VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Záznam streamovaného audia

Ročníkový projekt

2006 **Radovan Tůma**

Záznam streamovaného audia

Odevzdáno na Fakultě informačních technologií Vysokého učení technického v Brně
dne 2. května 2006

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Pavla Matějky
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

. . . . . .
Radovan Tůma
2. května 2006

Abstrakt

Skupina zpracování řeči na Fakultě Informačních technologií Vysokého učení technického v Brně poslední dobou dosahuje velmi dobrých výsledků v oblasti identifikace jazyků, pro další vývoj je zapotřebí získávat další data pro trénování a testování identifikačních nástrojů. Tato práce se zabývá získáváním potřebných dat. Jednak se zaměřuje na techniky zaznamenávání streamů internetových rádií a dále se snaží získávání mluvené řeči z těchto záznamů. Zvolený přístup se snaží využít nástrojů dostupných na fakultě nebo opensource programů. Prvním úkolem práce bylo získat záznamy z rádií vysílajících v požadovaných jazycích. Dále se snaží zhodnotit možnosti využití již existujícího fonémového rozpoznávače a programu ngram pro nalezení úseků řeči v získaných záznamech na základě jazykových modelů řeči a hudby.

Klíčová slova

záznam audio streamů, segmentace, phnrec, ngram

Poděkování

Tímto bych chtěl poděkovat za výborné vedení práce Ing. Pavlovi Matějkovi.

Abstract

Speech group at the Faculty of Information Technology have very good results in the language identification. For further improvements in this field it is necessary to get more data for training and testing identification tools. The main object of this project was to download streams from internet radios and to recognize speech blocks in received data. The first objective was to download stream in different languages, the second was to mark speech segments in the saved audio files using phonem recogniser and application ngram according to language models of speech and music. The project is trying to get best results using software avaible on our faculty and opensource applications.

Keywords

downloading audio streams, segmentation, phnrec, ngram

Obsah

Obsah 6

1 Úvod 8
1.1 Zaměření projektu . . . 8
1.2 Poznámky k použitým prostředkům . . . 8

2 Záznam streamovaného audia 9
2.1 Úvodem . . . 9
2.2 Vyhledávání zdrojů . . . 9
2.3 Nástroje pro nahrávání . . . 9
2.4 Průběh nahrávání . . . 10
2.5 Popis skriptů . . . 10

3 Segmentace 11
3.1 Úvod . . . 11
3.2 Trénování . . . 11
3.3 Segmentování . . . 12
3.4 Vývoj a ladění segmentátoru . . . 12
3.4.1 Stupeň ngramu . . . 13
3.4.2 Nastevení délky fragmentu a bloku . . . 13
3.4.3 Význam pauzy . . . 17
3.4.4 Nutnost postprocesingu . . . 17
3.4.5 Grafy . . . 17
3.5 Konečná podoba parametrů segmentace . . . 17

4 Vytvořené skripty a jejich použití 22
4.1 Zaznamenávání streamů . . . 22
4.1.1 str2wav . . . 22
4.2 Segmentace . . . 22
4.2.1 zpracovanitreninku.pl . . . 22
4.2.2 ucse . . . 22
4.2.3 segment.pl . . . 22
4.2.4 seg2sri.pl . . . 23

5 Závěr 24
5.1 Záznam . . . 24
5.2 Segmentace . . . 24

6 Apendix 1 - Co naleznete na CD **25**

Kapitola 1

Úvod

1.1 Zaměření projektu

Tento projekt se zaměřuje na získávání dat pro identifikaci jazyků(LID) vyvíjenou skupinou zpracování řeči na Fakultě informačních technologií. Úkolem bylo získat audio záznamy požadovaných jazyků z internetových rádií. tyto data jsou zapotřebá pro další testování a zdokonalování aplikace pro rozpoznávání jazyků. Jelikož pro rozpoznávání se ze stažených streamů používají pouze úseky mluveného slova, tak se práce v druhé části zaměřuje na označování úseků řeči v uložených zvukových nahrávkách.

1.2 Poznámky k použitým prostředkům

V rámci tohoto projektu jsem se zaměřil raději na co nejefektivnější využití již existujících programů namísto vývoje nových aplikací. V části zaměřené na záznam se využívá opensource multimediální přehravač Mplayer [1]. V druhé části, zaměřené na rozpoznání úseků řeči, je zkoumána možnost použití programů phnrec [3] a ngram [2].

Kapitola 2

Záznam streamovaného audia

2.1 Úvodem

Prvním zásadním úkolem tohoto projektu bylo zaznamenat audio streamy několika jazyků, jmenovitě: Hindi, Japonština, Vietnamština, Korejština a Mandarin(Čínština).

2.2 Vyhledávání zdrojů

Za účelem stahování těchto jazyků bylo zapotřebí najít internetová rádia, ze kterých půjde pořídit záznam. V tomto mi velice pomohlo několik internetových databází, zejména Multilingualbooks /citemultilingualbooks a Languages on the Web /citelonweb, obě stránky jsou zaměřené na podporu výuky cizích jazyků a obě mají i odkazy na rádia, kde se těmito jazyky mluví.
Pomocí výše uvedených stránek jsem tedy vyhledal rádia v požadovaných jazycích. Poté jsem tyto rádia vyzkoušel poslechem, zda kvalita streamu odpovídá požadavkům, tzn. není v nich příliš šumů a chyb zvuku způsobených velkou kompresí. Nakonec jsem ještě ověřil, zda se na nalezených radiích dostatečně mluví, protože základem práce bylo získat mluvené slovo.

