

Prohlášení

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

Prohlašuji, že jsem zadanou diplomovou práci zpracoval zcela samostatně, pouze s použitím literatury a pramenů, jejichž úplný seznam je její součástí a za odborného vedení vedoucího diplomové práce.

V Plzni dne

.....

podpis diplomanta

Anotace

Tato diplomová práce se zabývá vyšetřováním relativní účinnosti různých metod normalizace skóre v systémech verifikace řečníka podle hlasu. Rozpoznávání podle hlasu patří mezi tzv. biometrické identifikační techniky. Na úlohu rozpoznávání působí množství poruch. Pomocí normalizačních metod se je snažíme odstranit.

Experimenty byly založeny na různě velkých korpusech dat, mimo jiné korpus Timit a Yoho. Byly testovány off-line i on-line metody, metody založené na Bayesovském způsobu i pomocí standardizace rozložení pravděpodobnosti skóre. K porovnání se používal model pozadí.

Jako nejlepší normalizace vyšla metoda UCN (neomezený skupinový model). Chyba EER při normalizaci modelem pozadí byla redukována použitím normalizační metody UCN z původních 0,63% na 0,1%.

Klíčová slova:

Biometrické charakteristiky, normalizace skóre, normalizace Bayesovským způsobem, normalizace standardizací rozložení pravděpodobnosti skóre, MFCC, GMM, CN, UCN, UCEN, CIC, T-normalizace, Z-normalizace, model pozadí, SoAP

Annotation

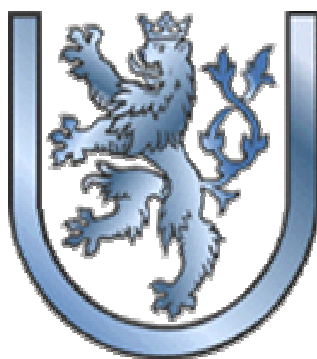
This diploma paper presents an investigation into relative effectiveness of various score normalization methods in speaker verification system. Speaker verification is so-called biometrical identification methods. For this task is affected much faults. It is possible to minimalize these faults with the aid of score normalization.

The experimental investigations are based on the use of various speech materials, e.g. Timit database or Yoho database. For tests of relative effectiveness were used off-line and on-line methods, Bayesian solution and Standardizing a score distribution. For comparing improvement of effectiveness was used a Background Model (BGM).

The best normalization was found Unconstraint Cohort Normalization (UCN). Equal Error Rate (EER) when using BGM was reduced from 0,63% to 0,1% at using UCN.

Key words:

Biometrical characteristics, score normalization, normalization with Bayesian solution, normalization with Standardizing a score distribution, MFCC, GMM, CN, UCN, UCEN, CIC, T-norm, Z-norm, Background Model, SoAP



**Západočeská Univerzita
Fakulta Aplikovaných Věd
Katedra Kybernetiky**

**Metody normalizace skóre
v úloze verifikace řečníka**

Diplomová práce

Obsah

1. Úvod	3
1.1. Cíle diplomové práce	3
1.2. Obsah jednotlivých kapitol	3
2. Úlohy v problému rozpoznávání řečníka	4
2.1. Biometrické metody verifikace	4
2.2. Rozdělení úlohy rozpoznávání	5
2.3. Postup rozpoznávání	6
2.4. Důvody nutnosti normalizace	6
3. Systém pro rozpoznání řečníka	8
3.1. Extrakce příznaků	8
3.1.1. Melovské keprální koeficienty (MFCC)	8
3.2. Reprezentace modelu řečníka	11
3.2.1. Směs Gaussovských rozložení (GMM)	11
3.3. Úloha rozpoznávání	13
3.3.1. Verifikace	13
3.3.2. Identifikace v uzavřené množině	13
3.3.3. Identifikace v otevřené množině	14
4. Normalizace skóre	15
4.1. Bayesovský způsob	15
4.1.1. Světový model	16
4.1.2. Skupinový model	16
4.1.3. Neomezený skupinový model	17
4.1.4. Skupinový model s extrapolací	17
4.1.5. Neomezený skupinový model s extrapolací	18
4.1.6. Robustní neomezený skupinový model s extrapolací	18
4.1.7. Virtuální řečník	18
4.2. Standardizace rozložení pravděpodobnosti skóre	19
4.2.1. Zero normalizace (Z-norm)	20
4.2.2. Testová normalizace (T-norm)	20
4.3. Výběr skupiny	21
4.3.1. Kritéria pro srovnání řečníků	21
4.3.2. Metody výběru skupiny	22
4.4. Druhy chyb ve verifikaci řečníka	23
5. Programové řešení	25
5.1. Stručný popis SoAP	25
5.2. Popis vytvořeného verifikačního modulu Verify_BGM2	26
5.2.1. Vstupní/výstupní soubory verifikačního modulu	26
5.3. Popis programů pro vytvoření kohorty	28
5.3.1. Sort_LogLike.exe	28
5.3.2. Sort_Models.exe	29
5.4. Výpočet EER CompEER_KW.exe	29

6. Experimenty	30
6.1. Testované korpusy dat	30
6.1.1. UWB-RobustVerify100CZ	30
6.1.2. DoD	30
6.1.3. Timit	31
6.1.4. Yoho	31
6.2. Použitá parametrizace	31
6.3. Použité modelování	32
6.4. Výsledky experimentů	33
6.4.1. UWB-RobustVerify100CZ	34
6.4.2. DoD	36
6.4.3. Timit	37
6.4.4. Yoho	38
6.5. Časová náročnost algoritmů	38
7. Závěr	40
Seznam literatury	41
Přílohy	42

1. Úvod

Verifikace řečníka je jedna z úloh v problému rozpoznávání řečníka podle hlasu. Jde o jednoznačné rozhodnutí o identitě člověka, v naší úloze označovaného za řečníka. Rozpoznávání podle hlasu patří mezi tzv. **biometrické identifikační techniky** [3,4]. Jsou to identifikační techniky založené na určitých vnitřních charakteristikách osoby (např. hlas, otisky prstů, geometrie ruky apod.). V úloze se předpokládá, že tyto vnitřní charakteristiky jsou pro každého jedinečné a nemohou se vyskytovat u nikoho jiného. Tím lze podle nich jednoznačně identifikovat danou osobu.

Systemy pro rozpoznávání osoby podle biometrických charakteristik mohou využít více různých vnitřních rysů člověka. Pokud je však dostupný pouze hlas rozpoznávané osoby (řečníka), musí se systém spolehnout při identifikaci pouze na vyslovenou promluvu. Takové úkoly jsou řešeny **systemem rozpoznání řečníka dle hlasu**.

Pro systémy provádějící identifikaci podle hlasu řečníka je nutná jejich automatická činnost. V praktických aplikacích totiž časová náročnost takové úlohy závisí na velikosti množiny rozpoznávaných řečníků (dochází k velkému množství porovnávání jednotlivých dat mezi sebou). Požadavkem je práce systému v reálném čase.

V průběhu rozpoznávání můžeme narazit na řadu problémů, které můžou znesnadnit řešení této úlohy. Nejobvyklejší jsou problémy a poruchy vznikající během nahrávání nebo vlivem komunikačního kanálu. Díky tomu se mohou lišit promluvy jednoho řečníka nahrané v různých časových okamžicích. Minimalizovat tyto poruchy lze mimo jiné i pomocí normalizace skóre, již se zabývá tato práce.

1.1. Cíle diplomové práce

Cílem této diplomové práce je implementovat různé metody normalizace skóre v úloze verifikace řečníka podle hlasu. Je nutné otestovat tyto metody na různých verifikačních úlohách, vyhodnotit jejich přínos, a porovnat jejich účinnost mezi sebou.

1.2. Obsah jednotlivých kapitol

Možné přístupy k úloze rozpoznávání řečníka jsou popsány v kapitole 2. Kapitola 3 rozebírá systém pro rozpoznávání řečníka a přichází s dekompozicí tohoto systému na tři podsystemy: extrakce příznaků, tvorba modelu a vlastní rozpoznávání. Kapitola 4 obsahuje rozbor jednotlivých metod normalizace pro zvýšení účinnosti v úloze verifikace. Kapitola 5 se věnuje popisu programového řešení zadaného problému, rozebírá jednotlivé programové moduly. Kapitola 6 se zabývá testovanými experimenty, popisuje použité korpusy dat a dává zhodnocení získaných výsledků.

2. Úlohy v problému rozpoznávání řečníka

2.1. Biometrické metody verifikace

Při **verifikaci** identity osoby jde o potvrzení nebo vyvrácení prohlašované identity. **Biometrie** je oblast měření zkoumané fyziologické charakteristiky osob (např. otisky prstů, hlas, sítnice oka, podpis, ...). **Biometrická verifikace** (autorizace) je tedy úloha potvrzení nebo vyvrácení proklamované identity osoby na základě jejich fyziologických charakteristik.

Postup biometrické verifikace je přibližně stejný u všech metod, nejprve je nutné uživatele zaregistrovat, tedy načíst poprvé jeho data (často několikrát, data posléze zprůměrována). Z dat jsou vybrány jen důležité informace obsahující identitu uživatele.

Při verifikaci uživatel zadá vzorek své biometrické charakteristiky a systém jej porovná s uloženou informací a vydá patřičné rozhodnutí (ano/ne).

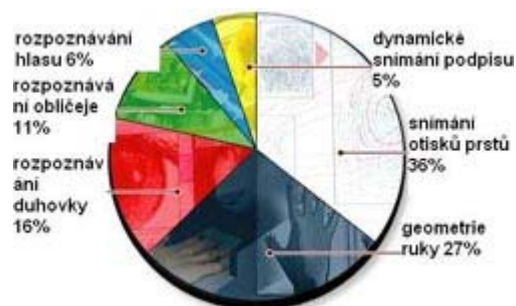
Výhody biometrik jsou nemalé, hlavním důvodem jejich používání je silná obrana proti zneužití. Biometrické charakteristiky jsou pro každou osobu jedinečné, obvykle nepřenositelné (spolu s ověřením aktuálnosti vzorku, např. přečtení aktuální hlášky systému). Je možné také tyto metody mezi sebou vzájemně kombinovat pro zvýšení spolehlivosti. Uživatel se nemusí obávat ztráty nebo odcizení.

Nevýhodou je technická i finanční náročnost a nutnost zabezpečení chybovosti (přijmutí nesprávného, zamítnutí správného uživatele).

Následující tabulky popisují známé biometrické verifikační metody a jejich využití. Jsou převzaté z článku Ing. Rita Pužmanová, CSc : „*Biometrické systémy v praxi*“ [3]

typ biometrie	výhody	Nevýhody	přesnost	Objem dat
otisk prstu	stálá a přesná, zavedená technologie	logické spojení odebírání otisků s kriminalitou	1:1000	240 – 1000 B
geometrie ruky	Střední přesnost (roste s 3D snímáním)	fyzický kontakt se snímačem	1:1000	9-20 B
sítnice	vysoká přesnost, mění se pouze nemocí či úrazem	přiblížit se k snímači a nechat si svítit do oka	1:1.000.000+	35 B
duhovka	vysoká přesnost; žádný fyzický kontakt, brýle nevadí	vysoká cena systému – vhodné pro vysoce bezpečné zóny	1:6.000.000	256-500 B
obličejové znaky	přijatelné pro uživatele – žádný fyzický kontakt	ne příliš přesná metoda		1 KB (100 B po kompresi)
DNA	vysoká přesnost	drahé; lze systém obelhat cizím vzorkem		
tvar ucha	netřeba fyzický kontakt	zatím se zkoumá		
rozprostření žil na zápěstí	fyzický kontakt	přijatelnost jako u snímání otisků prstů		
charakteristiky hlasu	přijatelné pro uživatele; vhodné i pro vzdálenou autentizaci po telefonu	méně přesné – vliv prostředí a stavu uživatele	1:50	
podpisové charakteristiky	naprosto přijatelné pro uživatele	méně přesné, může mít vliv únava nebo stres	1:50	
charakteristiky psaní na klávesnici	přijatelné pro uživatele, cenově dostupné	méně přesné - může mít vliv únava nebo stres		

Tabulka 1: Porovnání biometrických systémů



Obrázek 1. Podíl jednotlivých technologií biometrických systémů na trhu

	otisk prstu	obličej	dlaň	duhovka
podíl chybných odmítnutí	0,2 – 36 %	3,3 – 70 %	0 – 5 %	1,9 – 6 %
podíl chybných přijetí	0 – 8 %	0,3 – 5 %	0 – 2,1 %	pod 1%
doba transakce	9 – 19 s	10 s	6 – 10 s	12 s
velikost šablony	250 – 1000 B	84 – 1300 B	9 B	512 B
počet hlavních výrobců	25+	2	1	1
náklady na zařízení	Nízké	střední	střední	Vysoké
faktory ovlivňující výkon	špinavé, suché horké prsty	různá osvětlení, orientace obličeje, změny ve vzhledu	zranění ruky artritida, pocení	špatné vidění odrazy

Tabulka 2: Porovnání hlavních biometrik (podle General Accounting Office USA)

2.2. Rozdělení úlohy rozpoznávání

Existují dvě základní úlohy rozpoznávání řečníka – a to **identifikace** a **verifikace** [4,2]. V úloze identifikace řečníka hledáme k dané **promluvě** nejpravděpodobnější **model** řečníka z množiny referenčních řečníků. V úloze verifikace pak ověřujeme, zda daná promluva náleží apriorně danému řečníkovi.

Promluvou rozumíme nahrávku hlasu řečníka přicházející na vstup systému. Model je skupina parametrů charakterizující daného řečníka a je již uložen v systému.

Dále lze úlohu identifikace rozdělit podle velikosti množiny řečníků:

Identifikace v uzavřené množině - předpokládáme, že vstupní promluva patří někomu ze skupiny referenčních řečníků.

Identifikace v otevřené množině - vstupní promluva neznámého řečníka nikomu ze skupiny referenčních řečníků patřit nemusí. Tento problém lze chápat jako spojení identifikace v uzavřené množině následované verifikací.

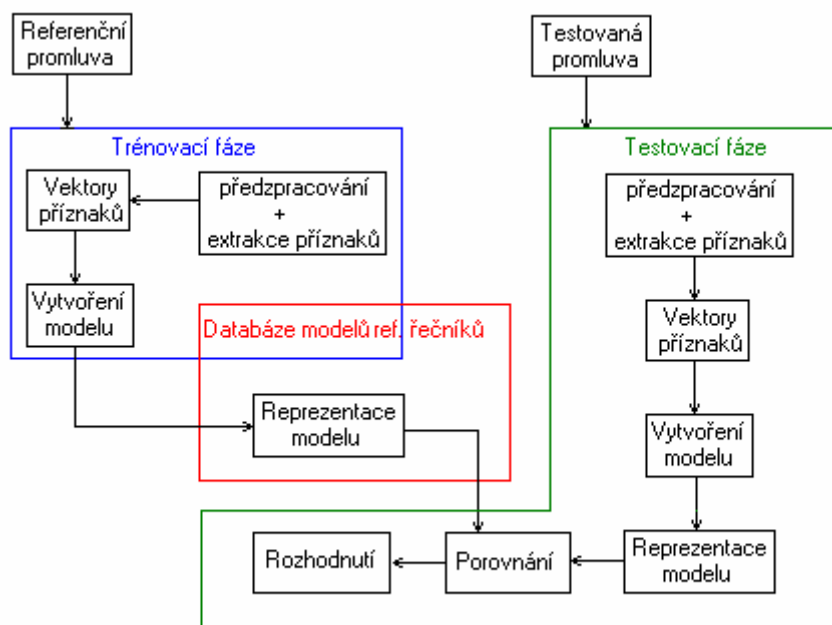
Další rozdělení úlohy rozpoznávání je v závislosti na obsahu vyslovované promluvy:

Textově nezávislé rozpoznávání – testovaná promluva neodpovídá trénovací promluvě. Řečník v tomto případě není nucen opakovat stejnou promluvu jako při trénovací fázi (je míněna shoda v obsahu promluvy). Testovací promluva může obsahovat neomezený text, popřípadě jen určité události, které se testují (např. hlásky slova, číslice).

Textově závislé rozpoznávání – zde je vyžadováno vyslovení stejné promluvy při testování, jako při trénovací fázi.

2.3. Postup rozpoznávání [1]

Rozpoznávání je rozděleno na dvě části (viz obr.2). **Třénovací fáze**, kdy jsou zpracovány referenční promluvy od každého řečníka a uloženy jejich referenční modely. Následuje **testovací fáze**, kdy je zpracována testovaná promluva. Samotné rozpoznání může být provedeno na základě shody testované promluvy s referenčním modelem, nebo srovnáním dvou modelů.



Obrázek 2. Schéma testovací a trénovací fáze

2.4. Důvody nutnosti normalizace [4]

Zpracovávaný řečový signál v sobě nese velké množství informace, mimo jiné

- individualitu řečníka (to nás zajímá, považujeme za konstantní),
- fonetický obsah vyslovené promluvy,
- okamžité pocity řečníka,
- jeho zdravotní stav,
- šum z okolí.

Při řešení problému, zda daná promluva patří některému z referenčních řečníků, nebo zda pochází od neznámého řečníka, se velmi projevují nežádoucí změny a poruchy v nahrané promluvě řečníka. Tyto změny mohou mít různé příčiny:

- Poruchy pocházející z okolního prostředí, které se projevují jako aditivní šum na pozadí nahrávky.
- Poruchy a změny komunikačního kanálu, které se projevují během nahrávání.
- Změny stavu hlasu řečníka.

