Sveučilište u Rijeci
Tehnički fakultet
Diplomski studij računarstva

Učenje akustičkih modela govora za raspoznavanje pomoću alata HTK/Julius

Izvješće

Kolegij: Računalna obrada govora i jezika
Studenti: Luka Šćulac, Jan Šubarić
Ak. god.: 2023./2024.

Rijeka, lipanj 2024.

Sadržaj

1 Uvod

U ovom radu opisujemo izradu projekta usmjerenog na učenje akustičkih modela za raspoznavanje govora pomoću alata HTK i Julius. Glavni cilj projekta bio je razviti akustičke modele koji mogu učinkovito pretvoriti zvučne zapise u tekstualni oblik koristeći skrivene Markovljeve modele (HMM). Takvi modeli imaju široku primjenu, od virtualnih asistenata do aplikacija za diktiranje. Kroz ovaj projekt koristili smo bazu snimljenog i transkribiranog govora kako bismo trenirali akustičke modele i testirali njihov rad.

Problem na kojem smo radili uključuje pretvaranje zvučnih zapisa u tekstualni oblik, što zahtijeva učinkovitu vezu između akustičkih značajki govora i fonetičkih jedinica. Ključna tehnologija za postizanje ovog cilja su skriveni Markovljevi modeli (HMM), koji su osnova za mnoge sustave za prepoznavanje govora. Cilj projekta bio je naučiti akustičke modele govora alatom HTK i testirati njihov rad koristeći programski alat Julius.

2 Metodologija

2.1 Baza podataka

Baza podataka VEPRAD [1] predstavlja opsežnu zbirku govornih hrvatskih vremenskih prognoza prikupljenih s nacionalnih radijskih emitiranja. Ova baza obuhvaća 644 prognoze, što čini ukupno 18 sati i 28 minuta snimljenog sadržaja. Svaka prognoza je uzorkovana sa 16 KHz i pohranjena u 16-bitnom PCM formatu, osiguravajući visoku kvalitetu zvuka.

Početni tekstovi prognoza prikupljeni su s web stranice Hrvatskog meteorološkog instituta, što odražava stvarne vremenske informacije u realnom vremenu. Proces transkripcije uključuje pažljivo slušanje govornih segmenata dok se ne identificiraju prirodni prekidi. Ovaj proces uključuje ne samo snimanje izgovorenih riječi, već i slučajnih zvukova poput tišine, uzdaha, šuškanja papira, kašljanja i ponovnog pokretanja.

Transkripcija se provodi u dva faze: prvo uključujući 11 muških i 12 ženskih govornika za čitanje skripti s označenim riječima, a potom profesionalni mete-

orolozi daju spontane izvještaje koji se također transkribiraju kako bi se obogatila baza podataka.

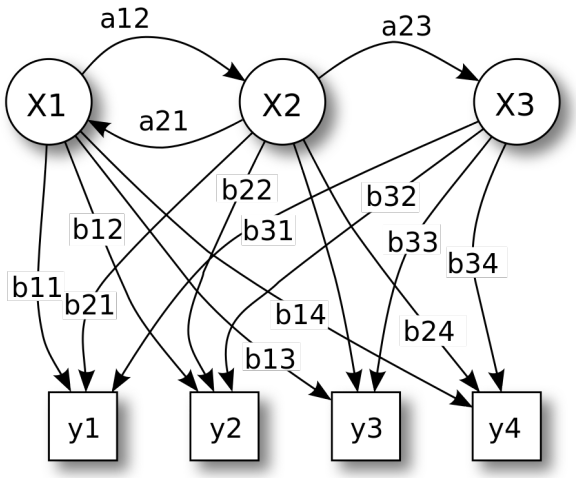
2.2 Akustički modeli

[2]Akustički modeli su ključni elementi u tehnologijama obrade govora, uključujući prepoznavanje govora i sintezu govora. Njihova osnovna svrha je mapiranje akustičnih značajki govornog signala na vjerojatnosti odgovarajućih fonetskih jedinica kao što su fonemi ili njihovi skupovi. Koriste se u sustavima za prepoznavanje govora kako bi se razumjelo što govornik izgovara na temelju njegovog akustičkog signala. Glavni izazov u prepoznavanju govora je različitost u izgovoru iste riječi od strane različitih govornika i varijacije u izgovoru istog govornika pod različitim uvjetima kao što su buka i šum.

Osnovni tip akustičkog modela je skriveni Markovljev model (HMM), koji je probabilistički model koji opisuje sekvencu događaja koja se događa u vremenu. Za prepoznavanje govora, HMM se koristi za modeliranje akustičnih svojstava govornog signala i mapiranje tih svojstava na fonetske jedinice

2.3 Skriveni Markovljevi modeli

[3]Skriveni Markovljevi modeli (HMM) su ključni elementi u tehnologijama obrade govora, uključujući prepoznavanje govora i sintezu govora. Osnovna svrha HMM-a je mapiranje akustičkih značajki govornog signala na vjerojatnosti odgovarajućih fonetskih jedinica, kao što su fonemi ili njihovi skupovi. HMM se koristi u sustavima za prepoznavanje govora kako bi se razumjelo što govornik izgovara na temelju njegovog akustičkog signala. Glavni izazov u prepoznavanju govora je raznolikost u izgovoru iste riječi od strane različitih govornika i varijacije u izgovoru istog govornika pod različitim uvjetima, kao što su buka i šum. HMM je probabilistički model koji opisuje sekvencu događaja koja se događa u vremenu, te se koristi za modeliranje akustičkih svojstava govornog signala i njihovo mapiranje na fonetske jedinice.



Slika 1: Schema skrivenog Markovljevog modela

Na slici 1, stanja (prikazana kao krugovi) predstavljaju fonetske jedinice, dok prijelazi (prikazani kao strelice) predstavljaju vjerojatnosti prijelaza između tih stanja. Skriveni Markovljevi modeli omogućuju precizno prepoznavanje govora unatoč varijabilnostima u akustičkim signalima.

