 40	poměr	< 30	poměr	< 20	poměr	< 10	poměr
Průměr všech	1049 (43)	4,1%	828 (42)	5,0%	598 (38)	6,4%	315 (36)	11,4%	73 (17)	23,3%
Medián všech	1024 (43)	4,2%	851 (43)	5,0%	658 (39)	5,9%	382 (33)	8,6%	130 (18)	13,9%
Průměr tří	920 (42)	4,6%	718 (40)	5,6%	469 (30)	6,4%	229 (25)	10,9%	36 (9)	25,0%
Medián tří	903 (42)	4,7%	733 (40)	5,5%	517 (32)	6,2%	250 (23)	9,2%	52 (10)	19,2%
Pomocí SA	905 (39)	4,3%	754 (37)	4,9%	577 (33)	5,7%	343 (25)	7,3%	137 (13)	9,5%

Tabulka 4.8: Počet chyb se skórem nižším než 50/40/30/20/10

Bohužel, spousta malých chyb je ohodnocena nízkým skóre, a proto jsou celková čísla poměrně vysoká. Pro představu jsou v posledním řádku tabulky uvedeny počty chyb, které bychom dostali, pokud bychom pro detekci velkých chyb využili pouze SA.

Je třeba dodat, že statistiky ze tří nejvyšších hodnot mají z principu vyšší ohodnocení, a proto jejich počet pod určitou hranicí skóre bude vždy nižší než u statistik počítaných ze všech hodnot skóre. Nejdůležitějším parametrem z výše uvedené tabulky je tedy poměr

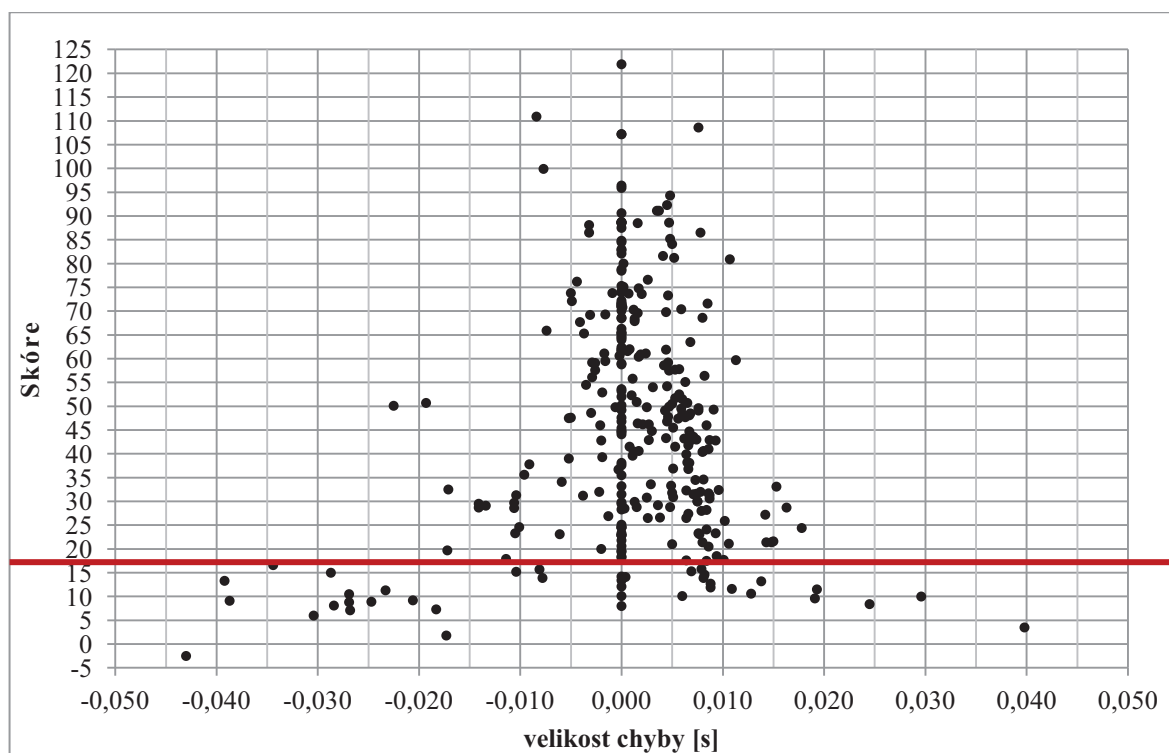
mezi celkovým počtem hranic, u kterých dojde k upozornění, a počtem chyb větších než 25 milisekund.

Nejzajímavější výsledky přináší prahy deset a dvacet. Nejlepší poměry u všech statistik dostáváme pro práh skóre rovný deseti, kde čtvrtina chyb, u kterých dojde k upozornění, je větších než 25 ms. V porovnání s hodnotami u prahu dvacet však dojde ke snížení počtu detekovaných chyb větších než 25 ms skoro ve všech případech o více než polovinu. Je tedy otázkou, zda chceme snížit pravděpodobnost špatného upozornění, tj. když se bude jednat o malou chybu, anebo jestli chceme detekovat většinu velkých chyb, ale za cenu velkého množství špatných upozornění. Ať tak nebo tak nejlepší výsledky pro rozlišení velkých a malých chyb dostáváme pro průměrové hodnoty.

4.3 Detekce velkých chyb pro jeden typ hranice

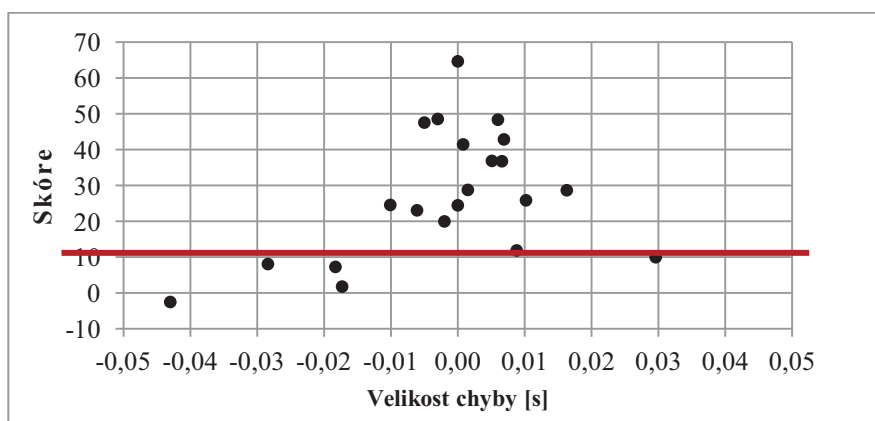
Zavedením statistických hodnot jsme docílili úspěšné identifikace velkých chyb, která je lepší, než kdybychom používali pouhou informaci o velikosti skóre automaticky určené hranice. Vzhledem k rozptřeni malých chyb i v nízké oblasti skóre však dochází k velkému počtu mylných detekcí, tj. za velkou chybu je někdy považována i malá chyba. Tyto nechtěné detekce by ale bylo možné zredukovat, pokud bychom k problému nepřistupovali jako k celku, ale práh bychom určovali zvlášť pro každý typ hranice. V poslední části tedy vyzkoušíme rozebrat závislost velikosti chyb a skóre na modelových typech hranic a to konkrétně na FRI-FRI, FRI-NLQ, FRI-PAU, FRI-VOW.

V předešlé části jsme experimentovali se čtyřmi druhy statistických hodnot a nejlepší výsledky nám přinášely průměrové hodnoty. Obecně lépe vycházel průměr ze všech hodnot, protože pro daný práh detekoval větší počet velkých chyb, a tak budeme nadále pracovat jen s touto statistickou hodnotou. Jako první uvedeme celkový pohled na závislost odchylek a skóre automaticky určených hranic ve spojení frikativ se všemi ostatními typy hranic a potom se podíváme na jednotlivé konkrétní typy.

