Umjetne neuronske mreže u prepoznavanju govora

Kresnik, Iva

SVEUČILIŠTE U ZAGREBU

FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Iva Kresnik

Zagreb, 2019.

SVEUČILIŠTE U ZAGREBU

FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Mentor: Student:

Prof. dr. sc. Dubravko Majetić Iva Kresnik

Zagreb, 2019.

Izjavljujem da sam ovaj rad u cijelosti napisala sama koristeći se dosad stečenim znanjem i navedenom literaturom.

Zahvaljujem se mentoru dr.sc. Dubravku Majetiću na suradnji prilikom pisanja ovog rada.

Također, hvala mami na svemu.

Iva Kresnik

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za završne ispite studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala i mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa:	
Ur.broj:	

ZAVRŠNI ZADATAK

Student: **Iva Kresnik** Mat. br.: 0035204121

Naslov rada na hrvatskom jeziku: **Umjetne neuronske mreže u prepoznavanju govora**

Naslov rada na engleskom jeziku: **Neural networks in speech recognition**

Opis zadatka:

Umjetne neuronske mreže često se koriste u rješavanju problema konverzije govora u tekst i obrnuto, te prepoznavanju govora. Prepoznavanje govora posebno je interesantno područje istraživanja u modernoj kognitivnoj robotici.

U radu treba načiniti sljedeće:

1. Opisati problem prepoznavanja odnosno govora.
2. Pokazati stanje u istraživačkom području primjene umjetnih neuronskih mreža u prepoznavanju govora.
3. Analizirati i usporediti algoritme umjetnih neuronskih mreža koji se koriste u metodama prepoznavanja govora.
4. Izvesti zaključke rada.

Zadatak zadan:
29. studenog 2018.

Rok predaje rada:
1. rok: 22. veljače 2019.
2. rok (izvanredni): 28. lipnja 2019.
3. rok: 20. rujna 2019.

Predviđeni datumi obrane:
1. rok: 25.2. - 1.3. 2019.
2. rok (izvanredni): 2.7. 2019.
3. rok: 23.9. - 27.9. 2019.

Zadatak zadao:

Prof.dr.sc. Dubravko Majetić

Predsjednik Povjerenstva:

Prof. dr. sc. Branko Bauer

SADRŽAJ

POPIS SLIKA

POPIS TABLICA

POPIS OZNAKA

A	matrica tranzicijskih vjerojatnosti
ANN	umjetna neuronska mreža (engl. *Artificial Neural Network*)
ASR	automatsko prepoznavanje govora (engl. *Automatic Speech Recognition*)
B	matrica emitiranih vjerojatnosti
BIAS	neuron bez ulaza s konstantnom izlaznom vrijednosti jednakom 1
DNN	duboke neuronska mreža (engl. *Deep Neural Network*)
E	funkcija cilja
GMM	Gaussov miješani model (engl. *Gaussian Mixed Model*)
HMM	skriveni Markovljevi modeli/lanci (engl. *Hidden Markov Models*)
h	skriveni sloj povratne neuronske mreže
I	broj ulaznih neurona uvećan za jedan
J	broj neurona u skrivenom sloju uvećan za jedan
LSTM	povratna mreža s dugom kratkotrajnom memorijom (engl. *Long Short-Term Memory*)
M	broj mogućih opservacija kod skrivenih Markovljevih modela
MLP	višeslojne unaprijedne neuronske mreže (engl. *Multi-Layer Perceptrons*)
N	broj mogućih stanja kod skrivenih Markovljevih modela
n	broj neurona ulaznog sloja, broj komponenti ulaznog vektora
net	rezultat funkcije sume
O	skup opservacija
O	izlazni podaci neuronske mreže
P	matrica prijelaznih vjerojatnosti
q_t	stanje skrivenog Markovljevog modela u vremenskom trenutku t
R	niz prepoznatih riječi
RTF	faktor stvarnog vremena (engl. *real-time factor*)
RNN	povratna/dinamička neuronska mreža (engl. *Recurrent Neural Network*)
S	skup stanja kod Markovljevih lanaca
T	vremenski skup kod Markovljevih lanaca
V	matrica težinskih faktora skrivenog sloja kod statičke neuronske mreže
V	skup svih opservacija

$\mathbf{W}$	matrica težinskih faktora neuronske mreže
WER	postotak pogrešno prepoznatih riječi (engl. *word error rate*)
w_j	težinski koeficijent j-tog ulaza neurona
$w(n)$	prozorska funkcija
X	skup slučajnih varijabli kod Markovljevih lanaca; niz akustičkih značajki
$X_m(f)$	Fourierova transformacija
$\mathbf{x}$	vektor ulaznih vrijednosti neuronske mreže
y	vrijednost izlaza umjetnog neurona
$\mathbf{y}$	izlazni vektor neuronske mreže
$\mathbf{Z}$	ulazni podaci neuronske mreže
γ	aktivacijska funkcija neurona
δ	odstupanje izračunate izlazne vrijednosti od željene izlazne vrijednosti
η	koeficijent brzine učenja
π	inicijalna raspodjela vjerojatnosti stanja kod skrivenih Markovljevih modela
$\sum$	funkcija sume umjetnog neurona

SAŽETAK

U ovom završnom radu razmatrana je primjena umjetnih neuronskih mreža u prepoznavanju govora. Na početku su definirani govor, jezik i pismo u pogledu lingvistike. Zatim su objašnjene osnove umjetnih neuronskih mreža i skrivenih Markovljevih modela. Prije razrade problema prepoznavanja govora, dan je kratki povijesni pregled razvitka sustava za prepoznavanje govora. Slijedi opis rada konvencionalnog sustava za prepoznavanje govora i moguća područja primjene umjetnih neuronskih mreža unutar njega. Spomenuti su i sustavi koji isključivo ovise o umjetnim neuronskim mrežama, ali koji su još uvijek u razvitku. Na kraju su navedeni aktualni problemi te moguća rješenja u budućnosti.

Ključni pojmovi: govor, umjetne neuronske mreže, sustav za prepoznavanje govora

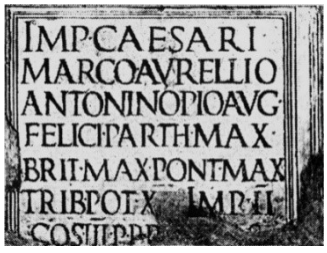
SUMMARY

This final thesis considers the use of artificial neural networks in speech recognition. It starts with the definition of speech, language and script in terms of linguistics. Then the basics of artificial neural networks and hidden Markov models are explained. Before explaining the speech recognition problem, a brief historical overview of the development of the speech recognition systems is given. The following is a description of the operation of a traditional speech recognition system and the possible applications of artificial neural networks within it. Systems that depend solely on artificial neural networks, but are still under development, are mentioned. Finally, current problems and possible solutions in the future are outlined.