2.4 Jezični modeli

Glavna funkcija jezičnih modela je modeliranje jezične strukture i vjerojatnosti različitih nizova riječi. Ovi modeli omogućuju računalima da razumiju i generiraju prirodni jezik na način sličan ljudskom razumijevanju.

Koriste se za:

- Automatsko prepoznavanje govora
- Strojno prevođenje
- Generiranje teksta
- Korekcija teksta

Jezični modeli često koriste tehnike iz područja statistike i strojnog učenja kako bi naučili vjerojatnosti pojavljivanja različitih riječi ili nizova riječi. Najčešće korištene tehnike uključuju n-gram modele, rekurentne neuronske mreže (RNN), LSTM (Long Short-Term Memory) mreže, transformerske mreže i slično.

3 Treniranje akustičkih modela

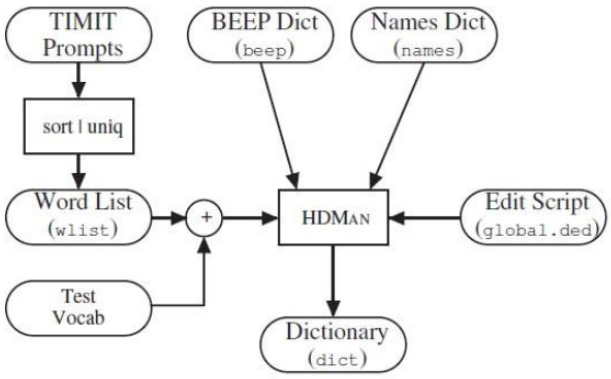
3.1 Priprema podataka

U ovom koraku, cilj je bio stvoriti fonetski balansiran rječnik za prepoznavanje govora pomoću HTK alata. Proces uključuje nekoliko ključnih koraka i stvara specifične datoteke potrebne za izgradnju akustičnog modela.

Jednostavnom python skriptom stvorena je datoteka *prompts.txt* koja sadrži sve transkripcije koje smo dobili uz našu bazu podataka. Datoteka sadrži računalnu adresu svake transkripcijske datoteke i transkripciju iz te datoteke. HTK koristi ASCII abecedu pa su svi dijakritički znakovi hrvatskog jezika zamijenjeni odgovarajućim zamjenskim znakovima poput: { , \, } , # , ^ i drugima. Također se pojavljuje oznaka <sil> koja označava kratku pauzu.

Korištenjem skripte *prompts2wlist.jl*, iz *prompts.txt* datoteke generirana je *wlist* datoteka. Ova datoteka sadrži sortirani popis jedinstvenih riječi koje se pojavljuju u *prompts.txt*. Zajedno sa popisom svih riječi skripta također automatski dodaje i nove oznake *SENT-END* i *SENT-START* koje će biti oznake za početak i kraj rečenica.

```
HDMan -A -D -T 1 -m -w wlist -n monophones1 -i -l
   dlog dict lexicon
```



Slika 2: [4]Prikaz HDMan naredbe

Korištenjem HTK naredbe HDMan2, *wlist* datoteka je prošla kroz leksikon, koji je zapravo fonetski rječnik također dobiven uz bazu podataka, kako bi se svakoj riječi dodao izgovor tj. fonetska raščlamba. Rezultat je *dict* datoteka, koja sadrži rječnik izgovora, i *monophones1* datoteka koja sadrži popis fonema korištenih u rječniku. *monophones1* datoteka zatim je kopirana u *monophones0* datoteku gdje je kratka pauza 'sp' uklonjena kako bi imali opciju treniranja sa i bez oznake za tišinu.

HTK alat ne može direktno obraditi datoteku *prompts.txt* pa zato moramo stvoriti Master Label File (MLF) koji sadrži sve unose. Za generiranje MLF datoteke koristimo skriptu *prompts2mlf.jl*. Ona stvara *words.mlf* datoteku.

Sljedeće, pokrećemo HLEd za proširenje naših transkripcija sa razine riječi na fonetsku razinu transkripcija, zamjenjujući svaku riječ njenim fonemima. Izlaz je nova datoteka *phones0.mlf* (bez kratkih pauza 'sp').

```
HLEd -A -D -T 1 -l * -d dict -i phones0.mlf
   mkphones0.led words.mlf
```

HLEd naredbu pokrenuti ćemo još jedanput da dobijemo datoteku *phones1.mlf* (sa kratkim pauzama 'sp').

```
HLEd -A -D -T 1 -l * -d dict -i phones1.mlf
   mkphones1.led words.mlf
```

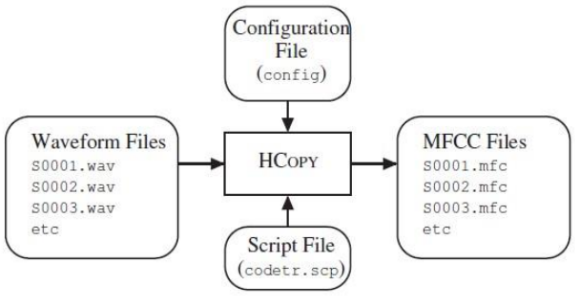
HTK nije tako učinkovit u obradi *wav* datoteka kao što je u njegovom internom formatu. Iz tog razloga bilo je potrebno pretvoriti sve audio datoteke u drugi format nazvan *MFCC* (Mel Frequency Cepstral Coefficients; općenito poznati kao 'vektori značajki').

```
HCopy -A -D -T 1 -C wav_config -S codetrain.scp
```

Naredbu HCopy koristit čemo za pretvorbu iz *wav* formata u *MFCC*. Ova naredba prima konfiguracijsku datoteku1 koja specificira sve potrebne parametre pretvorbe te *codetrain.scp* koji povezuje putanje svake *wav* datoteke sa svakom budućom *MFCC* datotekom.