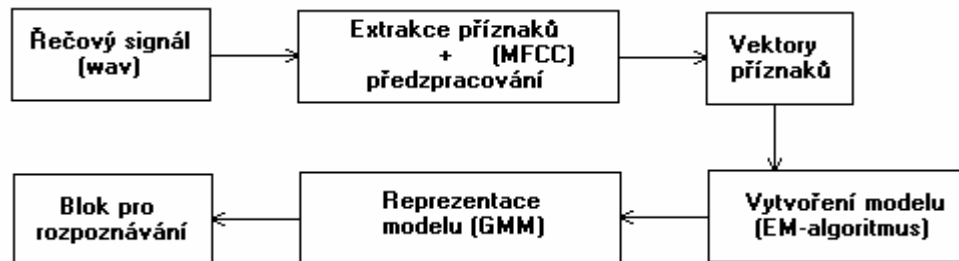
To vše způsobuje neshodu mezi testovanou promluvou a dříve nahranou trénovací promluvou. V praxi je nemožné shromáždit přesné informace o existenci a velikosti rušivých jevů.

Zatímco se změnami stavu hlasového ústrojí se rozpoznávací systémy vyrovnávají velmi těžko, chyby způsobené aditivním šumem nebo změnou komunikačního kanálu, lze minimalizovat. Jednou z možností odstranění takového rušení je i předzpracování nahrané promluvy.

Při určování podobnosti testované promluvy s referenčním řečníkem z dané množiny řečníků se působení těchto poruch projeví stejnou měrou u všech referenčních řečníků. Tedy jejich relativní vzdálenost (ve stavovém prostoru řečníků) od promluvy zůstává stejná. V těchto případech nejefektivnější cestou jak vyřešit tento problém je **normalizace**.

3. Systém pro rozpoznání řečníka

Systémy rozpoznávání řečníka se skládají ze tří základních subsystémů [1,2], které na sebe vzájemně navazují. Jde o blok parametrizace a předzpracování signálu (promluvy), následuje blok pro tvorbu modelu a poslední subsystém tvoří vlastní rozpoznávání (viz obr.3).



Obrázek 3. Svázanost dílčích modulů systému rozpoznávání

3.1. Extrakce příznaků

Jde o problém, jakým způsobem číselně popsat příchozí promluvu (v čase spojitý řečový signál). Nejprve je potřeba upravit signál (fáze předzpracování), tedy potlačit šum, vliv přenosového kanálu atd. Následně se nahrávka popíše množinou vektorů příznaků, které nesou informaci o identitě řečníka. Tato fáze je nazývána parametrizací.

Vstupní akustický signál můžeme rozdělit na mikrosegmenty s délkou cca 10-20ms, protože v tomto krátkém okamžiku považujeme parametry hlasového ústrojí za stacionární. Tyto krátké časové okamžiky popíšeme číselným vektorem příznaků pomocí tzv. **krátkodobých charakteristik**, jako jsou např. **Melovské keprální koeficienty (Mel-Frequency Cepstral Coefficients)** [5,9,10] nebo **Perceptivní lineární prediktivní analýza (Perceptive Linear Predictive Analyze)** [5]. Blíže popíšeme metodu MFCC, protože je použita v našem systému.

3.1.1. Melovské keprální koeficienty (MFCC)

Metoda MFCC je založena na respektování určitých vlastností vnímání zvuku člověkem. Jde především o respektování kritických pásem slyšení a vlivu subjektivního vnímání výšky tónů. MFCC využívá banky filtrů nelineárně rozmístěných ve frekvenční ose. Kritická pásma ovlivňují tvar filtrů (především šířku jednotlivých filtrů). Subjektivní vnímání šířky tónů ovlivňuje rozmístění jednotlivých filtrů ve frekvenční ose.

▪ Rozmístění filtrů

Poloha filtrů ve frekvenční ose [Hz] je nelineární, je závislá na subjektivním vnímání výšky tónu, jde tedy o nelineární rozmístění (viz obr.4). Různé experimenty dokázaly, že člověk vnímá výšku zvuku subjektivně a tato hodnota neodpovídá její fyzikální hodnotě udané v jednotkách Hz. Proto byla zavedena tzv. subjektivní výška zvuku udaná v jednotce *mel*. Její závislost na skutečné hodnotě v Hz můžeme matematicky zapsat ve tvaru

$$f_{mel} = 2595 \log_{10} \left(1 + \frac{f_{Hz}}{700} \right), \quad (1)$$

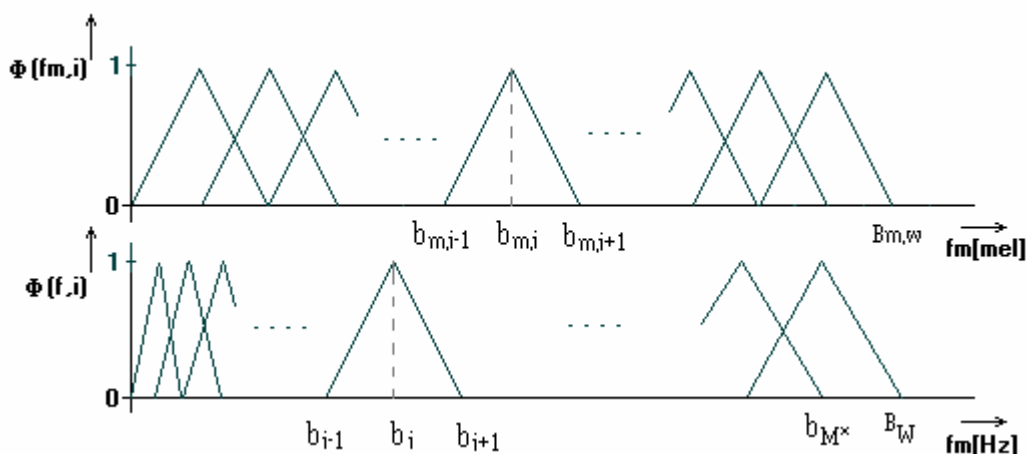
kde: f_{mel} subjektivní výška měřená v jednotce mel
 f_{Hz} frekvence měřená v jednotce Hz

▪ Tvar filtrů

Melovský filtr má trojúhelníkový tvar. Lze jej popsat vztahem

$$\Phi(f_m, i) = \begin{cases} 0 & \text{pro } f_m < b_{m,i-1} \\ \frac{f_m - b_{m,i-1}}{b_{m,i} - b_{m,i-1}} & \text{pro } b_{m,i-1} \leq f_m < b_{m,i} \\ \frac{b_{m,i+1} - f_m}{b_{m,i+1} - b_{m,i}} & \text{pro } b_{m,i} \leq f_m < b_{m,i+1} \\ 0 & \text{pro } b_{m,i+1} \leq f_m \end{cases}, \quad (2)$$

kde: $\Phi(f_m, i)$ funkční hodnota i -tého filtru ve frekvenci f_{mel}
 M^* počet filtrů
 $b_{m,i}$ je dáno vztahem $b_{m,0} = 0$, $b_{m,i} = b_{m,i-1} + \Delta_m$, pro $0 < i \leq M^*$
 $\Delta_m = B_{mw} / (M^* + 1)$
 (první filtr začíná v 0 mel a poslední končí v bodě B_{mw} mel, viz obr.3.)



Obrázek 4. Rozmístění melovských filtrů

Počet filtrů je doporučeno volit v závislosti na počtu kritických pásem. Při vzorkovací frekvenci F_s [Hz] je šířka přenášeného pásma B_w [Hz] podle Shannonova vzorkovacího teoremu rovna $F_s/2$.

▪ Postup výpočtu MFCC (viz obr.5)

Vstupní rozmluvu (akustický signál) rozdělíme na mikrosegmenty s délkou cca 30ms, protože v tomto krátkém okamžiku považujeme parametry hlasového ústrojí za stacionární.

Na signál aplikujeme Hammingovo okénko (provedeme konvoluci signálu s okénkem pro zlepšení vlastností ve spektrální oblasti).

Vypočteme krátkodobé výkonové spektrum $P(f)$ pomocí diskrétní Fourierovy transformace (DTF).

Výsledek filtrujeme bankou trojúhelníkových filtrů, jejichž jednotlivé odezvy lze vyjádřit vztahem

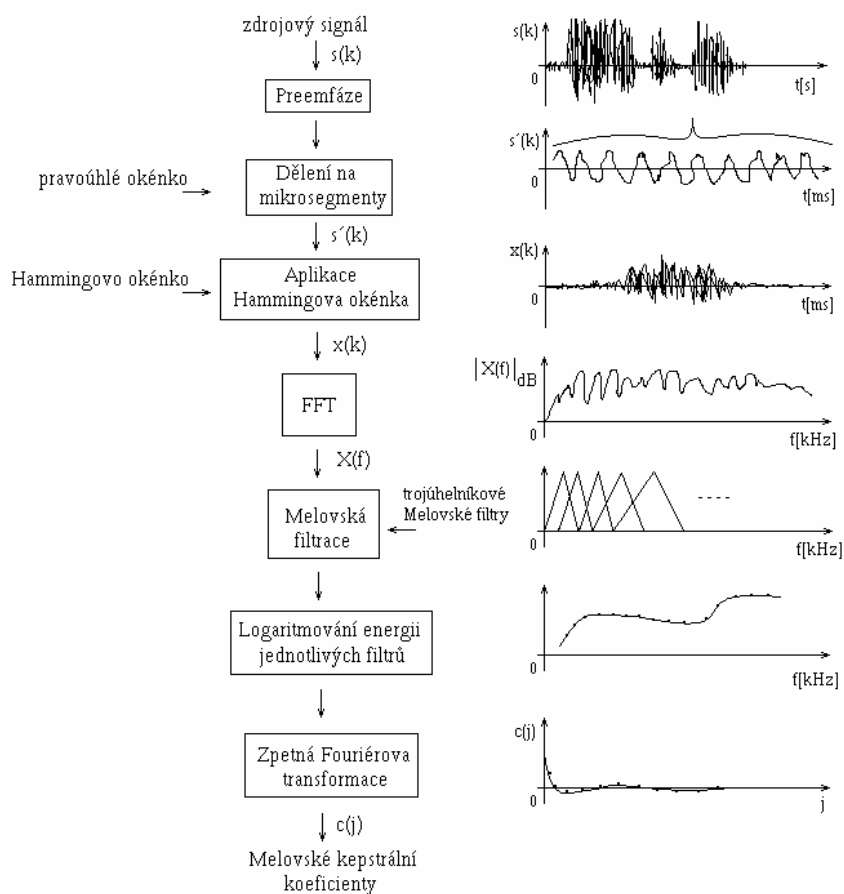
$$y_m = \sum_{f=b_{i-1}}^{b_{i+1}} P(f_{Hz}) \Phi(f_{Hz}, i), \quad i = 1, 2, \dots, M^*, \quad (3)$$

kde: M^* počet filtrů
 f_{Hz} frekvence v Hz

Aplikujeme keprální přístup, vypočteme logaritmy energií jednotlivých filtrů $y_m(i)$.
 Posledním krokem je zpětná Fourierova transformace. Vzhledem k tomu, že jsme vycházeli z výkonového spektra $P(f)$, které je pouze reálné a symetrické, zredukuje se výpočet zpětné Fourierovy transformace na zpětnou cosinovou transformaci (tedy diskrétní verzi, DCT).

$$c_m(j) = \sum_{i=1}^{M^*} y'_m(i) \cos\left[\frac{\pi}{M^*}(i-0,5)j\right], \quad \text{pro } i = 0, 1, \dots, N, \quad (4)$$

kde: $y'_m(i) = \log_{10} y_m(i)$ prvek keprální analýzy
 M^* počet melovských filtrů
 N počet melovských keprálních koeficientů $< M^*$



Obrázek 5. Postup výpočtu melovských keprálních koeficientů

3.2. Reprezentace modelu řečníka

Jestliže již máme k jednomu danému řečníkovi vytvořené n -dimenzionální vektory příznaků z trénovací fáze (obvykle je jich více, aby lépe popisovaly řečníka), můžeme z nich vytvořit jeho model. Řečnickovy vektory se vyskytují na určitém místě v n -dimenzionálním prostoru. Model by měl popisovat právě toto charakteristické rozložení vektorů příznaků.

Jednou z možností, jak vytvořit model, je použít Gaussovské rozložení pravděpodobnosti pro popis modelu. Tato metoda je pak nazývána **Směs Gaussovských rozložení (Gaussian Mixture Model)** [5,11].

3.2.1. Směs Gaussovských rozložení (GMM)

Několik posledních let je směs Gaussovských rozložení hlavním způsobem modelování řečníka pro systémy rozpoznávání podle hlasu.

GMM přístup předpokládá, že pravděpodobnostní rozložení vektorů příznaků může být modelováno váženou sumou hustot pravděpodobnosti a dáno rovnicí

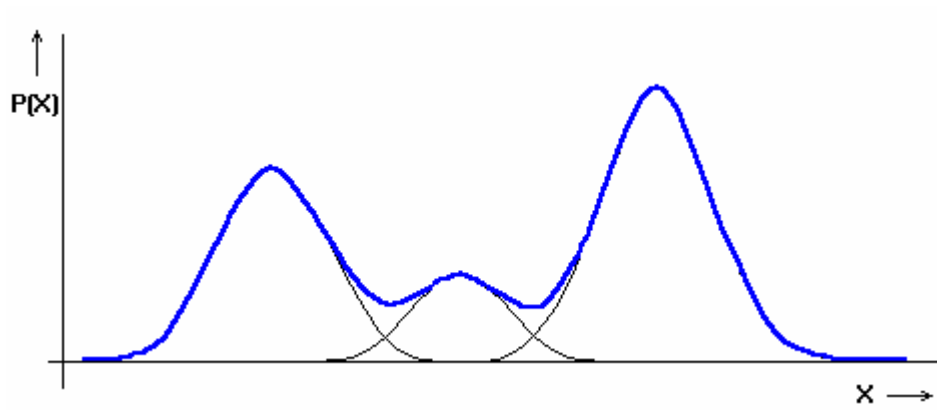
$$p(x / \lambda) = \sum_{i=1}^M \alpha_i N(x_i, \mu_i, C_i) , \quad (5)$$

kde:	x	n -dimenzionální vstupní vektor
	α_i	váha i -té složky, platí $0 \leq \alpha_i \leq 1$ a zároveň $\sum_{i=1}^M \alpha_i = 1$
	λ	model řečníka
	μ_i	střední hodnota, $\mu_i \in R^n$
	C_i	kovarianční matice řádu $n \times n$, pozitivně definitní (obvykle pouze diagonální pro zrychlení výpočtů)

$N(x_i, \mu_i, C_i)$ hustota Gaussovského rozložení pravděpodobnosti, která je jednoznačně dána střední hodnotou μ_i a kovarianční maticí C_i , tedy platí:

$$N(x_i, \mu_i, C_i) = \frac{1}{\sqrt{(2\pi)^p |C_i|}} \exp\left\{-\frac{1}{2}(x - \mu_i)^T C_i^{-1}(x - \mu_i)\right\}. \quad (6)$$

Výsledná hustota pravděpodobnosti $p(x / \lambda)$ je tedy dána váženou lineární kombinací M Gaussovských rozložení $N(x_i, \mu_i, C_i)$, jak ukazuje obr.6. Všechny parametry rozdělení jsou tedy reprezentovány modelem $\lambda = \{\alpha_i, \mu_i, C_i\}$, pro $i = 1, \dots, M$.



Obrázek 6. Směs Gaussovských rozložení pro $M=3$

Odhady parametrů λ jsou prováděny pomocí iterativního **EM-algoritmu** (**E**xpectation-**M**aximalization) [23].

3.3. Úloha rozpoznávání [1,2]

Shoda testovacích dat (ať jsou reprezentovány přímo nebo pomocí modelu) s referenčními je dána veličinou označovanou jako **skóre** (jde o míru podobnosti, je také nazývána anglicky **likelihood**).

Označíme skóre referenčního modelu λ_s pro promluvu O jako $p(O / \lambda_s)$. Způsob výpočtu skóre je závislý na typu modelu, popř. na typu obou porovnávaných modelů.

Hodnota skóre je pak zpracována dle typu řešené úlohy. Nejpoužívanější skóre pro systémy založené na stochastických modelech je podmíněná pravděpodobnost, že promluva O pochází od řečníka λ_s .

3.3.1. Verifikace

V systémech verifikace je rozhodnutí přijmout nebo odmítnout proklamovanou identitu s založeno na srovnání skóre se stanoveným prahem T .

$$p(O / \lambda_s) \begin{cases} \geq T \rightarrow O \in s \\ < T \rightarrow O \notin s \end{cases}, \quad (7)$$

kde:	O	testovaná promluva (vektor příznaků)
	s	hledaný řečník
	λ_s	referenční model proklamovaného řečníka
	$p(O / \lambda_s)$	skóre referenčního modelu λ_s pro testovou promluvu O
	T	předem určený práh rozhodování

Posuzujeme zde vlastně to, zda je velikost skóre testované promluvy O s referenčním modelem proklamovaného řečníka λ_s dostatečně vysoká.

