

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
Preddiplomski sveučilišni studij računarstva

Završni rad

Proširenje baze govornih snimaka VEPRAD

Rijeka, rujan 2021.

Piero Battelli
0069082557

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
Preddiplomski sveučilišni studij računarstva

Završni rad

Proširenje baze govornih snimaka VEPRAD

Mentor: prof. dr. sc. Ivo Ipšić

Rijeka, rujan 2021.

Piero Battelli
0069082557

SVEUČILIŠTE U RIJECI
TEHNIČKI FAKULTET
POVJERENSTVO ZA ZAVRŠNE ISPITE

Rijeka, 12. ožujka 2021.

Zavod: **Zavod za računarstvo**
Predmet: **Programiranje II**
Grana: **2.09.02 informacijski sustavi**

ZADATAK ZA ZAVRŠNI RAD

Pristupnik: **Piero Battelli (0069082557)**
Studij: **Preddiplomski sveučilišni studij računarstva**

Zadatak: **Proširenje baze govornih snimaka VEPRAD / Expansion of the Speech Database VEPRAD**

Opis zadatka:

Za potrebe razvoja sustava za automatsko rasponavanje hrvatskog govora pripremite dodatne snimke snimljenog govora. Nove snimke potrebno je formatirati u skladu sa postojećim snimcima u bazi VEPRAD, te odrediti njihovu transkripciju.

Rad mora biti napisan prema Uputama za pisanje diplomskih / završnih radova koje su objavljene na mrežnim stranicama studija.

Piero Battelli


Zadatak uručen pristupniku: 15. ožujka 2021.

Mentor:



Prof. dr. sc. Ivo Ipšić

Predsjednik povjerenstva za
završni ispit:



Izv. prof. dr. sc. Kristijan Lenac

Izjava o samostalnoj izradi rada

Izjavljujem da sam samostalno izradio ovaj rad.

Rijeka, rujan 2021.

Piero Battelli

Sadržaj

Popis slika	viii
Popis tablica	x
1 Uvod	1
1.1 Raspoznavanje govora	1
1.2 Povijest razvoja	2
2 Alati	4
2.1 Audacity	4
2.2 HTK	5
2.3 Julia	6
2.4 Julius	7
3 Prva faza - priprema podataka	8
3.1 Izrada rječnika	9
3.2 Snimanje podataka	10
3.3 Izrada transkripcija	12
3.4 Kodiranje audio podataka	14

Sadržaj

4	Druga faza - izrada monofonih HMM	15
4.1	Izrada “flat start” monofona	16
4.2	Ispravljanje modela tišine	18
4.3	Restrukturiranje treniranih podataka	19
5	Treća faza - izrada trifona povezanih stanja	21
5.1	Izrada trifona	22
5.2	Povezivanje stanja trifona	24
6	Rezultati	26
7	Zaključak	29
	Bibliografija	30
	Pojmovnik	32
	Sažetak	33

Popis slika

1.1	<i>Napredak preciznosti (word accuracy) raspoznavanja govora pomoću Google Machine Learning-a [3]</i>	3
2.1	<i>Primjer sučelja prilikom korištenja Audacity aplikacije</i>	5
2.2	<i>Pregled svih alata od kojih se Hidden Markov Model Toolkit (HTK) sastoji [6]</i>	6
2.3	<i>Julia pokrenuta u command prompt-u na Windows OS</i>	7
3.1	<i>Isječak prompts.txt datoteke korištene u projektu</i>	10
3.2	<i>Primjer naziva datoteka</i>	12
3.3	<i>Sadržaj Master Label File (MLF) datoteke</i>	13
3.4	<i>Sadržaj phones1 datoteke koja uz foneme uključuje i kratku pauzu</i> . .	14
4.1	<i>Primjer uporabe Skriveni Markovljevi Modeli (HMM) kod raspoznavanja govora [12]</i>	16
4.2	<i>Primjer izgleda datoteke macros</i>	17
4.3	<i>Izgled modela kratkih stanki short pause (SP)</i>	19
4.4	<i>Aligned.mlf datoteka sa vremenskim oznakama trajanja pojedinih glasova</i>	20
5.1	<i>Primjer trifona i trifona povezanih stanja [15]</i>	22
5.2	<i>Primjer sadržaja stats datoteke</i>	23

Popis slika

6.1	<i>Ručno izrađena transkripcija datoteke sm1305210710.wav</i>	27
6.2	<i>Grafički prikaz audio datoteke sm1305210710.wav i pripadajuća trans- kripcija</i>	27
6.3	<i>Manualna transkripcija sadrži riječ “dalmaciji”</i>	28
6.4	<i>Automatska transkripcija sadrži riječ “dalmaciju”</i>	28

Popis tablica

3.1	Tablica koja opisuje snimljene podatke	11
3.2	Tablica koja opisuje govornike	11

Poglavlje 1

Uvod

Tema ovog završnog rada je proširenje baze govornih snimaka VrEmenske Prognoze-RADio (VEPRAD) pomoću snimki vremenskih prognoza koje se temelji na dobavljanju potrebnih podataka, snimanju te transkribiranju istih. U nastavku će prije detaljnijeg opisa samog postupka biti objašnjeno što je to zapravo raspoznavanje govora kao znanstvena disciplina i povijest razvoja discipline.

1.1 Raspoznavanje govora

Raspoznavanje govora interdisciplinarno je područje koje objedinjuje znanja računarstva i komputacijske lingvistike za razvoj metodologija i tehnologija koje omogućuju prepoznavanje i prevađanje izgovorenog u tekst koristeći računalo[1]. Razlikujemo dvije vrste sustava za raspoznavanje govora. Podaci kojima je proširena baza VEPRAD dobiveni su pomoću “treniranja” što zapravo predstavlja čitanje teksta ili izoliranog vokabulara u sustav. Za potrebe ovog rada korištene su snimke vremenskih prognoza o čemu nešto više kasnije. Drugi pristup su sustavi nezavisni od govornika koji su nešto rjeđi u primjeni. Neke od aplikacija koje se temelje na raspoznavanju govora su sustavi za pretraživanje ključnih riječi, sustavi za obavljanje poziva, unos podataka, procesiranje govora u tekst što je bilo potrebno obaviti u sklopu ovog projekta i slično. Neke od ključnih značajki raspoznavanja govora su preciznost, brzina, akustični trening tj. prilagođavanje sustava na akustično okruženje i različite stilove

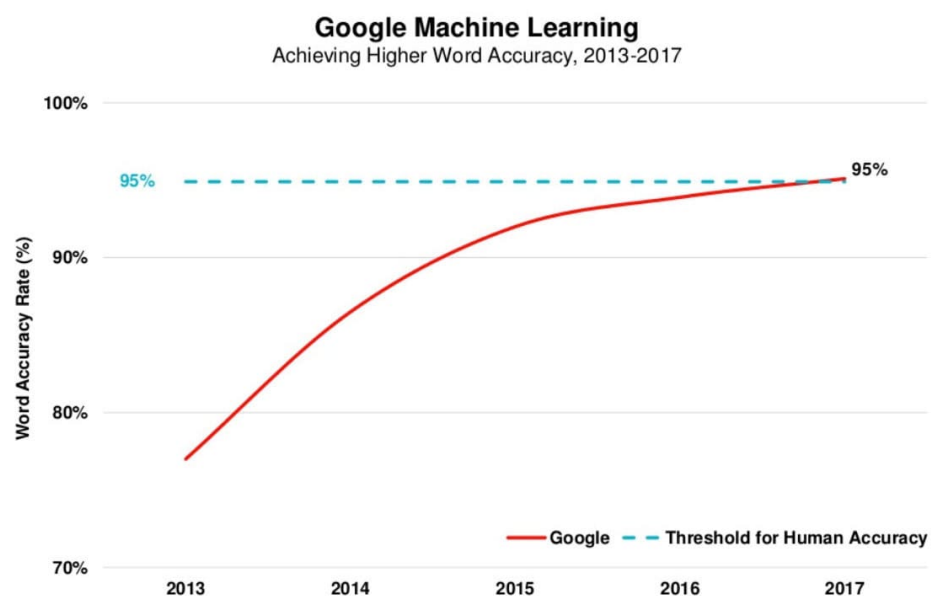
zvučnika te filtriranje riječi koje je potrebno ignorirati. S obzirom na to da su brzina i preciznost ključne značajke raspoznavanja govora razlikujemo nekoliko osnovnih algoritama od kojih se ističu Natural language Processing (NLP), HMM, neuronske mreže itd. Ovaj projekt temeljio se na treniranju koristeći HMM.

1.2 Povijest razvoja

Prvi sustavi za raspoznavanje govora nastali su pedesetih i šezdesetih godina prošlog stoljeća, ali s ciljem raspoznavanja brojeva. Jedan od takvih sustava bio je Audrey System koji je nastao 1952. godine u Bell Laboratories. Sedamdesetih godina prošlog stoljeća došlo je do velikog napretka zahvaljujući mnogobrojnim istraživanjima provedenim u Americi. Osamdesetih godina otkriće HMM označava procjenu vjerojatnosti da nepoznati zvuci zapravo čine riječ ili tekst dok se prije isključivo tražilo već definirane riječi i poznate nizove zvuka. Devedesetih godina eksponencijalno napreduje raspoznavanje govora zbog revolucije uzrokovane pojavom osobnih računala, a do 2000. godine sustavi za raspoznavanje govora dostižu jako visoku preciznost. U zadnjih desetak godina Google i IBM nalaze se na vrhu što se preciznosti i poticanja razvoja raspoznavanja govora tiče, a s obzirom na to da su troškovi istraživanja i razvoja istih sustava manji no ikada u budućnosti se očekuje veliki broj novih konkurenata.