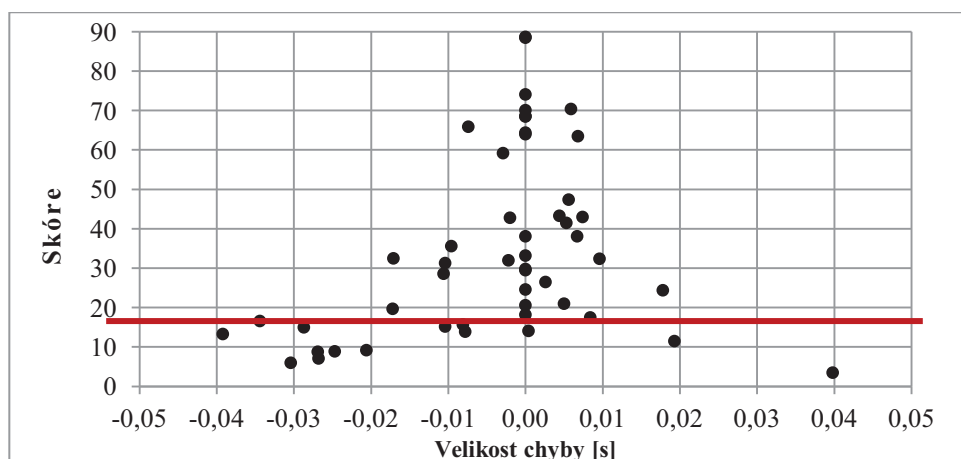


Obrázek 4.7: Závislost průměru hodnot skóre všech kandidátů na velikosti chyby pro skupinu FRI-*

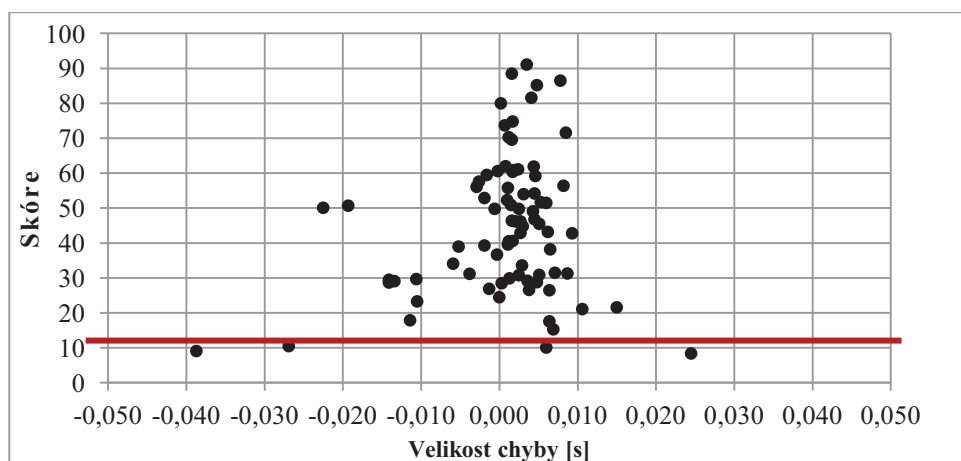
Z obrázku 4.7 máme ucelený pohled na skupinu frikativ a všech ostatních skupin a je dobře patrné, že pro detekci všech chyb nad 25 ms by stačila prahová hodnota skóre 17, jak je znázorněno červenou čarou. V tomto případě by počet celkových detekcí byl 39 a počet detekovaných chyb větších než 25 ms by byl 12, což je v poměru k celku přibližně 31 %. Na následujících obrázcích 4.8 až 4.10 bude zobrazen vždy jeden typ hranice a vyznačen příslušný práh, který se bude snažit zachytit co možná nejvíce chyb větších než 25 ms, ovšem s ohledem na co nejmenší počet chybných detekcí.



Obrázek 4.8: Závislost průměru hodnot skóre všech kandidátů na velikosti chyby pro skupinu FRI-FRI

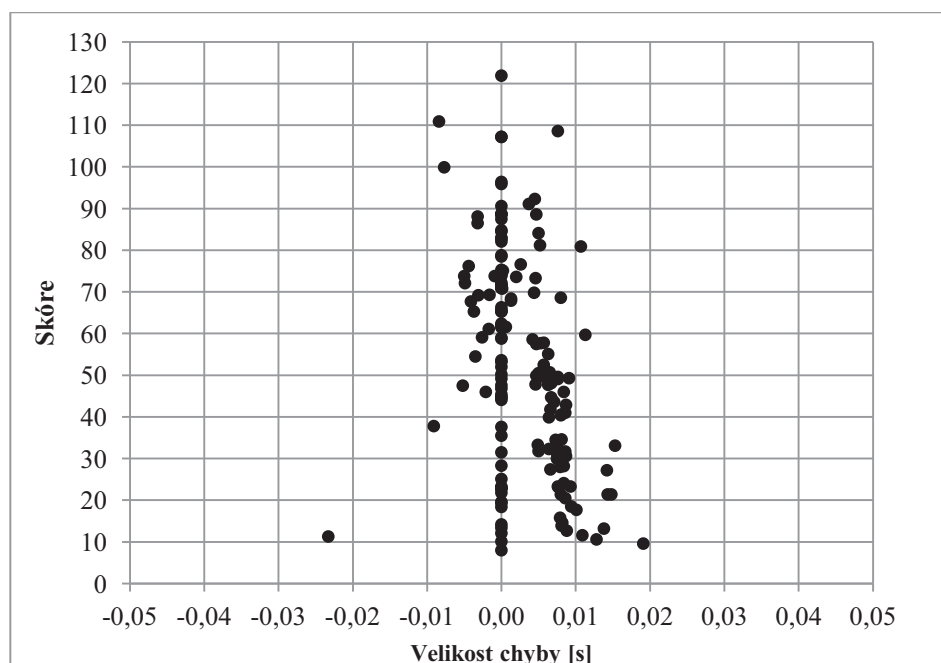


Obrázek 4.9: Závislost průměru hodnot skóre všech kandidátů na velikosti chyby pro skupinu FRI-NLQ



Obrázek 4.10: Závislost průměru hodnot skóre všech kandidátů na velikosti chyby pro skupinu FRI-PAU

Jak si můžeme všimnout na uvedených obrázcích, prahové hodnoty se pro různé hláskové skupiny liší a původní prahová hodnota použitá pro celou skupinu FRI-* může být v konkrétních případech ještě snížena. Tím dojde k omezení špatných detekcí a upozorňování na velké chyby tak bude přesnější. Obrázek 4.11 poslední skupiny FRI-VOW, který bude následovat, neobsahuje žádnou prahovou hodnotu, protože ze vzorků, které máme k dispozici, nevyplývá, že by docházelo k nějakým extrémně velkým chybám. Pro větší množství vzorků bychom možná nějakou větší chybu objevili a pak bychom se opět mohli pokusit stanovit nějakou prahovou hodnotu, podle které by se detekce řídila. Pro kompletní přehled jsou ještě v tabulce 4.9 sepsány souhrnné výsledky jednotlivých skupin.