3.3.2. Identifikace v uzavřené množině

Předpokládejme, že máme N řečníků, kteří poskytli trénovací promluvu a jejich statické modelové popisy jsou $\lambda_1, \lambda_2, \dots, \lambda_N$. Pokud O označíme vektor příznaků extrahovaný z testované promluvy, pak identifikace v uzavřené množině může proběhnout uvedeným způsobem:

$$s^* = \arg \max_{1 \leq n \leq N} \{p(O / \lambda_n)\}, \quad (8)$$

kde:	O	testovaná promluva (vektor příznaků)
	s^*	hledaný řečník
	λ_n	referenční model řečníka
	$p(O / \lambda_n)$	skóre referenčního modelu λ_n pro testovou promluvu O

Řečník s^* je tedy autorem testované promluvy. Jde vlastně o úlohu přiřadit testovanou promluvu jednomu z nám známých řečníků, a to tomu, který vykazuje minimální rozdíl mezi referenčním modelem a testovanou promluvou (maximální skóre).

3.3.3. Identifikace v otevřené množině

Tato úloha kombinuje úlohu identifikace i verifikace. Nejprve je vybrán řečník s^* , jehož model vykazuje nejvyšší skóre pro testovanou promluvu O . Poté je ověřeno, zda je autorem skutečně řečník s^* nebo se jedná o promluvu od řečníka mimo referenční skupinu.

Předpokládejme opět N řečníků v referenční skupině a jejich statické modelové popisy jsou $\lambda_1, \lambda_2, \dots, \lambda_N$. O označí vektor příznaků extrahovaný z testované promluvy. Pak identifikace v otevřené množině může proběhnout uvedeným způsobem:

$$\max_{1 \leq n \leq N} \{p(O / \lambda_n)\} \begin{cases} \geq T \rightarrow O \in \max_{1 \leq n \leq N} \{p(O / \lambda_n)\} \\ > T \rightarrow O \in U \end{cases}, \quad (9)$$

kde:	O	testovaná promluva (vektor příznaků)
	λ_n	referenční model řečníka
	$p(O / \lambda_n)$	skóre referenčního modelu λ_n pro testovou promluvu O
	T	předem určený práh rozhodování
	U	představuje neznámého řečníka (není v referenční skupině řečníků)

Promluva O je tedy přiřazena modelu řečníka, který získal maximální skóre ze všech řečníků v systému, jen pokud je velikost maximálního skóre větší než práh T . Pokud je skóre menší než práh T , řekneme, že promluva pochází od neznámého řečníka (zde označeného např. U).

4. Normalizace skóre

Absolutní hodnota skóre je velmi závislá na podmínkách nahrávání, promlouvaném textu apod. Tato závislost způsobuje obtížné stanovení hodnoty prahu při verifikaci. Pro překonání tohoto problému lze použít normalizační techniky.

Normalizaci lze provádět v zásadě dvěma způsoby, **Bayesovským způsobem** [6] nebo lze **standardizovat rozložení pravděpodobnosti skóre** [6,7].

4.1. Bayesovský způsob

Na úlohu se díváme z pravděpodobnostního hlediska. Porovnáváme pravděpodobnosti vyslovení testované promluvy O řečníkem λ_s s pravděpodobností vyslovení jiným řečníkem λ_u . Řešíme tedy rovnici:

$$p(\lambda_s / O) > p(\lambda_u / O) \rightarrow O \in \lambda_s \quad (10)$$

jinak $O \in \lambda_u$.

Aplikací Bayesova teorému na výše uvedenou rovnici dostáváme vztah:

$$\frac{p(O / \lambda_s)}{p(O / \lambda_u)} > \frac{P(\lambda_u)}{P(\lambda_s)} \rightarrow O \in \lambda_s \quad (11)$$

jinak $O \in \lambda_u$,

kde: $\frac{p(O / \lambda_s)}{p(O / \lambda_u)} = l(O)$ **verifikační skóre** vypočítané v prvním stupni , (12)

$$\frac{P(\lambda_u)}{P(\lambda_s)} = T$$
 apriori určený **práh** pro ověření (druhý stupeň). (13)

Hodnotu prahu je možné určit apriorně např. na základě odhadu poměru počtu oprávněných a neoprávněných osob, verifikační skóre je nutné spočítat.

Častěji užívaný tvar k vyjádření verifikačního skóre v praxi je

$$L(O) = \log p(O / \lambda_s) - \log p(O / \lambda_u) . \quad (14)$$

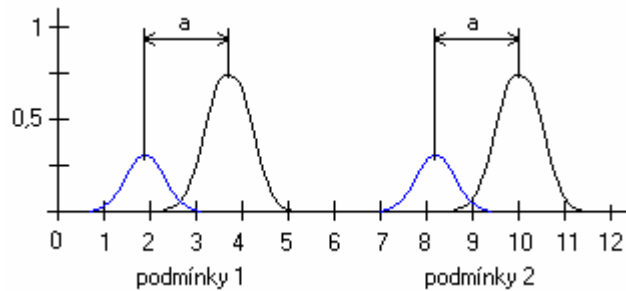
Zavedení logaritmu je motivováno faktem, že operace sčítání (resp. odečítání) je numericky mnohem snadnější než operace násobení (resp. dělení).

Zdůvodnění zavedení normalizačního faktoru $\log p(O / \lambda_u)$ je následné, pokud se změní podmínky nahrávání, posune se rozložení skóre $\log p(O / \lambda_s)$. Avšak rozložení skóre $\log p(O / \lambda_u)$ se posune stejným způsobem.

Pravděpodobnost $\log p(O / \lambda_u)$ ve výpočtu verifikačního skóre tedy reprezentuje dynamický práh, který je citlivý na změny v O v každém pokusu.

Výpočet $\log p(O / \lambda_s)$ není problém, protože je znám model referenčního řečníka λ_s . Jak ale určit model λ_u , který reprezentuje neznámého řečníka? Tento problém je řešen vhodnou aproximací modelu λ_u .

Motivací pro zavedení normalizačního faktoru $p(O/\lambda_s)$ je fakt, že změni-li se podmínky při nahrávání testované promluvy vůči podmínkám při nahrávání trénovací promluvy, změni se stejným způsobem rozložení skóre $\log p(O/\lambda_s)$ i rozložení skóre $\log p(O/\lambda_u)$. To ale znamená, že rozdíl mezi nimi zůstane zachován (viz obr.7).



Obrázek 7. Bayessonský přístup (změna nahrávacích podmínek)

4.1.1. Světový model λ_{WN} [6]

Jednou z možností jak aproximovat neznámý model λ_u je nahradit jej tzv. **světovým modelem** (někdy je také označen jako **model okolí**, anglicky **BackGround Model**). Je generovaný použitím promluv z velké populace řečníků. V rovnici tedy nahradíme:

$$p(O/\lambda_u) = p(O/\lambda_{WN}). \quad (15)$$

4.1.2. Skupinový model λ_{CN} [6]

V této metodě je každý zapsaný mluvčí přidružen se skupinou mluvčích $C = \{\lambda_s^{-1}, \dots, \lambda_s^{-K}\}$, jejichž modely jsou nejvíce konkurenční (jsou blízko referenčního řečníka λ_s v prostoru mluvčích). Výběr probíhá před vlastním testováním, skupina je vybrána **off-line**. Výpočet pak probíhá tímto způsobem:

$$\log p(O/\lambda_u) = \log p(O/\lambda_{CN}) = \frac{1}{K} \sum_{k=1}^K \log p(O/\lambda_s^{-k}) \quad (16)$$

nebo alternativa

$$\log p(O/\lambda_u) = \log p(O/\lambda_{CN}) = \log \left\{ \frac{1}{K} \sum_{k=1}^K p(O/\lambda_s^{-k}) \right\}. \quad (17)$$

Předpokládáme, že za referenčního řečníka se bude vydávat někdo s podobným hlasem. Řečník s velmi odlišným hlasem má daleko jak k modelu λ_s tak i k modelu λ_{CN} (pak je malý rozdíl $p(O/\lambda_s) - p(O/\lambda_u)$), to může vést k přijetí tohoto podvodníka. To se řeší vytvořením přidružené skupiny mluvčích, kde jen část řečníků je blízká k referenčnímu řečníkovi. Zbytek skupiny je od něj vzdálen daleko více (myšleno v prostoru mluvčích). Pro tento případ určujeme výsledné skóre ne jako průměr, ale jako maximum:

$$\log p(O/\lambda_u) = \log p(O/\lambda_{CN}) = \max_{k=1, \dots, K} \log p(O/\lambda_s^{-k}). \quad (18)$$

4.1.3. Neomezený skupinový model λ_{UCN} – [6]

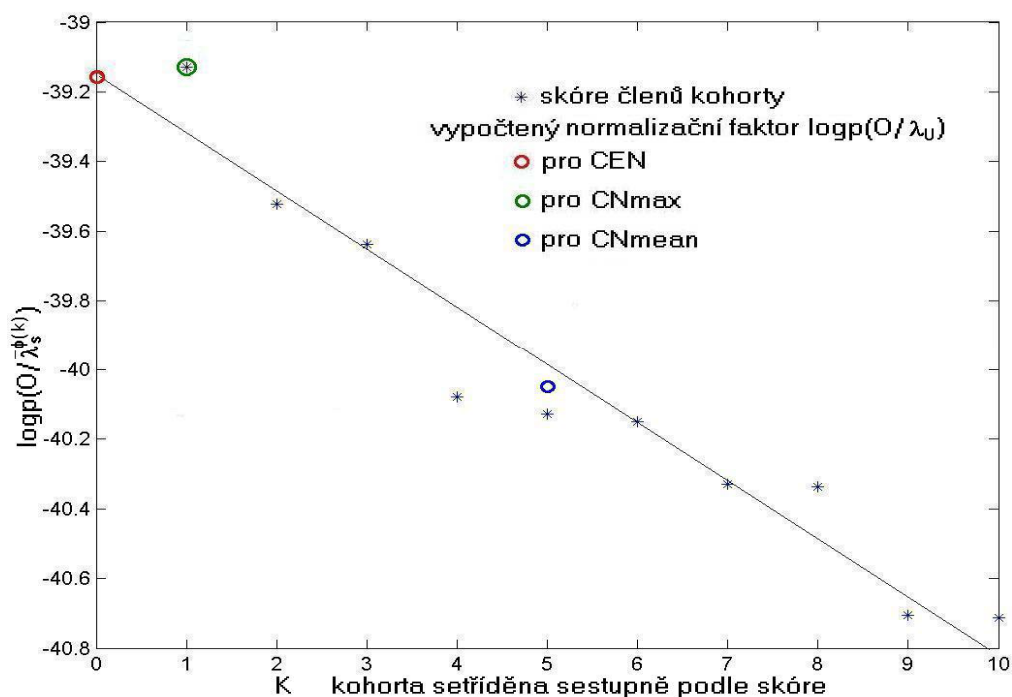
Vznikne spojením skupinového a světového modelu. Přidružená skupina modelů zde není vybrána apriorně, ale během testování (*on-line*). Skupina je tvořena K modely s nejvyšším skóre $\overline{\lambda}_s^{\phi(k)}$ pro testovanou promluvu (kromě λ_s). Výsledné skóre je pak dáno:

$$\log p(O / \lambda_U) = \log p(O / \lambda_{UCN}) = \frac{1}{K} \sum_{k=1}^K \log p(O / \overline{\lambda}_s^{\phi(k)}), \quad (19)$$

kde $\phi(i) \neq \phi(j)$ pro $i \neq j$ a $\overline{\lambda}_s^{\phi(1)}, \overline{\lambda}_s^{\phi(2)} \dots \overline{\lambda}_s^{\phi(K)}$ jsou modely, které přinášejí (hned po modelu λ_s) nejvyšší skóre k dané promluvě O .

4.1.4. Skupinový model s extrapolací λ_{CEN} – [12]

Zde není normalizační faktor (logaritmus pravděpodobnosti $\log p(O / \lambda_U)$) určen jako průměr logaritmů pravděpodobnosti modelů řečníků kohorty (jako je tomu u metody CN) ani brán jako maximum dané kohorty (viz metoda UCN). Místo toho je použita extrapolace této kohorty (viz obr.8). Extrapolace zohledňuje předpoklad, že model referenčního řečníka, pokud je řečník skutečně původcem nahrávky, dosahuje nejlepšího skóre. Narozdíl od maxima však extrapolace zohledňuje i trend kohorty. Proto extrapolace používající větší počet řečníků v kohortě je robustnější. Zároveň lze snadno měnit poměr mezi odmítnutými a přijmutými řečníky posouváním bodu extrapolace.



Obrázek 8. Poloha normalizačního faktoru vzhledem k ostatním členům kohorty pro metody CEN, CN (střední hodnota a maximum)

4.1.5. Neomezený skupinový model s extrapolací λ_{UCEN} – [12]

Počítá se naprosto stejně jako výše uvedená normalizace (tedy normalizace CEN). Změna je pouze v kohortě, ta je tvořena řečníky, kteří podávají největší hodnoty skóre při testování (tedy off-line). Jde tedy o kombinaci metod CEN a UCN.

4.1.6. Robustní neomezený skupinový model s extrapolací $\lambda_{ROBA-UCEN}$ – [12]

Normalizační faktor (logaritmus pravděpodobnosti $\log p(O / \lambda_{UCEN})$) získaný metodou UCEN je zde pro zvýšení robustnosti upraven. Tato úprava spočívá ve snížení pravděpodobnosti $\log p(O / \lambda_{UCEN})$ v závislosti na rozdílu skóre dS mezi apriorně vybranou kohortou a kohortou vybranou při testování.

$$\log p(O / \lambda_U) = \log p(O / \lambda_{UCEN}) - \sqrt{dS} \quad (20)$$

$$dS = \frac{1}{K} \sum_{k=1}^K \left[\log p(O / \lambda_s^{\phi_{UCN}^{(k)}}) - \log p(O / \lambda_s^{\phi_{CN}^{(k)}}) \right] \quad (21)$$

Metodu lze samozřejmě použít i na kohorty bez extrapolace (tedy ROBA-UCN).

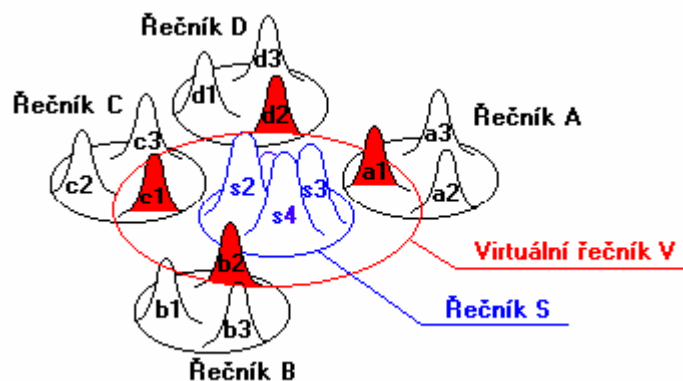
4.1.7. Virtuální řečník λ_{VS} – [13,14,1]

Skupinová normalizace CN využívá normalizační skóre $\log p(O / \lambda_U)$, které je vytvořeno ze skóre pro nejbližší podvodníky $\log p(O / \overline{\lambda_S})$. Tato skóre jsou ale vybírána na základě trénovací promluvy O . Tento výběr však nerespektuje různé podmínky při nahrávání.

Pokud je model řečníka vytvořen ze složek (jako je tomu u GMM nebo HMM), lze tento problém řešit pomocí virtuálního řečníka. V tom případě je model λ_U uměle vytvořen po složkách z nejbližších složek modelů vybraných řečníků. Ke každé složce x_m modelu

$\lambda_S = \{x_1, \dots, x_M\}$ je přiřazena nejbližší složka $\overline{x_m}$ od modelů z vybrané skupiny. Vznikne tak model neexistujícího (virtuálního) řečníka, který má nejbližší k referenčnímu řečníku ze všech vybraných modelů ze skupiny.

Jak ilustruje obr.9, je ke každé složce modelu referenčního řečníka S přiřazena nejbližší složka od řečníků A,B,C nebo D, z těch je pak sestaven model virtuálního řečníka V , přidružený k řečníkovi S .



Obrázek 9. Vytvoření virtuálního řečníka V k referenčnímu řečníku S

U verifikace využívající GMM modely je virtuální řečník vybrán ze všech možných složek potenciálních podvodníků. Pro referenčního řečníka popsaného M-složkovým jednodimenzionálním GMM modelem fE_S platí:

$$fE_S = \{w_m(S), N_m(S)\}_{m=1,2,\dots,M}, \quad (22)$$

kde: $w_m(S)$ statistická váha pro m-tou složku GMM
 $N_m(S)$ Gaussovo pravděpodobnostní rozložení pro m-tou složku GMM

Model GMM virtuálního řečníka fE_{V_S} přidružený k referenčnímu řečníkovi S, složený z K nejbližších podvodníků I, je pak reprezentován následně:

$$fE_{V_S} = \{w_m(V_S), N_m(V_S)\}_{m=1,2,\dots,M}, \quad (23)$$

kde: $w_m(V_S) = w_m(S)$ pro $m = 1, 2, \dots, M$
 $N_m(V_S) = N_n(I)$ pro $m = 1, 2, \dots, M$ a $n = 1, 2, \dots, K * M$

Statistická váha u modelu virtuálního řečníka je stejná jako u modelu referenčního řečníka. $N_n(I)$ je nejbližší složka GMM (z modelů K vybraných podvodníků) ke složce $N_m(S)$. Vzdálenost (podobnost) mezi jednotlivými složkami lze určit pomocí vzorce **Gaussova rozložení** nebo **Battacharyanovi míry** (více viz kapitola 3.3.1. Kritéria pro srovnání řečníků).