U nastavku rada biti će obrađena poglavlja vezana uz korištene alate, prikupljanje podataka, izradu monofona i trifona, rezultate i zaključak rada.

Poglavlje 1. Uvod



Slika 1.1 *Napredak preciznosti (word accuracy) raspoznavanja govora pomoću Google Machine Learning-a [3]*

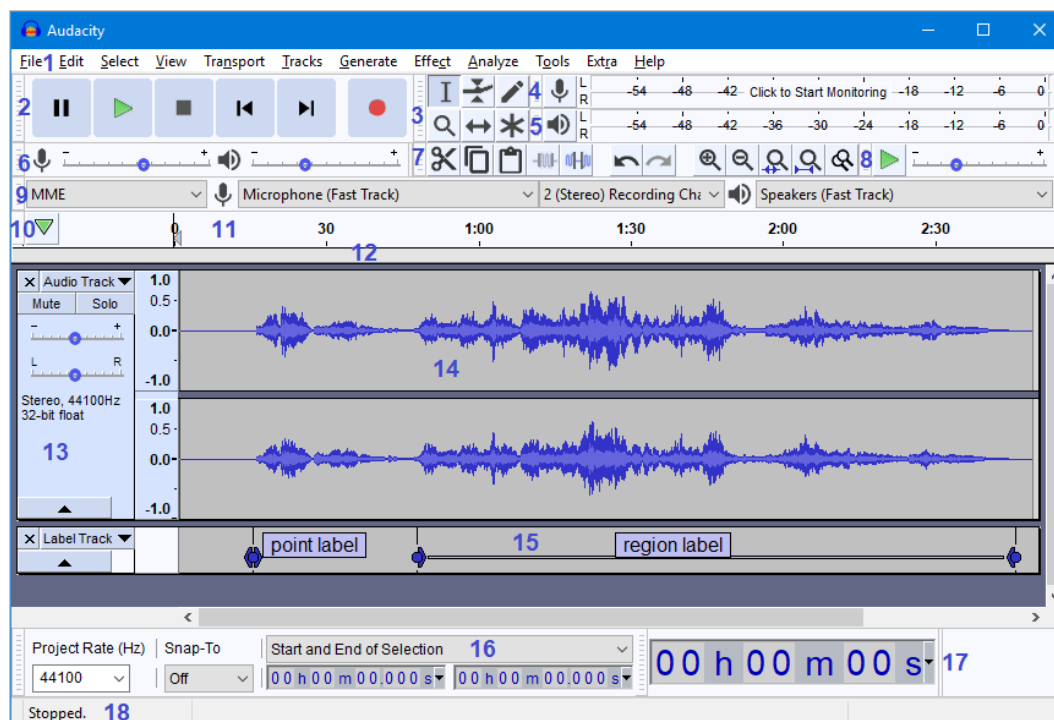
Poglavlje 2

Alati

U nastavku će biti ukratko opisani alati korišteni za izradu ovog projekta te dostupne hiperveze za preuzimanje istih na vaša računala. Navedene alate bilo je potrebno preuzeti na vlastito računalo kako bi se uspješno izradio projekt sa što manje poteškoća.

2.1 Audacity

Audacity je besplatan audio uređivač otvorenog koda te mu je jedna od osnovnih uloga snimanje različitih zvukova u koju se svrhu koristio i za ovaj projekt. Dostupan je na gotovo svim računalnim operacijskim sustavima te vrlo jednostavan za korištenje, a Audacity je korišten u prvoj fazi izrade projekta tj. za vrijeme dobavljanja samih audio zapisa vremenskih prognoza koji će u nastavku projekta biti korišteni za “treniranje”. Audacity je moguće preuzeti sa <https://www.audacityteam.org/download/>.

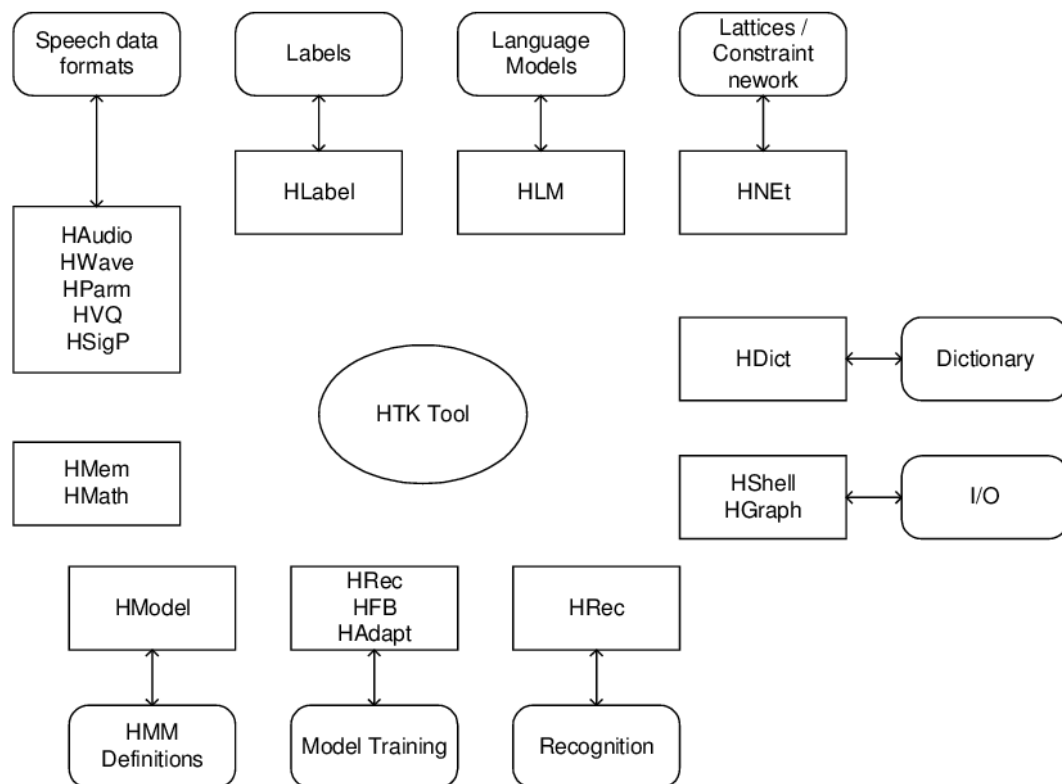


Slika 2.1 *Primjer sučelja prilikom korištenja Audacity aplikacije*

2.2 HTK

HTK je prenosivi skup alata korišten u drugoj fazi projekta radi izrade, korištenja i rada sa skrivenim Markovljevim modelima. Primarna uloga HTK je raspoznavanje govora te se sastoji od velike količine alata različite primjene. Otvorenog je koda, ali postoje limitacije unutar samih alata. HTK može se preuzeti sa <https://htk.eng.cam.ac.uk/download.shtml> nakon uspješne registracije, a od velike koristi je i HTK book koji sadrži detaljne instrukcije o tome kako koristiti alate koji su dostupni korisniku. HTK book moguće je preuzeti sa <https://htk.eng.cam.ac.uk/ftp/software/htkbook-3.5.alpha-1.pdf>.

Poglavlje 2. Alati



Slika 2.2 Pregled svih alata od kojih se HTK sastoji [6]

2.3 Julia

Julia je skriptni jezik visoke razine i performansi, a koristi se za tehničko računanje. Akustični model, o kojem detaljnije u nastavku, izrađen je koristeći skripte napisane u Julia-i. Programskom jeziku pristupa se putem terminala računala. Julia se može preuzeti sa <https://julialang.org/downloads/>.

Poglavlje 3

Prva faza - priprema podataka

Nakon uspješnog preuzimanja i instalacije potrebnih alata započinje implementacija i razvoj projekta. Prva faza razvoja predstavlja pripremu podataka. Svi Speech Recognition Engine (SRE) sastoje se od gramatike, akustičnog modela te dekodera.

Gramatika predstavlja datoteku koja sadrži predefinirane setove riječi koje nastojimo raspoznati tj. predstavlja ograničenja što SRE može raspoznati. Svaka riječ gramatike ima listu fonema od koje se ta riječ sastoji. Važno je napomenuti da u gramatici možemo isključivo koristiti riječi koje smo prethodno "trenirali" u našem akustičnom modelu stoga se može zaključiti da gramatika i akustični model jako ovise jedno o drugome.

Akustični model je datoteka koja sadrži statističku reprezentaciju svakog zvuka koji čini određenu izgovorenu riječ. Kao što je prethodno navedeno ono mora sadržavati sve riječi koje se nalaze u gramatici. SRE sluša i čeka niz zvukova koji čine riječ pohranjenu u gramatici što potvrđuje međuovisnost akustičnog modela i gramatike.

Dekoder je program koji na temelju izgovorenih zvukova traži iste u akustičnom modelu. Kada se pronade par dekodeer određuje foneme koji odgovaraju zvuku te ih pamti sve do prve stanke u govoru. Zatim se traže odgovarajući zvukovi u gramatici i ako se pronade par zvukova programu se vraća riječ ili fraza koja odgovara tom nizu fonema.