Listing 1: Konfiguracijska datoteka

```
SOURCEFORMAT = WAV
TARGETKIND = MFCC_0_D
TARGETRATE = 100000.0
SAVECOMPRESSED = T
SAVEWITHCRC = T
WINDOWSIZE = 250000.0
USEHAMMING = T
PREEMCOEF = 0.97
NUMCHANS = 26
CEPLIFTER = 22
NUMCEPS = 12
```



Slika 3: [4]Prikaz HCopy naredbe

3.2 Monofonski modeli

Treniranje akustičkog modela krećemo sa monofonskim modelima a kasnije prelazimo na trifonske. Kreirali smo direktorije od *hmm0* do *hmm15* jer ćemo imati 15 iteracija treniranja skrivenih Markovljevih modela.

Prije treniranja potrebno je ispravno popuniti direktorij *hmm0* inicijalnim datotekama. Prva naredba koju koristimo je HCompV.

```
HCompV -A -D -T 1 -C config -f 0.01 -m -S train.scp
   -M hmm0 proto
```

Ova naredba prima novu konfiguracijsku datoteku2 i kreira dvije potrebne datoteke *vFloors* i *proto*.

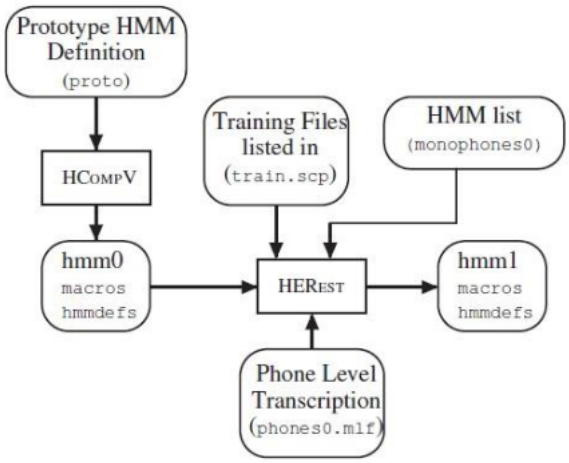
Listing 2: Konfiguracijska datoteka

```
TARGETKIND = MFCC_0_D_N_Z
TARGETRATE = 100000.0
SAVECOMPRESSED = T
SAVEWITHCRC = T
WINDOWSIZE = 250000.0
USEHAMMING = T
PREEMCOEF = 0.97
NUMCHANS = 26
CEPLIFTER = 22
NUMCEPS = 12
```

Nakon toga ručno kreiramo *hmmdefs* datoteku gdje definiramo inicijalne vrijednosti HMM strukture modela za svaki fonem posebno te za oznake tišine, početka i kraja rečenice, zvuka uzdaha papira i drugo. Isto tako kreiramo *macros* koja sadrži zajedničke definicije i makroe koji se koriste za sve modele HMM-a tijekom treniranja.

Popunjavanjem *hmm0* sa inicijalnim datotekama možemo pokrenuti proces treniranja skrivenih Markovljevih modela u iteracijama. Svaku iteraciju pozivamo naredbom HRest4.

```
 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
```



Slika 4: [4]Prikaz HRest naredbe

HRest naredba u procesu treniranja skrivenih Markovljevih modela (HMM) koristi se za reestimaciju parametara HMM-a na temelju novih podataka. U svakoj iteraciji koristi se za prilagodbu HMM modela, čime se postupno poboljšava točnost prepoznavanja fonema u zvučnim podacima tijekom treninga. Izlaz svake iteracije su nove datoteke *hmmdefs* i *macros* koji se pohranjuju u sljedeću *hmm* datoteku. U našem slučaju HRest pokrećemo 15 puta.

Nakon što smo izvršili 3 iteracije treniranja pokretanjem prethodno navedene naredbe uvodimo oznaku tišine 'sp' u model. Sada kod treniranja umjesto *phones0.mlf* koritimo *phones1.mlf* i tako pokrećemo još dvije iteracije treniranja. Također moramo naredbom HHed povezati našu oznaku tišine <sil> sa fonemom 'sp'. Nakon toga pokrečemo još dvije iteracije treniranja HRest naredbom.

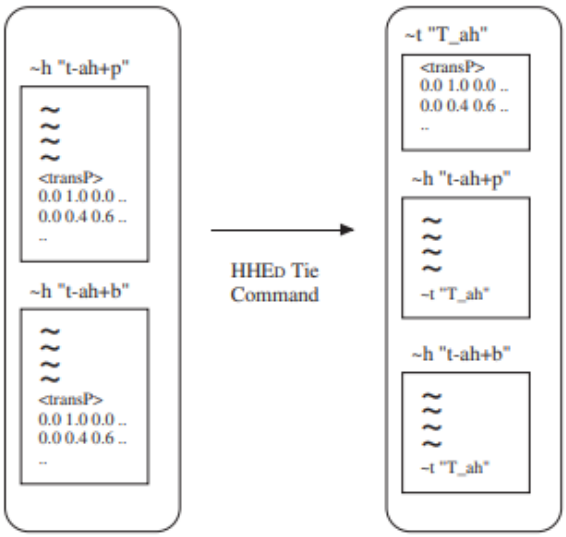
U sljedećem koraku, HVite naredba koristi se za ponovno poravnanje akustičkih podataka s pripadajućim transkripcijama. Ovo omogućava da se pronađe najbolja izgovorna varijanta za svaku riječ na temelju akustičkih podataka.

```
HVite -A -D -T 1 -l * -o SWT -b SENT-END -C config -H
   hmm7/macros -H hmm7/hmmdefs -i aligned.mlf -m -t
   250.0 150.0 1000.0 -y lab -a -I words.mlf -S
   train.scp dict monophones1> HVite_log
```

Ovim postupkom stvaramo *aligned.mlf* datoteku koju od sada također uključujemo u HRest naredbu i tako pokrećemo još 2 puta. Ovim korakom završavamo treniranje monofonskih modela i imamo 9 iteracija treniranja skrivenih Markovljevih modela.