4.2. Standardizace rozložení pravděpodobnosti skóre

Tyto metody mají za cíl transformovat každý tvar skóre $p(O / \lambda_U)$, vyplývající z různých podmínek při nahrávání, na normalizovaný tvar. Důvod, proč pracujeme s rozdělením skóre pro neznámého řečníka $p(O / \lambda_U)$ než s rozdělením skóre pravého řečníka $p(O / \lambda_S)$, je získat více spolehlivých odhadů pro transformační parametry.

Metody jsou uvedené níže (předpokládáme Gaussovo pravděpodobnostní rozdělení, tj.

$$p(O / \lambda_U) \approx N(\mu, \sigma^2).$$

4.2.1. Zero normalizace (Z-norm) [6,7]

Předpokládáme, že známe podmínky při nahrávání $\Phi(O)$ promluvy O , pak lze definovat standardizaci vztahem:

$$L(O) = \frac{\log p(O / \lambda_s) - \mu_z(\lambda_s, \Phi(O))}{\sigma_z(\lambda_s, \Phi(O))}, \quad (24)$$

kde $\mu_z(\lambda_s, \Phi(O))$ je střední hodnota a $\sigma_z(\lambda_s, \Phi(O))$ směrodatná odchylka, obě specifické pro prohlášenou identitu λ_s a nahrávací podmínky $\Phi(O)$. V trénovací fázi je pár μ_z a σ_z vypočítaný pro každého registrovaného řečníka a všechny možné provozní podmínky Φ .

Tato normalizační technika je s úspěchem používána na problém způsobený záměnou mikrofону popř. telefonního sluchátka (podle toho je metoda známa jako **handset normalization H-norm**). Tato metoda však nemůže být efektivně použita v neznámých provozních podmínkách.

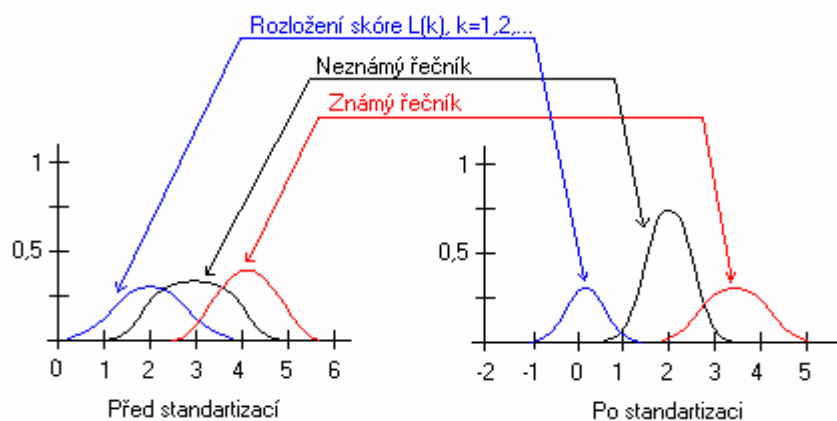
4.2.2. Testová normalizace (T-norm)

Tato metoda nevyžaduje žádnou znalost o podmínkách nahrávání. Vychází ze znalosti množiny možných podvodníků. Standardizaci definujeme vztahem :

$$L(O) = \frac{\log p(O / \lambda_s) - \mu_T(O)}{\sigma_T(O)}, \quad (25)$$

kde $\mu_T(O)$ je střední hodnota a $\sigma_T(O)$ směrodatná odchylka výstupů modelů z množiny podvodníků $C = \{\log p(O / \lambda_s^{-1}), \dots, \log p(O / \lambda_s^{-K})\}$. Tato metoda je velmi podobná metodě **neomezeného skupinového modelu**, používáme jen střední hodnotu a směrodatnou odchylku.

Motivací pro použití těchto dvou transformací je použití pevného prahu. Modrá čára (viz obr.10) je rozložení vzorového skóre $L(k)$, $k=1,2,\dots$, které odpovídá daným pracovním podmínkám $\Phi(O)$ (pro Z-norm) nebo je určeno z množiny podvodníků λ_s^{-k} , $k=1,\dots,K$ (pro T-norm). Po standardizaci pomocí transformace skóre pak může být porovnáno s pevným prahem.



Obrázek 10. Zobrazení rozdělení skóre před a po Z/T transformaci

4.3. Výběr skupiny

Při normalizaci skóre Bayesovským způsobem, ve skupinovém a neomezeném skupinovém modelu, bylo využíváno skupiny řečníků (tzv. kohorty). Jak takovou skupinu vybrat, tím se zabývá tato kapitola.

Skupinou modelů řečníků kohorty je K -nejbližších modelů k danému referenčnímu modelu. Tyto modely pak poskytují skóre podvodníků, blízkých ke skóre proklamovaného řečníka – referenčního modelu. Tím je poskytnuta ochrana proti přijetí podvodníků. Referenčním modelem (řečníkem) rozumíme model odpovídající řečníkovi, pro kterého skupinu vytváříme.

4.3.1. Kritéria pro srovnání řečníků

Používaná kritéria pro určení vzdálenosti mezi dvěma řečníky jsou následující:

▪ **Pair-wise kritérium (Rosenberg)** [15]

Nejbližší modely jsou ty, které maximalizují rovnici

$$d^P(\lambda_s, \lambda_j) = \frac{1}{2} (\log p(O / \lambda_s) - \log p(O / \lambda_j)) , \quad (26)$$

kde: O testovaná promluva (vektor příznaků)
 λ_j model testovaného řečníka
 λ_s referenční model proklamovaného řečníka

▪ **Divergence-like kritérium (Reynolds)** [16]

Nejbližší modely jsou ty, které minimalizují rovnici

$$d^P(\lambda_s, \lambda_j) = \log \frac{p(O_s / \lambda_s)}{p(O_s / \lambda_j)} - \log \frac{p(O_j / \lambda_j)}{p(O_j / \lambda_s)} , \quad (27)$$

kde: O_j sekvence vektorů příznaků testovacích dat přidružené k modelu λ_j
 λ_j model testovaného řečníka
 O_s sekvence vektorů příznaků testovacích dat přidružené k modelu λ_s
 λ_s referenční model proklamovaného řečníka

▪ **Gaussovo pravděpodobnostní rozložení**

Jde vlastně o vypočítání pravděpodobnosti, proto nejbližší modely jsou ty, které maximalizují rovnici:

$$p^G(N_m(S), N_n(I)) = \frac{1}{\sqrt{2\pi \det C}} e^{-\frac{1}{2}(\mu_{m,S} - \mu_{n,I})C^{-1}(\mu_{m,S} - \mu_{n,I})^T} , \quad (28)$$

kde: $N_m(S)$ m -tá složka modelu referenčního řečníka S
 $\mu_{m,S}$ vektor středních hodnot m -té složky modelu ref. řečníka I
 $N_n(I)$ m -tá složka modelu podvodníka I
 $\mu_{n,I}$ vektor středních hodnot n -té složky modelu podvodníka I
 C kovarianční matice

▪ **Battacharyanova míra** [17]

Nejbližší modely jsou ty, které minimalizují rovnici:

$$d^B(N_m(S), N_n(I)) = \frac{1}{8}(\mu_{m,S} - \mu_{n,I}) \left(\frac{C_{m,S} + C_{n,I}}{2} \right)^{-1} (\mu_{m,S} - \mu_{n,I})^T + \frac{1}{2} \log \frac{\frac{1}{2}|C_{m,S} + C_{n,I}|}{\sqrt{|C_{m,S} \cdot C_{n,I}|}}, \quad (29)$$

kde: $N_m(S)$ m-tá složka modelu referenčního řečníka S
 $C_{n,I}$ kovarianční matice n-té složky modelu ref. řečníka S
 $N_n(I)$ m-tá složka modelu podvodníka I
 $\mu_{n,I}$ vektor středních hodnot n-té složky modelu podvodníka I
 $C_{n,I}$ kovarianční matice n-té složky modelu podvodníka I

4.3.2. Metody výběru skupiny

Pro vytvoření skupiny řečníků C z referenční databáze se nejvíce používají tyto metody:

▪ **Metoda náhodných podvodníků RI (Random Impostor)** [2]

Do skupiny C je přiřazeno K náhodně vybraných řečníků z referenční množiny, pak $\log p(O / \lambda_U)$ je vypočteno jako

$$\log p(O / \lambda_U) = \log p(O / \lambda_{RI}) = \max_{k=1, \dots, K} \log p(O / \lambda^{\phi(k, RI)}), \quad (30)$$

kde: $\lambda^{\phi(k, RI)}$ k-tý náhodně vybraný model

▪ **Metoda nejbližších podvodníků CI (Closes Impostor)** [6]

Do skupiny C je přiřazeno K nejbližších řečníků k referenčnímu řečníkovi, pak $\log p(O / \lambda_U)$ je vypočteno jako

$$\log p(O / \lambda_U) = \log p(O / \lambda_{CI}) = \frac{1}{K} \sum_{k=1}^K \log p(O / \overline{\lambda}_S^{\phi(k, CI)}), \quad (31)$$

kde: $\overline{\lambda}_S^{\phi(k, CI)}$ k-tý nejbližší model k referenčnímu modelu S

Tento přístup však předpokládá, že za řečníka se bude vydávat někdo s podobným hlasem. Pokud má řečník velmi odlišný hlas než řečník s proklamovanou identitou, je modelován špatně jak modelem $\overline{\lambda}_S^{\phi(k, CI)}$, tak referenčním modelem λ_S . V tomto případě bude malý rozdíl mezi $\log p(O / \overline{\lambda}_S^{\phi(k, CI)})$ a $\log p(O / \lambda_S)$, což může vést k přijetí podvodníka.

Řešení této situace je modifikovat tuto metodu. Do skupiny C se nezahrnuje pouze K nejbližších řečníků, ale také M nejvzdálenějších (velmi odlišných) řečníků. Těchto M řečníků pak řeší výše uvedený problém. Velikost skupiny C je pak $N=K+M$, $\log p(O / \lambda_U)$ je pak určeno jako maximum z této skupiny namísto předchozího průměru

$$\log p(O / \lambda_U) = \log p(O / \lambda_{CI}) = \max_{n=1, \dots, N} \log p(O / \overline{\lambda}_S^{\phi(n, CI)}). \quad (32)$$

▪ **Metoda blízkých shluků podvodníků CIC** (*C*lose *I*mpostor *C*lustering) [1,2]

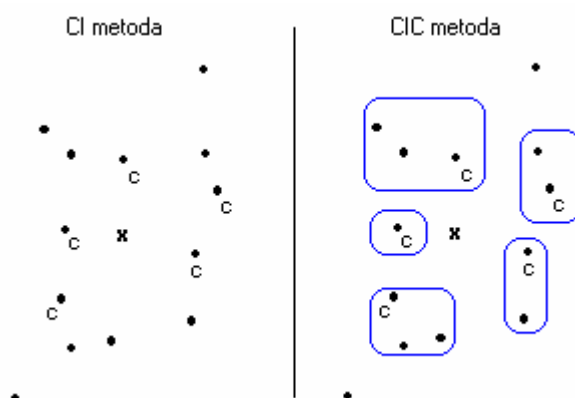
Tato metoda je vylepšením předchozí CI metody. Řeší problém, kdy nejbližší podvodníci nepokrývají celé okolí kolem referenčního modelu v prostoru řečníků, ale nechávají určitou část „nechráněnou“ (viz obr.11).

Metoda CIC spočívá ve vybrání K nejbližších řečníků (zde je K podstatně vyšší než u CI metody) do skupiny. Tato skupina řečníků je pak nashlukována do N shluků. Pro shlukování může být použita jakákoliv dostupná metoda. Z každého shluku je pak vybrán jeden řečník jako její reprezentant. Může jít o řečníka nejbližšího středu shluku, nebo nejbližšího referenčnímu řečníku.

Pro vybranou skupinu reprezentantů shluků je pak $\log p(O/\lambda_U)$ dáno vzorcem

$$\log p(O/\lambda_U) = \log p(O/\lambda_{CIC}) = \frac{1}{N} \sum_{n=1}^N \log p(O/\lambda_S^{\phi(n,CIC)}), \quad (33)$$

kde: $\lambda_S^{\phi(n,CIC)}$ model reprezentující n -tý shluk přidružený k modelu S

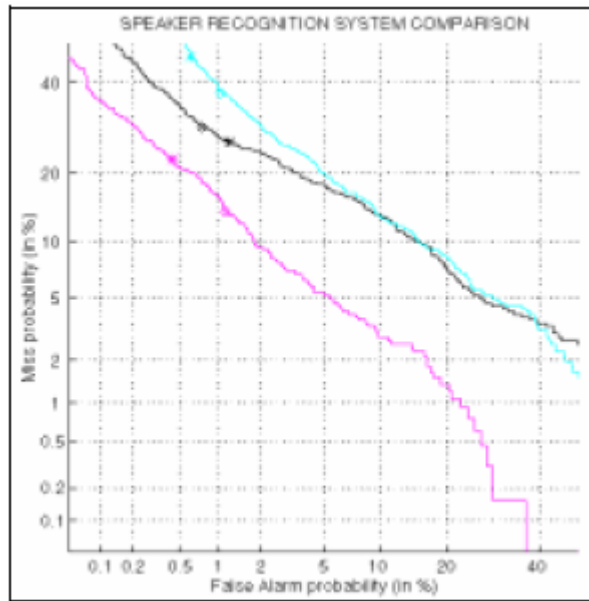


Obrázek 11. Porovnání metod CI a CIC

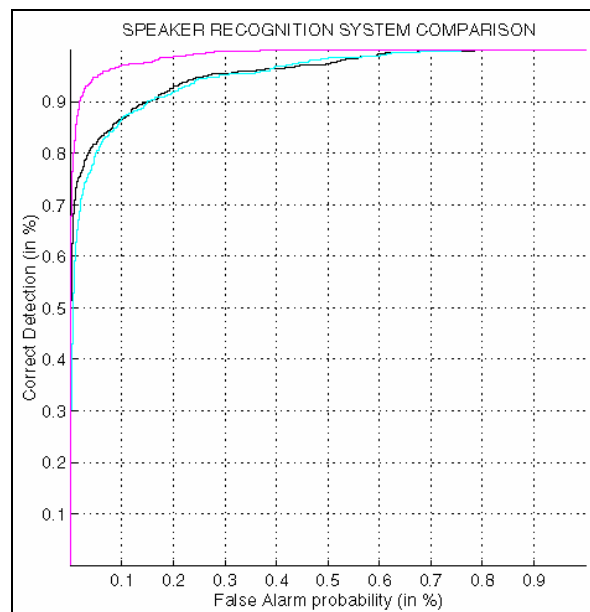
4.4. Druhy chyb ve verifikaci řečníka

Ve verifikačních systémech rozlišujeme dvě chyby [6,7]: chybné **přijetí nesprávného** řečníka (**F**alse **A**ceptance) a chybné **odmítnutí správného** řečníka (**F**alse **R**ejection). Tyto dvě chyby jsou na sobě závislé, jejich vzájemný vztah ukazuje **ROC-křivka** (**R**eceiver **O**perating **C**haracterics) [8]. Hodnoty pravděpodobnosti FA a FR je ale výhodnější vynášet jako kvantily normálního rozložení, křivka se nazývá **DET-křivkou** (**D**etection **E**rror **T**rade-off) [8]. Pro gaussovská rozložení FA a FR je DET-křivka lineární, v takovém zobrazení je pak snazší porovnání dvou podobných systémů (viz obr. 12 a 13).

Pro ohodnocení správné činnosti verifikačních systémů se používá skalární veličina **EER** (**E**qual **E**rror **R**ate). EER vyjadřuje procento správně rozpoznaných řečníků a odpovídá prahu, pro které jsou chyby FA a FR shodné. Používají se i jiná skalární ohodnocení, které mají jinak definovaný poměr FA a FR.



Obrázek 12. DET křivka pro vyhodnocení rozpoznání řečníka



Obrázek 13. ROC křivka pro stejná data jako obr. 11

5. Programové řešení

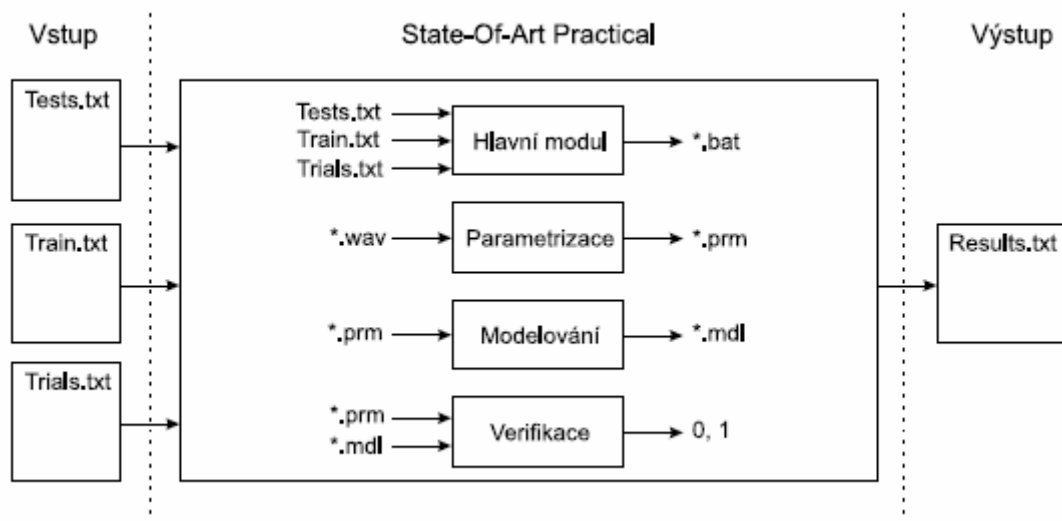
K ověření účinnosti jednotlivých normalizačních metod v úloze verifikace řečníka byla vytvořena sada programů provádějící výpočty na zadaných korpusech dat. Tyto programové moduly byly začleněny do verifikačního systému **SoAP** (**S**tate **o**f **A**rt **P**artical), který je vyvíjen katedrou kybernetiky na ZČU.