Key terms: speech, artificial neural networks, speech recognition system

1. UVOD

Sposobnost govora je umijeće svojstveno ljudskoj vrsti. Mnoga druga živa bića na ovom planetu komuniciraju međusobno proizvodeći određene glasove, no ta glasanja nisu usporediva s kompleksnošću kojom obiluje ljudski govor. Ne može se sa sigurnošću tvrditi kada su točno ljudski preci prešli s običnog životinjskog glasanja na primitivne osnove definiranog jezika. Ono što se sa sigurnošću zna jest da do razvitka kompleksnog govora ne bi došlo bez određenih fizičkih predispozicija svojstvenih ljudima. Prvenstveno se radi o dvije fizičke karakteristike; posebno oblikovane glasnice koje ljudima omogućuju proizvodnju širokog spektra glasova kao i veliku kontrolu nad njima te razvoj mozga o kojem još i danas puno ne znamo. Govor je omogućio stvaranje zajedničkog jezika unutar ljudske zajednice čime je olakšano prenošenje ideja, misli i znanja između članova zajednice. Jezik je omogućio ljudima da lakše stvaraju i barataju s mislima i zamišljenim konceptima te na taj način dao osnovu razmišljanja na kojoj su kasnije izgrađene sve društvene i prirodne znanosti. Ipak, vremenom su ljudi uočili problem nestalnosti govora; ono što je jednom izrečeno zauvijek se gubi u vremenu. Tako se javila ideja pisma koje omogućuje da jednom zapisane riječi ostaju zauvijek iste. Počeci pisma javljaju se već u 4. tisućljeću pr. Kr. na području Mezopotamije [1]. Razvitkom pisma javlja se nova ljudska odlika, pismenost. Pismenost, odnosno sposobnost čitanja i pisanja određenog jezika, zapravo je u suštini prepoznavanje govora. Kada ljudi pišu zapravo pretvaraju vlastite misli u riječi u obliku unutarnjeg monologa ili vanjskog govora i te glasove zapisuju u obliku znakova. Danas se teži tome da je cijelo čovječanstvo pismeno, što znači da je veliki postotak ljudi već u ranoj životnoj dobi sposoban vršiti funkciju pretvaranja barem jednostavnog govora u tekst. Jednom kada nauče, ljudi to čine s lakoćom, no za računalo to nije tako jednostavan problem. Umjetne neuronske mreže omogućile su da i strojevi u određenoj mjeri svladaju umijeće pretvaranja glasova u znakove, odnosno riječi i rečenice.



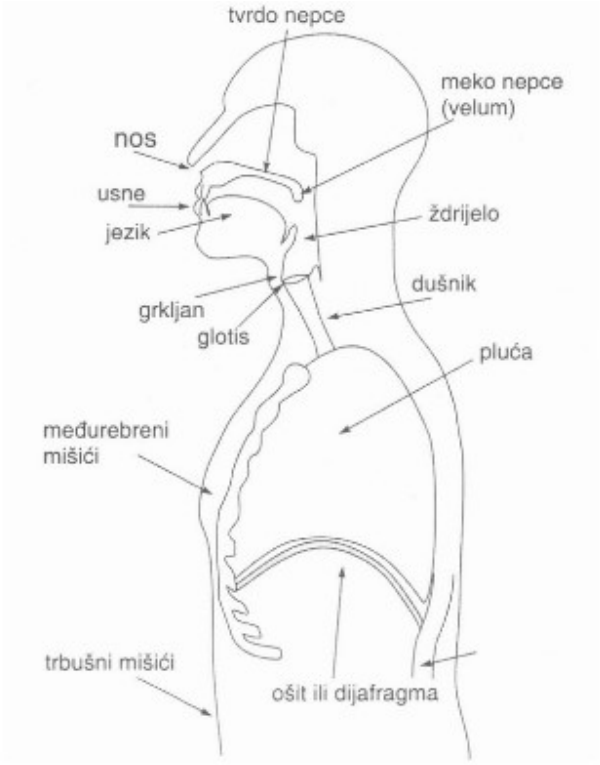
Slika 1.1 Primjer latiničnog pisma na starorimskom spomeniku [1]

2. GOVOR, JEZIK I PISMO

Ljudi prirodno posjeduju sposobnost komuniciranja glasovnim znakovima. Međuljudska komunikacija može se podijeliti na verbalnu i neverbalnu. U sklopu ovog rada naglasak će biti na verbalnoj komunikaciji. Ona se definira kao oblikovanje i prenošenje poruke jezičnim djelatnošću koja se sastoji od govora i jezika [2].

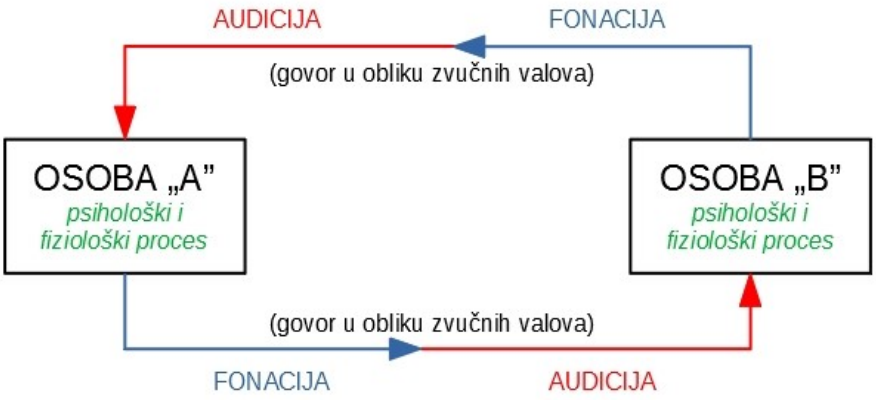
2.1. Govor

Povijesno gledano, govor uvijek prethodi jeziku te je nužno njegovo postojanje kako bi se jezik uopće uspostavio. Kao što je u uvidu spomenuto za ostvarenje ljudskog govora zaslužne su psihičke i fizičke predispozicije. Prije definiranja samog govora spomenut ćemo glasovni, odnosno govorni aparat i ukratko opisati uloge i rad organa koji sudjeluju u njemu.



Slika 2.1 Shematski prikaz organa koji su dio govorni aparata [3]

Na slici 2.1. shematski je prikazan govorni aparat. Mozak je početak i kraj govornog aparata. Živčani impulsi prenose podražaje iz mozga do govornih organa koji se pokreću i proizvode glasove, odnosno govor. Hrvatski jezik, kao i većina drugih, realizira se pri izdisaju. Nakon što se pluća napune zrakom pri udisaju, zračna struja iz pluća izlazi van kroz dušnik. Na kraju dušnika nalazi se grkljan u kojem su smještene glasnice. Prošavši kroz grkljan i glasnice, zračna struja dolazi do ždrijela koje vodi do usne i nosne šupljine. S obzirom na pokrete govornih organa u usnoj šupljini oblikuju se različiti glasovi. Prema zapisima zvučnih valova nastalih prilikom snimanja govora, prepoznaju se dva biološka faktora koja utječu na karakteristiku vala. Prvo, oblik zvučnog vala ovisi o načinu prolaska zraka kroz glasnice. Drugo, ovisi o obliku, odnosno pomacima samog govornog aparata prilikom stvaranja određenog zvuka. Važno je primijetiti da navedenim organima primarna svrha nije govor, već životno bitnije funkcije kao što su disanje, gutanje, žvakanje i pijenje.

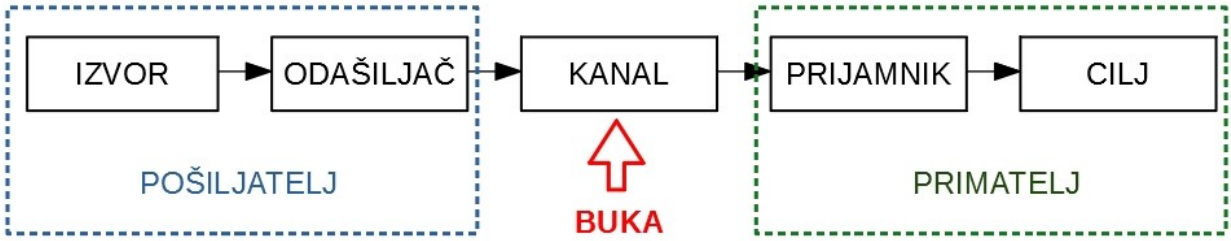


Slika 2.2 Shematski prikaz govornog kruga

Sada kada znamo kako ljudi stvaraju glasove potrebno je definirati govor. Lingvisti govor smatraju svakom konkretnom i pojedinačnom upotrebom jezika u obliku individualnog akta volje i inteligencije [2]. Govor se kod ljudi ostvaruje unutar govornog kruga koji se mora sastojati od minimalno dvije osobe. Postupak govora se može podijeliti na psihološki, fiziološki i fizički proces. Prva osoba u govornom krugu zamišlja pojam u mozgu, što spada pod psihološki proces. Zatim dolazi do fiziološkog procesa tj. podražaja u živčanom sustavu i na kraju je fizički proces, odnosno pokretanja govornih organa i stvaranja glasova. Poruka potom u obliku zvučnih valova

putuje do uha druge osobe. Druga osoba čuje zvučne valove (fizički proces) te dolazi do podražaja u živčanom sustavu (fiziološki proces) pa prepoznaje i povezuje s nekom mentalnom slikom ili idejom (psihološki proces). Potom se isti postupak ponavlja u drugom smjeru. Na slici 2.2 shematski je prikazan govorni krug. Fonacija i audicija fizički su procesi govornog kruga. Fonacija je stvaranje zvučnih valova, odnosno govorenje, a audicija je slušanje.