Važno je napomenuti da je u sklopu ovog projekta bilo potrebno isključivo proširiti bazu podataka s novim zapisima govora stoga nije bilo potrebno izraditi čitav SRE

Poglavlje 3. Prva faza - priprema podataka

u obliku aplikacije, ali su prethodno navedene komponente sve bile ključne da bi se do tih podataka došlo. U nastavku će biti opisane osnovne faze koje se odvijaju za vrijeme pripreme podataka, a to su:

- Izrada rječnika
- Snimanje podataka
- Izrada transkripcija
- Kodiranje audio podataka

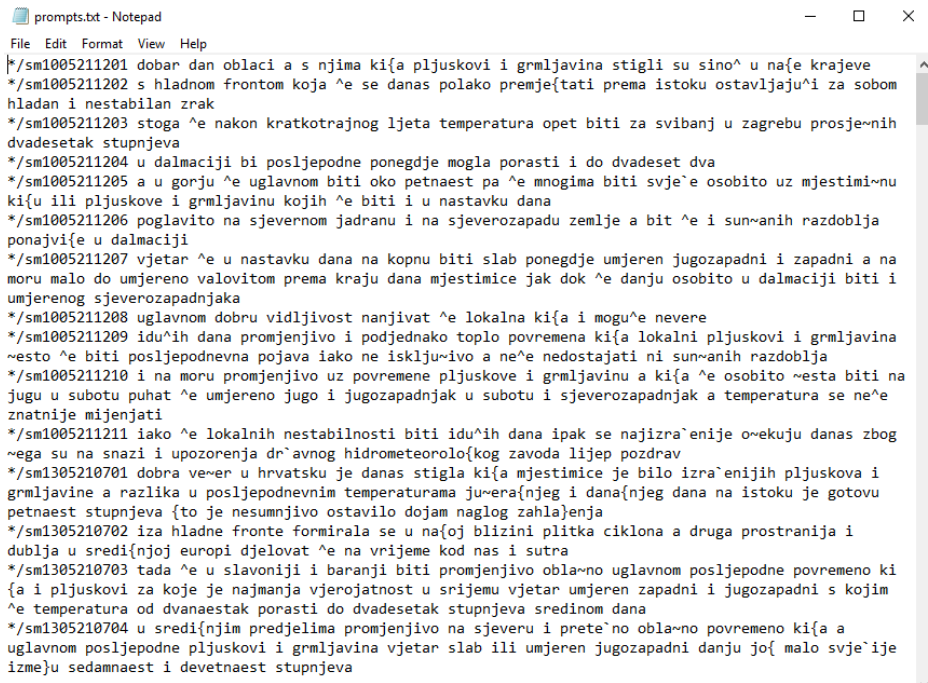
3.1 Izrada rječnika

Za izradu rječnika potrebno je napraviti sortiranu listu onih riječi koje se nalaze u gramatici. Lista se sastoji od jedne riječi po retku te uz riječ sadrži i izgovor riječi. Da bi HTK mogao sastaviti govor i transkripcije u akustični model potrebno je imati fonetski balansirani rječnik s najmanje trideset do četrdeset rečenica. Rječnik je fonetski balansiran ako se svaki fonem gramatike pojavljuje više puta, a ako se pojavljuje samo jednom potrebno je dodati još par uporaba tog fonema u novim rečenicama.

Prvi korak izrade rječnika u sklopu ovog projekta bila je izrada prompts.txt datoteke koja u prvom stupcu sadrži imena audio datoteka koje će biti naknadno snimljene dok se u nastavku nalaze ručno izrađene transkripcije onoga što treba snimiti u obliku audio datoteke. Važno je napomenuti kako kod ručno napisanih transkripcija zamjenjujemo dijakritičke znakove s odgovarajućim zamjenama kako bi ih mogli pravilno raspoznati npr. znak “C” koristi se umjesto “c”, “cc” zamjenjuje “ć” i slično.

Pokrenuvši Julia skriptu prompts2wlist.jl (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/bin/prompts2wlist.jl>) stvara se datoteka koja briše ime audio datoteke iz prvog stupca te sortira riječi prisutne u prompts.txt datoteci, svaku u svoj redak. Sljedeći korak je dodavanje podataka o izgovoru riječi za što se koristi HDMan komanda (HDMan -A -D -T 1 -m -w wlist -n monophones1 -i -l dlog dict ../lexicon/VoxForgeDict.txt) koja je dio HTK alata. Rezultat navedene akcije su dvije datoteke pod nazivom dict i monophones1. Dict

Poglavlje 3. Prva faza - priprema podataka



```
prompts.txt - Notepad
File Edit Format View Help
*/sm1005211201 dobar dan oblaci a s njima ki{a pljuskovi i grmljavina stigli su sino^ u na{e krajeve
*/sm1005211202 s hladnom frontom koja ^e se danas polako premje{tati prema istoku ostavljaju^i za sobom
hladan i nestabilan zrak
*/sm1005211203 stoga ^e nakon kratkotrajnog ljeta temperatura opet biti za svibanj u zagrebu prosje~nih
dvadesetak stupnjeva
*/sm1005211204 u dalmaciji bi posljedodne ponegdje mogla porasti i do dvadeset dva
*/sm1005211205 a u gorju ^e uglavnom biti oko petnaest pa ^e mnogima biti svje^e osobito uz mjestimi~nu
ki{u ili pljuskove i grmljavinu kojih ^e biti i u nastavku dana
*/sm1005211206 poglavito na sjevernom jadraniu i na sjeverozapadu zemlje a bit ^e i sun~anih razdoblja
ponajvi{e u dalmaciji
*/sm1005211207 vjetar ^e u nastavku dana na kopnu biti slab ponegdje umjeren jugozapadni i zapadni a na
moru malo do umjereno valovitom prema kraju dana mjestimice jak dok ^e danju osobito u dalmaciji biti i
umjerenog sjeverozapadnjaka
*/sm1005211208 uglavnom dobru vidljivost nanjivat ^e lokalna ki{a i mogu^e nevere
*/sm1005211209 idu^ih dana promjenjivo i podjednako toplo povremena ki{a lokalni pljuskovi i grmljavina
~esto ^e biti posljedpodnevna pojava iako ne isklju~ivo a ne^e nedostajati ni sun~anih razdoblja
*/sm1005211210 i na moru promjenjivo uz povremene pljuskove i grmljavinu a ki{a ^e osobito ~esta biti na
jugu u subotu puhat ^e umjereno jugo i jugozapadnjak u subotu i sjeverozapadnjak a temperatura se ne^e
znatnije mijenjati
*/sm1005211211 iako ^e lokalnih nestabilnosti biti idu^ih dana ipak se najizra^enije o~ekuju danas zbog
~ega su na snazi i upozorenja dr^avnog hidrometeorolo{kog zavoda lijep pozdrav
*/sm1305210701 dobra ve~er u hrvatsku je danas stigla ki{a mjestimice je bilo izra^enijih pljuskova i
grmljavine a razlika u posljedpodnevnim temperaturama ju~era{njeg i dana{njeg dana na istoku je gotovu
petnaest stupnjeva {to je nesumnjivo ostavilo dojam naglog zahla~enja
*/sm1305210702 iza hladne fronte formirala se u na{ojoj blizini plitka ciklona a druga prostranija i
dublja u sredi{njoj europskoj djelovat ^e na vrijeme kod nas i sutra
*/sm1305210703 tada ^e u slavonskoj i baranjskoj biti promjenjivo obla~no uglavnom posljedpodne povremeno ki
{a i pljuskovi za koje je najmanja vjerojatnost u srijemu vjetar umjeren zapadni i jugozapadni s kojim
^e temperatura od dvanaestak porasti do dvadesetak stupnjeva sredinom dana
*/sm1305210704 u sredi{njim predjelima promjenjivo na sjeveru i prete~no obla~no povremeno ki{a a
uglavnom posljedpodne pljuskovi i grmljavina vjetar slab ili umjeren jugozapadni danju jo{ malo svje~ije
izme~u sedamnaest i devetnaest stupnjeva
```

Slika 3.1 Isječak prompts.txt datoteke korištene u projektu

predstavlja rječnik koji također sadržava i izgovore svih riječi dok je monophones1 datoteka koja sadrži listu svih fonema korištenih u rječniku. S obzirom da nam kasnije treba još jedna lista fonema ona se stvara kopiranjem monophones1 u datoteku monophones0.

3.2 Snimanje podataka

U sklopu ovog projekta audio datoteke snimljene su iz vremenskih prognoza Hrvatske radiotelevizije. Korištene su vremenske prognoze koje su bile emitirane u 12 i 19 sati. Prognoze emitirane u 12 sati sadržavaju prognoze vremena za poslijepodne dok su prognoze emitirane u sklopu “Dnevnika” u 19 sati detaljnije te uz sutrašnje vrijeme govore i o prognozi idućih par dana. Putem web stranica HRTi te HRT vrijeme i promet na kojima su dostupne snimke starih vremenskih prognoza koristeći Audacity audio editor, koji je prethodno spomenut, snimljene su vremenske prognoze te

Poglavlje 3. Prva faza - priprema podataka

izrezane na manje audio datoteke koje se sastoje od nekoliko riječi ili fraza. Snimka vremenske prognoze reže se na manje snimke koje se sastoje od pojedinih rečenica ili fraza na temelju smislenih cjelina govora i kratkih pauza poput uzdaha koje se dešavaju tijekom ljudskog govora. Prilikom snimanja i rezanja datoteka važno je napomenuti da .wav datoteke dobivene tim postupcima moraju biti frekvencije uzorkovanja 16 000 Hz, u mono formatu (jedan kanal) te imenovane sukladno opisu baze VEPRAD. Naziv svake datoteke započinje sa spolom govornika (sm ili sz), datuma vremenske prognoze i njenog redoslijeda unutar tog dana (ddmmggrr) te završava rednim brojem zadanog izraza unutar prognoze. Prilikom rezanja vremenske prognoze na manje audio datoteke važno je pripaziti da razina zvuka ne prelazi gornju granicu od 1 te donju od -1 jer će to generirati iskrivljenje zvuka, a poželjno je pustiti kraću stanku prije i nakon izgovorenih fraza kako bi se dobro čule krajnje riječi. Uz snimanje podataka bilo je potrebno izraditi i ručno napisane transkripcije na temelju navedenih audio datoteka koje su nužne uz .wav i .lab datoteke za raspoznavanje govora, a iste te transkripcije koristile su se u prompts.txt datoteci. Transkripcije su u potpunosti izrađene ručno višestrukim slušanjem pojedinih prognoza te su transkripcije pohranjene u .txt formatu, ali slijede istu nomenklaturu kao i .wav datoteke koje sadržavaju audio zapise.