2.3 Nástroje pro nahrávání

Po vyhledání radií vhodných pro stahování bylo zapotřebí přijít na způsob zaznamenávání jejich vysílání. První metoda, kterou jsem zvolil, byla stahování streamů pomocí nástroje wget, a následný řevod zaznamenaných souborů pomocí programu sox. Tato metoda ovšem nebyla vhodná, zejména kvůli velkým paměťovým nárokům, kdy bylo zapotřebí po jistou dobu stažený záznam, než byl soxem převeden na požadovaný formát, další nevýhodou byla časová náročnost, kdy toto řešení neumožňovalo převádět záznam, dokud nebyl celý stažen. Z těchto důvodů jsem se rozhodl použít přehravač, který by uměl stahovaný stream rovnou ukládat do suboru požadovaného formátu(pcm s hlavičkou 16 khz, 16 bitů). Nejvodnějším přehravačem, díky splnění výše uvedených požadavků a také díky podpoře velkého množství audio formátů se ukázal Mplayer. Jeho velkou předností bylo, že nepotřeboval ukládat téměř žádná dočasná data a také jeho poměrně malé hardwerové nároky. Poté co jsem zvolil vhodný přehravač jsem potřeboval napsat skript, kterým by se ovládalo stahování. Rozhodl jsem se napsat jej v Bashi. Skript ve své konečné verzi umožňuje nastavení adresy streamu, jména souboru pro ukládání a požadované délky záznamu. Toto nastavení se provádí parametry při spouštění. Abych dosáhl vyšší efektivity práce při stahování, tak jsem vytvořil ještě druhý skript, který obsahoval pro několik streamů paralelně spuštění výše uvedeného skriptu na určitou dobu.

2.4 Průběh nahrávání

Nahrávání dat bylo náročné zejména na čas, za který se podaří získat dostatečný objem dat, dále bylo třeba řešit nespolehlivost některých rádií, takže bylo zapotřebí občas stažené soubory ručně zkontrolovat, zda obsahují srozumitelný záznam. Kromě těchto nutných úkonů proběhlo stahování v podstatě bez komplikací. Čímž byla splněna první část projektu.

2.5 Popis skriptů

Pro nahrávání byly použity dva skripty 1. jsem nazval str2wav a 2. tahej. První skript se zabývá zaznamenáváním streamu pouští se s parametry adresa streamu, jméno výstupního souboru a čas záznamu skript si po spuštění načte parametry a spustí na pozadí mplayer s přislušnými parametry. Po spuštění mplayeru se pomocí příkazů ps a grep zjistí jeho PID, pomocí příkazu sleep počká po dobu která je secifikována jedním z přijatých parametrů a poté posílá, killem mplayeru signál aby se ukončil.
Druhý skript tahej obsahuje adresy streamů, požadovanou délku záznamu a formátování názvu výstupních souborů. S těmito parametry volá skript str2wav díky volání na pozadí je schopen spustit několik instancí tohoto skriptu pro různé streamy paralelně. Mezi paralelním spuštěním několika instancí skriptu str2wav se používá krátký sleep(např. 1 vteřina), aby se zamezilo možnosti, že si dříve spuštěný skript načete omylem PID později spuštěného.

Kapitola 3

Segmentace

3.1 Úvod

Na výše uvedenou část práce úzce navazuje tzv. segmentace, kterou je v tomto případě myšleno rozdělení souboru na segmenty řeči a hudby. Zhlediska využití získaných dat pro identifikaci jazyků nás zejména zajímá správné označování segmentů řeči. Vše ostatní bude označováno jako hudba.
Pro segmentaci jsme se rozhodli ověřit možnost použití nástrojů phnrec a ngram, které se využívají při identifikaci jazyků. Pokud by toto řešení bylo použitelné, tak by jeho hlavní výhodou bylo, že celý soubor by se programem phnrec zpracoval pouze jednou a dále by se pracovalo již jen s jeho výstupem, což je zápis fonémů a jejich časování. A tudíž by šel výstup ze segmentování použít jako vstup pro LID.
Při segmentování se vychází z toho, že se vytvoří jazykové modely řeči a hudby. Dále se programem phnrec zpracuje testovaný zvukový záznam, čímž vznikne soubor se zápisem fonémů a informací o jejich časovém určení, tento soubor budu nadále zkráceně označovat, podle jeho často používané přípony, jako rec soubor. Rec soubor se dále rodělí na časové úseky. Pro každý úsek se vytvoří řetězec fonémů, který se pomocí programu ngram testuje na podobnost s jazykovým modelem řeči a hudby. Podle toho pro který model výjde větší pravděpodobnost se rozhodne zda se jedná o řeč nebo hudbu.