Dok ljudi komuniciraju međusobno u različitim manifestacijama govornog kruga, kada se u komunikaciju uključuje računalo stvari se mijenjaju. Primjer komunikacije između čovjeka i računala može se prikazati u obliku općeg komunikacijskom modela. Radi se o linearnom modelu komunikacije koji se sastoji od pošiljatelja, kanala i primatelja [2]. Kod prepoznavanja govora pošiljatelj je čovjek, kanal su zvučni valovi koji prenose poruku odnosno govor, a primatelj je računalo. Ovdje će biti prikazani osnovni dijelovi općeg komunikacijskom modela, a u trećem poglavlju će detaljno biti razrađen s aspekta prepoznavanja govora.



Slika 2.3 Shematski prikaz općeg modela komunikacije

Kao što se vidi na slici 2.3, pošiljatelj se u općem komunikacijskom modelu dijeli se na izvor i odašiljač. Iz izvora započinje proces komunikacije, kod ljudi je to misaoni proces koji se odvija u mozgu. Odašiljač je materijalni izvor koji ostvaruje namjeru komunikacije. Ako je riječ o verbalnoj komunikaciji, kod ljudi je to uvijek govorni aparat. Kanal za komunikaciju u sebi sadrži puno informacija. Prvenstveno se radi o mediju preko kojeg se šalje poruka. U verbalnoj komunikaciji to su zvučni valovi koji se mogu prenositi zrakom, ako je komunikacija izravna, ili posrednim medijem kao što su radio ili televizija, ako je komunikacija neizravna. Drugi sastavni dio kanala je kod koji predstavlja apstraktnu organizaciju znakova. Drugim riječima, kod je najčešće jezik kojeg poznaju i pošiljatelj i primatelj. U komunikacijskom kanalu može se pojaviti buka, odnosno

razne smetnje. Buka je također zvučnog karaktera, a može biti posljedica glasne okoline, veće udaljenosti između primatelja i pošiljatelja ili čak može nastati na izvoru komunikacije ako je pošiljatelj primjerice prehlađen ili ima govornu manu. Usprkos raznim smetnjama u većini komunikacije poruka se uspješno prenosi zahvaljujući redundanciji. Redundancija osigurava da ljudski jezik ima na raspolaganju više znakova nego što je potrebno za sporazumijevanje u uobičajenim i normalnim uvjetima. Primatelj se također sastoji od dva dijela, prijamnika i cilja. Zadatak prijamnika je prihvatiti materijaliziranu poruku i proslijediti je do cilja. U slučaju prepoznavanja govora, primatelj je zvučnik i dio memorije koji pohranjuje informacije o zvučnom zapisu, a cilj je algoritam koji će pomoću umjetnih neuronskih mreža pretvoriti dobivene informacije u tekst.

U opisu općeg komunikacijskom modela spomenuto je da pošiljatelj i prijamnik moraju imati zajednički jezik koji obje strane poznaju u dovoljnoj mjeri da bi komunikacija imala smisla. Ljudi uče jezike slušajući govor i pohranjujući iskustva u mozgu. Bez razumijevanja jezika, ljudi će čuti samo glasove koji im neće predstavljati ništa te se stoga komunikacija neće moći uspostaviti. Slično je i s umjetnim neuronskim mrežama, potrebno im je određeno poznavanje jezika koji se koristi pri prepoznavanju govora, no u manjoj mjeri nego što je potrebno prosječnom čovjeku.

2.2. Jezik

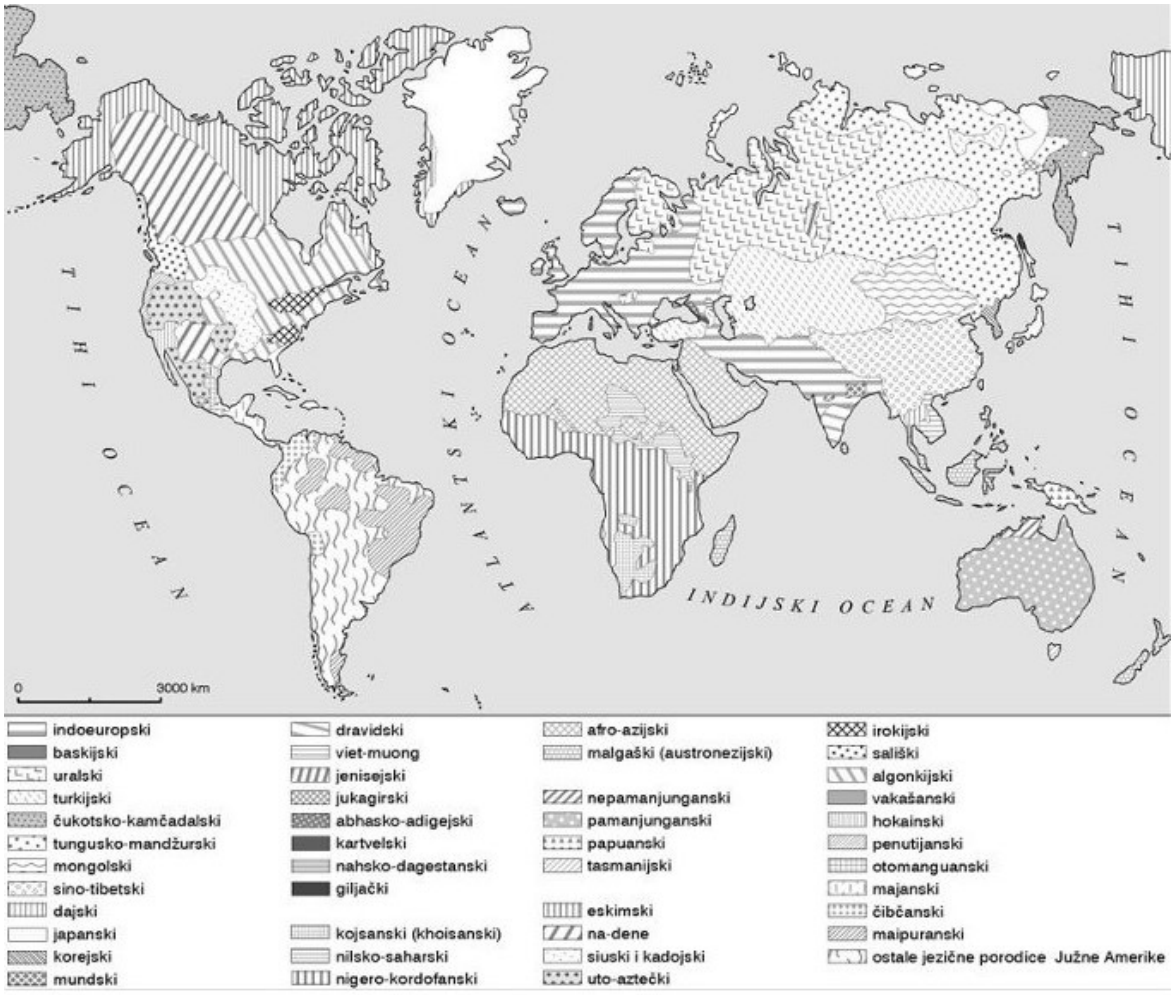
Već je spomenuto da iz govora proizlazi jezik, no jezik je također nužan kako bi govor bio smislen i ispunjavao svoju svrhu. F. de Saussure definira jezik kao društveni, zajednički dio jezične djelatnosti, izvan pojedinca koji ga sam ne može ni stvarati ni mijenjati; jezik postoji samo zahvaljujući svojevrsnom dogovoru sklopljenom između pripadnika zajednice. Jezik je toliko zasebna stvar da čovjek lišen moći govora može sačuvati jezik ako razumije glasovne znakove koje čuje [4].