Tablice prikazane u nastavku sadrže statistiku vezanu uz snimljene podatke, a ono što nije prikazano je da je ukupno snimljeno 1019 jedinstvenih riječi koje se mogu koristiti kod raspoznavanja govora.

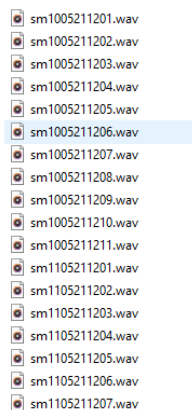
Tablica 3.1 Tablica koja opisuje snimljene podatke

broj prognoza	broj izrezanih snimki	ukupno trajanje snimki
13	151	32 min 31 s

Tablica 3.2 Tablica koja opisuje govornike

broj muških govornika	broj ženskih govornika	ukupno različitih govornika
4	5	9

Poglavlje 3. Prva faza - priprema podataka



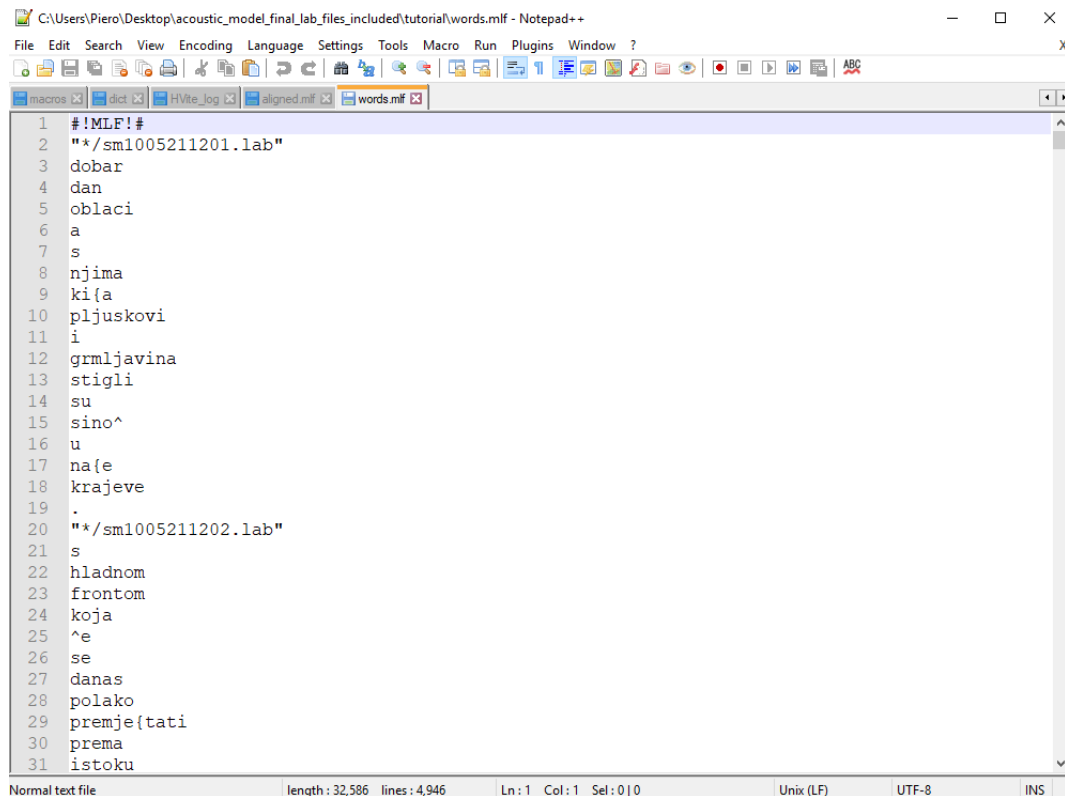
Slika 3.2 *Primjer naziva datoteka*

3.3 Izrada transkripcija

HTK alati ne mogu procesirati prompts.txt datoteku direktno, a za to postoje dva rješenja. Može se ručno izraditi “label” datoteka za svaki red prompts.txt datoteke, ali pristup koji je jednostavniji i ujedno korišten u sklopu ovog projekta je izrada MLF. MLF je datoteka koja se sastoji od label-a tj. oznake za svaki pojedini redak prompts.txt datoteke, a sve je pohranjeno u svega jednoj datoteci. Pokretanjem Julia skripte prompts2mlf (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/bin/prompts2mlf.jl>) generira se MLF datoteka iz prethodno izrađene prompts.txt datoteke.

Sljedeći korak izrada je transkripcija na razini fonema radi veće preciznosti. To se postiže korištenjem HLEd komande (C:>HLEd -A -D -T 1 -l * -d dict -i phones0.mlf mkphones0.led words.mlf). HLEd komanda pregledava svaku riječ naše MLF datoteke te traži foneme koji čine tu riječ unutar rječnika koji je zadan, a rezultat u obliku transkripcija na razini fonema zapisuje se u datoteku phones0.mlf. Osim phones0.mlf potrebno je generirati još jednu datoteku naziva phones1.mlf koja će uz

Poglavlje 3. Prva faza - priprema podataka



```
1 #!MLF!#
2 \"*/sm1005211201.lab\"
3 dobar
4 dan
5 oblaci
6 a
7 s
8 njima
9 ki{a
10 pljuskovi
11 i
12 grmljavina
13 stigli
14 su
15 sino^
16 u
17 na{e
18 krajeve
19 .
20 \"*/sm1005211202.lab\"
21 s
22 hladnom
23 frontom
24 koja
25 ^e
26 se
27 danas
28 polako
29 premje{tati
30 prema
31 istoku
```

Slika 3.3 Sadržaj MLF datoteke

foneme sadržavati i kratku pauzu “sp”. To se postiže korištenjem iste komande, ali se koristi skripta koja uključuje kratku pauzu. S obzirom da se radi o automatskoj transkripciji moguće su greške o kojima nešto više kasnije.

Poglavlje 3. Prva faza - priprema podataka

```
1  #!MLF!#
2  "*/sm1005211201.lab"
3  sil
4  d
5  o
6  b
7  a
8  r
9  sp
10 d
11 a
12 n
13 sp
14 o
15 b
16 l
17 a
18 c
19 i
20 sp
21 a
22 sp
23 s
24 sp
25 N
26 i
27 m
28 a
29 sp
30 k
31 i
```

Slika 3.4 Sadržaj *phones1* datoteke koja uz foneme uključuje i kratku pauzu

3.4 Kodiranje audio podataka

Posljednji korak pripreme podataka naziva se kodiranje audio podataka. HTK je efikasniji u procesiranju internih formata stoga je potrebno obaviti konverziju prethodno snimljenih .wav datoteka u Mel Frequency Cepstral Coefficients (MFCC) zapis. HCopy je alat HTK toolkit-a koji se koristi u tu svrhu (naredba HCopy -A -D -T 1 -C wav.config -S codetrain.scp) . Konverziju je moguće izraditi ručno za svaku datoteku, ali koristeći već postojeću konfiguracijsku datoteku (moguće preuzeti sa https://raw.githubusercontent.com/VoxForge/develop/master/tutorial/wav_config) automatski se obavlja konverzija za sve datoteke. Rezultat izvršavanja HCopy naredbe je niz .mfc datoteka koje odgovaraju datotekama navedenim u codetrain.scp skripti kojom smo prethodno definirali koje je sve datoteke potrebno konvertirati i gdje ih je potrebno pohraniti.