Obrázek 4.11: Závislost průměru hodnot skóre všech kandidátů na velikosti chyby pro skupinu FRI-VOW

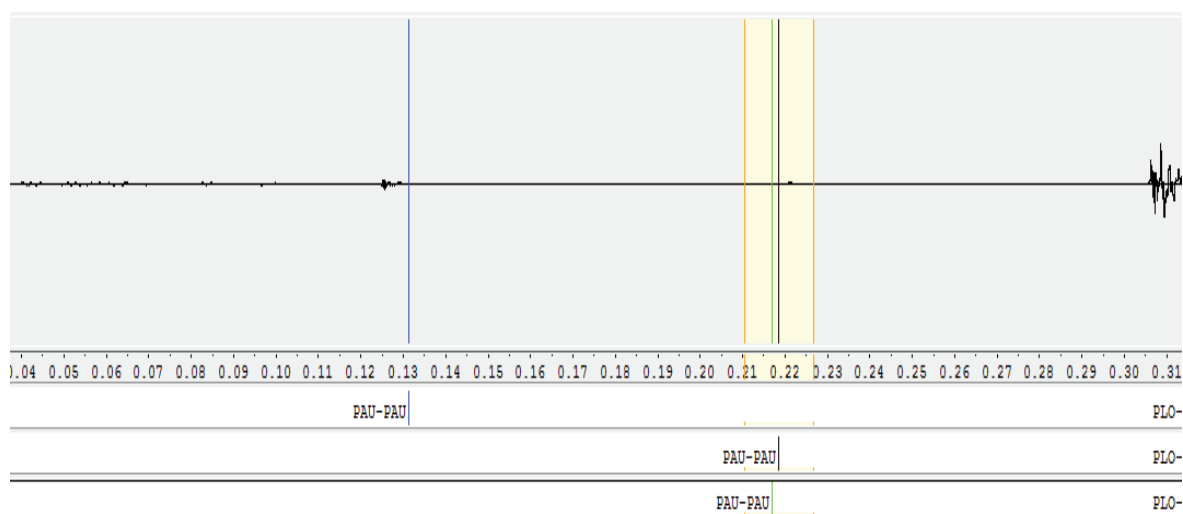
typ hranice	práh	detekovaných	nad 25 ms	poměr
FRI-FRI	11	5	3	60,0%
FRI-NLQ	17	14	7	50,0%
FRI-PAU	11	4	2	50,0%
FRI-VOW	-	-	-	-
celkem		23	12	52,2%
FRI-*	17	39	12	30,8%

Tabulka 4.9: Detekce velkých chyb u jednotlivých typů hranic

Vidíme, že určování prahů podle jednotlivých typů hranic značně zvyšuje poměr detekce velkých chyb. Smíchané chyby ze všech typů hranic mají různě vysoké skórové ohodnocení, které se napříč typy různě odlišuje. Když jsme se snažili získat jednu prahovou hodnotu pro všechny typy hranic najednou, bylo sice možné určit práh, pod kterým se vyskytovaly všechny velké chyby, avšak práh byl příliš vysoký. Díky tomu jsme detekovali i spoustu malých chyb, což bylo nežádoucí.

Původní šance, že detekovaná hranice bude velkou chybou, byla pro průměr hodnot skóre pouze 23,29 % (viz tabulka 4.8). Po zobecnění na typ FRI-* se šance zvýšila na 30,8 % a v konečném rozdělení na jednotlivé typy hranic se šance na detekci oproti původní hodnotě více než zdvojnásobila na výsledných 52,2 %.

Jako poslední je třeba zmínit největšího outliera, který se mezi výsledky vyskytoval, a to je automatická hranice s odchylkou cca 87 ms, které přísluší skóre o hodnotě 73,8. Tato odchylka patří pod typ PAU-PAU a je to první hranice ze začátku promluvy, než člověk vůbec začne mluvit. Na obrázku 4.12 je referenční hranice znázorněna modře, automatická hranice černě a opravená hranice zeleně. Žlutě je vyznačeno okolí, ze kterého se vybírali kandidáti na opravu.



Obrázek 4.12: Největší outlier

Vzhledem k tomu, že signál je všude téměř konstantní, akustické vlastnosti hranic budou víceméně shodné, ať je hranice umístěna prakticky kamkoliv. V takovém případě může automatická hranice dostat poměrně vysoké ohodnocení, i když se nachází velmi daleko od referenční hranice a s takovou chybou si pak bohužel tento přístup neporadí.

V tomto případě bychom tedy možná mohli diskutovat o tom, jestli referenční hranice opravdu musela být umístěna tak daleko od samotného začátku promluvy a zda zde nedošlo k nějaké chybě v referenčních datech. Referenční data jsou však pro nás jasně daná, protože byla určena člověkem-expertem, a i když se může stát, že dojde k nějaké nekonzistenci, nám nezbývá než těmto datům důvěřovat.

5 Závěr

Seznámili jsme se s algoritmy automatické fonetické segmentace řeči a s možnostmi její automatické korekce. Víme, že fonetická segmentace řeči je proces, při kterém vznikají řečové jednotky, které jsou základním stavebním prvkem v syntéze řeči pomocí konkatenční metody. Kvalita fonetické segmentace tak ovlivňuje kvalitu výsledné syntetizované řeči a proto je obecnou snahou co nejpřesněji určit hranice řečových jednotek. Algoritmy, které provádějí segmentaci, však mohou při určování hranic řečových jednotek udělat chybu, a z toho důvodu bývá snahou výslednou segmentaci ještě opravovat pomocí korekčních programů.

Prvním cílem této práce bylo vyvinutí metod, které by zkvalitnily práci korekčního programu navrženého v rámci mé bakalářské práce, který pracuje na bázi regrese. Program byl rozvinut tak, aby opravoval všechny typy hláskových skupin a byla otestována řada postupů, které měly za cíl jeho vylepšení. Mezi testované postupy patřilo zavedení dynamiky, omezení maximální odchylky, dvoufázové vyhledávání parametrů pro regresní SPM, cross-validace a oprava hranic mezi konkrétními hláskami. Dále pak byly provedeny další experimenty na hranicích mezi konkrétními hláskami týkající se opravy podle velikosti skóre, úpravy maximální odchylky, opravy dalších konkrétních přechodů a snížení počtu složek cross-validace. Jak bylo v práci rozebráno, ne všechny experimenty měly příznivý dopad na výsledek korekce, a proto některé z nich nebyly do programu zahrnuty. Většina provedených experimentů nás však posunula blíže k lepšímu výsledku korekce. Z celkových 1570 opravovaných hranic se jich zlepšilo 637 (40,6 %), zhoršilo 293 (18,7 %) a nezměnilo 640 (40,8 %). Celková chyba segmentace tak klesla o 26,3 % z původních 9,9580 s na 7,3377 s.

Druhým cílem práce bylo prozkoumat činnost korekčního programu v kontextu klasifikační úlohy. Došlo k výměně rozhodovací složky, kdy jsme místo regresního SPM dosadili klasifikátor, a úlohu korekce jsme tak převedli na klasifikační problém. Byly otestovány tři typy klasifikátorů – SVM klasifikátor, klasifikátor podle k-nejbližších sousedů a klasifikátor navržený podle metody extremely randomized trees. Všechny

testované klasifikátory podávaly oproti regresnímu přístupu o něco horší výsledky ve velikosti celkové segmentační chyby. Také počet správně opravených hranic, který se pohyboval jen okolo 20 %, byl nižší než u regresního přístupu. Na druhou stranu však došlo k menšímu počtu zhoršených hranic. Ten se u korekce pomocí klasifikátorů pohyboval jen kolem 5 % a byl tak výrazně menší než u regresního přístupu. Z těchto výsledků tedy usuzujeme, že korekce jako klasifikační úloha by byla vhodná tam, kde by byl kladen důraz na co nejmenší počet zhoršených hranic.