3.3 Trifonski modeli

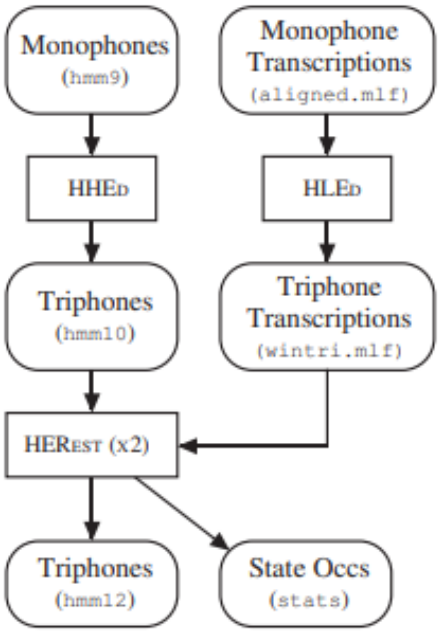
Nakon što je kroz devet iteracija treniranja dobiven monofonski akustički model, počinjmo razvoj trifonskog akustičkog modela. S obzirom da se trifonski modeli oslanjaju na kontekst fonema, dodjeljujući različite značajke monofonu ovisno o okruženju, trifonski modeli postaju značajno robusniji i točniji u raspoznavanju govora sa smanjenim pogreškama za razliku od prethodnih monofonskih modela. Trifoni se definiraju kao L-X+R forma, gdje L označava fonem koji okružuje zadani fonem s lijeve strane, X je odabrani fonem čije se značajke određuju, dok R označava desni fonem koji okružuje X. Kod trifonskih modela dolazi do dijeljenja stanja trifona jer su određene instance trifonskih stanja dovoljno slične pa se podaci mogu dijeliti između različitih grupa trifona. Prethodno opisani proces nazivamo povezivanje stanja (eng. Tying states) različitih trifona koji dijele zajedničke parametre.



Slika 5: Povezivanje stanja prijelaznih matrica [4]

Prvi korak stvaranja trifonskih modela uključuje konverziju monofonskih transkripcija zapisanih u *aligned.mlf* datoteci. Konverzija transkripcija u trifonski oblik se vrši pomoću naredbe HLEd i unaprijed definirane skripte *mktri.led*. Ovim korakom nastaju dvije nove datoteke, od kojih jedna sadrži popis svih trifona,

a druga transkripcije u trifonskom obliku. Kako bi se povezali različiti HMM modeli, na način da dijele isti skup parametara, stvorena je unaprijed definirana skripta *mktri.he*d koja sadrži naredbe za kloniranje 'CL' te naredbu za povezivanje stanja 'TI'. Koroštena je naredba HHEd koja se koristi za povezivanje modela preko određenih HMM parametara te je dva puta pokrenuta HERest naredba za podešavanje i procjenu novih modela.



Slika 6: Stvaranje trifonskih modela [4]

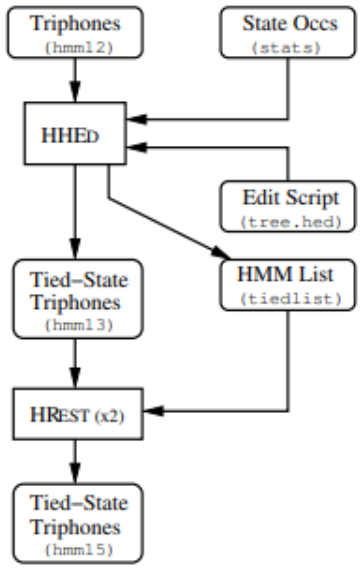
Prilikom stvaranja trifonskih modela, u pozadini se koriste binarna stabla odluke za odabir ispravnog modela fonema. U fonetskom stablu odluke, modeli čine stablo, a pitanja za odlučivanje u stablu čine značajke. Odlučivanje se vrši postavljanjem da/ne fonetskih pitanja o kontekstu fonema nakon čega se odabire pripadajući model. Stabla su definirana komandom 'TB', dok su sva moguća fonetska pitanja definirana komandom 'QS'. Korištenjem unaprijed definirane skripte i naredbom HDMan nad cijelim fonetskim rječnikom, stvorene su dvije datoteke koje sadrže izgovor trifonskih grupa i sve moguće kombinacije trifona.

```
TR 0

QS "L_Front"              { "p-*","b-*","f-*","m-*","v-*","e-*","i-*" }
QS "L_Central"            { "t-*","d-*","s-*","z-*","S-*","Z-*","c-*","cc-*","C-*","dz-*","n-*","N-*","l-*","L-*","r-*","j-*","a-*" }
QS "L_Back"               { "k-*","g-*","h-*","o-*","u-*" }
QS "D_Front"              { "*+p","*+b","*+f","*+m","*+v","*+e","*+i" }
QS "D_Central"            { "*+t","*+d","*+s","*+z","*+S","*+Z","*+c","*+cc","*+C","*+dz","*+n","*+N","*+l","*+L","*+r","*+j","*+a" }
QS "D_Back"               { "*+k","*+g","*+h","*+o","*+u" }
QS "L_Vowel"              { "i-*","e-*","a-*","o-*","u-*" }
QS "L_Vowel_Unstressed"   { "i-*","e-*","a-*","o-*","u-*" }
QS "L_Vowel_Front"        { "e-*","i-*" }
QS "L_Vowel_Central"      { "a-*" }
QS "L_Vowel_Back"         { "o-*","u-*" }
QS "L_Vowel_High"         { "i-*","u-*" }
QS "L_Vowel_Medium"       { "o-*","e-*" }
QS "L_Vowel_Low"          { "a-*" }
QS "D_Vowel"              { "*+i","*+e","*+a","*+o","*+u" }
QS "D_Vowel_Unstressed"   { "*+i","*+e","*+a","*+o","*+u" }
QS "D_Vowel_Front"        { "*+e","*+i" }
```

Slika 7: Dio fonetskih pravila za grupiranje trifona u hrvatskom jeziku - **tree.hed**

Posljednji korak u stvaranju trifonskih modela uključuje kreiranje skripte *tree.hed*. *Tree.he*d skripta sadrži fonetska pravila za grupiranje trifona na hrvatskom jeziku koja se kasnije koriste za stvaranje fonetskih stabla pretrage i definiranje mogućih prijelaza između različitih trifona. Koristi se naredba HHEd koja kreira *tiedlist* datoteku koja sadrži sve moguće kombinacije fonema sukladno fonetskim pravilima hrvatskog jezika. Treniranje je završeno pokretanjem naredbe HERest koja automatski podešava težine i prijelazna stanja dobivenih modela. Posljednje istrenirani model se koristi u alatu Julius za raspoznavanje govora.