Poglavlje 4

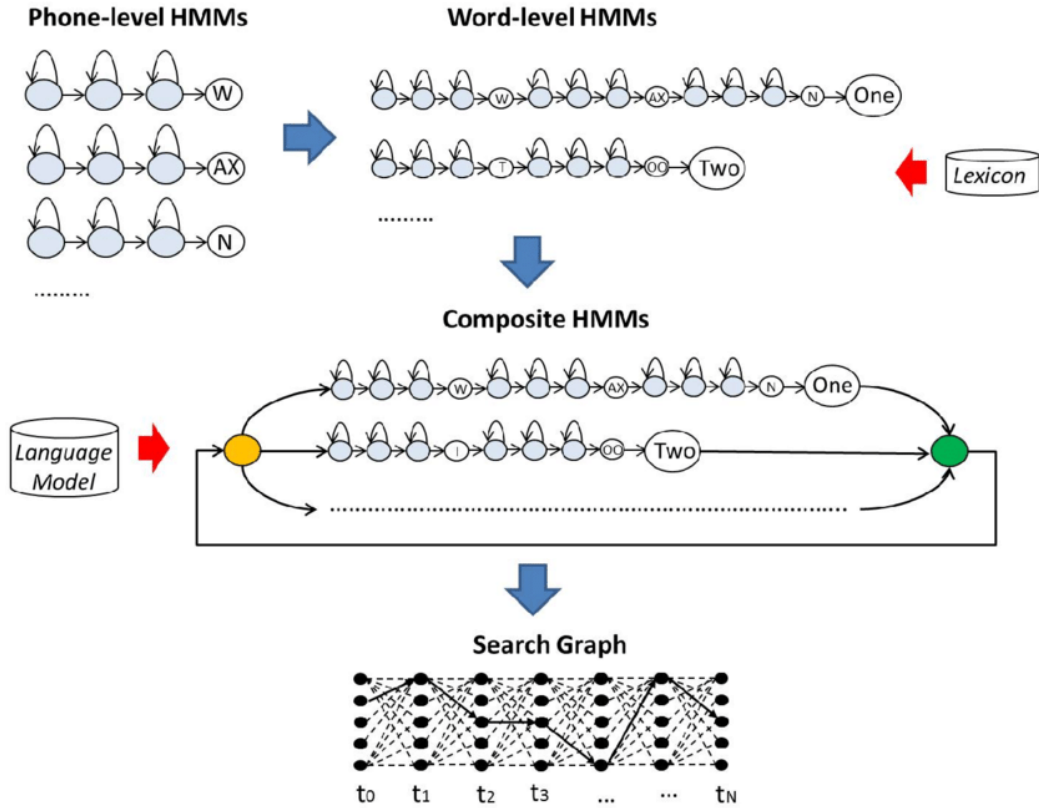
Druga faza - izrada monofonih HMM

Moderni sustavi za raspoznavanje govora temelje se na stohastičkim modelima HMM. HMM su statistički modeli koji modeliraju sekvencu statističkih modela pojedinih glasova. Koriste se u raspoznavanju govora jer se signal govora može promatrati u kratkom vremenskom razdoblju (npr. 10 ms) kao stacionarni signal ili kratkotrajni stacionarni signal te na taj način govor možemo aproksimirati kao stacionarni proces. Još jedan od razloga zbog čega se u raspoznavanju govora koriste HMM je što su vrlo jednostavni za implementaciju te se lako automatizira njihovo “treniranje”. [1]

U nastavku će biti opisane osnovne faze koje se odvijaju za vrijeme izrade monofonih HMM, a to su:

- Izrada “flat start” monofona
- Ispravljanje modela tišine
- Restrukturiranje treniranih podataka

Poglavlje 4. Druga faza - izrada monofonih HMM



Slika 4.1 Primjer uporabe HMM kod raspoznavanja govora [12]

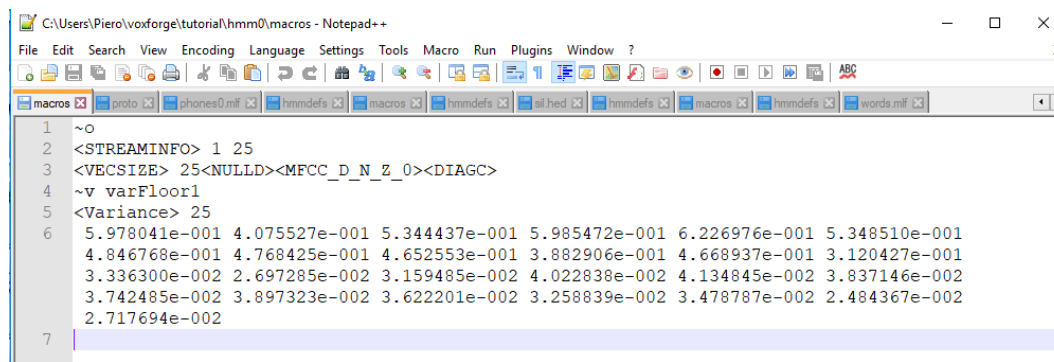
4.1 Izrada “flat start” monofona

Prvi korak kod treniranja naših HMM je definicija modela prototipa pod nazivom “proto” (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/tutorial/hmm0/proto>) te konfiguracijske datoteke “config” (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/tutorial/config>). Uz to potrebno je dati do znanja HTK gdje se nalaze MFCC datoteke izrađene u prošlom koraku što se postiže novom train.scf datotekom. Zatim stvaramo novu mapu pod nazivom hmm0 te izvršavanjem funkcije HCompV (HCompV -A -D -T 1 -C config -f 0.01 -m -S train.scf -M hmm0 proto) koja se koristi za inicijalizaciju parametara HMM dobivamo novu verziju datoteke proto i datoteku vFloors koja sadržava informacije o varijaciji prisutnoj u našim HMM.

Poglavlje 4. Druga faza - izrada monofonih HMM

Sljedeći korak izrada je same hmmdefs datoteke koja sadrži prethodno navedene “flat start” monofone. Za to je potrebno kopirati monophones0 datoteku u tek izrađenu hmm0 mapu te preimenovati tu datoteku u hmmdefs. Za svaki fonem unutar datoteke potrebno je okružiti ga dvostrukim navodnicima, dodati “ h” prije samog fonema i kopirati od 5.linije do kraja sadržaj proto datoteke i dodati ga nakon svakog fonema. Ovime je uspješno napravljena hmmdefs datoteka koja sadrži “flat start” monofone.

Posljednji korak ove faze je izrada “macros” datoteke. U mapi hmm0 stvaramo novu datoteku pod nazivom macros, unutar nje kopiramo sadržaj datoteke vFloors i kopiramo prve 3 linije iz datoteke proto na vrh macros datoteke.



```
1 ~o
2 <STREAMINFO> 1 25
3 <VECSIZE> 25<NULLD><MFCC_D_N_Z_0><DIAGC>
4 ~v varFloor1
5 <Variance> 25
6 5.978041e-001 4.075527e-001 5.344437e-001 5.985472e-001 6.226976e-001 5.348510e-001
  4.846768e-001 4.768425e-001 4.652553e-001 3.882906e-001 4.668937e-001 3.120427e-001
  3.336300e-002 2.697285e-002 3.159485e-002 4.022838e-002 4.134845e-002 3.837146e-002
  3.742485e-002 3.897323e-002 3.622201e-002 3.258839e-002 3.478787e-002 2.484367e-002
  2.717694e-002
7
```

Slika 4.2 *Primjer izgleda datoteke macros*

Prije prelaska na sljedeći korak potrebno je napraviti 9 novih mapa naziva od hmm1 do hmm9. Prethodno generirane “flat start” monofone potrebno je ponovno procijeniti koristeći HERest alat (primjer naredbe HERest -A -D -T 1 -C config -I phones0.mlf -t 250.0 150.0 1000.0 -S train.scp -H hmm0/macros -H hmm0/hmmdefs -M hmm1 monophones0). To je alat koji se koristi za ponovnu procjenu parametara skupa HMM, a u ovom koraku ponovna se procjena vrši za modele iz mape hmm0

te se novi skup modela zapisuje u mapu `hmm1` te se to ponavlja i za `hmm1` i `hmm2` mapu.

4.2 Ispravljanje modela tišine

U posljednjem koraku izrađeni su HMM koji nisu uključivali SP model tišine. SP odnosi se na tip kratke pauze do kojih dolazi između riječi izgovorenih u normalnom govoru. No, u prošlom su koraku izrađeni sil modeli tišine koji se odnose na duže pauze tipično na početku i kraju rečenice. Potrebno je napraviti novi SP model unutar `hmmdefs` koji će koristiti središnje stanje sil modela i zatim ih moramo povezati. Za povezivanje dva modela koristit će se HHED (naredba `HHed -A -D -T 1 -H hmm4/macros -H hmm4/hmmdefs -M hmm5 sil.hed monophones1`) alat i tako im omogućiti korištenje istog središnjeg stanja.

Najprije moramo kopirati sadržaj mape `hmm3` u mapu `hmm4`. Sada koristeći neki od text editora radimo SP model tako da kopiramo sil model i preimenujemo ga u SP, brišemo stanja 2 i 4, mijenjamo vrijednost `<NUMSTATES>` u 3, `<STATE>` u 2, `<TRANSP>` u 3 te u novu matricu dimenzija 3x3 upisujemo potrebne vrijednosti.

Nakon toga pokrećemo alat HHED koji se koristi za “povezivanje” SP stanja sa središnjim stanjem sil modela što znači da će veći broj HMM dijeli iste parametre. Za povezivanje stanja koristi se skripta `sil.hed` (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/tutorial/sil.hed>) te HHED alat. Dobivamo nove verzije (nove procjene) datoteka `hmmdefs` i `macros` u mapi `hmm5`. Posljednji korak kod ispravljanja modela tišine je pokretanje naredbe `HE-Res` dva puta te tako dobivamo nove verzije datoteka `hmmdefs` i `macros` u mapi `hmm6` i `hmm7`.