3.2 Trénování

Prvním důležitým předpokladem pro správnou funkčnost segmentace je mít dobře natrénované modely řeči a hudby. První trénování proběhlo na základě již dříve oanotovaných zvukových záznamů, které jsem měl k dispozici. Trénování probáhalo pomocí skriptu, který z mlf(master label file) zjistil ve kterých časových úsecích se nalézá řeč, pro tyto úseky vybral z recu daného zvukového záznamu příslušné fonémy a tyto přidal do řetězce pro trénování řeči. Obdobným způsobem byl získán řetězec pro hudbu. Z obou řetězců byly programem ngram vytvořeny jazykové modely. Tím sem získal základní jazykový model pro testování segmentace. Jak se později ukázalo, tak takto natrénované modely nevykazovali uspokojivou rozpoznávací schopnost mezi řečí a hudbou. Pro další zpřesnění trénování jazyového modelu řeči proto bylo později použito několik set písniček různých interpretů a žánrů. Pro zpřesnění modelu řeči jsem použil trénovací řetězec pro Maďarštinu, který mi poskytl vedoucí projektu Ing. Pavel Matějka. Dalším důležitým parametrem trénování bylo rozhodnout, jaký řád zpracování programem ngram použít, ze začátku jsem zkoušel 2. řád, což znamená, že model vyjadřoval četnost používání dvojic fonémů. Později jsem přešel na 3. řád. Tato změna sice způsobila několika násobné zpomalení procesu rozpoznávání, ale odměnila se vysokým

nárustem úspěšnosti segmentace. Jelikož po všech úpravách stále docházelo k velké chybovosti, byl model hudby ještě rozšířen o řetězce odpovídající kombinaci řeči a hudby z původních trénovacích audio souborů.

3.3 Segmentování

Segmentování se provádí pomocí skriptu napsaného v perlu, který dostává jako vstup rec soubor daného záznamu, tudíž tento rec je zapotřebí vytvořit předem. Dále pro segmentování potřebujeme jazykový model řeči a hudby, jejichž vytvořením se zabívala předchozí podkapitola. Základní přístup k segmentování je ten, že se snažíme rozložit soubor na malé kousky(fragmenty) a u každého rozhodneme, zda se jedná o řeč či hudbu. Abychom mohli takové rozhodnutí provézt, tak potřebujeme sekvenci fonémů, kterou porovnáváme sjazykovými modely. Jelikož fragmenty jsou ětšinou příliš krátké a obsahují tudíž příliš málo fonémů pro kvalifikované rozhdnutí, volí se přístup, že se každý fragment rozšiřuje v pravo i vlevo o nějaký časový úsek. Experimentálně bylo zjištěno, že aby byl tento způsob testování použitelný, měl by mít jeden úsek rozsah více než 10 sekund. Tyto parametry, tzn. velikost 1 fragmentu a použité okolí, které udávám v počtu fragmentů jsem zvolil jako klíčové pro zkoumání nejlepšího způsobu testování.

3.4 Vývoj a ladění segmentátoru

V předchozích kapitolách jsem stručně shrnul problematiku trénování a segmentování. Nyní je čas popsat vývoj segmentátoru jak byl asi od začátku prováděn. Prvním úkolem bylo zpracovat všechny zvukové soubory, které jsem měl použít pro trénování a pro testování programem phnrec. V této části stačilo změnit formát souborů na raw 8kHz 16 bitů, k čemuž byl použit program sox. Na takto upravené soubory jsem použil nástroj phnrec a jeho výstupy jsem si uložil pro další zpracování.
Dále bylo zapotřebí vytvořit jazykové modely, k čemuž byl napsán skript v perlu, který z rec souborů a mlf souborů trénovacích záznamů vypsal fonémy do trénovacích řetězců pro řeč a hudbu. Dále stačilo jen na tyto řetězce zavolat program ngram, který vytvořil požadované jazykové modely.
Po provedení výše uvedených kroků již nic nebránilo tomu, abych mohl začít vyvíjet skript pro segmentaci samotnou. Zvolil jsem si, že skript napíši v jazyce perl, protože tento jazyk umožňuje poměrně jednoduše pracovat s textovými soubory, umožňuje jednoduchou práci s poli a hlavně s regulárními výrazy. Základem segmentování bylo procházení rec daného rec souboru a porovnávání řetězců získaných z jeho částí s jazykovými modely řeči a hudby. Bylo zapotřebí zvolit vhodný přístup z hlediska rozdělení rec souboru na vhodně dlouhé části. Ale jelikož pro věrohodné posouzení podobnosti znaků z daného časového úseku s jazykovými modely je zapotřebí blok o délce desítek vteřin, byl zvolen přístup překrývání těchto bloků, tzn.: posuzování začíná prvním celým blokem od začátku souboru a další bloky jsou o nějaký menší časový úsek posunuty. Každý blok rozhoduje o fragmentu (tak nazývám úsek, o který se posunuje a který je nejmenším časovým úsekem pro který skript umí rozhodnout, zda se jedná o hudbu nebo o řeč). Fragmenty nastavuji řadově na jednotky vteřin a bloky obsahují řádově desítky fragmentů. Pro lepší vysvětlení následuje obrázek. Pro všechny fragmenty na začátku rec souboru a na jeho konci a tudíž nemají dostatečné okolí pro kvalifikované rozhodnutí, určuji zda se jedná o řeč či hudbu na základě fragmentů následujících nebo předcházejících. Hlavním problémem vývoje segmentování bylo rozhodnout o velikosti fragmentu a bloku, tak aby rozhodovací řetězec byl dostatečně dlouhý, pro rozhodnutí a zároveň co nejkratší, aby nedocházelo k velkému ovlivnění sousedícími částmi souboru. Dále jde o velikost fragmentu, aby přechody mezi hudbou a řečí byly co nejjemnější.