Slika 8: Stvaranje trifonskih modela 2 [4]

4 Stvaranje jezičnog modela

U ovom poglavlju će biti opisan proces stvaranja jezičnog modela za potrebe raspoznavanja govora u alatu Julius. Kao što je prethodno objašnjeno, jezični model predstavlja reprezentaciju samog jezika tako što vraća vjerojatnosti pojavljivanja određenih kombinacija riječi unutar rečenice. Za potrebe rada alata Julius, kreirati će se bigram i obrnuti trigram jezični model. Jezični modeli su stvoreni prateći opsiane korake za stvaranje jezičnih modela u HTK knjizi. [4]

4.1 Priprema podataka

Kao i prilikom procesa treniranja akustičkih modela, prvi korak u postupku stvaranja n-gram jezičnih modela zahtjeva prilagodbu i procesuiranje podataka. Ovaj korak uključuje kreiranje skripte koja će sve postojeće transkripcije spojiti u jednu tekstualnu datoteku te dodijeliti oznake za početak <s> i kraj </s> rečenice.

```
<s> biometeorolo{ka prognoza za sutra <sil> uvjeti ^e biti razmjerno povoljni </s>
<s> manje tegobe mogu osje^ati samo vrlo osjetljivi ljudi i kroni~ni bolesnici </s>
<s> vremenska prognoza za poslijepodne na jadranu prete`ito a u unutra{njosti djelomice sun~ano </s>
<s> potkraj dana sve obla~nije a u unutra{njosti mogu^ slab snijeg </s>
<s> vjetar slab do umjeren sjeverni i sjeverozapadni </s>
<s> na sjevernom jadranu slaba i umjerena u dalmaciji i jaka bura </s>
<s> temperatura izme}u jedan i {est na jadranu od sedam do jedanaest stupnjeva </s>
<s> biometeorolo{ka prognoza za sutra </s>
<s> utjecaj vremena u ve^em dijelu zemlje bit ^e razmjerno nepovoljan </s>
<s> zato bi osjetljivije osobe mogle biti slabije <raspolo`ene> te imati smetnje u koncentraciji </s>
<s> tegobe mogu osje^ati i kroni~ni bolesnici </s>
<s> biometeorolo{ka prognoza za sutra utjecaj vremena u ve^em dijelu zemlje bit ^e razmjerno nepovoljan <uzdah> </s>
<s> zato bi osjetljivije osobe mogle biti slabije raspolo`ene <uzdah> </s>
```

Slika 9: Transkripcije za stvaranje jezičnog modela

Korak u pripremi podataka za stvaranje obrnutog trigramskog modela uključuje skriptu koja će obrnuti redoslijed postojećih pojedinačnih transkripcija. Primjerice, rečenicu " <s> Lijep je dan </s>" potrebno je obrnuti u "</s> dan je Lijep <s>". [5] Nakon zamjene redoslijeda riječi, proces pripreme podataka je završen te se može započeti kreiranje jezičnog modela s obrnutim trigramom.

```
</s> poslijepodne osobito to i snijega malo s ponegdje obla~no prete`ito unutra{njosti u sutra za prognoza vremenska <s>
</s> uvjeti biometeorolo{ki nepovoljni prevladavati ^e unutra{njosti u fronte pribli`avanja zbog sutra za prognoza biometeorolo{ka <s>
</s> <uzdah> stupnjeva trideset i pet dvadeset izme}u temperatura najvi{a slab uglavnom vjetar <s>
</s> <uzdah> danas nego vru^e manje malo i podru~ju kopnenom u a sun~ano prete`ito <uzdah> tjedna ovog kraja do vrijeme <s>
</s> osam trideset minus osam tri minus karlovac kupa {est sedamdeset {est sedam radenci kupa <s>
</s> jugo jadranu na a jugozapadnjak umjeren ve^inom ^e puhat <s>
</s> <uzdah> grmljavinom s pljusak ili ki{a kratkotrajna mjestimice naoblaku promjenljivu uz sun~ano djelomice <uzdah> poslijepodne za prognoza vremenska <s>
</s> bura olujna i mjestimice dijelu sjevernom na a jaka jadranu na <uzdah> sjeveroisto~njak i sjevernjak umjeren no^i tijekom <s>
</s> <uzdah> ~etiri dvadeset do devetnaest od jadranu na sedamnaest i dvanaest izme}u temperatura najni`a <s>
</s> {est dva {est dvadeset karlovac kupa jedan devet jedan devedeset radenci kupa <s>
</s> <uzdah> kilometara dvadeset do deset vidljivost <uzdah> ~etiri do tri otvorenome na tri do dva more ~vorova pet dvadeset do petnaest jugo <s>
```

Slika 10: Obrnute transkripcije za stvaranje obrnutog trigrama

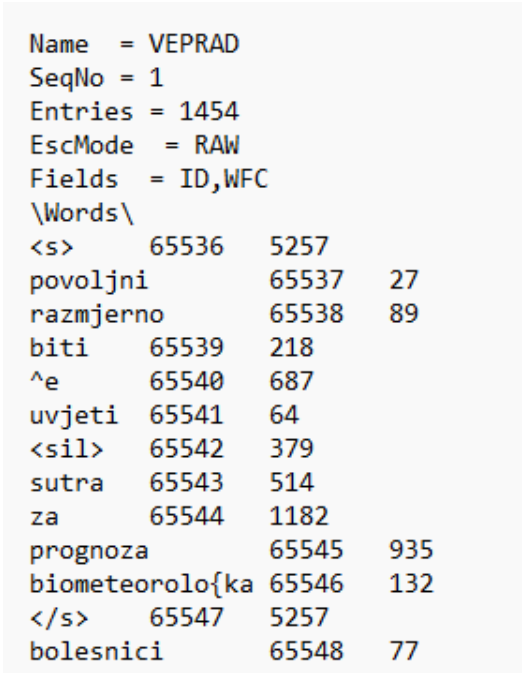
Nastavak u pripremi podataka uključuje kreiranje prazne prototipske datoteke *empty.wmap* koja sadrži samo zaglavlje s određenim parametrima. Zaglavlje datoteke je prikazano u nastavku:

```
Name    = Holmes
SeqNo   = 0
Entries = 0
EscMode = RAW
Fields  = ID,WFC
\Words\
```