Poglavlje 4. Druga faza - izrada monofonih HMM

```
844 ~h "sp"
845 <BEGINHMM>
846 <NUMSTATES> 3
847 <STATE> 2
848 <MEAN> 25
849 -5.032526e+000 -2.679660e+000 7.229495e-001 3.664983e+000 4.741467e+000 1.347394e+000
850 3.515197e+000 2.863642e+000 1.281811e+000 -6.980046e-001 2.071212e+000 2.491851e+000
851 1.108491e-001 1.321773e-001 1.055339e-002 -4.292671e-003 -3.799241e-003 9.028691e-002
852 7.624830e-002 -1.203338e-001 -4.442824e-002 6.100859e-002 4.963951e-002 -2.524987e-002
853 -1.273459e-001
854 <VARIANCE> 25
855 1.256283e+001 1.084950e+001 1.554436e+001 1.619645e+001 2.403161e+001 2.303516e+001
856 2.198750e+001 2.741690e+001 2.690071e+001 2.212668e+001 3.109196e+001 1.702169e+001
857 8.000259e-001 6.871839e-001 1.075315e+000 1.055975e+000 1.725897e+000 1.673275e+000
858 1.775787e+000 2.175568e+000 1.977203e+000 1.898801e+000 1.790181e+000 1.492105e+000
859 7.969059e-001
860 <GCONST> 8.578795e+001
861 <TRANSP> 3
862 0.0 1.0 0.0
863 0.0 0.9 0.1
864 0.0 0.0 0.0
865 <ENDHMM>
866
```

normal text file length: 82,509 lines: 858 Ln: 844 Col: 8 Sel: 0|0 Unix (LF) UTF-8 INS

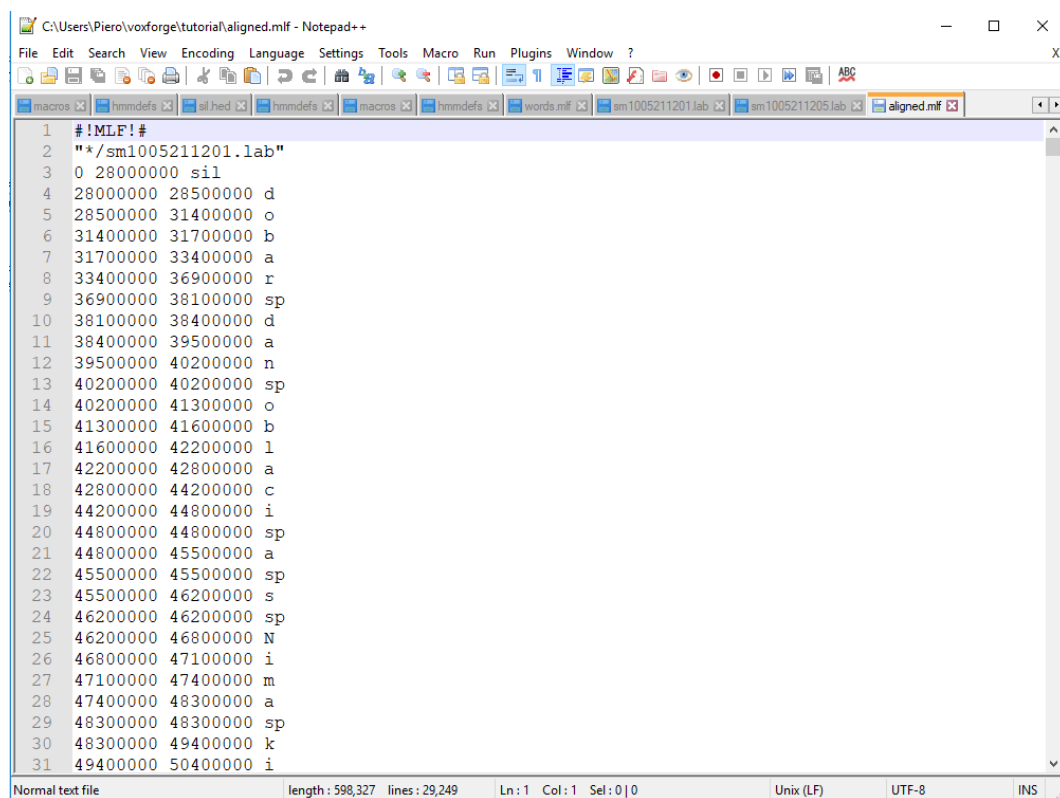
Slika 4.3 Izgled modela kratkih stanki SP

4.3 Restrukturiranje treniranih podataka

Ključna naredba za posljednju fazu izrade monofonih HMM je HVite (primjer naredbe HVite -A -D -T 1 -l * -o SW -b SENT-END -C config -H hmm15/macros -H hmm15/hmmdefs -i aligned.mlf -m -y lab -a -I words.mlf -S train.scp dict monophones1 > HVite.log). HVite uspoređuje datoteku govora s nizom HMM i vraća transkripcije za svaku datoteku. Izvršavanjem naredbe HVite s odgovarajućim argumentima dobiva se datoteka aligned.mlf koja sadrži popis .lab datoteka, njihov sadržaj sa vremenom početka i kraja svakog pojedinog glasa te sil i sp pauzama. Datoteke sa .lab ekstenzijom moguće je generirati ručno ili korištenjem odgovarajućeg argumenta prilikom korištenja HVite naredbe.

Ponovno se dva puta izvršava naredba HERest i tako se u mapama hmm8 i hmm9 dobivaju nove verzije datoteka hmmdefs i macros. Monofoni su dovoljni za precizno raspoznavanje govora, ali se preciznost dodatno može povećati stvaranjem trifona što će biti opisano u trećoj i ujedno posljednjoj fazi.

Poglavlje 4. Druga faza - izrada monofonih HMM



```
1 #!MLF!#
2 "*/sm1005211201.lab"
3 0 28000000 sil
4 28000000 28500000 d
5 28500000 31400000 o
6 31400000 31700000 b
7 31700000 33400000 a
8 33400000 36900000 r
9 36900000 38100000 sp
10 38100000 38400000 d
11 38400000 39500000 a
12 39500000 40200000 n
13 40200000 40200000 sp
14 40200000 41300000 o
15 41300000 41600000 b
16 41600000 42200000 l
17 42200000 42800000 a
18 42800000 44200000 c
19 44200000 44800000 i
20 44800000 44800000 sp
21 44800000 45500000 a
22 45500000 45500000 sp
23 45500000 46200000 s
24 46200000 46200000 sp
25 46200000 46800000 N
26 46800000 47100000 i
27 47100000 47400000 m
28 47400000 48300000 a
29 48300000 48300000 sp
30 48300000 49400000 k
31 49400000 50400000 i
```

Slika 4.4 *Aligned.mlf* datoteka sa vremenskim oznakama trajanja pojedinih glasova

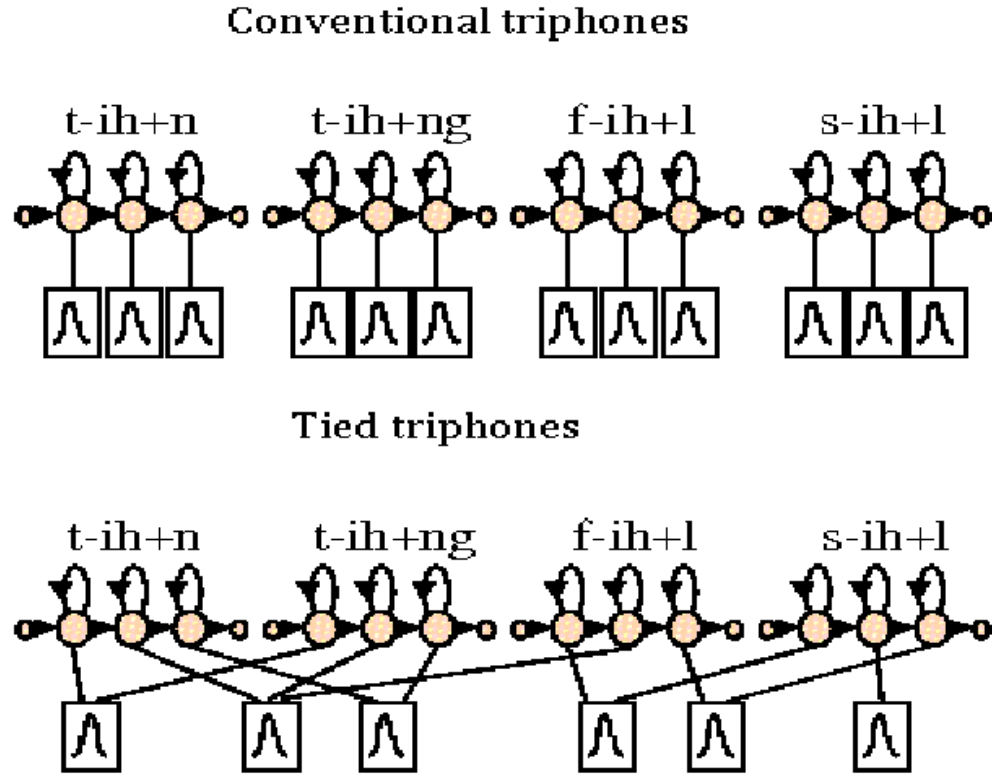
Poglavlje 5

Treća faza - izrada trifona povezanih stanja

Do sada su izgovori riječi bili predstavljeni serijom fonema odnosno monofona. Trifon je naziv za skup od 3 fonema. Da bismo generirali trifon iz monofona potrebno je da lijevi “L” fonem prethodi X fonem, a X fonem se nalazi prije desnog “R” fonema. Trifoni se deklariraju u obliku “L-X+R”. Trifoni se generiraju kako bi se povećala preciznost raspoznavanja jer oni uz foneme uzimaju u obzir i kontekst tog monofona u rečenici. Uporabom trifona smanjuje se vjerojatnost pogreške do koje dolazi zamjenom dva slična zvuka.