Dále se kapitola zabývá funkcí jednotlivých programů v rámci systému SoAP, udává požadavky na jejich bezproblémový chod a možnost jejich samostatného spuštění.

5.1. Stručný popis SoAP

Citace z dokumentace k systému viz [18] Ing Vaněk, J., Ing Padrta, A., „*Speaker Verification System SoAP*“, dokumentace k systému, Plzeň, 2005.

Verifikační systém SoAP je složen ze samostatně spustitelných modulů, které mají pevně definované rozhraní. Byl navržen zejména pro testování funkčnosti nových modulů. Schéma systému je znázorněno na obr.14.



Obrázek 14. Schéma verifikačního systému SoAP

- **Hlavní modul** generuje dávkové soubory, které ve správném pořadí a s odpovídajícími parametry používají zbývající moduly. Provede, v souladu se vstupními soubory, vyhodnocení a výsledek uloží do zvoleného výstupního souboru.

- **Modul parametrizace** provede extrakci z dané promluvy *.wav a uloží výsledek do souboru *.prm.

- **Modul modelování** nejprve načte soubor s parametrizací *.prm, provede modelování a výsledek uloží do souboru *.mdl.

- **Verifikační modul** zajišťuje určení míry shody mezi testovací a trénovací nahrávkou (mezi vstupní parametrizovanou promluvou a modelem prohlašované identity). Výsledkem je číslo z intervalu $\langle 0,1 \rangle$, kde 1 značí úplnou shodu a 0 úplnou rozdílnost. Funkce a předpoklady pro spuštění tohoto modulu jsou náplní zbývajících částí této kapitoly.

5.2. Popis vytvořeného verifikačního modulu Verify BGM2

Verifikační modul je součástí systému SoAP, je ale možná i jeho samostatná činnost, bez závislosti na zbytku modulů. Program *Verify_BGM2.exe* provádí nadefinované pořadí testů mezi vstupní parametrizovanou promluvou a zadaným modelem identity. Ověřuje, patřil-li příchozí promluva řečníkovi, za kterého se vydává.

Pro svou činnost používá různé metody normalizace. Jaký způsob normalizace se použije pro danou úlohu udává vstupní inicializační soubor. Výsledky verifikace ukládá do výstupního souboru.

Program je spouštěn příkazem:

Verify_BGM2.exe File.ini Trials.txt Data Model CohortFile Train.txt Results.txt, kde:

File.ini .. konfigurační soubor

Trials.txt .. soubor testů

Data .. cesta k souborům s daty, tedy k parametrizovaným nahrávkám *.prm
(př: "Param_Test\%04d.prm")

Model .. cesta k souborům s modely, tedy k souborům typu *.gmm
(př: "Models\%04d.gmm");

CohortFile .. cesta k souborům s kohortou pro každého řečníka
(př: "Cohort\OffLineCohort%03d.txt")

Train.txt ... soubor s čísly jednotlivých modelů řečníků

Result.txt .. cílový soubor, obsahuje výsledky jednotlivých testů

Jednotlivé soubory budou rozebrány a popsány v následující kapitole.

5.2.1. Vstupní/výstupní soubory verifikačního modulu

- **Konfigurační soubor** verifikačního modulu udává parametry verifikace. Při spuštění programu *Verify_BGM2.exe* se tyto informace zapíší do paměti. Struktura tohoto souboru s možným nastavením je uvedena níže:

[COMPARE-BGM2]	Informace o modelu pozadí (B ack G round M odel).
BGMDir = "BGModels"	Jméno adresáře obsahující BGM.
BGMinTXTformat = 1	Nastavení zda je BGM uložen v textovém formátu.
AverageN = 1	Počet nejlepších BGMModelů, jejichž výsledky se průměrují.
FinalDecision = 0	Pro 1 je výsledek 1 nebo 0 (jinak hodnota z intervalu <0,1>).
Thresh1 = 0.0	Hodnota prahu T, o kterou musí skóre řečníka překročit skóre normalizace, aby byl vektor přiřazen řečníkovi.
Thresh2 = 50.0	Minimální procento vektoru, které musí být přiřazeno řečníkovi, aby mu patřila celá promluva (má smysl pouze pro nastavení FinalDecision = 1).
NumberOfModels=630	Počet modelů v testu.
Normalization = 1	Typ normalizace: pro 0-WM, 1-UCN, 2-CN, 3-Tnorm, 4-UCEN, 5-CEN, 6-CN_N.
NumberOfCohort = 1	Počet členů kohorty.
NejmensiCohort = 3	Počet nejmenších členů kohorty (z konce souboru s kohortníky), je uvažováno jen pro normalizaci typu CN_N.
Zrychleni = 2	Kolik vzorku se bude v předvýběru přeskakovat.
Predkohort = 0.2	Kolik procent z všech řečníků bude předvýběr.

[MODEL-DB-GMM]	Parametry pro načítání modelu.
MixturesNumber = 0	Ukončovací podmínka shlukování (pro hodnotu 0 se bude shlukovat do dosazení kritéria).
NormalizeJ = 0	Normalizace kritéria (nejen ukončovacího) počtem vektoru ve shluku (je-li nastaveno na 1).
FinalJValue = 500.0	Ukončovací podmínka shlukování - hodnota kritéria (je-li takto shlukováno).
FinalJSumm = 0	Je-li nastaveno na 1, pak se počítá celkové kritérium (součet), jinak se bere max. kritérium.
FinalRecluster = 0	Zapnutí úplného přeshlukování na konec.
UseCF = 0	Bude využita informace o činiteli jistoty.
SaveTxt = 1	Formát uložení (TXT/BIN).

- **Soubor s testy** obsahuje jednotlivé testy daného experimentu. Na každém řádku souboru je uloženo číslo promluvy a číslo proklamované identity (modelu), např.:

```
001 001
001 002
001 003
001 004
...
```

- **Data** značí cestu k parametrizovaným nahrávkám *.prm, které byly předchozími moduly systému SoAP zpracovány ze vstupních **testových** nahrávek (*.wav). Jsou uloženy v jednom adresáři a mají číselné jméno (např. 001.prm).

- **Model** značí cestu k souborům s modely *.gmm, které byly předchozími moduly systému SoAP zpracovány ze vstupních **trénovacích** nahrávek (*.wav). Jsou uloženy v jednom adresáři a mají číselné jméno (např. 001.gmm).

- **CohortFile** značí cestu k souborům s kohortou pro každého řečníka. Soubor má na každém řádku uloženo číslo jiného řečníka (kohortníka), který má k němu blízko. V souboru jsou kohortníci seřazeni podle vzdálenosti od referenčního řečníka.

Tyto soubory jsou předvytvoreny pomocí programu Sort_LogLike.exe nebo Sort_Models.exe (budou popsány v další kapitole). Příklad části jednoho souboru OfflineCohort001.txt (pro řečníka číslo 001):

```
1
5
21
...
```

- **Train.txt** je soubor s číslem jednotlivých modelů řečníků na řádkách, např.:

```
001
002
003
004
...
```

-Result.txt je cílový soubor, obsahuje výsledky jednotlivých testů experimentu. Na každém řádku je uloženo výsledné skóre (číslo z intervalu $<0,1>$) pro jeden test ze souboru testů, např.:

0.617647
0.772021
0.826923
0.820896
0.922652
...

5.3. Popis programů pro vytvoření kohorty

Programy **Sort_LogLike.exe** a **Sort_Models.exe** slouží k vytvoření kohorty (množiny nejbližších řečníků = kohortníků k referenčnímu řečníku) jednotlivě pro všechny řečníky korpusu.

Pro daného referenčního řečníka se spočte vzdálenost jeho modelu k ostatním modelům řečníků a ti se pak seřadí podle této vzdálenosti a vytvoří kohortu. Čísla seřazených kohortníků jsou uložena v souboru s názvem referenčního řečníka. Všechny soubory s kohortami k jednotlivým řečníkům jsou uloženy v jednom adresáři.

5.3.1. Sort_LogLike.exe

Program provádí seřazení kohortníků (nejbližších řečníků) pomocí vzdálenosti podle **Rosenberga** či **Reynoldse** (viz kapitola 4.3.1.).

Program je spouštěn příkazem:

Sort_LogLike.exe File.ini Train.txt Data Model CohortFile , kde
File.ini .. konfigurační soubor
Train.txt .. soubor s čísly modelů
Data .. cesta k souborům s daty, tedy k parametrizovaným nahrávkám *.prm
(př: "Param_Test\%04d.prm")
Model .. cesta k souborům s modely, tedy k souborům typu *.gmm
(př: "Models\%04d.gmm");
CohortFile .. cesta k výstupním souborům s kohortou pro každého řečníka
(př: "Cohort\OffLineCohort%03d.txt")

Obsah konfiguračního souboru:

[COMPARE-CN]	Parametry pro porovnání modelů.
NumberOfModels=630	Počet modelů v testu.
Sort=0	Typ kritéria pro určení vzdálenosti ref. model – model (0-pro Rosenberg , cokoliv jiného pro Reynolds).

Ostatní soubory jsou shodné s popisem v kapitole 5.2.1.

5.3.2. Sort Models.exe

Program provádí seřazení kohortníků (nejbližších řečníků) podle vzdálenosti jednotlivých složek jejich modelů podle **Battacharryan míry** (viz kapitola 4.3.1.). Výsledná míra je pak dána součtem od jednotlivých složek.

Program je spouštěn příkazem:

Sort_Models.exe File.ini Train.txt Model CohortFile , kde

File.ini .. konfigurační soubor

Train.txt .. soubor s čísly modelů

Model .. cesta k souborům s modely, tedy k souborům typu *.gmm

(př: "Models\%04d.gmm");

CohortFile .. cesta k výstupním souborům s kohortou pro každého řečníka

(př: "Cohort\OffLineCohort%03d.txt")

Obsah konfiguračního souboru:

[COMPARE-CN]	Parametry pro porovnávání modelů.
NumberOfModels=630	Počet modelů v testu.
[MODEL-DB-GMM]	Parametry pro načítání modelu.
MixturesNumber = 0	Ukončovací podmínka shlukování (pro hodnotu 0 se bude shlukovat do dosazení kritéria).
NormalizeJ = 0	Normalizace kritéria (nejen ukončovacího) počtem vektoru ve shluku (je-li nastaveno na 1).
FinalJValue = 500.0	Ukončovací podmínka shlukování - hodnota kritéria (je-li takto shlukováno).
FinalJSumm = 0	Je-li nastaveno na 1, pak se počítá celkové kritérium (součet), jinak se bere max. kritérium.
FinalRecluster = 0	Zapnutí úplného přeshlukování na konec.
UseCF = 0	Bude využita informace o činiteli jistoty.
SaveTxt = 1	Formát uložení (TXT/BIN).

Ostatní soubory jsou shodné s popisem v kapitole 5.2.1.

5.4. Výpočet EER CompEER KW.exe

Výpočet chyby EER verifikačního testu je prováděn pomocí programu **CompEER_KW.exe**, jehož tvůrcem je ing. Aleš Padrta. Program je spouštěn příkazem:

CompEER_KW.exe Result.txt CorrectResults.txt Min Step Max EER.txt det.txt roc.txt ,
kde:

Result.txt .. soubor s výsledky jednotlivých testů, je vytvořen souborem Verify_BGM.exe

CorrectResults.txt .. soubor se správnými výsledky (po řádcích 1 nebo 0)

Min .. minimální uvažovaná hodnota prahu

Step.. počet bodů ROC/DET křivky

Max .. maximální uvažovaná hodnota prahu

EER.txt .. cílový soubor s údajem o velikosti chyby EER daného souboru testů.

det.txt .. cílový soubor s body DET-křivky

roc.txt .. cílový soubor s body ROC-křivky

6. Experimenty

Účelem provedených experimentů bylo zjistit vliv normalizačních metod na úspěšnost rozpoznávání v úloze verifikace řečníka v uzavřené množině. Bylo provedeno několik experimentů s různými korpusy dat pro různé druhy normalizací v závislosti na velikosti kohorty pro normalizaci.

Výsledky testů by měly ukázat nejen jaká z normalizačních metod je nejlepší, ale také pro jaké nahrávací podmínky je jaká z metod výhodnější, či jaká je optimální velikost kohorty.

6.1. Testované korpusy dat

Jednotlivé korpusy se lišily v počtu řečníků, nahrávacích podmínkách i ve velikosti testovací/trénovací promluvy.

6.1.1. UWB-RobustVerify100CZ [19]

Jde o nahrávky 100 českých řečníků, na které jsou vytvořeny modely. Ty jsou natrénovány z 20 vět. Model pozadí je natrénován z těchto řečníků, od každého dvě věty. Test je verifikací každého řečníka s každým. Pro test je použita jedna věta. Věty pro model pozadí, pro model řečníka a pro testy jsou různé.

Vzorkovací frekvence je 44,1kHz. Pro účely tohoto korpusu byly nahrávky převzorkovány na 8kHz a uloženy s 16bitovou přesností.

Model je tvořen 32 složkami GMM, stejně tak i model pozadí.

Experiment obsahuje šest testovacích sad **A** až **F**. Sada **A** je s neupravenými nahrávkami získanými přes jeden mikrofon. Z těchto dat jsou natrénovány modely. V sadách **B** až **F** jsou testovací nahrávky buď modifikovány, nebo pořízeny jiným způsobem.

V sadě **B** je k testovacím nahrávkám přimíchán aditivní šum v náhodně zvoleném poměru SNR 15 nebo 20dB. Náhodně je vybrán druh šumu ze čtyř prostředí: auto, vlak, ulice a metro.

V sadě **C** je simulováno zkreslení kanálem. To spočívá ve zkreslení frekvenční přenosové charakteristiky ve čtyřech pásmech na náhodnou hodnotu mezi 0 – 30dB.

Sada **D** je kombinací sady **B** a sady **C**, tedy nahrávky ze sady **B** jsou zkresleny průchodem kanálu.

Nahrávky ze sady **E** jsou pořízeny paralelně jiným mikrofonem, a kromě zkreslení kanálem je zde přítomen i stacionární aditivní šum.

Nahrávky sady **F** jsou pořízeny přes telefon. Byly pořízeny v jinou dobu než předchozí, proto neobsahuje nahrávky všech řečníků, ale jen 74 dostupných.

6.1.2. DoD [20]

Tento korpus byl nahrán při dnech otevřených dveří na katedře kybernetiky ZČU v roce 2005. Obsahuje 42 řečníků, kde test je verifikací každého řečníka s každým. Celý experiment tedy obsahuje 1764 porovnání. Délka trénovací nahrávky je přibližně dva odstavce textu. Délka testovací nahrávky je v délce jedné věty až jednoho odstavce. Dat pro modely pozadí je málo, protože většina řečníků namluvila pouze dvě promluvy.

Kvalita nahrávek je průměrná, byl použit hlavový mikrofon, nahrávka obsahuje běžný hluk při dni otevřených dveří. Vzorkovací frekvence je 8kHz.

Model je tvořen 32 složkami GMM, model pozadí je 256 složkový GMM.

6.1.3. Timit [21]

Tento korpus obsahuje promluvy od 630 řečníků, kteří reprezentují hlavní dialekty americké angličtiny. Obsahuje 3 testovací promluvy a 5 trénovacích promluv, ze kterých je vytvářen model řečníka. Každá promluva je v rozsahu jedné věty.

Test je verifikací všech modelů řečníků se svou testovací promluvou a následně s náhodně vybranou množinou odlišných testovacích promluv. Celý experiment obsahuje 32130 porovnáání.

Vzorkovací frekvence je 16kHz.

Model je tvořen 8 složkami GMM, model pozadí je 256 složkový GMM.

6.1.4. Yoho [22]

Tato databáze řečníků byla vytvořena na zakázku americká vlády. Jde o 138 řečníků (108 mužů a 30 žen) z různých částí USA. Data byla nasbírána v 3 měsíčním období v reálném kancelářském prostředí. S každým řečníkem proběhla vždy 4 sezení s 24 frázemi v každém z nich. Dále bylo 10 testovacích sezení s každým řečníkem, v každém sezení 4 fráze. Fráze obsahuje vždy tři různá dvojčíferná čísla přečtená za sebou.

Z toho vyplývá, že máme pro každého řečníka 4 trénovací nahrávky a 10 testovacích. Máme tedy také 4 modely pro jednoho řečníka.

Test je verifikací poloviny z počtu řečníků v korpusu (tedy 69 řečníků), jejich 4 modelů s každou ze svých 10ti testovacích promluv a následně s náhodně vybranou množinou odlišných testovacích promluv. Do kohorty je vybíráno z druhé poloviny řečníků, tedy z těch co nejsou testováni. Celý experiment obsahuje 8280 porovnáání.

Vzorkovací frekvence je 8kHz.

Model je tvořen 16 složkami GMM, stejně tak i model pozadí.