Neki od važnijih parametara zaglavlja uključuju naziv, SeqNo argument koji predstavlja sekvencijalni broj i povećava se prilikom svakog sljedećeg dodavanja novih riječi u mapu riječi. Također, WFC parametar označava opciju praćenja frekvencije pojavljivanja pojedinih riječi. Prazna prototipska datoteka se kreira pomoću naredbe:

```
LNewMap -f WFC Holmes empty.wmap
```

Nakon definiranja početne datoteke sa zadanim parametrima, potrebno je obraditi zadane i unaprijed pripremljene transkripcije. Obrada postojeći transkripcija se vrši naredbom LGPrep pri čemu se komandi navodi put do datoteke s deskripcijama, određuje se maksimalna veličina n-grama i navodi se prototipska datoteka. Kao rezultat navedene naredbe, nastaje datoteka gram.0 koja sadrži monograme, bigrame i trigrame jer je za veličinu n-grama odabran broj tri te datoteka *wmap* koja sadrži riječi transkripcija.



Slika 11: Datoteka *wmap* nakon pokretanja naredbe LGPrep

Sadržaj datoteke *gram.0* se može provjeriti korištenjem alata LGList. Datoteka sadrži dobivene n-grame zajedno s njihovom frekvencijom pojavljivanja u transkripcijama. U nastavku je prikazana prethodno opisana naredba LGPrep za stvaranje *gram.0* datoteke:

```
LGPrep -T 1 -a 100000 -b 200000 -d holmes.0 -n 3 -s
   "VEPRAD" empty.wmap prompts.txt
```

Posljednji korak u pripremi podataka za stvaranje jezičnih modela uključuje korištenje alata LGCopy. Navedena naredba će kreirati završnu *n-gram* datoteku zajedno s mapom riječi povezanu u datoteku *data.0* koja se zatim koristi u daljnjem koraku za stvaranje jezičnog modela.

4.2 Proces kreiranja i prilagodbe n-gram jezičnih modela

Nakon što su podaci pripremljeni za daljnju obradu, jezični modeli se kreiraju koristeći *data.0* datoteku. Dodatno, prije samog kreiranja jezičnih modela može se provesti opcionalni korak mapiranja ne-vokabularnih riječi (eng. OOV Mapping), odnosno ograničavanja broja riječi u rječniku. Proces stvaranja modela je

jednostavan i provodi se koristeći naredbu LBuild. Naredba LBuild se može koristi prilikom zasebnog stvaranja modela unigrama, bigrama i trigrama. Unaprijed definirani format nastalih modela je ARPA format, ali s obzirom da je nastale modele potrebno pretvoriti u binarni format za alat Julius, argument '-f TEXT' u naredbi LBuild označava izlazni format u tekstualnom obliku. U nastavku je primjer naredbe za stvaranje bigramskog jezičnog modela koristeći LBuild:

```
LBuild -T 1 -c 2 1 -n 2 -l lm/ug holmes.0/wmap lm/bg1
   holmes.1/data.0
```

Obrnuti trigramski jezični model je stvoren ponavljanjem istog procesa, ali nakon zasebne pripreme podataka opisane u prethodnom poglavlju.
Stvaranjem potrebnih jezičnih modela, bigrama i obrnutog trigrama, posljednji korak uključuje pretvorbu modela u binarni format pogodan za robusnije i brže korištenje s alatom Julius. Pretvorba i povezivanje jezičnih modela u binarni format se vrši koristeći naredbu mkbingram. [6] U nastavku je prikazana navedena naredba koja je korištena za stvaranje završnog jezičnog modela u binarnom formatu za upotrebu s alatom Julius:

```
mkbingram -nlr bg1 -nrl reversed_trigram
   output.bingram
```

5 Rezultati

U ovom poglavlju će detaljno biti prikazani rezultati raspoznavanja govora pomoću istreniranih akustičkih i jezičnih modela. Rezultati će se prikazati pomoću određenih metrika te će se zasebno provjeriti preciznost i robusnost monofonskog kao i trifonskog akustičkog modela. Za testiranje akustičkog modela korišten je alat Julius uz pomoć kojeg se vršila analiza rezultata raspoznavanja govora na zvučnim zapisima snimljenim uživo. Za testiranje je bio potreban mikrofon i konfiguracijska datoteka potrebna alatu Julius za rad. Detaljna statistička analiza i evaluacija akustičkog i jezičnog modela uključivala je izgovor različitih rečenica vezanih uz vremensku prognozu te korištenje zvučnih zapisa namijenjenih za fazu testiranja.