Svaki trifon ima vlastitu definiciju HMM, ali česti je slučaj veći broj trifona koji imaju slična stanja stoga je moguće dijeliti te podatke između skupa trifona. Proces dijeljenja podataka naziva se povezivanje (slično povezivanju središnjih stanja iz prošlog koraka). Povezivanje se koristi kako bi veći broj trifona HMM moglo dijeliti iste parametre, a to se provodi zbog bolje procjene novih parametara. U nastavku će biti opisane osnovne faze izrade trifona povezanih stanja, a to su:

- Izrada trifona
- Povezivanje stanja trifona



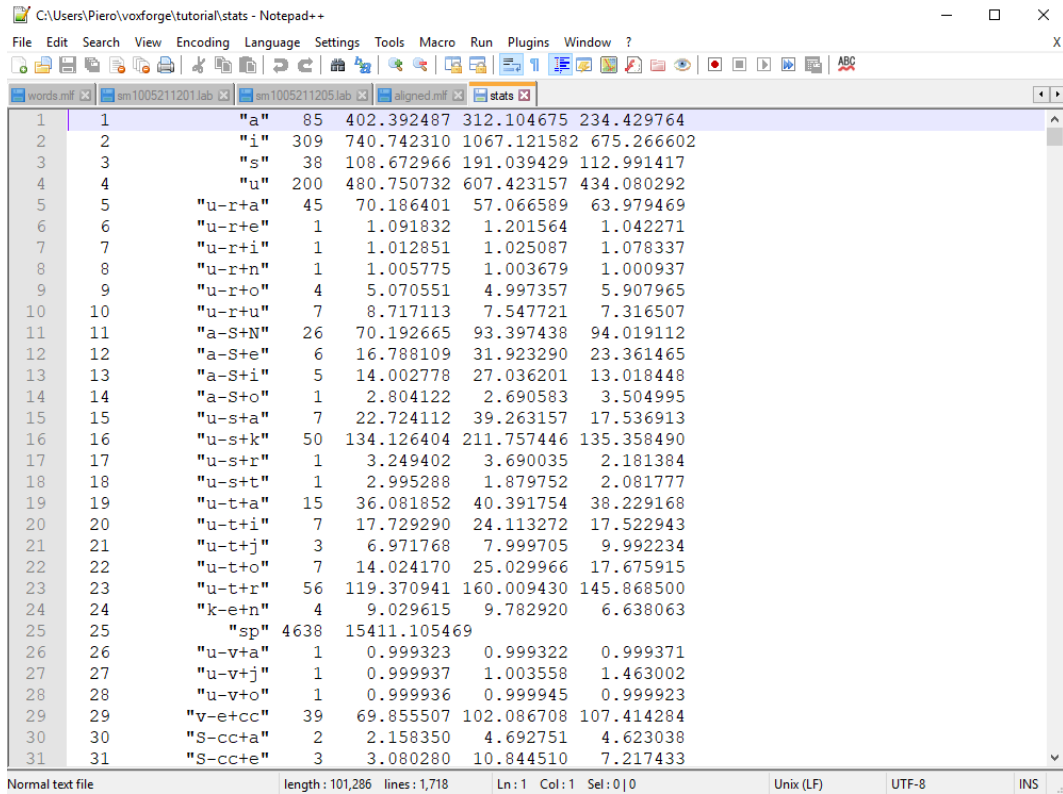
Slika 5.1 *Primjer trifona i trifona povezanih stanja* [15]

5.1 Izrada trifona

Konverzija monofonih transkripcija iz datoteke `aligned.mlf` u ekvivalentni set trifona ostvaruje se koristeći naredbu `HLEd` (primjer naredbe `HLEd -A -D -T 1 -n triphones1 -l * -i wintri.mlf mktri.led aligned.mlf`). `HLEd` je uređivač kojim se upravlja “label” datotekama. Koristeći skriptu `mktri.led` (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/tutorial/mktri.led>) te odgovarajuću `HLEd` naredbu generiraju se dvije datoteke pod nazivom `wintri.mlf` te `triphones1`. `Wintri.mlf` je MLF trifonska datoteka, a `triphones1` sadrži popis svih trifona koje koristimo za treniranje. Sljedeći korak je izvršavanje Julia skripte `mktrihed.jl` (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/>

Poglavlje 5. Treća faza - izrada trifona povezanih stanja

bin/mktrihed.jl) koja generira mktri.hed datoteku. Navedena datoteka sastoji se od komandi “CL” i “TT” koji se koriste za povezivanje stanja između različitih HMM. Potrebno je izraditi tri nove mape hmm10, hmm11 i hmm12 te ponavlja se postupak sličan postupku izrade monofona. Koristeći naredbu HHed povezuju se odabrani HMM, a rezultat te akcije su nove procjene u obliku datoteka hmmdefs i macros. Zatim se dva puta izvršava HERest naredbe s odgovarajućim argumentima za ponovnu procjenu i u mapama hmm11 i hmm12 generiraju se nove hmmdefs i macros datoteke. Važno je napomenuti da kod posljednje HERest naredbe koristimo zastavicu stats koja nam generira stats datoteku potrebnu za izvršenje posljednjeg koraka.



1	1	"a"	85	402.392487	312.104675	234.429764
2	2	"i"	309	740.742310	1067.121582	675.266602
3	3	"s"	38	108.672966	191.039429	112.991417
4	4	"u"	200	480.750732	607.423157	434.080292
5	5	"u-r+a"	45	70.186401	57.066589	63.979469
6	6	"u-r+e"	1	1.091832	1.201564	1.042271
7	7	"u-r+i"	1	1.012851	1.025087	1.078337
8	8	"u-r+n"	1	1.005775	1.003679	1.000937
9	9	"u-r+o"	4	5.070551	4.997357	5.907965
10	10	"u-r+u"	7	8.717113	7.547721	7.316507
11	11	"a-S+N"	26	70.192665	93.397438	94.019112
12	12	"a-S+e"	6	16.788109	31.923290	23.361465
13	13	"a-S+i"	5	14.002778	27.036201	13.018448
14	14	"a-S+o"	1	2.804122	2.690583	3.504995
15	15	"u-s+a"	7	22.724112	39.263157	17.536913
16	16	"u-s+k"	50	134.126404	211.757446	135.358490
17	17	"u-s+r"	1	3.249402	3.690035	2.181384
18	18	"u-s+t"	1	2.995288	1.879752	2.081777
19	19	"u-t+a"	15	36.081852	40.391754	38.229168
20	20	"u-t+i"	7	17.729290	24.113272	17.522943
21	21	"u-t+j"	3	6.971768	7.999705	9.992234
22	22	"u-t+o"	7	14.024170	25.029966	17.675915
23	23	"u-t+r"	56	119.370941	160.009430	145.868500
24	24	"k-e+n"	4	9.029615	9.782920	6.638063
25	25	"sp"	4638	15411.105469		
26	26	"u-v+a"	1	0.999323	0.999322	0.999371
27	27	"u-v+j"	1	0.999937	1.003558	1.463002
28	28	"u-v+o"	1	0.999936	0.999945	0.999923
29	29	"v-e+cc"	39	69.855507	102.086708	107.414284
30	30	"S-cc+a"	2	2.158350	4.692751	4.623038
31	31	"S-cc+e"	3	3.080280	10.844510	7.217433

Slika 5.2 *Primjer sadržaja stats datoteke*

5.2 Povezivanje stanja trifona

Posljednji korak izrade akustičnog modela koji se može pouzdano koristiti za raspoznavanje govora je povezivanje stanja trifona. Jedan od osnovnih problema prethodno izrađenog akustičnog modela koji se bazira na trifonima je što ne može djelovati nad trifonima koji nisu trenirani. Ovaj problem rješava se stablom odluke. Pomoću fonetskog stabla odluke organiziramo modele u obliku stabla, a parametre koje prosljeđujemo nazivamo pitanjima. Zatim dekodier postavlja pitanje i na temelju konteksta fonema odlučuje koji model koristiti za raspoznavanje govora. Fonetsko stablo odluke je zapravo binarno stablo kod kojeg svaki čvor ima pridruženo “da/ne” fonetsko pitanje. Pitanje za svaki čvor odabire se tako da se maksimizira vjerojatnost treniranih podataka.

Kod povezivanja trifona potrebno je najprije dobiti datoteke dict-tri i fulllist0. Dict-tri je rječnik koji se ovoga puta sastoji od trifona, a fulllist0 je lista svih trifona koji su prisutni u akustičnom modelu. Potrebno je koristiti skriptu maketrip-hones.ded (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/tutorial/maketrip-hones.ded>) te odgovarajuću HDMan (koristi se za stvaranje rječnika izgovora) naredbu da bi dobili te dvije datoteke. Nad dobivenom fulllist0 datotekom izvodimo Julia skriptu fixfulllist.jl (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/bin/fixfulllist.jl>) i time dobivamo fulllist datoteku kojoj su dodani monofoni na početak te izbrisani duplikati.

Sljedeći korak je korištenje HTK skripte tree.hed (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/tutorial/tree1.hed>). Radi se o skripti koja sadrži fonetska kontekstualna pitanja koje će HTK koristiti kako bi odabrao trifone. Uz tree.hed skriptu potrebna je i Julia skripta mkclscript.jl (moguće preuzeti sa <https://raw.githubusercontent.com/VoxForge/develop/master/bin/mkclscript.jl>) koja ažurira tree.hed datoteku.

Da bi dobili konačne HMM ponavlja se sličan postupak kao i ranije. Potrebno je napraviti mape hmm13, hmm14 i hmm15. Izvršavanjem HHed naredbe s odgovarajućim argumentima dobivaju se datoteke hmmdefs, macros i tiedlist u hmm13 mapi. Preostaje još dva puta pokrenuti odgovarajuću HREst naredbu kako bi se

Poglavlje 5. Treća faza - izrada trifona povezanih stanja

generirale ažurirane hmmdefs i macros datoteke u mapama hmm14 i hmm15. Hmmdefs datoteka iz mape hmm15 i tiedlist dovoljni su za precizno raspoznavanje govora i time završava izrada akustičnog modela.