6.2. Použitá parametrizace

Způsob parametrizace promluvy je ***Param_KW*** (označení modulu v systému SoAP), který je založen na MFCC. Nastavení jednotlivých parametrů a jejich popis viz následující:

PreemK = 0.97	Koeficient preemfáze , pro zvýraznění vyšších frekvencí.
WinLength = 25	Délka Hammingova okna v ms.
ShiftLength = 10	Délka posunu okna (menší rovno WinLength) v ms.
NumFilters = 25	Počet pásmových filtrů.
MinFreq = 200	Minimální frekvence při rozmíst'ování filtrů.
MaxFreq = 0	O kolik má být maximální frekvence menší než je maximální skutečná.
Shape = 0.0	Tvar filtru 0.0 trojúhelník, 1.0 obdélník, něco mezi – lichoběžník.
NumCepst = 20	Počet keprálních koeficientů.
Delta = 1	Pro 1 se počítají Delta koeficienty.
DeltaDelta = 0	Pro 1 se počítají DeltaDelta koeficienty (Delta pak musí být také 1).
BlackmanFilter = 4	Provede filtraci Blackmanovým oknem, pro 1 - filtrace spektra, 2 - filtrace log-spectra, 3 - filtrace keprsta, 4 - filtrace keprsta i s delta a acel.
BlackLength = 11	Délka Blackmannova okna.
DownSample = 3	Poměr případného finálního downsamplu vektoru příznaků, ponechá pouze každý 3-tí vektor příznaků.
VAD = 1	Zapne detektor reci, ponechá jen řečové příznakové vektory.
minVADrez = 0.05	Relativní minimální počet vektorů označených jako řeč.

6.3. Použité modelování

Pro modelování byl použit *Model_DBEMGMM* (označení modulu v systému SoAP).
Jde o algoritmus EM. Nastavení jeho jednotlivých parametrů a jejich popis následuje zde:

[MODEL-DB-GMM]

MixturesNumber = xx	Počet složek (závisí na korpusu).
FinalJValue = 500.0	Ukončovací podmínka shlukování.
FinalJSumm = 0	Celkove kriterium se bere jako max. kriterium.
FinalRecluster = 0	Úplné přeshlukování je vypnuto.
UseCF = 0	Nebude využita informace o činiteli jistoty .
SaveTxt = 1	Formát uložení (1-TXT/0-BIN).

[MODEL-EM-GMM]

Mixtures = xx	Počet složek (závisí na korpusu).
ReestNumber = 16	Počet reestimací.
MinLike = 0.01	Vektory s nižší ppstí nejsou zahrnuty do reestimace.
UseCF = 0	Nebrat ohled na činitel jistoty.
SplitCoef = 0.5	Koeficient při dělení složky na dvě.
SaveTxt = 1	Formát uložení (1-TXT/0-BIN).

6.4. Výsledky experimentů

Zkoumanými faktory ovlivňujícími úspěšnost verifikace jsou použité normalizační metody a velikost skupiny řečníků N , které při normalizaci tvoří kohortu. Pro porovnání dosaženého zlepšení můžeme brát v úvahu chybu při normalizaci světovým modelem pozadí WM (je uvedena v grafech jako konstantní přímka, protože není závislá na velikosti kohorty).

Při posuzování optimální normalizační metody nesmíme zapomenout na časovou náročnost algoritmů.

Diskutovaným faktorem může být také ideální poměr blízkých a vzdálených řečníků v kohortě. Pro metodu CN (tedy offline vybranou kohortu) byl v těchto experimentech volen poměr 1:1. Označení N_N tedy znamená $2N$ řečníků, z nichž N je nejbližších a N nejvzdálenějších k referenčnímu řečníkovi.

Zde jsou uvedeny normalizační metody, popřípadě jejich nastavení, které jsou použité v experimentech. Detailní popis normalizačních technik je v kapitole 4 Normalizace skóre.

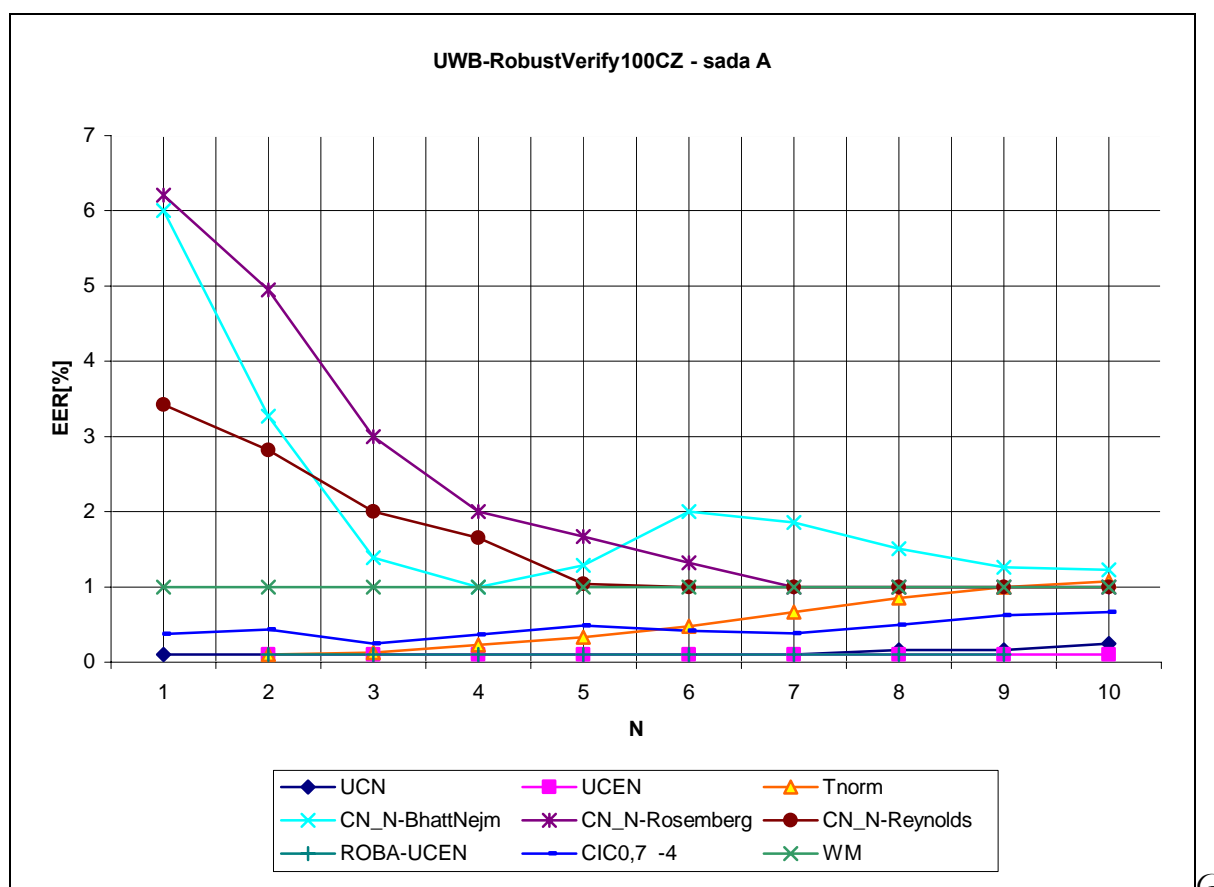
<i>UCN</i>	Neomezený skupinový model.
<i>UCEN</i>	Neomezený skupinový model s extrapolací.
<i>Roba-UCEN</i>	Robustní neomezený skupinový model s extrapolací.
<i>Tnorm</i>	T normalizace.
<i>CIC 0,7</i>	Neomezený skupinový model se shlukováním pomocí shlukovací metody Mini-Max, kde parametr $q=0.7$ určuje maximální vzdálenost řečníků ve shluku.
<i>CN_N – Rosenberg</i>	Skupinový model, kde kohorta byla vybrána pomocí programu Sort_LogLike.exe. Pro výpočet vzdálenosti mezi jednotlivými modely bylo použito <i>par-wise (Rosenbergovo) kritérium</i> . Výsledné normalizační skóre je určeno jako maximum skóre řečníka z kohorty, ne jako u výše uvedených normalizací průměr z kohorty.
<i>CN_N – Reynolds</i>	Skupinový model, kde kohorta byla vybrána pomocí programu Sort_LogLike.exe, pro výpočet vzdálenosti mezi jednotlivými modely bylo použito <i>divergence-like (Reynoldsovo) kritérium</i> . Výsledné normalizační skóre je určeno jako maximum.
<i>CN_N – BattNejm</i>	Skupinový model, kde kohorta byla vybrána pomocí programu Sort_Models.exe, pro výpočet vzdálenosti mezi jednotlivými složkami modelu byla použita <i>Battacharyanova míra</i> . Výsledné normalizační skóre je určeno jako maximum.
<i>WM</i>	Světový model pozadí.

6.4.1. UWB-RobustVerify100CZ

Již podle dosažených výsledků ze sady A (viz. Graf 1.) lze normalizační metody rozdělit do dvou kategorií podle toho, zda byla normalizační kohorta vybrána před nebo až při verifikačním experimentu (*on-line* nebo *off-line* kohorta).

Do první kategorie (výběr kohorty *off-line*) patří normalizace CN_N –BattNejm, CN_N –Rosenberg, CN_N –Reynolds. Z grafu je vidět, že jejich úspěšnost roste (chyba EER klesá) s velikostí kohorty. I přesto ale nedosahují ani úspěšnosti WM metody. Z toho lze usuzovat, že předvýběr kohorty před samotným testováním není moc přesný. Se zvětšující se kohortou se ale zvyšuje šance, že výsledné maximální skóre řečníka z kohorty bude blízko skóre skutečného řečníka dané promluvy.

Opakem je druhá kategorie (výběr kohorty *on-line*), kam patří normalizace UCN, UCEN, Roba-UCEN, CIC0.7-4, Tnorm. Ty mají opačný trend. Jejich úspěšnost klesá s velikostí kohorty, protože výsledné skóre je zde určeno jako průměr od všech členů kohorty. Dosahují větších úspěchů než metoda WM.



Graf 1. Chyba verifikace v závislosti na velikosti kohorty pro korpus **UWB-RobustVerify100C**, sada **A**

Z testů pro korpus UWB-RobustVerify100C (kompletní tabulky výsledků všech sad v Příloze 1) vychází nejlépe metoda UCN, a to pro malou velikost kohorty (1-5 řečníků). Snížení chyby využitím normalizace UCN vůči WM je ukázáno v tabulce č. 3.

Na druhou stranu, normalizační metoda CIC0,7-4 dává horší výsledky než normalizace UCN, ze které je odvozena. Protože jde jen o složitější postup úpravy výsledku z UCN, který ale nevede ke zlepšení účinnosti, v dalších experimentech se již s touto normalizací nepočítá. Pro

zlepšení CIC normalizace by bylo výhodné zkusit jinou shlukovací metodu či měnit parametry pro automatické shlukování.

Stejně je to i s normalizační metodou ROBA-UCEN, která dává podobné výsledky jako UCEN. V dalších experimentech již není použita.

Metody CN_N sice nedosahovali zlepšení UCN normalizace, ale zašuměné sady B až F dosahovali lepších výsledků jak normalizace světovým modelem WM. Nejlepší úspěšnost normalizace CN_N byla pro data s aditivním šumem (sada B). Bohužel nelze s jistotou říci, jaké z kritérií výběru nejbližších členů kohorty je nejlepší (zda Rossenberg, Reynolds či Battacharrya).

Sada	A	B	C	D	E	F
WM	1	11	6,8554	18	4	13,6985
UCN	0,1	6	1,7695	11,982	1	9,7793

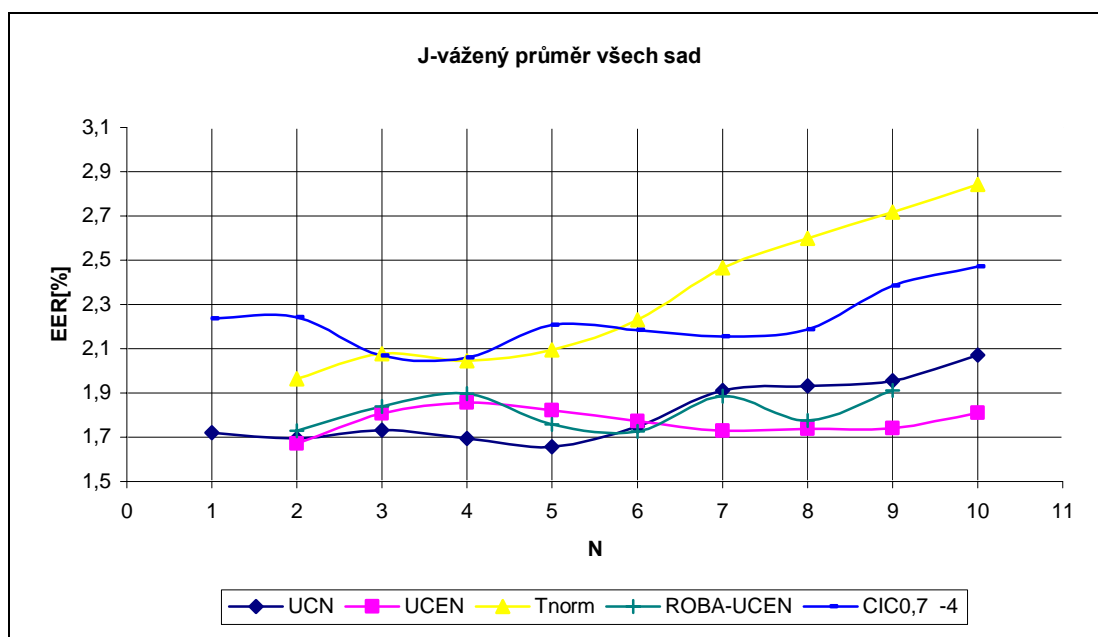
Tabulka 3. Chyba verifikace EER[%] pro korpus *UWB-RobustVerify100C*

Pro účely porovnání jednotlivých metod mezi sebou, nezávisle na sadě, jsem použil vážený průměr:

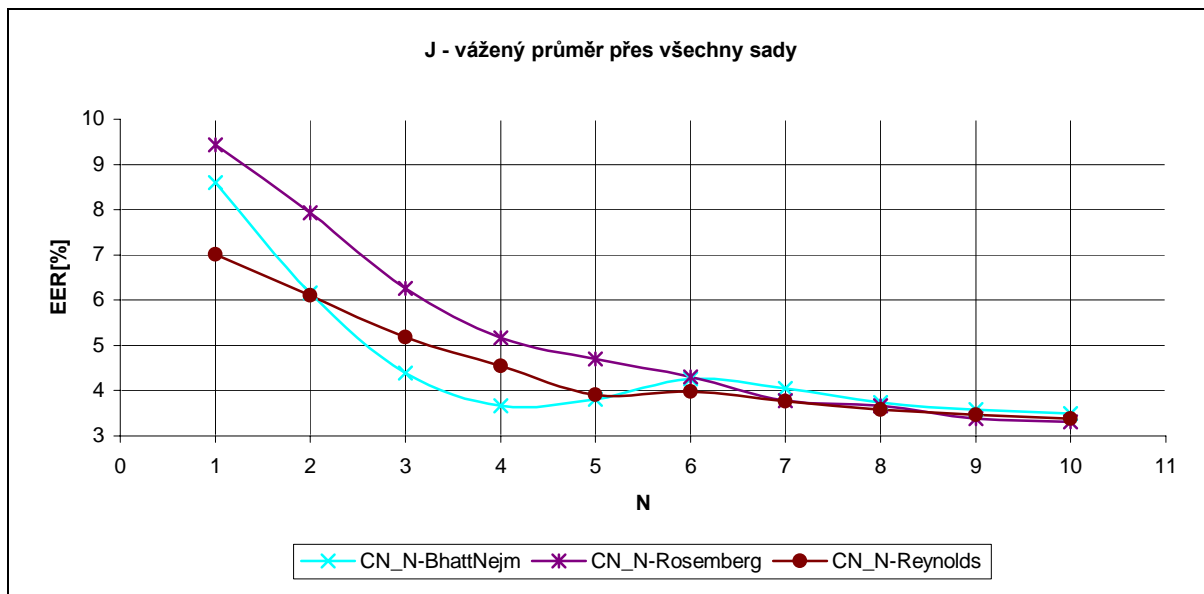
$$J_N^{norm} = \frac{\sum_{sada=A}^F \frac{EER_N^{norm}(sada)}{EER_{wm}(sada)}}{\sum_{sada=A}^F \frac{1}{EER_{wm}(sada)}}, \quad (34)$$

kde: *norm* druh normalizace (UCN,CN_N,...)
N počet řečníků v kohortě
sada typ sady (A až F)

Výsledky on-line a off-line metod jsou zobrazeny zvláště kvůli většímu rozlišení (viz. Graf č.2 a Graf č.3). Je zde opět vidět, že nejlepší metodou je UCN pro malý počet řečníků (1-5) v kohortě. Velmi blízko jsou výsledky metod UCEN a ROBA-UCEN (vycházejí ze stejného principu).