Ova lingvistička definicija jezika može se razdvojiti na dijelove. Prvenstveno, jezik je osnovno sredstvo komunikacije među ljudima. On je posrednik između misli i glasa, odnosno zvuka. Jezik je također društveni proizvod, nastao je i postoji unutar određene jezične zajednice. Konvencionalan je tj. ima skup pravila te ga je potrebno učiti jer nije urođen ljudima kao što je

sam govor. Osnovna jedinica jezika je jezični znak koji povezuje pojam i akustičku sliku. Intuitivno bi bilo reći da jezični znak, primjerice riječ, povezuje stvar i ime, no to nije točno. Veza između dviju mentalnih slika, akustičke slike koja predstavlja zbir glasova koje čujemo i pojma koji je zapravo mentalna slika. Primjerice, pri pomisli na pojam *šuma*, u glavi se stvara slika određene šume. U pravilu će svaka osoba stvoriti drugačiju sliku šume. Kada poželimo tu sliku prenijeti u vanjski svijet, glasovima *š*, *u*, *m* i *a* ćemo stvoriti akustičku sliku koja se povezuje s tim pojmom. Ta veza je jezični znak.

No, jezik nije samo skup jezičnih znakova. Jezik je kompleksan sustav koji uključuje i gramatiku, odnosno pravila po kojima se jezični znakovi povezuju u veće cjeline sa svrhom ostvarenja komunikacije. Gramatika se sastoji od morfologije i sintakse. Morfologija se bavi vrstama riječi, kao što su imenice, glagoli, pridjevi itd., te njihovom sklonidbom, odnosno konjugacijom. Sintaksa se bavi međusobnim odnosima između vrsta riječi prilikom oblikovanja rečenica. U jeziku su također bitni sintagmatski odnosi, odnosno sličnosti i razlike. Jezik je linearnog karaktera, zbog čega ne možemo istovremeno izgovoriti dvije riječi. Linearne kombinacije dviju ili više susjednih riječi nazivaju se sintagmom. Nalazeći se u sintagmi, riječ dobiva vrijednost samo zbog odnosa prema prethodnoj ili sljedećoj riječi. No sintagme ne vrijede samo za riječi, već i za skupine riječi pa i za čitave rečenice. Gramatika i sintagmatski odnosi uvelike pomažu umjetnim neuronskim mrežama pri prepoznavanju govora, kao što se je objašnjeno u petom poglavlju.

Prije no što se definira pismo te njegov razvitak i važnost, nužno je spomenuti ogromno bogatstvo jezika na planetu Zemlji. Pri pomisli na sve jezike ovog svijeta, odmah se zapaža njihova mnogobrojnost i raznovrsnost. U svijetu se priča nekoliko tisuća jezika koji međusobno imaju nekih sličnosti ili pak nemaju nikakvih. Najraširenija skupina jezika je indoeuropska u koju spadaju i slavenski jezici, pa tako i hrvatski. Jezici unutar te skupine imaju više ili manje sličnosti, dok primjerice jezici sinotibetske skupine, kao što je kineski, nemaju baš nikakvih poveznica s njima. Također postoje i razlike unutar jednog jezika. U slučaju da jezik posjeduje pismo, onda ima i književni jezik koji se razlikuje od govornog jezika. Govorni jezik često ovisi o području govornika pa tako unutar jezika razlikujemo mnoge dijalekte. Može se reći da dijalekata ima koliko ima i mjesta, sela i gradova. U visoko civiliziranim društvima javljaju se i posebno specijalizirani jezici kao što su pravnički jezik, znanstveni ili pomorski.



Slika 2.4 Karta raspodjele jezičnih skupina u svijetu [1]

2.3. Pismo

Jezik i pismo su dva različita sustava znakova. Pismo je sustav grafičkih znakova koji predstavljaju elemente govornog jezika (glasove, slogove, riječi) i služe za pisanje. Može se reći da pismo postoji samo kako bi vizualno predstavljalo jezik. Dok se govor uči spontano, pismo se mora posebno učiti. Za razliku od jezika u obliku govora, pismo se čini kao nešto što je trajno i čvrsto što bolje osigurava jedinstvo jezika kroz vrijeme. Pismo je nepokretno u vremenu dok se jezik neprestano mijenja pa stoga dolazi do razlika u zapisanim i izgovorenim glasovima, što se može zapaziti primjerice u francuskom jeziku.

Općenito, pisma se dijele u dva velika sustava, ideografski i fonetski. U ideografskom sustavu riječ je predočena jednim jedinim znakom bez povezanosti sa zvukovima od kojih se sastoji. Klasičan primjer takvog sustava je kinesko pismo. S druge strane, težnja fonetskog sustava je da dosljedno reproducira slijed zvukova koji u riječi slijede jedan drugoga. Fonetska pisma mogu biti slogovna ili abecedna; drugim riječima, temelje se na najmanjim elementima govora [4]. U ovom radu govori se o fonetskim pismima čiji je prototip grčki alfabet, a u koja spada i latinica te time i hrvatska abeceda. Kao što im i samo ime kaže, fonetska pisma kao osnovu za povezivanje govora i pisma koriste najmanju jedinicu govora, glas. Glas ili fonem je skup akustičkih svojstva koja se opažaju istodobno. Fonemi su stalne i apstraktne jedinice jezika, dok su fonovi njihovo promjenjivo ostvarenje, odnosno konkretni glasovi. Ideja fonetskog pisma je da jedno slovo predstavlja jedan glas. No često se radi o više slova koja samostalno predstavljaju jedan glas, a u određenoj kombinaciji predstavljaju drugi. Pri prepoznavanju različitih fonema u govoru, ljudski mozak traži razlike između pojedinačnih fonema.

Hrvatski standardni jezik ima 32 fonema, a hrvatska abeceda ima 30 grafema, odnosno slova. To znači da hrvatsko pismo nije u potpunosti fonetsko jer se neki fonemi predstavljaju kombinacijom više slova. 27 fonema je predstavljeno jednim slovom, 3 fonema dvoslovom (*nj*, *lj*, *dž*), a dvoglasnik *ie* ili dvoslovom *je* ili troslovom *ije*. Fonetsko bilježenje (fonetska notacija) je dosljedan i sustavan način bilježenja govornih glasova. Svaki različiti glas ima vlastiti znak i bilježi se ili u uglatim zagradama ili u kosim zagradama kako bi se naglasilo da se ne radi o abecednom zapisu riječi. Za potrebe fonetskog bilježenja koristi se fonetska abeceda, a najpoznatija je međunarodna fonetska abeceda (engl. *International Phonetic Alphabet, IPA*). Ova abeceda razlikuje se od slavističke pa tako i hrvatske te je stoga proširena za potrebe hrvatskog jezika dodatnim simbolima. Fonetski zapis 32 fonema u hrvatskom standardnom jeziku vidi se na slici 2.5.