Posledním cílem bylo navrhnout metodiku značení velkých segmentačních chyb. Velké chyby jsou závažným problémem, protože korekční program je z principu své funkčnosti neumí efektivně opravovat. Proto by bylo dobré na ně alespoň upozornit a případně k nim pak přistupovat individuální opravou. V průběhu práce byla zjištěna určitá závislost velkých chyb na velikosti skóre, které predikuje regresní SPM. Ukázalo se totiž, že velké chyby na rozdíl od malých chyb nedostávají tak velké skóre, a proto byla navržena detekce velkých chyb podle nízkého skóre. Bohužel se stává, že i některé malé chyby dostanou nízkou hodnotu skóre a některé velké chyby dostanou naopak vyšší hodnotu skóre, což komplikuje práci detektoru. Tento postup byl nejprve otestován na hranicích frikativ a všech ostatních hláskových skupin, tj. FRI-*, a výsledkem byla detekce velkých chyb s úspěšností 30,8 %. Takto malá úspěšnost byla způsobena tím, že hranice mezi různými hláskovými skupinami dostávají různě vysoké skóre. Následně bylo provedeno testování pro jednotlivé hranice typu FRI-FRI, FRI-NLQ, FRI-PAU, FRI-VOW. Tady už úspěšnost detekce byla téměř dvojnásobná a pohybovala mezi 50 až 60 %.

Byly úspěšně splněny všechny cíle práce a v jejím průběhu byly navrženy i některé další postupy na zlepšení korekce, které byly nad rámec této práce (část 3.1.6 poslední odstavec a část 3.1.7 poslední odstavec v rámci opravy dle skóre). Možný potenciál dalšího zlepšení by se mohl skrývat v navržené metodě pro detekci velkých chyb, kdyby se podařilo jejím dalším zpracováním dosáhnout ještě větší detekční úspěšnosti. Lepších výsledků při detekci bychom možná mohli dosáhnout při provedení detekce velkých chyb pouze na hranicích mezi konkrétními hláskami. U některých typů hranic jsme se také setkali s velmi malým počtem referenčních hranic, a proto by mohl ke zlepšení výsledků přispět i větší počet referenčních dat, protože bychom mohli kvalitněji natrénovat jednotlivé estimátory.

Citovaná literatura

- [1] J. PSUTKA, L. MÜLLER, J. MATOUŠEK a V. RADOVÁ, Mluvíme s počítačem česky, Praha: Academia, 2006.
- [2] J. PSUTKA, Komunikace s počítačem mluvenou řečí, Praha: Academia, 1995, p. 287.
- [3] C.-C. CHANG a C.-J. LIN, „LIBSVM : a library of support vector machines,“ 2011. [Online]. Available: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [4] C.-Y. LIN a J.-S. JANG, Automatic Phonetic Segmentation by Score Predictive Model for the Corpora of Mandarin Singing Voices, 2007, p. 2155.
- [5] „MetaCentrum VO,“ [Online]. Available: <http://metavo.metacentrum.cz/>.
- [6] M. MATURA, Automatická korekce fonetické segmentace řeči : bakalářská práce, Plzeň: Západočeská univerzita, Fakulta aplikovaných věd, 2010.
- [7] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research* 12, pp. 2825-2830, 2011.

Přílohy

A Tabulky

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I 0	1,5916	1,5847	7,062	0,187	1	5%	19	90%	1	5%	21
J-I 1	0,4494	0,4312	8,636	7,687	3	13%	19	83%	1	4%	23
J-I 2	0,2238	0,2001	7,907	0,000	3	14%	19	86%	0	0%	22
J-I 3	0,2786	0,2195	14,312	6,233	5	19%	19	73%	2	8%	26
J-I 4	1,5490	1,5331	10,406	2,421	2	9%	18	82%	2	9%	22
J-I 5	1,7302	1,7199	10,313	0,000	1	5%	20	95%	0	0%	21
J-I 6	1,3493	1,3572	0,000	3,937	0	0%	19	90%	2	10%	21
J-I 7	1,5757	1,5685	7,250	0,000	1	5%	20	95%	0	0%	21
J-I 8	1,9930	1,9858	7,250	0,000	1	4%	22	96%	0	0%	23
J-I 9	2,4054	2,4054	0,000	0,000	0	0%	21	100%	0	0%	21
PRŮMĚR:	1,3146	1,3005	7,314	2,047	1,7	8%	19,6	89%	0,8	4%	22,1

Tab. 1: Konkrétní hranice J-I s opravou podle skóre většího než 50

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I 0	1,5916	1,5765	7,656	0,187	2	10%	18	86%	1	5%	21
J-I 1	0,4494	0,4312	8,636	7,687	3	13%	19	83%	1	4%	23
J-I 2	0,2238	0,1611	12,544	0,000	5	23%	17	77%	0	0%	22
J-I 3	0,2786	0,2426	14,312	8,890	5	19%	17	65%	4	15%	26
J-I 4	1,5490	1,4958	11,625	2,421	5	23%	15	68%	2	9%	22
J-I 5	1,7302	1,6795	16,896	0,000	3	14%	18	86%	0	0%	21
J-I 6	1,3493	1,3413	7,895	5,229	3	14%	15	71%	3	14%	21
J-I 7	1,5757	1,5258	11,550	7,812	5	24%	15	71%	1	5%	21
J-I 8	1,9930	1,9586	10,141	6,125	4	17%	18	78%	1	4%	23
J-I 9	2,4054	2,3792	20,094	6,969	2	10%	17	81%	2	10%	21
PRŮMĚR:	1,3146	1,2792	12,135	4,532	3,7	17%	16,9	76%	1,5	7%	22,1

Tab. 2: Konkrétní hranice J-I s opravou podle skóre většího než 40

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
J-I 0	1,5916	1,6638	8,265	13,156	4	19%	9	43%	8	38%	21
J-I 1	0,4494	0,4436	8,589	15,250	6	26%	14	61%	3	13%	23
J-I 2	0,2238	0,2186	12,272	23,512	10	45%	7	32%	5	23%	22
J-I 3	0,2786	0,3833	14,312	16,019	5	19%	10	38%	11	42%	26
J-I 4	1,5490	1,5921	12,539	15,926	8	36%	5	23%	9	41%	22
J-I 5	1,7302	1,7977	13,141	28,760	8	38%	7	33%	6	29%	21
J-I 6	1,3493	1,3964	9,312	10,410	5	24%	7	33%	9	43%	21
J-I 7	1,5757	1,6125	12,571	13,868	7	33%	5	24%	9	43%	21
J-I 8	1,9930	2,0164	11,437	18,045	9	39%	7	30%	7	30%	23
J-I 9	2,4054	2,2901	13,662	7,104	10	48%	8	38%	3	14%	21
PRŮMĚR:	1,3146	1,3415	11,610	16,205	7,2	33%	7,9	36%	7,0	32%	22,1

Tab. 3: Konkrétní hranice J-I s opravou podle skóre většího než 20

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
n-a 0	0,8594	0,8060	8,508	7,375	8	44%	8	38%	2	11%	18
n-a 1	2,1194	2,0783	7,458	9,314	8	47%	7	41%	2	12%	17
n-a 2	1,0340	0,9589	10,734	0,000	7	41%	10	59%	0	0%	17
n-a 3	0,9133	0,8254	11,356	14,312	9	53%	7	41%	1	6%	17
n-a 4	0,8114	0,7494	7,752	0,000	8	47%	9	53%	0	0%	17
n-a 5	2,3273	2,2468	8,947	8,938	11	61%	5	28%	2	11%	18
n-a 6	3,0832	3,0140	9,119	12,876	9	53%	7	41%	1	6%	17
n-a 7	2,0192	1,9570	8,884	0,000	7	39%	11	61%	0	0%	18
n-a 8	0,5251	0,4866	7,604	7,188	6	33%	11	61%	1	6%	18
n-a 9	2,7576	2,7243	7,882	7,001	6	35%	9	53%	2	12%	17
PRŮMĚR:	1,6450	1,5847	8,824	6,700	7,9	45%	8,4	48%	1,1	6%	17,4