5.1 Konfiguracija i pokretanje alata Julius

U nastavku će biti opisana konfiguracijska datoteka potrebna za pokretanje alata Julius te sam proces testiranja akustičkih modela i raspoznavanja govora.

```
-input mic          # microphone input
-v julius_dict      # pronunciation dictionary
-mapunk <sil>
-h hmm15/hmmdefs    # acoustic HMM
-hlist tiedlist     # HMMList to map logical phone to
   physical
-d language_model.bingram
-smpFreq 16000      # sampling rate (Hz)
-spmodel "sil"      # name of a short-pause silence
   model
-b 4000
-lmp 12 -6
-lmp2 12 -6
-fallback1pass
-multipath
-iwsp
-iwcd1 max
-no_ccd
-sepnum 150
-b2 360
-n 40
-s 2000
-m 8000
-lookuprange 5
-sb 80
-forcedict
-lv 100             # level threshold (0-32767)
-logfile julius.log
-quiet
```

U konfiguracijskoj datoteci iznad navedene su postavke potrebne za uspješno pokretanje alata Julius za uživo prepoznavanje govora, a bitnije linije su i dodatno obilježene komentarom. [7]

Sadržaj audio datoteke:

```
jutarnja temperatura izmedu deset i petnaest
   stupnjeva na jadranu od sesnaest do dvadeset a
   najvisa dnevna od dvadeset do dvadeset pet
   stupnjeva
```

Izlaz alata Julius:

```
pass1_best: <s> jutarnja temperatura izme}u <greska>
   stupnjeva na jadranu od {est do <greska> a najvi{a
   dnevna od dvadeset do dvadeset pet stupnjeva </s>
sentence1: <s> jutarnja temperatura izme}u deset i
   petnaest stupnjeva na jadranu od {esnaest do
   dvadeset a najvi{a dnevna od dvadeset do dvadeset
   pet stupnjeva </s>
```

Testiranjem akustičkog modela uživo, izgovarajući različite rečenice vezane uz vremensku prognozu na mikrofon, može se zaključiti da je model zaista dobar u prepoznavanju rečenica čak i prilikom susreta s raznim glasovima različitog naglaska i jačine. Pojavljuju se dvije varijacije na izlazu dobivenih rezultata raspoznavanja:

- **pass1best**: Ovo predstavlja najbolju hipotezu dobivenu nakon prve faze pretraživanja. Ova faza obično koristi manje resursa i radi bržu, ali potencijalno manje preciznu analizu govornog signala.
- **sentence1**: Ovo je rezultat dobiven nakon druge faze pretraživanja, gdje se koristi preciznija i detaljnija analiza kako bi se poboljšala točnost prepoznavanja. Ova faza koristi više resursa i često je preciznija od prve faze.

Osim što je model testiran pomoću mikrofona na ulazu, dodatno je korišten testni skup zvučnih zapisa za testiranje modela. Kako bi se dobili rezultati testiranja modela na zvučnim zapisima, potrebno je podesiti nekoliko konfiguracijskih parametara unutar datoteke *sample.jconf*. Umjesto ulaza koji je ranije bio postavljen na mikrofon, odabrana je datoteka kao ulazni medij za testiranje. Također, dodan je parametar za ispis rezultata testiranja u zasebne datoteke. [7] Na slikama u nastavku su prikazani rezultati testiranja modela na unaprijed definiranom zvučnom zapisu. Prva slika se odnosi na rezultate dobivene korištenjem monofonskog akustičkog modela, hmm9 bez tiedlist datoteke.

```
1  sentence1: <s> vremenska prognoza za poslijepodne na jadranu prete`ito a u unutra{njosti djelomice
   sun~ano </s>
2  wseq1: <s> vremenska prognoza za poslijepodne na jadranu prete`ito a u unutra{njosti djelomice sun~ano
   </s>
3  phseq1: sil | v r e m e n s k a | p r o g n o z a | z a | p o s l i j e p o d n e | n a | j a d r a n u
   | p r e t e Z i t o | a | u | u n u t r a S N o s t i | d j e l o m i c e | s u n C a n o | sil
4  cmscore1: 0.908 0.998 0.991 0.790 0.498 0.998 0.964 0.983 0.368 0.431 1.000 0.999 0.316 1.000
5  score1: -14706.159180 (AM: -14488.592773  LM: -217.565979)
```

Slika 12: Rezultati raspoznavanja govora - **monofonski akustički model hmm9**

U ispisu rezultata raspoznavanja zvučnog zapisa možemo vidjeti nekoliko redaka. Oznaka 'sentence1' se odnosi na procjenu najbolje odgovarajuće rečenice iz jezičnog modela. Oznake 'phseq1' i 'wseq1' označavaju raspoznate sekvence akustičkog modela u obliku fonema i riječi, dok se oznaka 'cmscore1' odnosi na pouzdanost prepoznavanja pojedinih riječi. Što je vrijednost 'cmscore1' veća, model je sigurniji prilikom točnog prepoznavanja određene riječi. Na primjeru je vidljivo da je pouzdanost modela za većinu riječi iznad 90%, dok je za riječi "uvjeti" i "povoljni" model najnesigurniji, dok ukupna točnost prepoznavanja (score1) uključuje sumu rezultata akustičkog modela (AM) i jezičnog modela (LM).

```
1  sentence1: <s> vremenska prognoza za poslijepodne na jadranu prete`ito a u unutra{njosti djelomice
   sun~ano </s>
2  wseq1: <s> vremenska prognoza za poslijepodne na jadranu prete`ito a u unutra{njosti djelomice sun~ano
   </s>
3  phseq1: sil | v r e m e n s k a | p r o g n o z a | z a | p o s l i j e p o d n e | n a | j a d r a n u
   | p r e t e Z i t o | a | u | u n u t r a S N o s t i | d j e l o m i c e | s u n C a n o | sil
4  cmscore1: 0.951 0.999 0.996 0.696 0.668 0.999 0.981 0.989 0.499 0.306 1.000 0.999 0.410 1.000
5  score1: -13890.614258 (AM: -13673.047852  LM: -217.565979)
```

Slika 13: Rezultati raspoznavanja govora - **trifonski akustički model hmm15**

Razlike u rezultatima testiranja monofonskog i trifonskog akustičkog modela uključuju nešto veću preciznost i sigurnost trifonskog modela prilikom prepoznavanja određenih riječi u testnom primjeru.