/i, e, a, o, u, i̯e; v, m, n, ń, l, l̩, r̩, r, j;
p, t, k, b, d, g, c, č, ć, ǯ, ǵ, f, s, š, h, z, ž/

Slika 2.5 Fonetski zapis 32 fonema u hrvatskom standardnom jeziku [5]

Fonemi se dijele s obzirom na mjesto nastanka u usnoj šupljini i akustičkim svojstvima. Akustička svojstva fonema su: vokalnost, šumnost, kompaktnost, difuznost, gravisnost, akutnost, nazalnost, neprekidnost, zvučnost, stridentnost i napetost [5]. Podjela fonema u hrvatskom jeziku prikazana je u tablici 2.1.

Tablica 2.1 Podjela glasova (fonema) u hrvatskom standardnom jeziku [5]

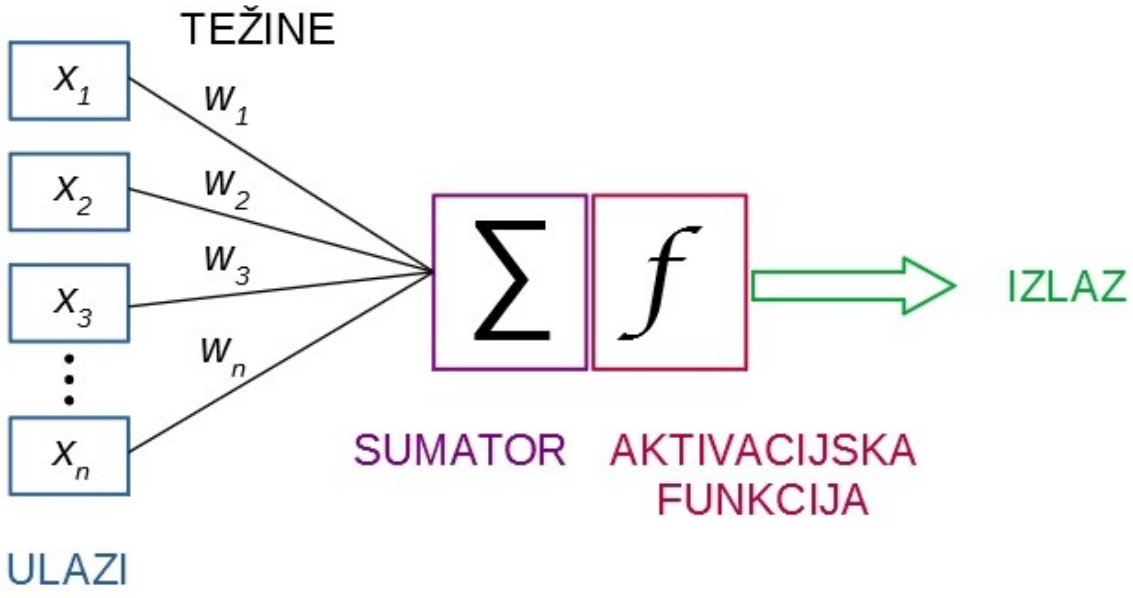
po vrsti zapreke / po mjestu tvorbe			samoglasnici						suglasnici													
									sonanti				šumnici									
													tjesnačni			poluzatvoreni					zatvo-reni	
ispred tvrdog nepca	dvousneni (bilabijalni)									m											p	b
	zubnousneni (labiodentalni)								v				f									
	zubni (dentalni)									n				s	z	c	ʒ				t	d
	nadzubni (alveolari)							r̥			l	r										
na tvrdom nepcu	nep-čani	prednjo-tvrdo-nepčani	i		i̯e	e	i̯		j	ń	l̥			ś	ź					ć	ʒ́		
		stražnjo-tvrdo-nepčani												š	ž			č	ǯ				
iza tvrdog nepca	srednjonepčani			a		ə																
	mekonepčani		u		o					ŋ			x	h	ɣ						k	g

Ljudima nije problem odmah jasno prepoznati i razdvojiti foneme jednom kada čuju riječ, no računalu je. Suština problema je što je svaki proizvedeni glas jedinstven i neponovljiv. Isti glas može biti izgovoren na nebrojno mnogo načina. Može varirati u duljini trajanja, glasnoći, jasnosti, i tonu glasa osobe koja ga izgovara, kao i o položaju unutar izgovorene riječi. Bez obzira na različito izgovaranje istog fonema zbog razlika u dobi, spolu, naglasku ili emocionalnom stanju govornika, ljudi u pravilu uspješno razlikuju foneme. Cilj je korištenja umjetnih neuronskih mreža pri prepoznavanju govora da računalo jedan te isti fonem uvijek isto i točno prepozna bez obzira ne sve njegove moguće varijacije, baš kao što to ljudi s lakoćom čine.

3. UMJETNE NEURONSKE MREŽE

Prije detaljnog opisa umjetnih neuronskih mreža koje se koriste pri prepoznavanju govora, potrebno je objasniti općeniti model umjetnih neuronskih mreža te kako one u osnovi funkcioniraju.

Kao što im i samo ime kaže, umjetne neuronske mreže (engl. *Artificial Neural Networks, ANN*) stvorene su kako bi oponašale rad stvarnih neuronskih mreža, odnosno rad ljudskog mozga. Ideja je bila kreirati model koji će biti sposoban procesirati informacije analogno ljudskom mozgu, tj. model koji će prihvaćati, obrađivati, generirati, pohranjivati i prenositi informacije. Po uzoru na biološki neuron, 1943. McCulloch i Pitts su osmisli jednostavan model umjetnog neurona. Kao što je ljudski mozak građen od bioloških neurona, tako su i umjetne neuronske mreže građene od umjetnih neurona. Dok se pretpostavlja da ljudski mozak sadrži preko 100 milijardi neurona koji su povezani s preko 100 trilijuna veza, umjetne neuronske mreže mogu se sastojati od samo jednog umjetnog neurona do onoliko koliko omogućuju hardverske mogućnosti računala. Na slici 3.1. prikazana je jednostavna struktura umjetnog neurona.



Slika 3.1 Pojednostavljeni prikaz strukture umjetnog neurona

Iz slike 3.1 vidljivo je da se model umjetnog neurona ugrubo sastoji od četiri dijela – ulaza, sumatora, aktivacijske funkcije i izlaza. Aktivnost umjetnog neurona modelira se kao zbroj ulaza pomnoženih nekim faktorom (težinama), a ovisi o broju ulaza, odnosno veza s okolinom neurona, intenzitetu tih veza te pragu osjetljivosti [6]. Stanje neuronske mreže je u svakom trenutku opisano skupom aktivnosti svakog pojedinog neurona u mreži, a to stanje se mijenja u vremenu s obzirom na ulazne podatke. Okolinu neurona čine ostali neuroni umjetne neuronske mreže i/ili okruženje te mreže. Intenzitet veza ovisi o težinskom faktoru koji će kasnije biti detaljno objašnjen. Prag osjetljivosti se definira kao stanje koje neuron mora dosegnuti kako bi mogao generirati signal koji će proslijediti ostalim neuronima u okolini. Svi signali unutar neuronske mreže mogu biti kontinuirani ili diskretni te deterministički ili stohastički, ovisno o tipu i zahtjevima mreže.

Tablica 3.1 Usporedba dijelova i njihovih uloga kod biološkog i umjetnog neurona

BIOLOŠKI NEURON	**UMJETNI NEURON**
tijelo	sumator
dendriti	ulazi u sumator
akson	izlaz sumatora
prag osjetljivosti	aktivacijska funkcija
sinapsa, sinaptičke veze	težinski faktori

U tablici 3.1 vidi se usporedba građe biološkog neurona s građom umjetnog neurona. Biološki neuron sastoji se od tijela, dendrita i aksona. Sinapsa je malen prostor između neurona preko kojeg se odvija komunikacija među neronima. Kod umjetnog neurona tijelo postaje sumator, dendriti postaju ulazi u sumator iz okoline neurona, a akson izlaz sumatora. Izlaz sumatora povezan je s ulazom aktivacijske funkcije koja kod umjetnog neurona preuzima ulogu funkcije praga osjetljivosti. Ona na svom izlazu daje izlaz cijelog neurona. Aktivacijske funkcije dijele se na linearne i nelinearne. Kod linearnih aktivacijskih funkcija izlaz sumatora se množi s određenim faktorom, a kod nelinearnih se izlaz sumatora prenosi na izlaz neurona preko određenog nelinearnog pojačanja. Postoji više različitih nelinearnih aktivacijskih funkcija. Klasični primjeri aktivacijskih funkcija su funkcije praga osjetljivosti, sigmoidalne i hiperbolične funkcije.