Tab. 4: Konkrétní hranice n-a s opravou podle skóre většího než 50

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
n-a 0	0,8594	0,8160	8,508	8,250	8	44%	7	33%	3	17%	18
n-a 1	2,1194	2,0661	7,983	9,314	9	53%	6	35%	2	12%	17
n-a 2	1,0340	0,9342	9,976	0,000	10	59%	7	41%	0	0%	17
n-a 3	0,9133	0,8254	11,356	14,312	9	53%	7	41%	1	6%	17
n-a 4	0,8114	0,7328	7,858	0,000	10	59%	7	41%	0	0%	17
n-a 5	2,3273	2,2468	8,947	8,938	11	61%	5	28%	2	11%	18
n-a 6	3,0832	3,0212	9,119	10,032	9	53%	6	35%	2	12%	17
n-a 7	2,0192	1,9622	8,884	5,188	7	39%	10	56%	1	6%	18
n-a 8	0,5251	0,4596	8,070	7,188	9	50%	8	44%	1	6%	18
n-a 9	2,7576	2,7028	8,599	7,001	8	47%	7	41%	2	12%	17
PRŮMĚR:	1,6450	1,5767	8,930	7,022	9,0	52%	7,0	40%	1,4	8%	17,4

Tab. 5: Konkrétní hranice n-a s opravou podle skóre většího než 40

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
n-a 0	0,8594	0,7971	9,667	8,250	9	50%	6	29%	3	17%	18
n-a 1	2,1194	2,0661	7,983	9,314	9	53%	6	35%	2	12%	17
n-a 2	1,0340	0,8989	11,262	0,000	12	71%	5	29%	0	0%	17
n-a 3	0,9133	0,8254	11,356	14,312	9	53%	7	41%	1	6%	17
n-a 4	0,8114	0,7139	8,865	0,000	11	65%	6	35%	0	0%	17
n-a 5	2,3273	2,2468	8,947	8,938	11	61%	5	28%	2	11%	18
n-a 6	3,0832	3,0212	9,119	10,032	9	53%	6	35%	2	12%	17
n-a 7	2,0192	1,9550	8,672	5,188	8	44%	9	50%	1	6%	18
n-a 8	0,5251	0,4596	8,070	7,188	9	50%	8	44%	1	6%	18
n-a 9	2,7576	2,7028	8,599	7,001	8	47%	7	41%	2	12%	17
PRŮMĚR:	1,6450	1,5687	9,254	7,022	9,5	55%	6,5	37%	1,4	8%	17,4

Tab. 6: Konkrétní hranice n-a s opravou podle skóre většího než 30

Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2239	3,256	5,085	5	24%	13	62%	3	14%	21
FRI-NLQ	0,4870	0,4253	8,137	9,271	11	22%	37	73%	3	6%	51
FRI-PAU	0,4619	0,4113	3,392	2,230	32	40%	22	28%	26	33%	80
FRI-VOW	0,6668	0,4386	5,602	3,619	53	33%	91	56%	19	12%	163
NLQ-FRI	0,1255	0,1046	5,347	0,438	4	19%	16	76%	1	5%	21
NLQ-NLQ	0,2144	0,2144	0,000	0,000	0	0%	29	100%	0	0%	29
NLQ-PAU	0,3617	0,2255	7,441	4,183	20	43%	24	51%	3	6%	47
NLQ-VOW	1,9411	1,2881	9,047	5,454	77	34%	140	62%	8	4%	225
PAU-FRI	0,0438	0,0646	0,683	3,914	4	27%	5	33%	6	40%	15
PAU-NLQ	0,1119	0,1061	2,673	2,584	8	53%	1	7%	6	40%	15
PAU-PAU	0,4933	0,4799	2,685	0,000	5	21%	19	79%	0	0%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2439	4,819	4,655	5	23%	14	64%	3	14%	22
PLO-NLQ	0,1874	0,0690	7,222	5,756	18	34%	33	62%	2	4%	53
PLO-PAU	0,1746	0,1371	5,899	3,907	11	33%	15	45%	7	21%	33
PLO-VOW	0,8970	0,4072	6,385	5,778	93	49%	77	41%	18	10%	188
VOW-FRI	0,6728	0,6395	7,155	6,589	24	15%	116	72%	21	13%	161
VOW-NLQ	1,4056	1,2996	8,500	10,357	21	12%	146	84%	7	4%	174
VOW-PAU	1,1957	0,9327	6,902	7,060	78	32%	126	52%	39	16%	243
VOW-VOW	0,0386	0,0386	0,000	0,000	0	0%	5	100%	0	0%	5
Skóre > 50	9,9581	7,7496	4,757	4,044	469	30%	929	59%	172	11%	1570

Tab. 7: Výsledky korekce všech kombinací skupin podle skóre většího než 50

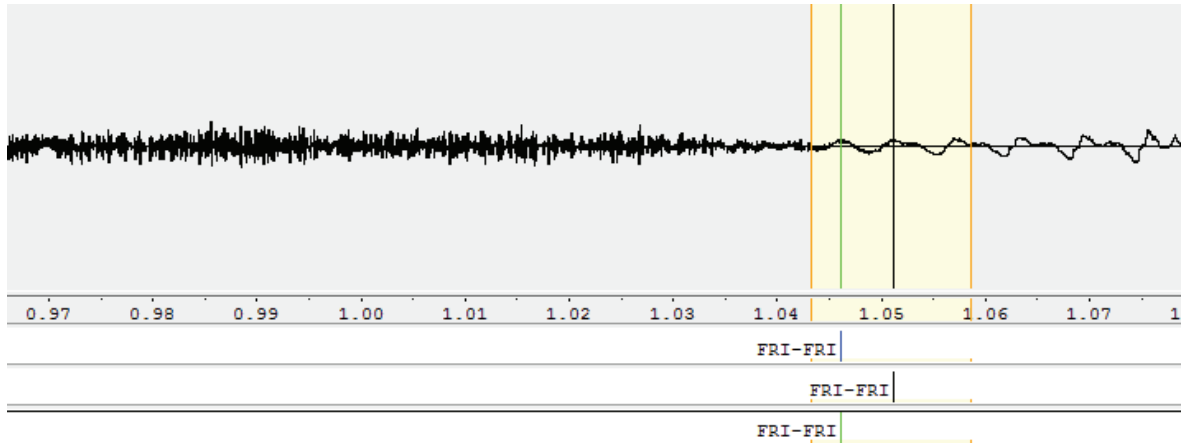
Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2231	4,138	5,429	7	33%	9	43%	5	24%	21
FRI-NLQ	0,4870	0,4175	8,571	8,426	14	27%	31	61%	6	12%	51
FRI-PAU	0,4619	0,4205	3,352	2,299	35	44%	12	15%	33	41%	80
FRI-VOW	0,6668	0,3900	5,758	3,619	60	37%	84	52%	19	12%	163
NLQ-FRI	0,1255	0,0820	8,790	0,438	5	24%	15	71%	1	5%	21
NLQ-NLQ	0,2144	0,2206	0,000	6,250	0	0%	28	97%	1	3%	29
NLQ-PAU	0,3617	0,2310	7,369	7,696	24	51%	17	36%	6	13%	47
NLQ-VOW	1,9411	1,2013	9,295	6,731	89	40%	123	55%	13	6%	225
PAU-FRI	0,0438	0,0691	1,168	3,891	5	33%	2	13%	8	53%	15
PAU-NLQ	0,1119	0,1061	2,673	2,584	8	53%	1	7%	6	40%	15
PAU-PAU	0,4933	0,5083	7,457	24,258	11	46%	9	38%	4	17%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2321	3,590	4,655	10	45%	9	41%	3	14%	22
PLO-NLQ	0,1874	0,0641	7,095	5,756	19	36%	32	60%	2	4%	53
PLO-PAU	0,1746	0,1519	5,827	4,824	13	39%	9	27%	11	33%	33
PLO-VOW	0,8970	0,3963	6,452	5,649	96	51%	71	38%	21	11%	188
VOW-FRI	0,6728	0,6311	7,755	6,541	29	18%	104	65%	28	17%	161
VOW-NLQ	1,4056	1,2206	9,348	9,516	32	18%	130	75%	12	7%	174
VOW-PAU	1,1957	0,9108	6,969	6,824	83	34%	117	48%	43	18%	243
VOW-VOW	0,0386	0,0386	0,000	0,000	0	0%	5	100%	0	0%	5
Skóre > 40	9,9581	7,5151	5,280	5,769	540	34%	808	51%	222	14%	1570