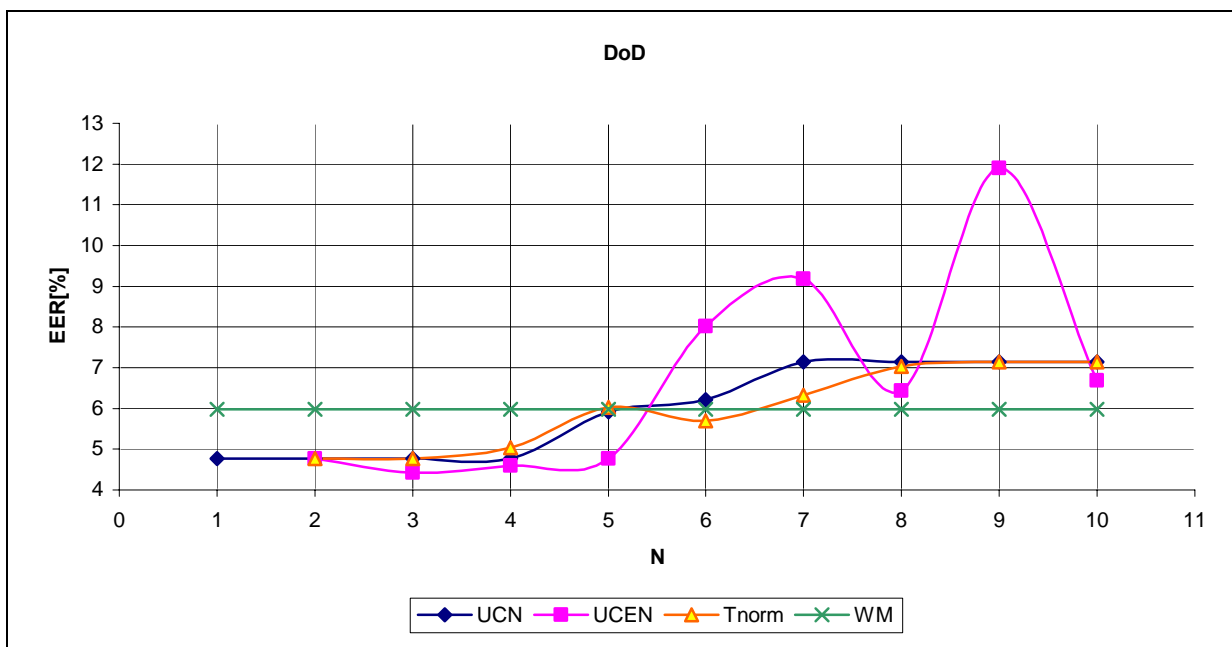
Graf 2. Chyba verifikace v závislosti na velikosti kohorty pro korpus *UWB-RobustVerify100C*, vážený průměr všech sad (J), on-line metody



Graf 3. Chyba verifikace v závislosti na velikosti kohorty pro korpus **UWB-RobustVerify100C**, vážený průměr všech sad (J), off-line metody

6.4.2. DoD

Výsledky jsou podobné jako u korpusu UWB-RobustVerify100C sady A. Metody CN_N nedosahují ani úspěšnosti metody WM (kompletní tabulka výsledků v Příloze 2). Na rozdíl od metod založených na výběru kohorty on-line, kde je patrné zlepšení účinnosti verifikace. Nejlepší výsledek EER dává metoda UCEN a to 4,4201%, oproti WM 5,98%. Pro vyšší počet N je však úspěšnost mnohem horší (viz graf č. 4).

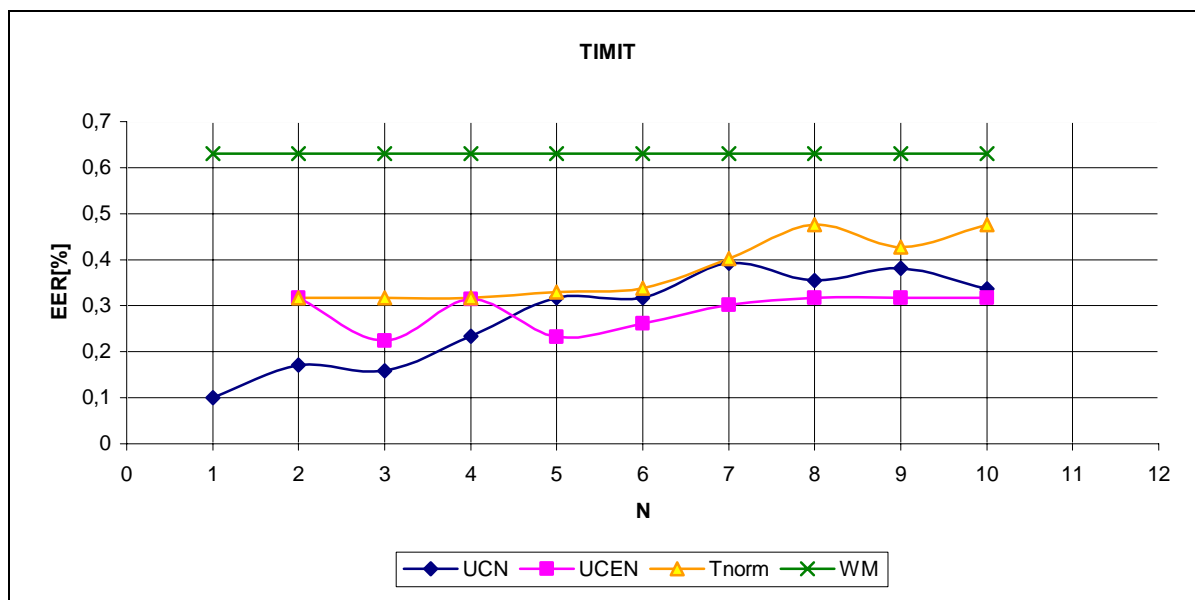


Graf 4. Chyba verifikace v závislosti na velikosti kohorty pro korpus **DoD**

Vítězem je opět spíše metoda UCN (4,7619%), která má velmi podobné výsledky jako UCEN, je ale stabilnější. Optimální počet členů kohorty je 1-4.

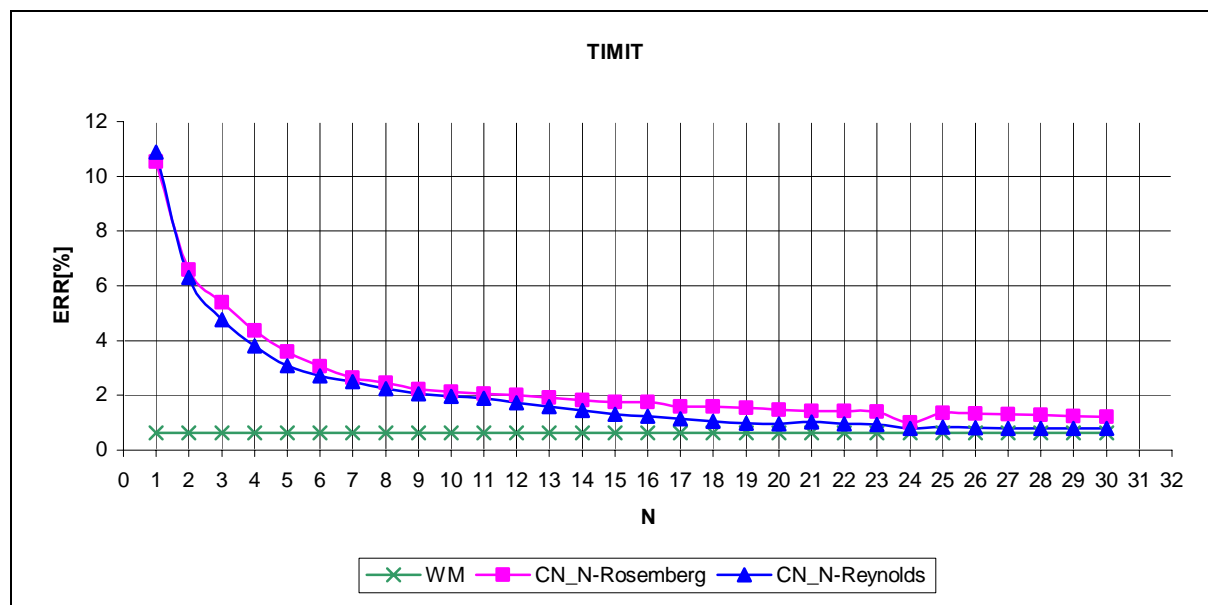
6.4.3. Timit

Na tomto korpusu jsme opět dosáhli pomocí on-line metod značného zlepšení účinnosti verifikace (viz graf č.5). Nejlepší metodou je UCN, kde ERR = 0,1% , vůči WM 0,63% (kompletní výsledky v Příloze 3)



Graf 5. Chyba verifikace v závislosti na velikosti kohorty pro korpus **Timit**

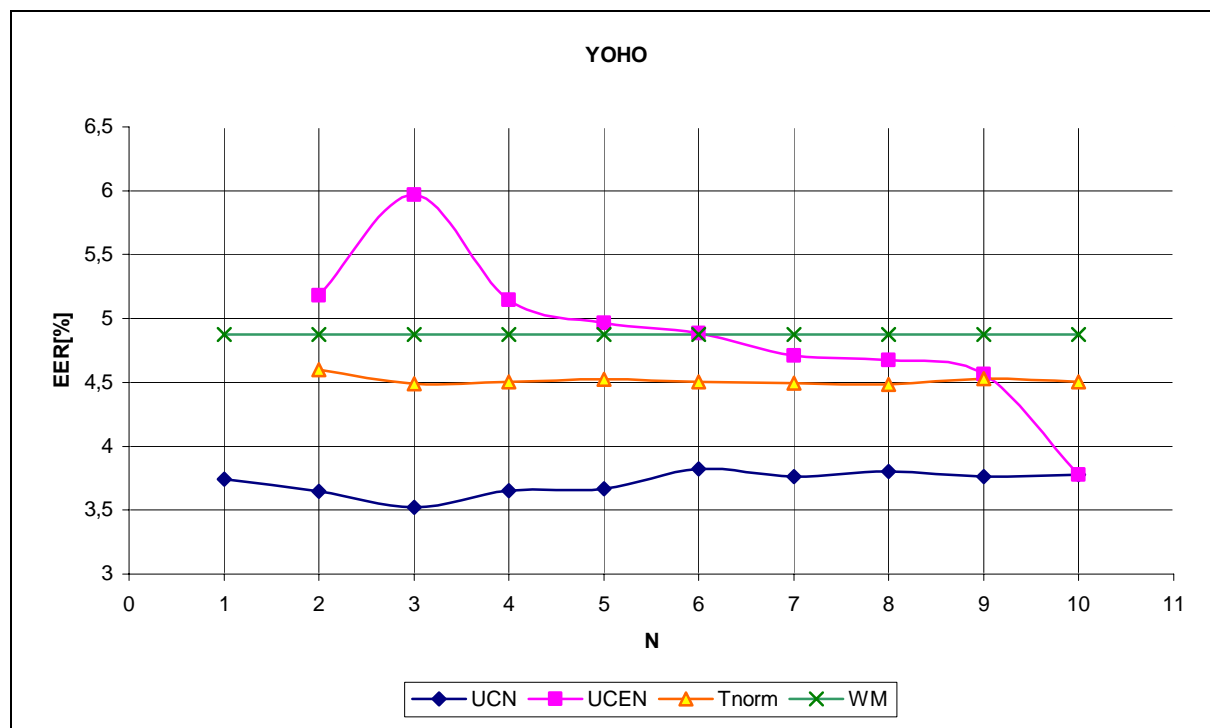
Pro metodu CN_N je chyba EER stále větší než WM (jak ukazuje graf č.6). Při zvyšování počtu členů kohorty chyba klesá exponenciálně. Ale i pro hodnotu N=30, tedy 30 nejbližších a 30 nejvzdálenějších řečníků (60 členů kohorty), je chyba větší než WM a pokles už není tak strmý. Metoda CN_N je sice časově rychlejší, ale nepřináší žádné zlepšení účinnosti. Pro velký počet řečníků je výhodnější WM, jehož výpočet je nejrychlejší z testovaných metod a dává lepší výsledky.



Graf 6. Chyba verifikace v závislosti na velikosti kohorty

6.4.4. Yoho

V korpusu Yoho (na rozdíl od všech ostatních) je pro účely kohorty vyhraněna skupina řečníků, kteří se testu nezúčastní. To však ovlivňuje úspěšnost normalizačních metod. Ty už z principu do své kohorty vybírají pouze N nejbližších řečníků k řečníku referenčnímu, nikoliv však samotného referenčního řečníka. Z toho vyplývá, že úspěšnost potvrzení správné identity se nemění, avšak úspěšnost zamítnutí nesprávného modelu je tím zhoršena (nemáme k dispozici model, který skutečně k promluvě patří). Z tohoto hlediska se dají očekávat horší výsledky rozpoznávání.



Graf 7. Chyba verifikace v závislosti na velikosti kohorty pro korpus *Yoho*

Z výsledků v grafu č.7 je vidět, že opět nejmenší chybovost je u normalizace UCN. Oproti WM s $EER=4,8731\%$ snižuje tato metoda chybovost až na $3,5235\%$. I metoda Tnorm, a pro větší počet řečníků i metoda UCEN, dávají lepší výsledky než metoda WM (kompletní výsledky v Příloze 4).

6.5. Časová náročnost algoritmů

Verifikace řečníka je časově náročný proces, jehož vyhodnocení hraničí se zpracováním v reálném čase (což je ale obvykle náš požadavek na tuto úlohu). Normalizace obecně velmi prodlužuje dobu trvání výpočtu, protože se provádí velké množství porovnání (obzvláště u online metod). Doba verifikace je silně závislá na počtu řečníků v systému (viz. tabulka č.5).

Pro urychlení výpočtu skóre normalizační kohorty (tedy pro nalezení nejbližších N kohortníků) v online metodách se používá následující postup:

- Výpočet skóre je proveden pouze pro každou Z -tou složku testovacího vektoru promluvy (s rostoucím Z tedy klesá jak přesnost výpočtu, tak i časová náročnost).
- Nalezne se určitý počet nejbližších řečníků porovnáním jednotlivých nepřesných skóre. Velikost tohoto předvýběru je dána číslem M , které je bráno procentně z počtu všech možných řečníků, z kterých je vybírána kohorta.
- K těmto nejbližším řečníkům je znovu vypočítáno skóre, ale již pro celý vektor testovací promluvy.
- Následně se z předvybraných řečníků vybere nejlepších N , které reprezentují normalizační kohortu.

Časovou náročnost (a zároveň i úsporu při použití předvýpočtů) ukazuje tabulka č. 4. Údaje v tabulce jsou v sekundách. Výpočet délky trvání byl proveden pro korpusy Timit a DoD s metodou UCN s kohortou 5 řečníků. Protože metody UCEN a Tnorm jsou založeny na stejném principu (vždy je testovaná promluva porovnána se všemi modely pro nalezení nejbližších řečníků), dá se předpokládat i u nich stejná časová náročnost. Nároky na čas těchto metod nejsou závislé ani na velikosti kohorty, protože je nutné vždy vypočítat vzdálenost testovací promluvy ke všem modelům.

Při předvýběru je vybíráno 20% řečníků z celého korpusu, tedy $M=0,2$. Délka trvání výpočtu testů u jednotlivých korpusů je nepřímo závislá na faktoru Z , tedy na urychlení výpočtu. Faktor Z lze zvyšovat, dokud jsou poskytované výsledky testů shodné (nezvyšuje se chyba).

Z	1	2	4	6
Timit	63534,12	34329,71	22565,68	18854,12
DoD	1164,53	925,01	825,34	724,31

Tabulka 4. Délka trvání [s] výpočtu korpusu Timit a DoD v závislosti na Z^*

Z tabulky č.4 tedy vyplývá, že úspora času je největší u korpusů s velkým počtem řečníků (velký počet porovnání), kdy se doba trvání výpočtu sníží až na třetinu. Výsledky zůstávají ale neměnné.

To souvisí také s velikostí předvýběru M . Snažíme se M nastavit dostatečně velké, aby se v předvýběru objevila celá předpokládaná kohorta nejbližších řečníků. Čím větší bude Z , tím bude první výběr nejbližších řečníků nepřesný. Proto s rostoucím Z by mělo růst i M , tedy skupina řečníků pro druhý (přesný) výběr cílové kohorty.

Tabulka č.5 ukazuje časovou náročnost jednotlivých testů s různými korpusy. Údaje v tabulce jsou v sekundách. Použitá metoda normalizace je UCN s $N=5$ řečníků v kohortě a zrychlením $Z=2$. Je zde uvedena i délka provedení jednoho dílčího testu (jedné verifikace).

Korpus	Yoho	Timit	DoD	UWB-RV100CZ
Celkové trvání	5000,84	34329,71	925,01	7177,38
Jeden dílčí test	0,6040	1,0685	0,5244	0,7177

Tabulka 5. Délka trvání [s] výpočtu jednotlivých korpusů*

Z výsledku je vidět, jak velikost korpusu zpomaluje verifikaci (viz tabulka č.5). Ale i přes velký počet porovnání jsou metody normalizace zvládnutelné v reálném čase. Doba verifikace jedné promluvy zřetelně nepřesahuje 1 sekundu, což odpovídá od 6% do 30% reálného času (údaj je závislý na počtu řečníků, ze kterých je vybírána kohorta).

*Pozn: Výpočty prováděny na počítači AMD 1,8GHz s OS WinXp, metodou UCN s 5ti kohortníky.

7. Závěr

Cílem této diplomové práce bylo porovnat mezi sebou různé normalizační metody. Na úlohu verifikace působí mnoho chyb, které jsme se pomocí normalizace snažili odstranit. Měly být nalezeny nejlepší metody a současně i podmínky, kdy je vhodné je použít.

K tomuto účelu bylo vytvořeno několik programů, které měly na určených korpusech dat provést verifikační testy, tedy zjistit, zda daná promluva byla vyslovena řečníkem s proklamovanou identitou. Získané pravděpodobnostní skóre jsme pak normalizovali pro následné porovnání s pevně stanoveným prahem.