Sinaptičke veze kod bioloških neurona povezuju neuron s njegovom okolinom. Ulogu sinapsa kod umjetnih neurona preuzimaju težinski faktori. Oni povezuju izlaze jednog neurona s ulazima drugog neurona. Svaki težinski faktor ima realnu vrijednost koja se kreće od $-\infty$ do $+\infty$, iako se u nekim slučajevima koriste ograničeni intervali mogućih vrijednosti. Uz iznos težinskog faktora veže se već spomenuti intenzitet veze, a uz predznak se veže karakter veze. Veće vrijednosti težinskih faktora svojstvo su snažnijih i bitnijih veza [7]. Također, težinski faktori mogu biti konstantni ili varijabilni. Konstantan težinski faktor može biti pozitivan ili negativan, pri čemu pozitivne vrijednosti pobuđuju mrežu, a negativni inhibiraju veze između neurona. U slučaju kad je iznos težinskog faktora jednak nula, veza s okolinom ne postoji. Varijabilni težinski faktor modelira se pomoću određene funkcije. Težinski faktori mijenjaju se tijekom procesa učenja, a na kraju učenja ostaju nepromjenjivi.

Tablica 3.2 Vrste podjela umjetnih neuronskih mreža

PODJELA PO:	**VRSTA UMJETNE NEURONSKE MREŽE**
broju slojeva neuronske mreže	jednoslojne
	višeslojne
smjeru signala	unaprijedne
	povratne
metodi učenja	nadgledano (supervizorno)
	nenadgledano (nesupervizorno)
	polu nadgledano (polu supervizorno)
topologiji mreže	nestrukturirane
	slojevite
	povratne
	modularne

Umjetne neuronske mreže se međusobno razlikuju prema dva aspekta. Prvo se gleda struktura veza među neuronima te veza između neurona i okoline neuronske mreže. Kao drugo se promatra metodologija određivanja intenziteta veza, odnosno metode učenja mreže. Općenito, umjetne neuronske mreže mogu se podijeliti s obzirom na broj slojeva neuronske mreže, smjer signala, metodu učenja te topologiju mreže.

Prema tablici 3.2 vidi se da postoje jednoslojne i višeslojne umjetne neuronske mreže. Višeslojne umjetne neuronske mreže sastoje se od ulaznog i izlaznog sloja te određenog broja sakrivenih slojeva između njih. Ulazni sloj nalazi se u prvom sloju mreže i prima podražaje, odnosno ulaze. Skriveni slojevi nalaze se unutar mreže i obrađuju podatke, a izlazni sloj nalazi se na posljednjem mjestu neuronske mreže i prosljeđuje dobiveni signal dalje.

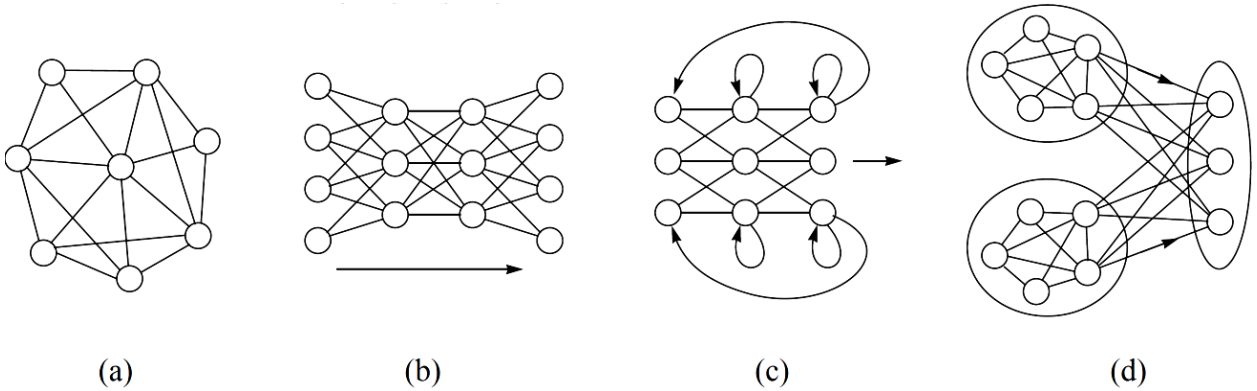
Neuronske veze su uglavnom jednosmjerne, no postoje i mreže sa specifičnim topologijama gdje nisu točno određeni ulazni i izlazni neuroni pa su veze dvosmjerne. Unaprijedne i povratne neuronske mreže razlikuju se po smjeru kretanja signala. Kod unaprijednih signal putuje samo u jednom smjeru, od ulaza do izlaza mreže, dok kod povratnih mora postojati barem jedna povratna petlja po kojoj signal putuje u suprotnom smjeru.

Učenje je zajednička karakteristika svih tipova umjetnih neuronskih mreža. Učenje mreže predstavlja mijenjanje težinskih faktora čime se utječe na sposobnost i topologiju mreže. Može biti nadgledano (engl. *supervised*), nenadgledano (engl. *unsupervised*) ili polu nadgledano učenje (engl. *semi-supervised*). Kod nadgledanog učenja potreban je vanjski utjecaj, odnosno ljudska kontrola, a podaci za učenje su strukturirani u parove ulaza i izlaza. Učiteljev je zadatak pratiti proces učenja i eksplicitno ispravljati pogreške koje mreža čini. Ovakva vrsta učenja prvenstveno se koristi kod unaprijednih perceptronskih te kod povratnih neuronskih mreža. Kod nenadgledanog učenja mreža mora potpuno sama pronaći pravila među nestrukturiranim podacima za učenje, a pogodno je za zadatke klasifikacije i preslikavanja ulaznih podataka. Polu nadgledano učenje je metoda u kojoj ulogu „učitelja“ zamjenjuje „kritičar“. Drugim riječima, čovjek u ulozi kritičara prati proces učenja mreže i evaluira njeno ponašanje kao „dobro“ ili „loše“, od čega dolazi drugi naziv za ovu metodu učenja, učenje s poticajem (engl. *reinforcement learning*). Većina mreža

upotrebljava jednu od ove tri metode učenja, iako postoje hibridne mreže koje koriste kombinaciju nadgledanog i nenadgledanog učenja.

Algoritmi učenja mogu se podijeliti i prema broju koraka učenja. Postoje dvije osnovne skupine, učenje u jednom koraku i iterativno učenje. Najčešće se mreže uče kroz iterativne postupke što zahtjeva višestruki prolaz kroz skup podataka za učenje; slično kao što kaže latinska poslovica – *repetitio est mater studiorum*. Određivanje brzine učenja je jako važno jer u slučaju da je brzina premala, učenje bi moglo trajati beskrajno, a ako je brzina prevelika, moglo bi pri iteracijama doći do gubitka otprije stečenog znanja u obliku vrijednosti težinskih faktora. U nedostatku analitičke metode, brzina učenja se u pravilu određuje empirijski isprobavanjem različitih vrijednosti.