Obrázek 3.1: Obrázek k vysvětlení rozdělení souboru rec na řetězce

3.4.1 Stupeň ngramu

Dalším důležitým faktorem je jak byly natrénovány modely, na čemž velmi záleží délka zpracování (pro trigramy je porovnání ngramem náročnější než pro digramy apod.). Také na použití digramů nebo trigramů záleží nastavení velikosti bloku, protože pro trigram je zapotřebí více fonémů pro smysluplné rozhodování. Nicméně trigramy se odměňují mnohem vyšší mírou úspěšnosti rozhodnutí. Jelikož náročnost zpracování mezi unigramy, digramy, trigramy ... narůstá exponenciálně byly ve větší míře testovány maximálně trigramy. Unigramy se naopak při prvníc testech ukázaly jako prakticky nepoužitelné, takže byly vyřazeny. Budu se zde tedy zabývat pouze digramy a trigramy. Takže zpracováním modelů programem ngram s parametrem order 2 a order 3.

3.4.2 Nastevení délky fragmentu a bloku

Pominuly nutnost rozhodnutí o stupní nramu, je hlavním testovaným parametrem nastavení délky bloku, jehož důležitost jsem zmínil výše. Postupně bylo testováno několik nastavení, dosažené výsledky znázorňuje následující tabulka, jedná se o srovnání správně anotace všech souborů a anotace, kterou jsem prováděl já. řádky tabulky obsahují mým skriptem určené třídy a sloupce správné třídy, procenta v každém políčku tabulky určují míru shody. tabulky mají prázdné řádky pro třídy které jsem neimplemetnoval. zkratky znamenají: S - speech(řeč), M - music(hudba), SM - speech and music(hudba s muzikou dohromady), N - noise(neurčitelný hluk), SI - silence(ticho). Za úspěch se považuje označování SM, N a M jako M a S jako S. SI nemá valný význam, protože záleží na nastavení konkrétního segmentátoru, jak dlouhé úseky ticha bude brát jako ticho a ne jako malou přestávku v řeči nebo hudbě. Já používám jako ticho úseky trvající minimálně 1 vteřinu. označení SM jako S nemusí nutně znamenat chybu, protože se většinou, dle mých testů s poslechem jedná o hlasitou řeč se slabým hudebním podkresem, který se téměř úplně ztrácí při převodu záznamů z původních 16 kHz na 8 kHz.

```
ngram 2. radu
frame = 0.5s block size = 20
M N S SI SM
M 10.64% 15.60% 41.53% 28.41% 3.82%
N
S 1.35% 4.94% 44.90% 6.24% 42.57%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 0.5s block size = 50
 M N S SI SM
M 11.79% 16.60% 43.32% 27.84% 0.45%
N
S 1.33% 4.96% 44.60% 6.65% 42.45%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 15
 M N S SI SM
M 12.01% 17.62% 36.81% 29.71% 3.85%
N
S 1.33% 4.85% 45.47% 6.45% 41.90%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 20
 M N S SI SM
M 11.59% 16.99% 39.01% 28.70% 3.71%
N
S 1.34% 4.88% 45.21% 6.48% 42.10%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 40
 M N S SI SM
M 11.38% 15.11% 46.99% 26.51% 0.00%
N
S 1.34% 5.11% 44.09% 6.73% 42.73%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 2s block size = 30
 M N S SI SM
M 11.38% 15.11% 46.99% 26.51% 0.00%
N
S 1.34% 5.11% 44.09% 6.73% 42.73%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
```

```
frame = 4s block size = 10
 M N S SI SM
M 11.38% 15.11% 46.99% 26.51% 0.00%
N
S 1.34% 5.11% 44.09% 6.73% 42.73%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
ngram 3. radu
frame = 0.5s block size = 20
 M N S SI SM
M 7.25% 16.33% 34.05% 19.64% 22.73%
N
S 1.44% 3.90% 47.01% 6.62% 41.03%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 0.5s block size = 50
 M N S SI SM
M 6.61% 16.79% 39.52% 18.03% 19.05%
N
S 1.49% 3.51% 45.79% 6.78% 42.42%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 15
 M N S SI SM
M 6.83% 16.84% 36.02% 18.78% 21.52%
N
S 1.48% 3.62% 46.65% 6.69% 41.57%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 20
 M N S SI SM
M 7.55% 16.54% 35.67% 18.75% 21.49%
N
S 1.44% 4.01% 46.47% 6.99% 41.09%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 40
 M N S SI SM
M 9.20% 16.62% 37.12% 22.14% 14.91%
N
S 1.39% 4.49% 45.78% 6.84% 41.50%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
```

```
frame = 2s block size = 30
 M N S SI SM
M 7.32% 12.20% 29.53% 17.98% 32.96%
N
S 1.46% 4.95% 48.02% 7.08% 38.48%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 4s block size = 10
 M N S SI SM
M 7.01% 12.66% 28.27% 18.27% 33.80%
N
S 1.47% 4.76% 48.55% 6.89% 38.33%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
lm rozsiren o Noise a SM jako music
ngram 2. radu
frame = 4s block size = 10
 M N S SI SM
M 6.19% 9.98% 22.07% 14.84% 46.92%
N
S 1.10% 4.85% 53.72% 6.85% 33.48%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 15
 M N S SI SM
M 7.56% 14.19% 34.08% 17.43% 26.74%
N
S 0.98% 3.82% 47.81% 6.52% 40.88%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
ngram 3. radu
frame = 4s block size = 10
 M N S SI SM
M 3.29% 8.74% 27.34% 10.36% 50.26%
N
S 1.25% 1.81% 77.01% 6.96% 12.97%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
frame = 1s block size = 15
 M N S SI SM
M 3.56% 9.75% 28.20% 10.77% 47.73%
N
S 1.10% 1.12% 69.52% 6.76% 21.50%
SI 0.00% 0.07% 0.05% 99.88% 0.00%
SM
```