5.2 Evaluacija akustičkog modela

Nakon što je završni akustički model testiran koristeći alat Julius s mikrofon u stvarnom vremenu, dodatno su provedene evaluacijske metrike za točnost i preciznost raspoznavanja govora monofonskog i trifonskog akustičkog modela. Korištene su metrike WER (engl. Word Error Rate) i SER (engl. Sentence Error Rate) koje daju sveobuhvatan pregled ukupnih grešaka u prepoznavanju govora te na taj način određuju preciznost samih akustičkih modela. WER je najčešće korištena

metrika za procjenjivanje točnosti prepoznavanja govora, a odnosi se na postotak grešaka na razini riječi. WER se računa koristeći formulu u nastavku:

$$WER = \frac{S + D + I}{N}$$

gdje je

- **S (engl. Substitutions)**: broj zamjena riječi, kada je jedna riječ zamijenjena drugom.
- **D (engl. Deletions)**: broj izostavljenih riječi, kada je riječ iz referentnog teksta izostavljena u stvorenoj transkripciji.
- **I (engl. Insertions)**: broj umetnutih riječi, kada je dodana riječ koja nije u referentnom tekstu.
- **N**: ukupan broj riječi u referentnom tekstu.

S druge strane, SER je metrika koja se odnosi na postotak rečenica koje su prepoznate s greškom. SER metrika se računa pomoću formule u nastavku:

$$SER = \frac{N_s}{N_t}$$

gdje je

- **Ns**: broj rečenica koje sadrže barem jednu grešku.
- **Nt**: ukupan broj rečenica.

Korištenjem ovih metrika mogu se identificirati slabosti sustava i poboljšati nedostaci akustičkih modela, čime se povećava ukupna točnost i robusnost sustava za raspoznavanje govora. U nastavku su prikazani rezultati WER i SER metrika monofonskog i trifonskog akustičkog modela na određenom testnom skupu podataka. Testiranje i dobivanje prethodno navedenih metrika se može provesti automatski korištenjem HTK alata HResult ili ručno pisanjem vlastite skripte za računanje točnosti modela.

Evaluacija akustičkog modela izvedena je pisanjem Python skripte gdje su se uspoređivale izlazne vrijednosti akustičkog modela na testnim zvučnim zapisima i njihove odgovarajuće transkripcije.

```
Average WER: 0.06
Average SER: 0.52
Average Similarity: 76.31%
```

Prosjek WER-a od 0.06 ukazuje na vrlo nisku razinu grešaka zamjene, brisanja i umetanja riječi u odnosu na referentne transkripcije, što sugerira visoku preciznost modela. Međutim, prosječni SER od 0.52 pokazuje da postoji prostor za poboljšanja u točnosti prepoznavanja cijelih rečenica. Također, prosječna sličnost od 76.31% ističe koliko su hipoteze modela slične referentnim rečenicama, što je ključno za realne primjene u različitim scenarijima prepoznavanja govora. Sve datoteke i rezultati projekta su dostupni na Github poveznici na repozitorij projekta. [8]

6 Zaključak

U ovom je radu proveden razvojni proces akustičkih modela za prepoznavanje govora pomoću alata HTK i Julius. Cilj istraživanja bio je stvoriti modele koji bi mogli uspješno prepoznati i pretvoriti govorne signale na hrvatskom jeziku u tekst koristeći skrivene Markovljeve modele.

Obrađena je baza podataka VEPRAD, koja uključuje meteorološke prognoze na hrvatskom jeziku. Nakon toga, korišten je HTK za obradu audio zapisa kako bi ih se pretvorilo u format pogodan za analizu, izdvajajući kepstralne značajke i stvarajući akustičke modele.

Početak treniranja obuhvatio je korištenje monofonskih modela, gdje je primjenjivana iterativna metoda HMM modeliranja kako bi se unaprijedila točnost prepoznavanja fonema. Kasnije su ovi modeli razvijeni u trifonske oblike, što je omogućilo bolje raspoznavanje i razumijevanje fonema pomoću konteksta unutar rečenica.

Konačno, uspješno su prikazani rezultati koji pokazuju koliko su skriveni Markovljevi modeli važni u obradi govornih signala te kako se primjenjuju u stvarnim okruženjima kao što su sustavi za raspoznavanje govora. Mogući daljnji razvoj obuhvaća optimizaciju parametara modela, proširenje baze podataka i primjenu naprednijih tehnika strojnog učenja za još preciznije rezultate.

Ovaj rad stvara osnovu za daljnja istraživanja i primjene u području obrade govora, potičući napredak tehnologija koje unapređuju komunikaciju između ljudi i računala putem govornog sučelja.

Literatura

[1] S. Martinčić-Ipšić and I. Ipšić, "Veprad: A croatian speech database of weather forecasts," *25th International Conference on Information Technology Interfaces, ITI 2003. proceedings.*, 2003. [Online]. Available: https://www.croris.hr/crosbi/publikacija/prilog-skup/496523

[2] Avinash. (2023) Acoustic model. [Online]. Available: https://medium.com/@avinashmachinelearninginfo/acoustic-model-14a1c8939497

[3] D. Kralj, "Jezično modeliranje u sustavima za raspoznavanje govora," Diplomski rad, Sveučilište u Rijeci, Tehnički fakultet, Rijeka, 2021. [Online]. Available: https://urn.nsk.hr/urn:nbn:hr:190:555478

[4] S. Young, G. Evermann, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.5)*. Microsoft Corporation and Cambridge University Engineering Department, 2015, all Rights Reserved. First published December 1995.

[5] VoxForge. (2006) How to create reverse 3 gram for julius? [Online]. Available: https://www.voxforge.org/home/forums/message-boards/general-discussion/how-to-create-reverse-3-gram-for-julius3/5

[6] L. Akinobu. (2024) Julius: Open-source large vocabulary continuous speech recognition engine. [Online]. Available: https://github.com/julius-speech/julius

[7] A. LEE, *The Julius Book 4.1.5*. Julius project team, Nagoya Institute of Technology, 2010.

[8] Github repozitorij projekta: Učenje akustičkih modela govora za raspoznavanje pomoću alata htk/julius. [Online]. Available: https://github.com/lukasculac/Learning-Acoustic-Models-for-Speech-Recognition-Using-HTK-Julius-Tools