Experimenty byly prováděny na několika rozdílných sadách dat. Jednotlivé korpuse se lišily jak do počtu řečníků v nich obsažených, tak i druhem poruchy, kterou byla nahrávka znehodnocena.

Výsledky testů prokázaly účinnost jednotlivých normalizačních metod. Většina z nich vedla k podstatnému zmenšení chybovosti verifikační úlohy. Ne však všechny metody byly stejně úspěšné.

Metody založené na skupinovém modelu, kde řečníci byly do skupiny (kohorty) vybírání před testováním (tedy off-line), nedosahovaly žádaných zlepšení vůči metodě světového modelu WM. Výjimkou byla zašuměná data s pomocí aditivního šumu. Účinnost těchto normalizací se zvyšovala s počtem řečníků v kohortě, tím se ale také princip těchto metod blížil principu WM, aniž by obvykle dosáhl jeho úspěšnosti. Z výsledků tedy vyplývá, že normalizace světovým modelem je účinnější, než off-line normalizační metody založené na skupinovém modelu.

Naopak normalizace s kohortou nejbližších řečníků vybíranou až během testování (on-line) vedla k značnému snížení chyby verifikace ERR. Nejlepší metodou byla normalizace UCN, kdy bylo pozorováno podstatné zlepšení jak u zašuměných tak i u nezašuměných dat. Ideální velikost kohorty byla zjištěna u všech testovaných korpusech 1 až 5 řečníků. Pro větší kohortu se u všech on-line metod účinnost zhoršovala. To je dáno tím, že výsledné skóre kohorty se počítá průměrem jednotlivých skóre jejich členů (na rozdíl od maxima v off-line metodách).

Dalším významným kritériem těchto metod je časová náročnost algoritmů, protože předpokládáme jejich využití v reálném čase. On-line metody jsou v tomto ohledu nejnáročnější. Dochází zde k porovnání promluvy se všemi potencionálními modely řečníků pro kohortu (obvykle všichni řečníci v systému), což vede k velkému počtu testů. Rychlost výpočtu se odvíjí od počtu řečníků v systému.

Zjistili jsme, že výpočet algoritmu se zrychleným předvýpočtem je časově zvládnutelný, aniž by se zhoršila kvalita výsledku, a to i pro velké korpuse dat. I s použitím normalizace UCN, která vyšla v testování nejlépe, trvá verifikace pod 1s, což odpovídá 6% až 30% délky testované promluvy (v závislosti na počtu řečníků vybíraných pro kohortu).

Výsledná absolutní účinnost verifikace byla pro korpuse s nízkou kvalitou nahrávek velmi špatná, a to i po provedení normalizace. Naopak pro data bez velkého šumu se účinnost blížila 99%. Další zvýšení úspěšnosti verifikace je možné docílit pomocí lepší filtrace nahrávek a použitím propracovanějších metod pro parametrizaci či tvorbu modelů.

Seznam literatury

- [1] Ing. Padrta, A. : „*Expertní systém pro rozpoznávání řečníka*“, rigorózní práce, Plzeň, 2004
- [2] Prokop, V. : „*Normalizace logaritmické pravděpodobnosti pomocí skupinového modelu v úloze rozpoznávání řečníka*“, Diplomová práce, Plzeň, 2005
- [3] Ing. Pužmanová, R., CSc : „*Biometrické systémy v praxi*“
<http://www.systemonlin.cz/site/bezpecnost>
- [4] Ing. Radová V. : „*Rozpoznávání řečníka*“, přednášky, Plzeň, 2005
- [5] Prof. Ing. Psutka, J. : „*Systémy vnímání a porozumění*“, přednášky, Plzeň, 2005
- [6] Sivakumaran, P., Fortuna, J., Ariyaeinia, M. A. : „*Score Normalisation Applied to Open-Set, Text-Independent Speaker Identification*“, Eurospeech, Ženeva, 2003
- [7] Auckenthaler, R., Carey, M., Lloyd-Thomas, H. : „*Score Normalization for Text-Independent Speaker Verification Systems*“, Digital Signal Processing 10 (42-54), UK, 2000,
<http://www.ling.mq.edu.au/ling/units/slp806/papers/score-normalisation.pdf>
- [8] Martin, A., Doddington, G., Kamm, T., Ordowski, M., Przybocki, M. : „*The DET curve in assessment of detection task performance*“, Eurospeech, Rhodes, 1997,
- [9] Kranthi Kiran, P. : „*Speech recognition using MFCC and Dynamic Time Warping*“, Kharagpur, Indian Institut of Technologi, India
- [10] Xu Shao, Miller, B. : „*Clean speech reconstruction from noisy Mel-frequency cepstral coefficients using a sinusoidal model*“ Norwich, School of Information Systems, UK, 2002 ,
http://www.dcs.shef.ac.uk/~xu/download/sx_icassp_03_1.pdf
- [11] Boccardi, F., Drioli, C. : „*Sound morphing with Gaussian Mixture Models*“, University of Padova, Italy, 2001 , <http://www.csis.ulie/dafx01/proceedings/papers/boccardi.pdf>
- [12] Ing Vaněk, J., Ing Padrta, A., „*Robustní rozpoznávání řečníka. Třetí studie. Normalizace skóre pro verifikaci*“, Plzeň, 2004
- [13] Toshihiro Isobe., Jun-ichi Takahashi : „*Text-independent Speaker Verification using Virtual Speaker Based Kohort Normalization*“, NTT Data Corporation, Tokyo, Japan
- [14] Jung Gon Kim, Woo-Yong Choi, Sung Bum Pan, Hyung Soon Kim : „*A New Score Normalization Metod for Speaker Verification with Virtual Impostor Model*“, Electronics and Telecommunications Research Institute, Korea
- [15] Rosenberg, A. E., DeLong, J., Lee, C. H., Juang, B. H., Soong, F. K. : „*The Use of Kohort Normalized Score for Speaker Verification*“, Proc. ICSLP 92, pp. 599-602, 1992
- [16] Reynolds, D. A., Quatieri, T. F., Dunn, R. B. : „*Speaker Verification using Adapted Gaussian Mixture Model*“, Digital Signal Processing 10, pp. 19-41, 2000
http://www.ll.mit.edu/IST/pubs/000101_Reynolds.pdf
- [17] Fukunaga, K. : „*Introduction on Statistical Pattern Recognition (Sekond Edition)*“, Academia Press, Inc, San Diego, 1990
- [18] Ing Vaněk, J., Ing Padrta, A. : „*Speaker Verification System SoAP*“, dokumentace k systému, Plzeň, 2005
- [19] Ing Vaněk, J. : „*Robustní parametrizace pro rozpoznávání řečníka. Druhá studie*“, Plzeň, 2004
- [20] Ing Vaněk, J., Ing Padrta, A. : „*Informace o korpusu DoD*“
- [21] Stránky s dokumentací ke korpusu Timit:
<http://www.mpi.nl/world/tg/corpora/timit/timit.html>
- [22] Stránky s dokumentací ke korpusu Yoho:
<http://www.biometrics.org/REPORTS/ICASSP95.html>
- [23] Dellaert, F. : „*The Expectation Maximalization Algorithm*“, Technical Report number GIT-GVU-02-20, Georgia Institute of Technology, 2002

Přílohy

Hodnoty ve všech tabulkách jsou EER[%].

Příloha 1: Testy na korpusu UWB-RobustVerify100CZ

Sada A

N	UCN	UCEN	Tnorm	CN_N - BattNejm	CN_N - Rosenberg	CN_N - Reynolds	ROBA - UCEN	CIC0,7-4	WM
1	0,1			6	6,2074	3,4216		0,378319	1
2	0,1	0,1	0,1	3,2675	4,9492	2,8229	0,1	0,434343	1
3	0,1	0,1	0,13131	1,3893	3	2	0,1	0,248878	1
4	0,1	0,1	0,22902	1	2	1,6481	0,1	0,368344	1
5	0,1	0,1	0,32876	1,2824	1,6726	1,0394	0,1	0,481714	1
6	0,1	0,1	0,47529	2	1,3234	1	0,1	0,416464	1
7	0,1	0,1	0,66452	1,8597	1	1	0,1	0,383838	1
8	0,15994	0,1	0,85314	1,5033	1	1	0,1	0,497995	1
9	0,16162	0,1	1	1,2562	1	1	0,1	0,618484	1
10	0,24535	0,1	1,0689	1,2222	1	1		0,6606	1

Sada B

N	UCN	UCEN	Tnorm	CN_N - BattNejm	CN_N - Rosenberg	CN_N - Reynolds	ROBA - UCEN	CIC0,7-4	WM
1	6			14,526	16	13,315		5,871136	11
2	6	6	7	12,701	17,624	11,765	5	6	11
3	6	6	7,8412	12,88	15,113	12,048	6	5,961419	11
4	6	6	8	12	12,495	9,7637	7	5,331372	11
5	5	6	7,4345	11	11,787	10	6	5,504558	11
6	5	6	7,5872	10,572	11	11,672	6	6	11
7	6	6,2878	8	10	10,81	10	7	6,046984	11
8	5,9509	6,4662	8	9,0096	10,009	9,2339	6	6	11
9	6	6,6122	7,1776	9	9,1286	8,1781	7	6,62779	11
10	6,9631	6,8012	7,7467	9	8,2845	8		6,801779	11

Sada C

N	UCN	UCEN	Tnorm	CN_N - BattNejm	CN_N - Rosenberg	CN_N - Reynolds	ROBA - UCEN	CIC0,7-4	WM
1	2,2098			11,137	12,805	10,861		3,037826	6,86
2	1,7695	2,2958	3	9,5973	9	10,83	3,639759	3,41527	6,86
3	2	2,8098	3,7027	8	8,6626	9	3,541215	3,642977	6,86
4	1,9177	3	3	6,0085	9	8,5037	3,727273	2,532374	6,86
5	2,0668	3	3,1098	5	9	7	1,961383	2,851102	6,86
6	2,3224	2,5518	3,1039	5,4611	8	7	2	2,883783	6,86
7	2,7697	2,1623	3,321	5	7	7	2,350123	2,386679	6,86
8	2,7659	2,117	3,8824	5	6,9815	7	2,825855	3	6,86
9	3	2,0276	3,8963	5	6,0829	7	2,479153	3	6,86
10	3	2,3754	3,8754	5	5,2325	7		3	6,86

Sada D

N	UCN	UCEN	Tnorm	CN_N - BattNejm	CN_N - Rosenberg	CN_N - Reynolds	ROBA - UCEN	CIC0,7-4	WM
1	11,982			20,486	22,187	20,926		12,87175	18
2	13,384	11,051	13	16,419	22	18,535	11,88445	13	18
3	13	11,899	13	15	21	19,563	12,85142	11	18
4	12,681	11,984	12	15	19	19	11	11,65932	18
5	13	12	12	18	18	18,273	12	11	18
6	13	12,026	12,376	17	16,747	18	12	11,2032	18
7	13,466	11,801	13	17	16	18,71	12,40454	12,4377	18
8	13,036	12	12,83	17,372	16	17,982	10,80622	11,7314	18
9	13,891	11,82	13	17,062	15	18	12	11,82046	18
10	13	12,115	12	16,59	16,274	17,55		12	18

Sada E

N	UCN	UCEN	Tnorm	CN_N - BattNejm	CN_N - Rosenberg	CN_N - Reynolds	ROBA - UCEN	CIC0,7-4	WM
1	1			9,6652	12,272	9		2,144758	4
2	1	0,90909	1	7,9615	9,4144	7,1307	0,262626	2	4
3	1,2567	1	1	5,6858	8,0095	5,9255	0,388294	2	4
4	1,355	1	1	4,0657	6,1616	4,4209	0,481714	2,087797	4
5	1,535	1	1,0287	4	4,9522	4,1006	1	2,318814	4
6	1,6495	1	1,3115	4,5765	5	4	1	2,10101	4
7	2	1	1,5496	3,9655	4	3,4923	1,009827	2,029266	4
8	2	1	1,3451	3,4459	3,6333	3,08	0,930732	2	4
9	1,7695	1	1,6025	3,7201	3	2,8622	1,278641	2,151052	4
10	1,7535	1	1,9503	3,4022	3	2,5796		2,311784	4

Sada F

N	UCN	UCEN	Tnorm	CN_N - BattNejm	CN_N - Rosenberg	CN_N - Reynolds	ROBA - UCEN	CIC0,7-4	WM
1	12,277			19,142	19,178	23,288		13,77473	13,7
2	11,52	12,054	14,041	16,676	18,906	21,575	13,4459	12,62471	13,7
3	11,319	13,071	13,699	15,282	17,808	19,249	13,69863	12,4187	13,7
4	10,543	13,699	13,612	15,264	17,808	19,178	14,45281	12,82827	13,7
5	9,7793	12,928	13,699	15,682	17,808	17,912	13,60835	13,35123	13,7
6	10,959	12,692	13,291	14,331	17,643	18,451	12,81952	13,69863	13,7
7	10,838	12,329	13,624	15,068	16,731	17,122	14,04918	13,69863	13,7
8	10,851	12,21	13,699	15,969	16,438	15,946	13,35928	12,32877	13,7
9	10,959	12,416	14,355	15,068	15,981	15,328	13,71687	13,69863	13,7
10	11,919	12,826	15,068	15,068	16,082	15,068		14,14278	13,7

Všechny sady – vážený průměr J

N	UCN	UCEN	Tnorm	CN_N - BattNejm	CN_N - Rosenberg	CN_N - Reynolds	ROBA - UCEN	CIC0,7-4
1	1,720811			8,603094	9,428644	7,013362		2,236087
2	1,695059	1,67241	1,963177	6,154751	7,936542	6,103744	1,729011	2,242142
3	1,733312	1,808021	2,077903	4,39385	6,256628	5,1887	1,840518	2,067623
4	1,695053	1,8565	2,045548	3,671932	5,165913	4,542042	1,898464	2,060603
5	1,656555	1,822207	2,093757	3,811307	4,701817	3,910937	1,759163	2,206028
6	1,75067	1,771963	2,230791	4,266877	4,307892	3,980044	1,727002	2,185447
7	1,912146	1,728844	2,464134	4,044935	3,785016	3,77173	1,885903	2,154358
8	1,931945	1,736259	2,597491	3,741656	3,668272	3,586621	1,774195	2,187042
9	1,955501	1,739522	2,718709	3,579202	3,384519	3,466184	1,910227	2,385311
10	2,071802	1,81024	2,84316	3,49272	3,308601	3,385197		2,472298

Příloha 2: Testy na korpusu DoD

N	UCN	UCEN	Tnorm	CN_N- Rosenberg	CN_N- Reynolds	WM
1	4,7619			16,667	21,429	5,98
2	4,7619	4,7619	4,7619	10,44	9,5238	5,98
3	4,7619	4,4201	4,7619	7,1429	8,6538	5,98
4	4,7619	4,5909	5,0366	7,1429	5,633	5,98
5	5,9201	4,7619	6,031	7,1429	6,177	5,98
6	6,2137	8,0214	5,6911	7,1429	7,1429	5,98
7	7,1429	9,1829	6,3298	7,1429	7,1429	5,98
8	7,1429	6,434	7,0291	7,1429	7,1429	5,98
9	7,1429	11,905	7,1429	7,1429	7,1429	5,98
10	7,1429	6,6783	7,1429	6,8643	6,4608	5,98

Příloha 3: Testy na korpusu Timit

N	UCN	UCEN	Tnorm	WM	CN_N- Rosenberg	CN_N- Reynolds
1	0,1			0,63	10,524	10,885
2	0,17073	0,31746	0,31746	0,63	6,5913	6,3107
3	0,15873	0,22379	0,31746	0,63	5,3968	4,752
4	0,23315	0,31429	0,31746	0,63	4,3773	3,8147
5	0,31746	0,23281	0,32962	0,63	3,6037	3,0738
6	0,31746	0,26123	0,3383	0,63	3,0659	2,6984
7	0,39177	0,30218	0,40188	0,63	2,6372	2,4971
8	0,35556	0,31746	0,47619	0,63	2,4499	2,243
9	0,38161	0,31746	0,4266	0,63	2,2222	2,0635
10	0,33651	0,31746	0,47619	0,63	2,1162	1,9723
11				0,63	2,0635	1,8889
12				0,63	2,0066	1,7315
13				0,63	1,9048	1,5873
14				0,63	1,8225	1,4496
15				0,63	1,746	1,3088
16				0,63	1,746	1,2352
17				0,63	1,5873	1,1506
18				0,63	1,5873	1,0516
19				0,63	1,5415	0,987
20				0,63	1,4621	0,95238
21				0,63	1,4347	1,0219
22				0,63	1,4286	0,95238
23				0,63	1,4089	0,94381
24				0,63	1,0121	0,80502
25				0,63	1,3504	0,83677
26				0,63	1,3281	0,81082
27				0,63	1,2974	0,79365
28				0,63	1,2754	0,79365
29				0,63	1,2484	0,79365
30				0,63	1,2179	0,79365

Příloha 4: Testy na korpusu Yoho

N	UCN	UCEN	Tnorm	WM
1	3,7439			2,25
2	3,649	5,1812	4,6014	2,25
3	3,5235	5,9661	4,4882	2,25
4	3,6522	5,1449	4,5047	2,25
5	3,6674	4,9638	4,5248	2,25
6	3,8224	4,8872	4,503	2,25
7	3,7629	4,7101	4,4928	2,25
8	3,8043	4,6739	4,4866	2,25
9	3,7629	4,5652	4,529	2,25
10	3,7772	3,7777	4,5039	2,25