3.4.3 Význam pauzy

Pro zpracování segmentace je v tomto případě, kdy se zpracovávají poměrně velké bloky důležité nastavení pauzy, skript je totiž implementován tak, že zpracovává úseky rozdělené pauzou zadané délky, jejíž optimální hodnota byla nalezena kolem 1 vteřiny, pokud by se pauza nepoužila, dojde k velkému zhoršení, protože po pauze často následuje změna z řeči na hudbu, která ba se jinak projevyla ve výstupu segmentace až se zpožděním. Význam puazy je v tom, že se zpřesní některé přechody mezi hudbou a řečí.

3.4.4 Nutnost postprocesingu

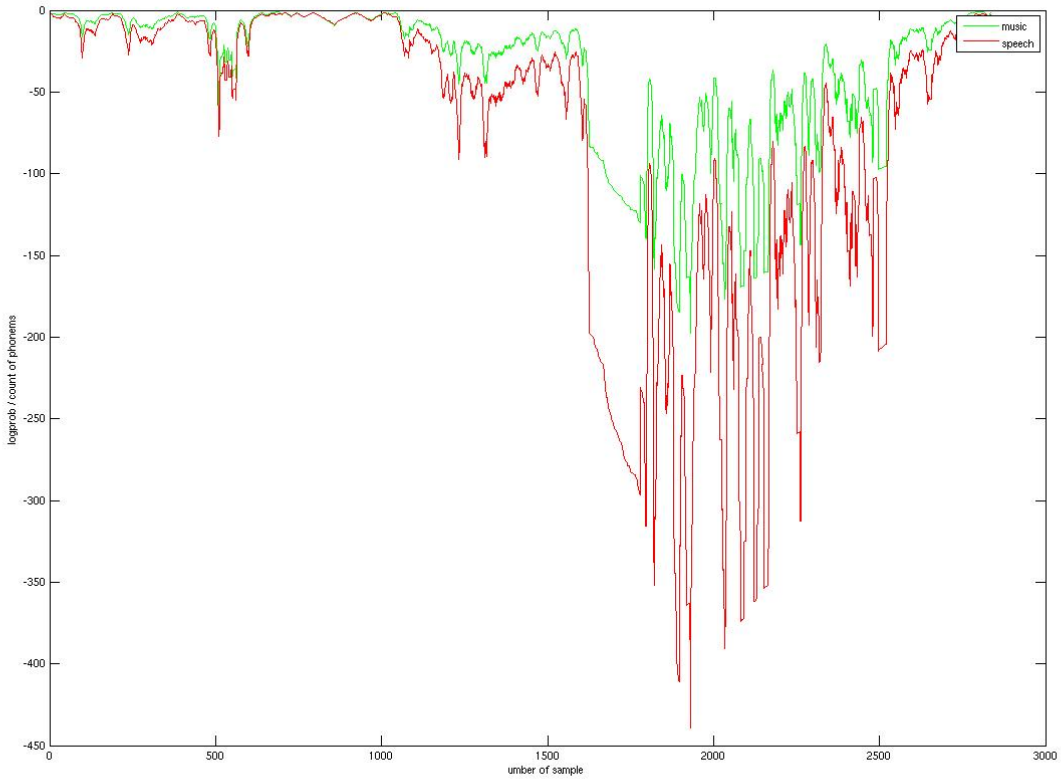
Při vývoji skriptu jsem se zabýval i nutností postprocesingu, tzn. úpravy výstupů segmentace za účelem dosažení lepších výsledků. Hlavní motivací pro úvahy o nutnosti postprocesingu byla možnost, že při zpracování bde docházet k označování malých úseků zpracovávaného záznamu za řeč uprostřed hudby nebo naopak. Případně by mohlo docházet k častému střídání tříd. žádná z těchto obav se však neprojevila a post procesing tedy nebyl použit. Hlavním důvodem, proč narozdíl od segmentace jinými způsoby k těmto jevům nedochází je velikost bloku, který určuje, zda je daný fragment řeč nebo hudba a tím, že se bloky překrývají, tudíž se téměř nestane aby malý kousek hudby, například nějaký gong nebo znělka uprostřed řeči, byl vyhodnocen jako hudba a naopak. Tento malý kousek se totiž ztratí ve velikosti bloku a blok je tudíž vždy podobnější třídě, kterou obsahuje z větší části.