Tab. 8: Výsledky korekce všech kombinací skupin podle skóre většího než 40

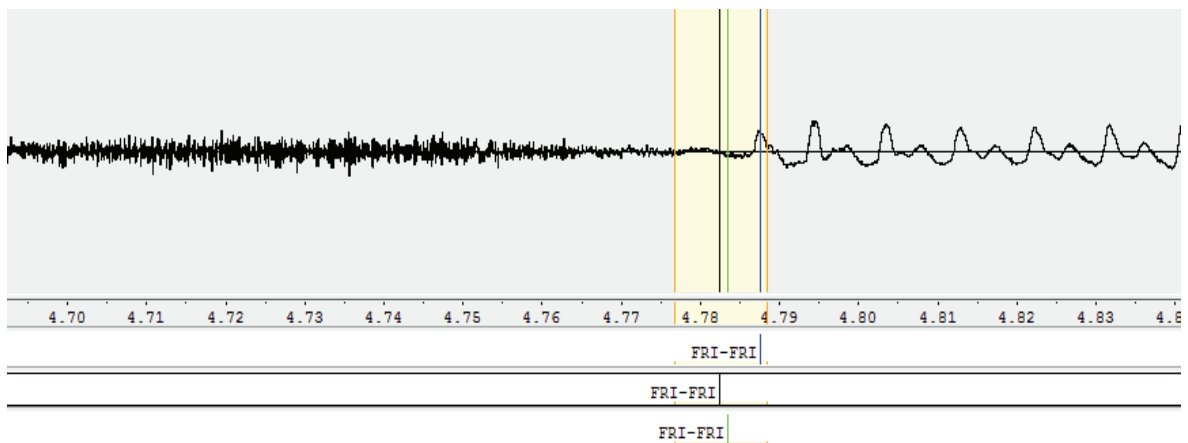
Typ hranice	Čas chyby [s]		Průměrné [ms]		Kolik hranic bylo						Celkem hranic
	ref-auto	ref-fix	zlepšení	zhoršení	zlepšeno		nezměněno		zhoršeno		
FRI-FRI	0,2249	0,2167	4,667	4,159	8	38%	6	29%	7	33%	21
FRI-NLQ	0,4870	0,3529	9,717	8,426	19	37%	26	51%	6	12%	51
FRI-PAU	0,4619	0,4534	3,352	2,789	35	44%	6	8%	39	49%	80
FRI-VOW	0,6668	0,3624	5,975	3,711	64	39%	78	48%	21	13%	163
NLQ-FRI	0,1255	0,0714	9,085	0,438	6	29%	14	67%	1	5%	21
NLQ-NLQ	0,2144	0,2076	7,292	7,562	3	10%	24	83%	2	7%	29
NLQ-PAU	0,3617	0,2314	7,509	8,208	25	53%	15	32%	7	15%	47
NLQ-VOW	1,9411	1,1334	9,567	6,519	96	43%	112	50%	17	8%	225
PAU-FRI	0,0438	0,0821	1,674	5,368	6	40%	0	0%	9	60%	15
PAU-NLQ	0,1119	0,1061	2,673	2,584	8	53%	1	7%	6	40%	15
PAU-PAU	0,4933	0,5516	7,608	24,619	15	63%	2	8%	7	29%	24
PAU-VOW	0,0000	0,0000	0,000	0,000	0	0%	0	0%	0	0%	0
PLO-FRI	0,2540	0,2379	3,325	5,929	12	55%	6	27%	4	18%	22
PLO-NLQ	0,1874	0,0660	7,044	6,483	20	38%	30	57%	3	6%	53
PLO-PAU	0,1746	0,1596	5,452	5,104	14	42%	7	21%	12	36%	33
PLO-VOW	0,8970	0,4140	6,649	6,495	100	53%	60	32%	28	15%	188
VOW-FRI	0,6728	0,6535	7,429	6,861	34	21%	93	58%	34	21%	161
VOW-NLQ	1,4056	1,1223	9,930	8,456	43	25%	114	66%	17	10%	174
VOW-PAU	1,1957	0,9048	6,999	6,760	86	35%	111	46%	46	19%	243
VOW-VOW	0,0386	0,0386	0,000	0,000	0	0%	5	100%	0	0%	5
Skóre > 30	9,9581	7,3656	5,797	6,024	594	38%	710	45%	266	17%	1570

Tab. 9: Výsledky korekce všech kombinací skupin podle skóre většího než 30

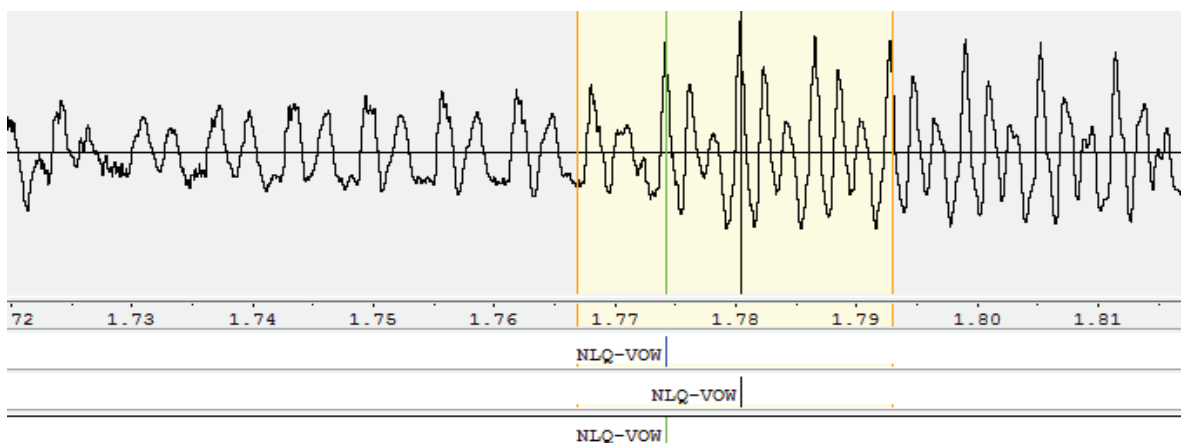
B Obrázky



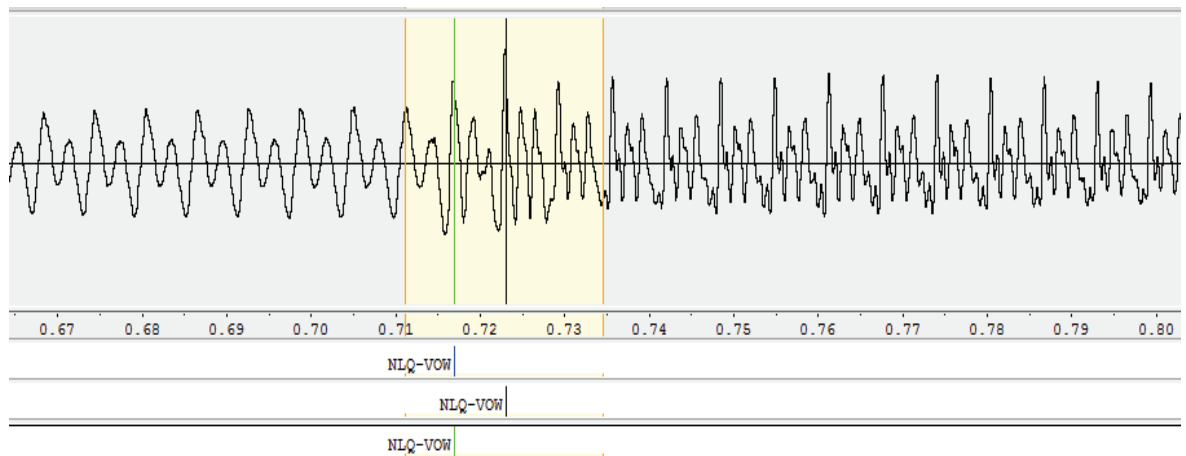
Obr. 1: Druhá malá chyba pro hranici FRI-FRI (žlutě oblast pro výběr kandidátů)