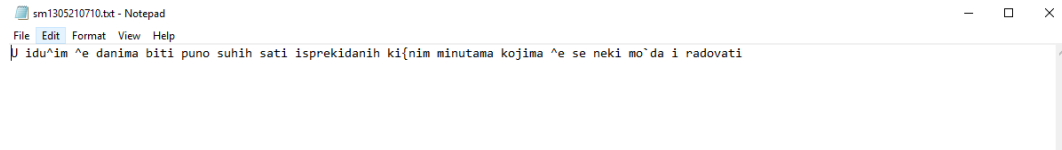
Poglavlje 6

Rezultati

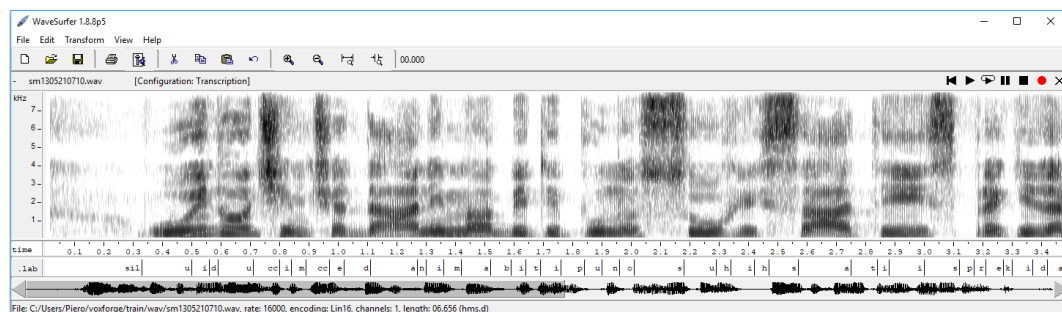
Nakon što su uspješno provedene faze izrade akustičnog modela (priprema podataka, izrada monofonih HMM te izrada trifona povezanih stanja) imamo na raspolaganju tri vrste datoteka. Datoteke sa .txt ekstenzijom predstavljaju ručno izrađene transkripcije koje su izrađene na temelju riječi ili rečenica koje se želi raspoznati. Datoteke sa .wav ekstenzijom predstavljaju audio datoteke snimljene i izrezane na manje datoteke koristeći Audacity audio editor. Te posljednje, datoteke sa .lab ekstenzijom, koje sadrže vremenske oznake početka i kraja pojedinih glasova i foneme s kratkim pauzama SP te dugim pauzama sil. Nakon što su dobivene te tri datoteke moguće je uspješno provoditi raspoznavanje govora treniranih riječi, a tim se datotekama proširuje sadržaj baze govornih snimaka VEPRAD. Osim proširenja baze govornih snimaka moguće je i vrlo jednostavno koristeći Julius SRE izraditi jednostavnu “dialog manager” aplikaciju koja će raspoznavati izgovorene riječi i mogu se implementirati dodatne mogućnosti poput ključnih riječi i slično. U nastavku je prikazana ručno izrađena transkripcija datoteke sm1305210710.wav nakon čega sljede ista .wav datoteka prikazana zajedno sa njenom automatskom transkripcijom.

Kao što je ranije spomenuto postotak preciznosti raspoznavanja govora sve je veći, no i dalje nećemo uvijek dobiti točnu transkripciju. Do grešaka kod transkripcije dolazi zbog loše kvalitete audio zapisa, šuma u pozadini, brzog govora koji otežava razumijevanje, limitirani vokabular i slično. U nastavku je prikazana jedna od grešaka transkripcije gdje je zadnji fonem riječi pogrešno transkribiran.

Poglavlje 6. Rezultati

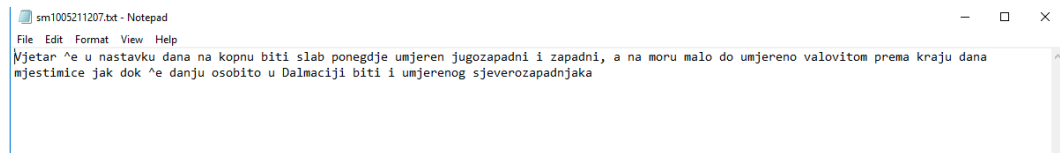


Slika 6.1 Ručno izrađena transkripcija datoteke *sm1305210710.wav*



Slika 6.2 Grafički prikaz audio datoteke *sm1305210710.wav* i pripadajuća transkripcija

Poglavlje 6. Rezultati



Slika 6.3 *Manualna transkripcija sadrži riječ “dalmaciji”*



Slika 6.4 *Automatska transkripcija sadrži riječ “dalmaciju”*

Poglavlje 7

Zaključak

Cilj ovog rada bio je proširenje baze govornih snimaka VEPRAD i on je uspješno odrađen. Raspoznavanje govora je disciplina koja je u svega nekoliko godina jako napredovala i taj trend će se nastavljati u budućnosti. U radu je objašnjeno korištenje Audacity audio editora, prikupljanje i priprema podataka potrebnih za izradu akustičnog modela, što su to i u koje se svrhe koriste monofoni i trifoni i slično. Nakon uspješno obavljenog rada stečeno je puno znanja u vezi alata potrebnih za raspoznavanje govora, HMM te njihove uloge u raspoznavanju govora te se sa malo uloženog truda može nastaviti sa obradom snimaka govora. S obzirom na to da je broj podataka limitiran rezultati nisu optimalni, ali dodavanjem novih snimki može se povećati preciznost raspoznavanja.

Bibliografija

- [1] Definicija raspoznavanje govora preuzeta sa Wikipedia-e, s Interneta, https://en.wikipedia.org/wiki/Speech_recognition, kolovoz 2021.
- [2] IBM, s Interneta, <https://www.ibm.com/cloud/learn/speech-recognition>, kolovoz 2021.
- [3] Povijest razvoja raspoznavanja govora, s Interneta, <https://sonix.ai/history-of-speech-recognition>, kolovoz 2021.
- [4] Audacity - Audio Editing Software, s Interneta, <https://www.audacityteam.org/>, kolovoz 2021.
- [5] Hidden Markov Model Toolkit - building and manipulating Hidden Markov Models, s Interneta, <https://htk.eng.cam.ac.uk/docs/docs.shtml>, kolovoz 2021.
- [6] HTK Toolkit, s Interneta https://www.researchgate.net/figure/HTK-toolkit-overview_fig15_27342930, kolovoz 2021.
- [7] Julia programski jezik, s Interneta, <https://julialang.org/>, kolovoz 2021.
- [8] Open-Source Large Vocabulary CSR Engine Julius, s Interneta, http://julius.osdn.jp/en_index.php, kolovoz 2021.
- [9] HRTi - pristup snimkama vremenskih prognoza, s Interneta, <https://hrti.hrt.hr/home>, kolovoz 2021.
- [10] HRT - Vrijeme, dodatne vremenske prognoze, s Interneta, <https://vrijeme-i-promet.hrt.hr/vrijeme>, kolovoz 2021.
- [11] VoxForge - Create Acoustic Model, s Interneta, <http://www.voxforge.org/home/dev/acousticmodels/windows/create/htkjulius/tutorial>, kolovoz 2021.

Bibliografija

- [12] ResearchGate HMM, s Interneta, https://www.researchgate.net/figure/An-example-of-Hidden-Markov-Models-HMMs-for-speech-recognition_fig7_321902421, kolovoz 2021.
- [13] Introduction to HTK toolkit, s Interneta, <https://homepage.iis.sinica.edu.tw/~whm/course/Speech-NTUT-2004S/slides/HTKToolkit.pdf>, kolovoz 2021.
- [14] HTKBook for HTK3, s Interneta, <http://www.seas.ucla.edu/spapl/weichu/htkbook/>, kolovoz 2021.
- [15] ResearchGate Triphones, s Interneta, https://www.researchgate.net/figure/State-typing-of-triphones_fig4_271583694, kolovoz 2021.
- [16] Problems With Automated Transcription, s Interneta, <https://speakai.co/problems-with-automated-transcription/>, kolovoz 2021.
- [17] WaveSurfer, s Interneta, <https://sourceforge.net/projects/wavesurfer/>, kolovoz 2021.

Pojmovnik

HMM Skriveni Markovljevi Modeli. viii, 2, 15–19, 21, 23, 24, 26, 29, 33

HTK Hidden Markov Model Toolkit. viii, 5, 6, 9, 12, 14, 16, 24

LVCSR large vocabulary continous speech recognition. 7

MFCC Mel Frequency Cepstral Coefficients. 14, 16

MLF Master Label File. viii, 12, 13, 22

NLP Natural language Processing. 2

SP short pause. viii, 18, 19, 26

SRE Speech Recognition Engine. 8, 26, 33

VEPRAD VrEmenske Prognoze-RADio. 1, 11, 26, 29

Sažetak

Ovim radom nastoji se na što jednostavniji način opisati što je to zapravo raspoznavanje govora i u kakve se svrhe koristi te povijest razvoja same discipline. Opisuju se ključni alati korišteni za raspoznavanje govora ili izradu vlastitog SRE, a ukratko je opisan čitav postupak izrade akustičnog modela koji objedinjuje prikupljanje i pripremu podataka, izradu monofonih HMM te izradu trifona povezanih stanja kako bi se povećala preciznost i efikasnost raspoznavanja. Pomoću datoteka dobivenih kao rezultat rada moguće je raspoznavati trenirane riječi, a i vrlo lako implementirati razne aplikacije koje se temelje na istome poput “dialog manager-a”.

Ključne riječi — raspoznavanje govora, SRE, HMM, trifon

Abstract

This thesis tries to explain what speech recognition is, its uses and a brief history of its development. A description of the key tools used for speech recognition or the creation of your own speech recognition engine SRE is given alongside a brief description of the entire process used to build an acoustic model which includes the gathering and preparation of data, generating monophone HMM and generating tied-state triphones to increase recognition accuracy and efficiency. Using the data we’ve gathered as a result of the process we can recognise the “trained” words or phrases but also easily implement different applications based on speech recognition such as a simple dialog manager.

Keywords — speech recognition, speech recognition engine, HMM, triphone