3.4.5 Grafy

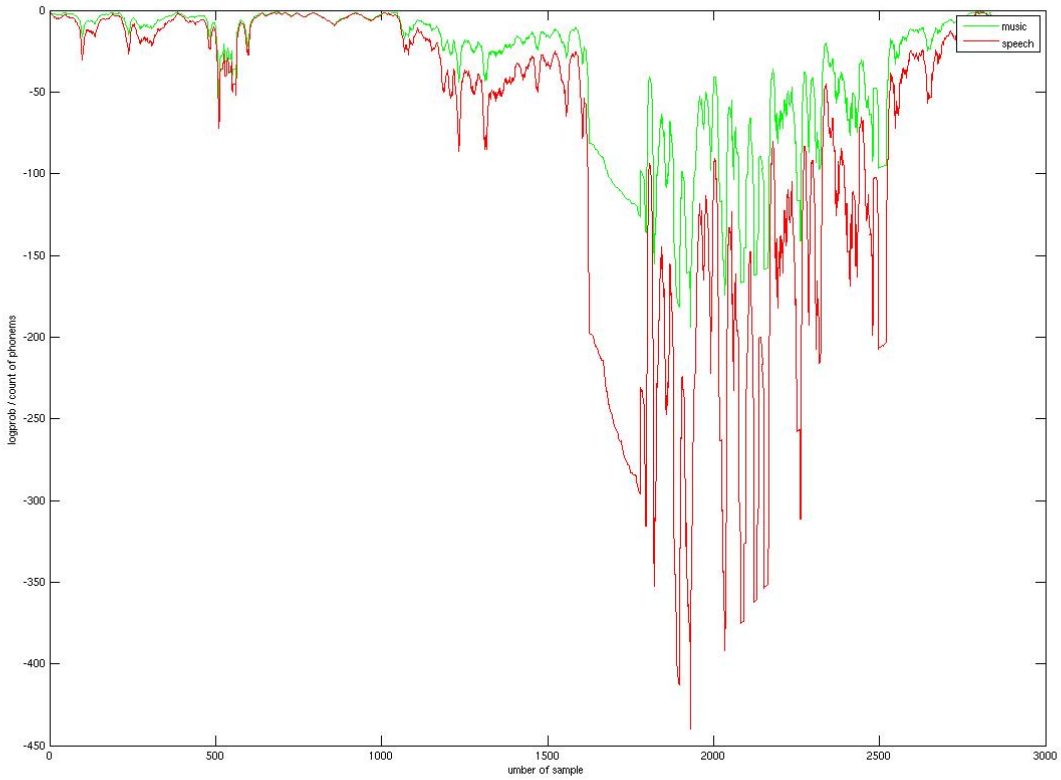
Pro lepší pochopení fungování segmentace založené na phnrecu jsem v průběhu vývoje upravil skript tak, aby vypisoval pro jednotlivé testované soubory hodnoty pravděpodobnosti shody s modely do souboru, který byl poté v matlabu vykreslen jako graf. Tyto grafy měly sloužit k demonstraci možností přesné detekce přechodů mezi řečí a hudbou. Ukázalo se, že možnosti detekce přesných míst změny jsou u této metody segmentace velmi omezené. Pro ilustraci následuje několik grafů. Hodnoty pravděpodobnosti, které vratil program ngram, jsou poděleny počtem fonémů, ze kterých byly získany, takže by se dalo říci, že se jedná o normované hodnoty pravděpodobnosti.