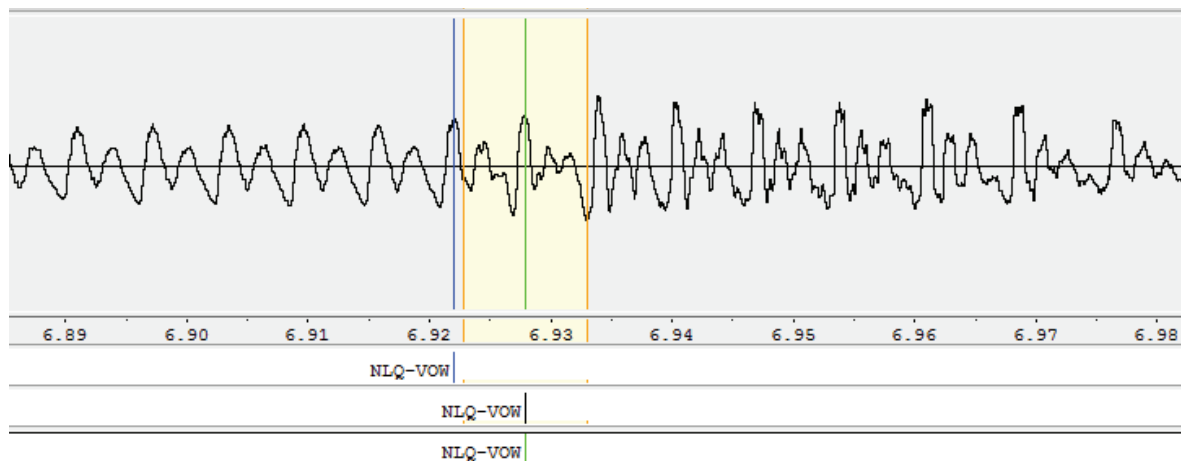
Obr. 2: Třetí malá chyba pro hranici FRI-FRI (žlutě oblast pro výběr kandidátů)



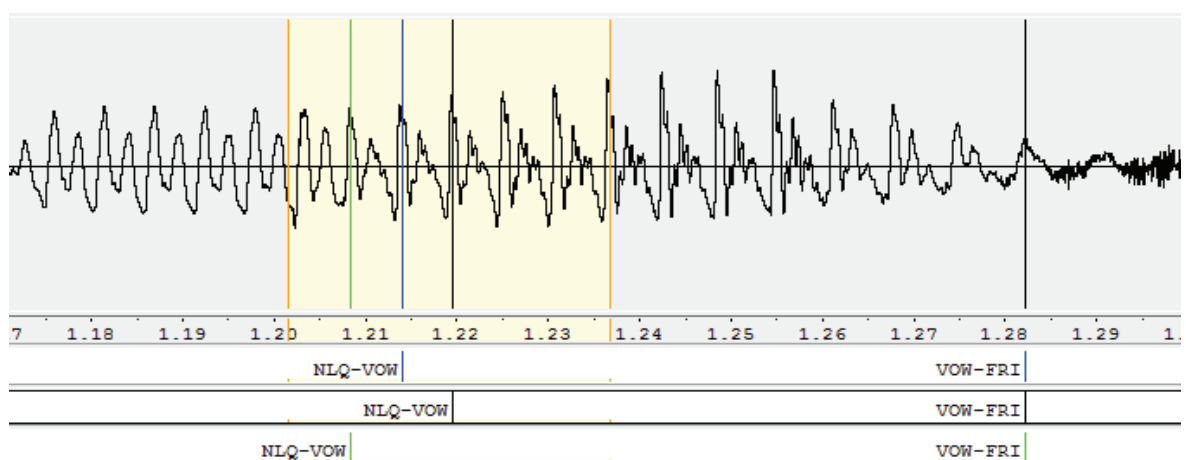
Obr. 3: První malá chyba pro hranici NLQ-VOW (žlutě oblast pro výběr kandidátů)



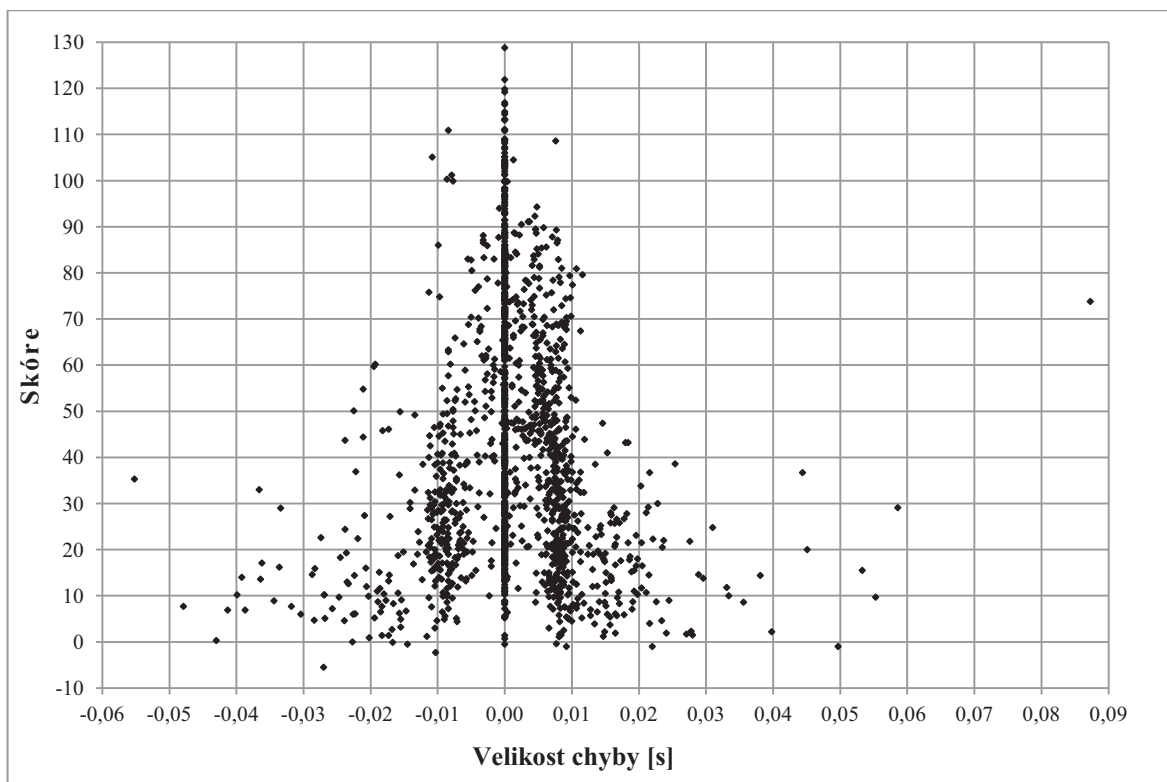
Obr. 4: Druhá malá chyba pro hranici NLQ-VOW (žlutě oblast pro výběr kandidátů)