3.5 Konečná podoba parametrů segmentace

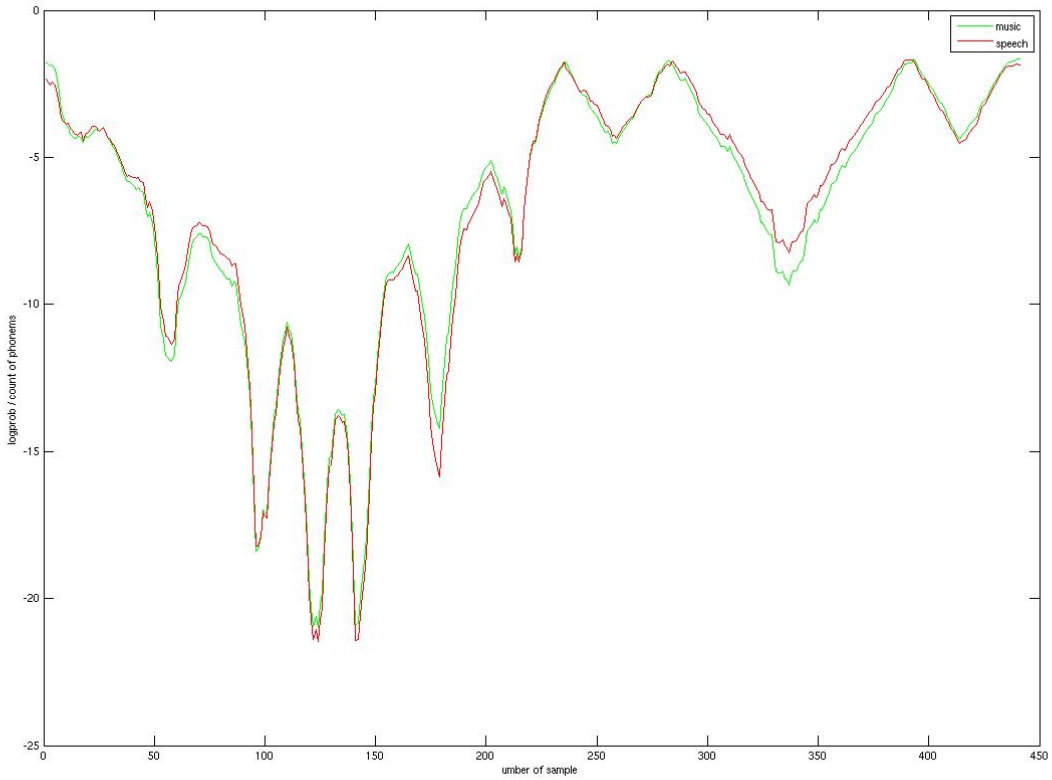
Jelikož výše segmentování po částech, chtěl bych teď vše upřesnit a shrnout volené parametry a metody použité v konečné podobě projektu.
Trénování bylo nakonec provedeno na dodaných trénovacích záznamech v trvání několika hodin, ke kterým jsem měl k dispozici jejich oanotování v tzv. master label file, tzn. označení tříd a jim náležících časů v daných nahrávkách. Jednalo se o záznamy v různých jazycích. Pro trénování hudby z nich byly použity třídy speech, noise a speech and music. Pro řeč potom třída speech. Dále byla hudba trénována na přibližně 2 GB hudebních skladeb ve formátu mp3 s bitrate většinou 64 až 128. Pro spřesnění trénování hudby byl dále použit řetězec používaný pro rozpoznávání maďarštiny. Natrénování je prováděno programem ngram 3. řádu.
Pro segmentaci byla použita velikost fragmentu 1 vteřina a velikost bloku 15 fragmentů s rozdělovací pauzou o velikosti 1 vteřiny. testování ngramem na podobnost bylo prováděno pro trigramy. Segmentovací skript na základě předloženého rec souboru a modelů řeči a hudby vytváří master label file ve formátu ’časod časdo třída’, kde časod a časdo jsou ve stovkách ns.



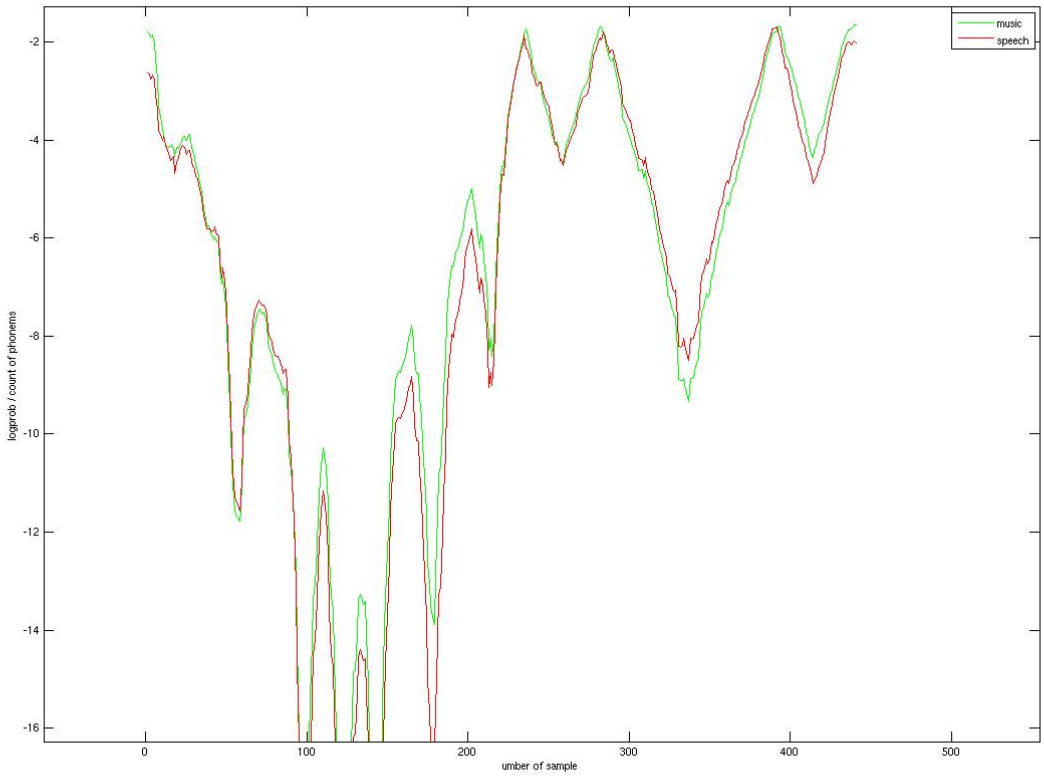
Obrázek 3.2: Soubor recording11_1, frame 1s blok 15 frame, digram



Obrázek 3.3: Soubor recording11_1, frame 1s blok 15 frame, trigram



Obrázek 3.4: Soubor recording15_3, frame 1s blok 15 frame, digram



Obrázek 3.5: Soubor recording15_3, frame 1s blok 15 frame, trigram

Kapitola 4

Vytvořené skripty a jejich použití

Výstupem projektu kromě této zprávy je i několik skriptů, které naleznete na přiloženém CD.

4.1 Zaznamenávání streamů

4.1.1 str2wav

Skript v bashi pro stahování streamu z internetu a jeho zaznamenávání do souboru pcm který používá 16-ti bitový formát a 16 kHz, stream zaznamenává po určenou dobu.
Spouští se s parametry:
-h : vypíše nápovědu
-s stream : adresa streamu
-o outputfile : jméno souboru kam ukládá stream
-t time : čas ve vteřinách po který se bude záznam provádět

4.2 Segmentace

4.2.1 zpracovanitreninku.pl

Skript, ktrý na základě předloženého mlf souboru, který je spojením několika původních mlf a kde je definovaným způsobem vždy uveden soubor ze kterého vychází pr odpovídající rec soubory vytvoří základ trénovacích řetězců.

4.2.2 ucse

Jednoduchá dávka, která volá ngram na trénovací řetězce řeči a hudby, čímž vzniknou jazykové modely řeči a hudby.

4.2.3 segment.pl

Hlavní skript, jehož vytvoření a nalezení ideálních parametrů pro správnou funkci bylo stěžejním úkolem této části projektu. Jako parametry příjmá délku fragmentu ve stovkách ns a počet fragmentů v bloku.
použití: segment.pl [-f delka -b velikost souborrec] -f delka: délkafragmentu
-b velikost: velikostbloku
souborrec: rec soubr, který má zpracovat

4.2.4 seg2sri.pl

Skript, který umožňuje využití master label file a k němu příslušného rec souboru pro identifikaci jazyků. Na základě předloženého mlf souboru a požadavku na třídu S nebo M vypíše na výstup odřádkované řetězce které nalezl v jednotlivých úsecích odpovídajících dané třídě.
použití: seg2sri.pl T file.rec file.mlf
T: třída S(speech - řeč) nebo M(music - hudba)

Kapitola 5

Závěr

Závěrem bych chtěl stručně zhodnotit dosažené výsledky.

5.1 Záznam

Co se týče záznamu, tak zde není moc co hodnotit, byl použit v podstatě 1 jednoduchý skript a dávka na jeho spouštění pro různé streamy. Důležité je, že skript plní svou funkci a dle mého názoru je celkem šetrný na hardwerové nároky. Díky použití programu mplayer si dobře poradí s širokou škálou formátů audio souborů. Hlavní prací na této části bylo nalezení streamů v poněkud exotických jazycích, což ovšem bylo prováděno bez použití inteligentních agentů, či jiného za tímto účelem vyvinutého software, takže se v podstatě nejednalo o vědeckou práci, která by si zasloužila hlubší rozbor.

5.2 Segmentace

Segmentace byla náročnější částí tohoto projektu a její výsledky se jistě dají nějakým způsobem ohodnotit, ideální bude srovnání s jinou segmentační metodou. Ve stejném čase jako já pracoval na naší fakultě na projektu segmentace Jan Hovorka, student magisterského programu, jelikož jsme oba testovali naše přístupy k segmentaci na stejných datech, dalo by se toto srovnání považovat za relevantní. Moje dosahovaná úspěšnost správného označení, lépe řečeno, jaké procento z toho, co jsem prohlásil za řeč je opravdu řeč, byla 69.52%(97.78% i se třídou SM a SI) proti 95.85% u Jana Hovorky. Toto hodnocení, za pedpokladu, že fonémy obsažené v úsecích SM budou jen tak málo zkreslené hudbou, aby bylo použitelné pro rozpoznávání jazyků, vyznívá lépe pro mně. Ovšem pokud se podíváme na celkovou úspěšnost nalezení řeči, tak moje úspěšnost je jen asi 70%, kdežto Jan Hovorka dosahuje přes 90%. Z tohoto srovnání plyne, že segmentace založená na fonémovém rozpoznávači dosahuje mnohem horších výsledků než jiné metody segmentace, což je dáno tím, že fonémový rozpoznávač nebyl pro tyto účely nikdy určen. Výhodou mého řešení je hlavně to, že používá stejné nástroje, které se používají pro rozpoznávání jazyků, k čemuž by jeho použití mělo nakonec sloužit.

Kapitola 6

Apendix 1 - Co naleznete na CD

Na přiloženém CD naleznete všechny výše zmíněné skripty, dále jazykové modely, které byly použity. Také zde jsou umístěny trénovací řetězce a rec soubory, na kterých si můžete vyzkoušet funkčnost skriptů.
Samozřejmě zde najdete i soubor README se stručným popisem všech souborů.

Literatura

[1] WWW stránky. Mplayer hq. http://www.mplayerhq.hu.

[2] WWW stránky. ngram. http://www.speech.sri.com/projects/srilm/.

[3] WWW stránky. phnrec. http://www.fit.vutbr.cz/research/groups/speech/index.php?id=phnrec.