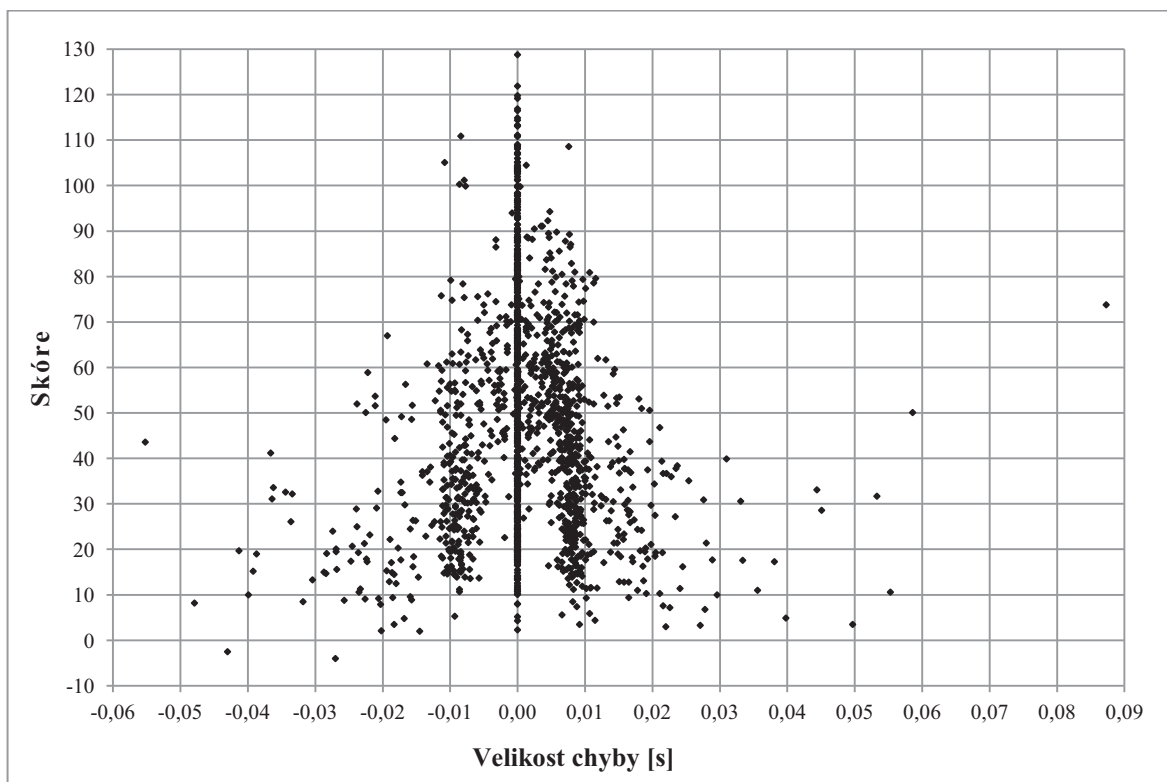
Obr. 5: Třetí malá chyba pro hranici NLQ-VOW (žlutě oblast pro výběr kandidátů)



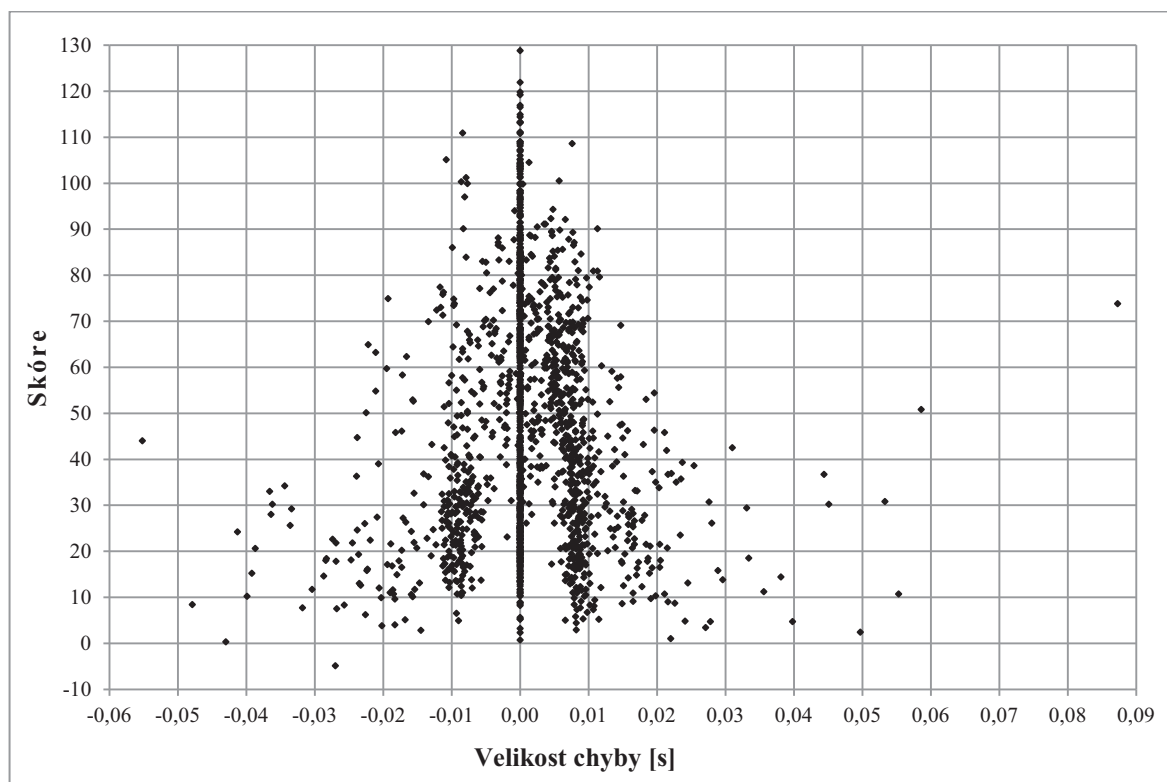
Obr. 6: Čtvrtá malá chyba pro hranici NLQ-VOW (žlutě oblast pro výběr kandidátů)



Obr. 7: Závislost mediánu hodnot skóre všech kandidátů na velikosti chyby



Obr. 8: Závislost průměru tří nejvyšších hodnot skóre na velikosti chyby



Obr. 9: Závislost mediánu tří nejvyšších hodnot skóre na velikosti chyby

C Obsah přiloženého CD

složka *klasifikace* s podsložkami:

- data – řečové promluvy a soubory s pitchmarky
- KOREKCE – dílčí opravené hranice a jejich třídy a odchylky
- pomocne_programy – dodatečné skripty a exe soubory užívané při práci programu
- skripty – vlastní program pro automatickou korekci
- TEST – předzpracovaná data k opravě
- TRENINK – předzpracovaná data pro trénink SPM
- vstup – vstupní MLF soubory
- VYSTUP – konečné výstupy

složka *regrese* s podsložkami:

- data – řečové promluvy a soubory s pitchmarky
- KOREKCE – dílčí opravené hranice a jejich skóre a odchylky
- pomocne_programy – dodatečné skripty a exe soubory užívané při práci programu
- skripty – vlastní program pro automatickou korekci
- TEST – předzpracovaná data k opravě
- TRENINK – předzpracovaná data pro trénink klasifikátorů
- vstup – vstupní MLF soubory
- VYSTUP – konečné výstupy

soubor *Diplomova_prace.pdf*

soubor *Uzivatelaska_prirucka.pdf*