S obzirom na način povezivanja neurona, postoje četiri topologije neuronskih mreža. Nestrukturirane, odnosno neuređene, mreže uspješne su u zadacima generaliziranja i rekonstruiranja djelomičnih ulaznih podataka (engl. *pattern completion*). Ako nestrukturirana mreža sadrži barem jednu povratnu vezu, što je moguće, onda je ona također i povratna mreža. Slojevite mreže, čiji su najpoznatiji primjer konvencionalne unaprijedne mreže, dobre su za zadatke klasifikacije i asocijacije (engl. *pattern association*). Povratne mreže mogu u osnovi biti slojevite ili nestrukturirane, ali obavezno moraju sadržavati barem jednu povratnu vezu. Ističu se u zadacima koji prate aktivnosti mreže u vremenu (engl. *pattern sequencing*). Modularne mreže su sastavljene od više jednostavnih modela triju prethodno spomenutih mreža, pri čemu mreže ne moraju biti iste topologije. Na slici 3.2 shematski su prikazane različite topologije mreža.



Slika 3.2 Različite topologije neuronskih mreža [7]:
(a) nestruktuirana, (b) slojevita, (c) povratna i (d) modularna

Za rješavanje problema prepoznavanja govora danas se najčešće koriste kombinacije unaprijednih ili povratnih neuronskih mreža sa skrivenim Markovljevim lancima. U nastavku ovog poglavlja ukratko će biti objašnjeni principi rada unaprijednih i povratnih neuronskih mreža, kao i najčešće upotrebljavani algoritam učenja, algoritam učenja rasprostiranjem pogreške.

3.1. Unaprijedne neuronske mreže

Unaprijedne neuronske mreže (engl. *feedforward neural network*) su jedne od najpopularnijih topologija čiji su neuroni složeni po slojevima te signal putuje u samo jednom smjeru, od ulaza do izlaza, bez ikakvih povratnih petlji. U ovim mrežama izlaz svakog neurona jednog sloja povezan je s ulazom neurona idućeg sloja.

3.1.1. Jednoslojne unaprijedne neuronske mreže

Najjednostavniji model unaprijedne neuronske mreže sastoji se od jednog umjetnog neurona. Ako taj umjetni neuron kao aktivacijsku funkciju koristi binarnu funkciju praga osjetljivosti, onda se još naziva i perceptron. Binarna funkcija praga osjetljivosti može postići samo dvije vrijednosti, 1 ili 0, a definirana je izrazom

$$y_j = f(z_j) = \begin{cases} 1 \text{ ako je } z_j > \text{prag} \\ 0 \text{ ako je } z_j \leq \text{prag} \end{cases}, \tag{3.1}$$

gdje je y_j vrijednost izlaza neurona, a z_j je sumirana vrijednost otežanih ulaza. Kada ulazna vrijednost prelazi prag osjetljivost, perceptron je aktivan i daje izlaz jednak 1 te šalje dalje signal. Kada ulaz ne prelazi prag osjetljivosti, perceptron je neaktivan i izlaz mu je 0.

Prva neuronska mreža koja je osmišljena bila je jednostavna jednoslojna perceptronska mreža (engl. *Single Layer Perceptron*). Sastoji se od više perceptrona povezanih u neuronsku mrežu, a ima samo jedan ulazni i jedan izlazni sloj. Primjenjuje se isključivo za učenje problema linearne klasifikacije.

Na slici 3.3 prikazana je struktura jednoslojne mreže perceptrona na kojoj se vidi da su neuroni ulaznog sloja u potpunosti povezani s neuronima izlaznog sloja, a neuroni unutar istog sloja nisu međusobno povezani. Ulazni podaci i težinski faktori se mogu vektorski zapisati:

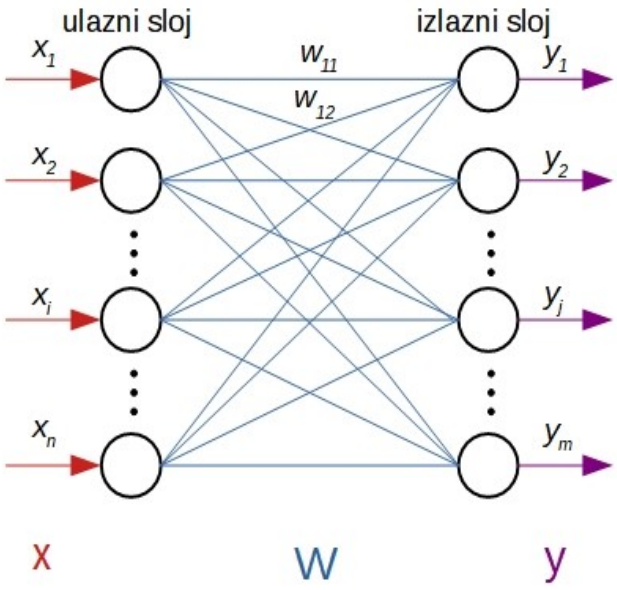
$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \\ \vdots \\ x_n \end{bmatrix}, \tag{3.2}$$

$$\mathbf{W} = [w_{11} \quad w_{12} \quad \cdots \quad w_{ij} \quad \cdots \quad w_{nm}]. \tag{3.3}$$

Izlazni vektor računa se prema sljedećem izrazu:

$$\mathbf{y} = \mathbf{W}^{\mathbf{T}}\mathbf{x}. \tag{3.4}$$

Općenito, dimenzije ulaznog vektora **x** su n x 1, matrica težinskih faktora **W** je dimenzija n x m, a izlazni vektor **y** je dimenzija m x 1, pri čemu je n broj ulaznih neurona, a m broj izlaznih. Izraz (3.4) vrijedi samo kod mreža čiji su svi slojevi međusobno povezani.



Slika 3.3 Jednoslojna mreža perceptrona

Predmet učenja umjetne neuronske mreže su težine izlaznog sloja w_{ij}, a učenje se provodi pomoću skupa za učenje, koji se sastoji od većeg broja uzoraka opisanih ulaznim vektorima i njima odgovarajućih željenih izlaznih vrijednosti. Perceptronske mreže se mogu učiti pomoću delta pravila koje je također osnova za učenje većine modela neuronskih mreža. Delta pravilo računa pogrešku između dobivene vrijednosti izlaza i željene vrijednosti, a može se zapisati izrazom:

$$\delta_j = T_j - A_j \,, \tag{3.5}$$

gdje je T_j željena izlazna vrijednost (engl. *Target output*), A_j izračunata vrijednost izlaza (engl. *Actual output*) izlaznog neurona *j*, a odstupanje δ_j je njihova razlika. U slučaju da postoji odstupanje od željenih vrijednosti, ono se koristi za korigiranje podesivih težina mreže prema sljedećim izrazima:

$$\Delta_{ij} = \eta \delta_j x_i, \tag{3.6}$$

$$w_{ij}(n+1) = w_{ij}(n) + \Delta_{ij}. \tag{3.7}$$

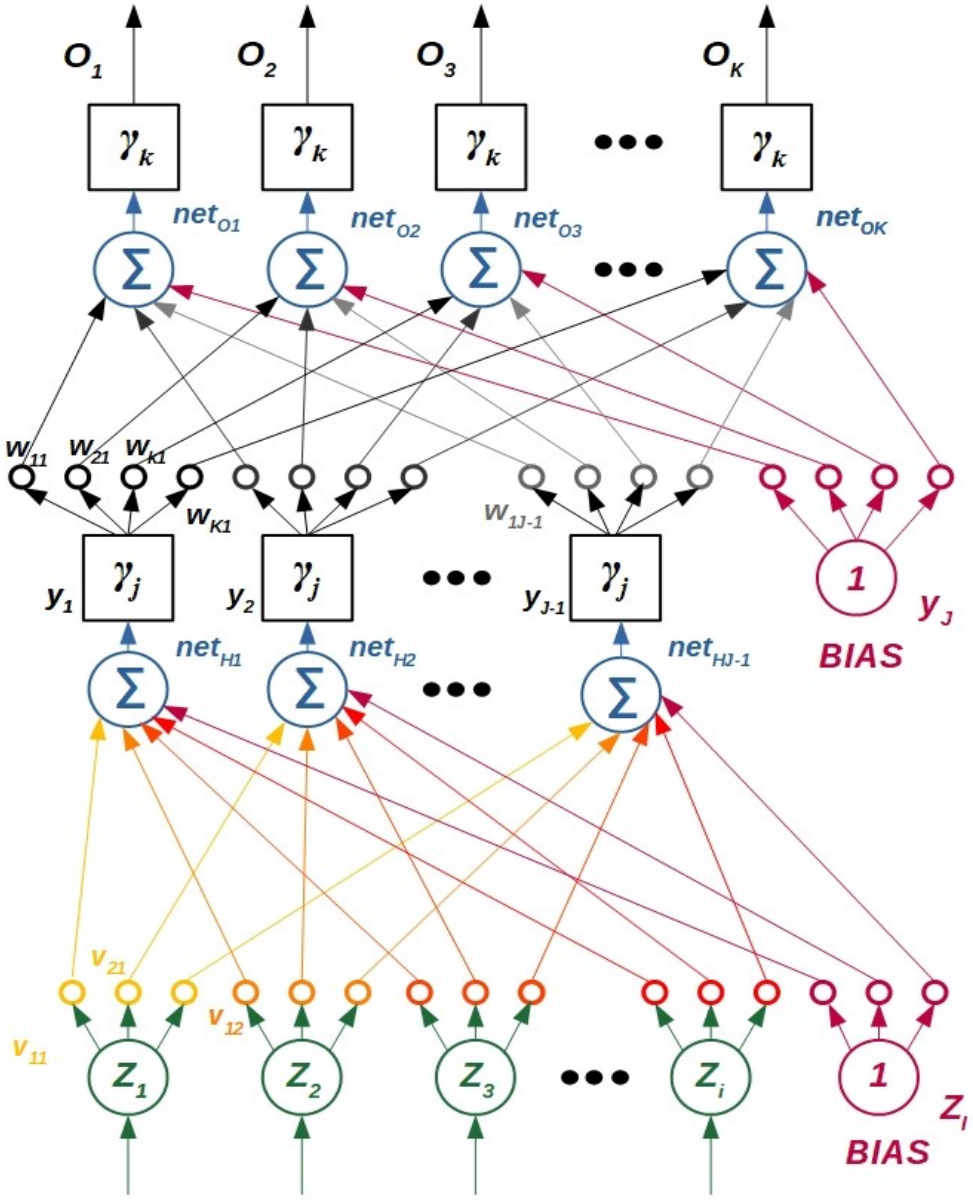
Simboli u prethodnim izrazima redom predstavljaju: *i* je indeks neurona ulaznog sloja, *j* je indeks neurona izlaznog sloja, Δ_{ij} je faktor promjene težina, η je koeficijent brzine učenja, $w_{ij}(n+1)$ je težina veze između neurona *i* ulaznog sloja i neurona *j* izlaznog sloja poslije podešavanja, a $w_{ij}(n)$ je težina prije podešavanja.

3.1.2. Višeslojne unaprijedne neuronske mreže

Višeslojne perceptronske neuronske mreže (engl. *Multi-Layer Perceptrons, MLP*) nastaju dodavanjem slojeva perceptrona. Slojevi su međusobno povezani opterećenim težinskim faktorima. Ulazni i izlazni sloj mreže su u direktnoj komunikaciji s okolinom, dok skriveni slojevi nisu. Broj skrivenih slojeva nije ograničen, ali treba uzeti u obzir da svaki dodatni skriveni sloj usporava učenje mreže pa se stoga najčešće koriste jedan do dva skrivena sloja u mrežama. Signal i dalje putuje isključivo u jednom smjeru, od ulaza do izlaza. Unaprijedne mreže s većim brojem skrivenih slojeva često se nazivaju i duboke neuronske mreže (engl. *Deep Neural Networks, DNN*).

3.1.2.1. Statička unaprijedna višeslojna mreža

Statička unaprijedna višeslojna mreža (engl. *Static Neural Network*) je najpoznatija i najčešće upotrebljavana neuronska mreža, a shematski je prikazana na slici 3.4. Svi slojevi mreže u potpunosti su umreženi s izuzetkom BIASa. Težinski faktori između ulaznog i skrivenog sloja označavaju se s v_{ji}, a težinski faktori između skrivenog i izlaznog sloja s w_{kj}. Izlazi neuronske mreže označeni su s O_k.



Slika 3.4 Model unaprijedne statičke mreže [6]

Statička unaprijedna višeslojna mreža građena je od statičkih neurona. Standardni model statičkog neurona sadrži dvije podfunkcije, funkciju sume Σ i aktivacijsku funkciju γ. Kao i svi neuroni, statički neuron posjeduje više ulaza i samo jedan izlaz. Svaki neuron u mreži ima „poseban" ulaz povezan sa zasebnim neuronom s oznakom BIAS. Riječ je o ulazu jedinične vrijednosti koji je potreban za odvijanje procesa učenja mreže. Broj sakrivenih slojeva je proizvoljan, a najčešće se koristi jedan ili dva sakrivena sloja. Broj neurona u skrivenom sloju je također proizvoljan; u pravilu ovisi o zadatku mreže te se određuje eksperimentalno. Funkcija sume statičkog neurona definira se sljedećim izrazom:

$$net = \sum_{j=1}^{J} w_j u_j \,. \tag{3.8}$$

Rezultat funkcije sume je vrijednost *net* koja se pomoću aktivacijske funkcije γ preslikava na izlaznu vrijednost neurona *y*:

$$y = \gamma(net). \tag{3.9}$$

3.2. Povratne neuronske mreže

Uz unaprijedne neuronske mreže, u današnjim sustavima za prepoznavanje govora koriste se i povratne neuronske mreže (engl. *Recurrent Neural Networks, RNN*). To su neuronske mreže koje u sebi sadrže povratne petlje što omogućuje informacijama da ostanu unutar mreže. Drugim riječima, povratne mreže omogućuju da se prethodne informacije povežu s trenutnim zadatkom. Zbog svoje lančane prirode povratne neuronske mreže dobro rade s nizovima i listama. Stoga su jako dobro primjenjive za sve zadatke obrade ljudskog govora, među koje spada i prepoznavanje govora.

Za razliku od unaprijednih neuronskih mreža, povratne neuronske mreže na ulazu obrađuju podatke u obliku niza, a težine mreže su međusobno vremenski ovisne. Postoji više standardnih oblika povratnih mreža. Konvencionalna povratna neuronska mreža ima skriveni sloj koji se može označiti s:

$$h_t^i = f\left(W^i h_t^{i-1} + U^i h_{t-1}^i + c^i\right), \tag{3.10}$$

gdje $f(\cdot)$ predstavlja nelinearnu funkciju (npr. sigmoidalnu), *i* je sloj mreže, *t* je oznaka broja segmenta ili vremenski indeks, a ulaz *x* je ekvivalentan izlazu nultog sloja, $h_t^i = x_t$. Slojevi u povratnim mrežama ovise o trenutnom ulazu te izlazu prijašnjeg vremenskog koraka.

Postoje izvedbe povratnih neuronskih mreža koje provode povratnu vezu u oba smjera, a koriste se u situacijama u kojima latentnost nije bitna. Ovakve mreže nazivaju se dvosmjerne neuronske mreže (engl. *bidirectional neural network*). Kod ovakvih mreža svaki sloj ima skupinu parametara koja se koristi za obradu niza prema naprijed u vremenu i drugu skupinu parametara koja služi za obradu niza unazad. Ova dva izlaza se mogu spojiti i proslijediti na ulaz idućeg sloja.

Matematički zapis ovog postupka glasi:

$$\overrightarrow{h_t^i} = f\left(W_b^i h_t^{i-1} + U^i h_{t-1}^i + c_f^i\right), \tag{3.11}$$

$$\overleftarrow{h_t^i} = f\left(W_b^i h_t^{i-1} + U^i h_{t+1}^i + c_b^i\right), \tag{3.12}$$

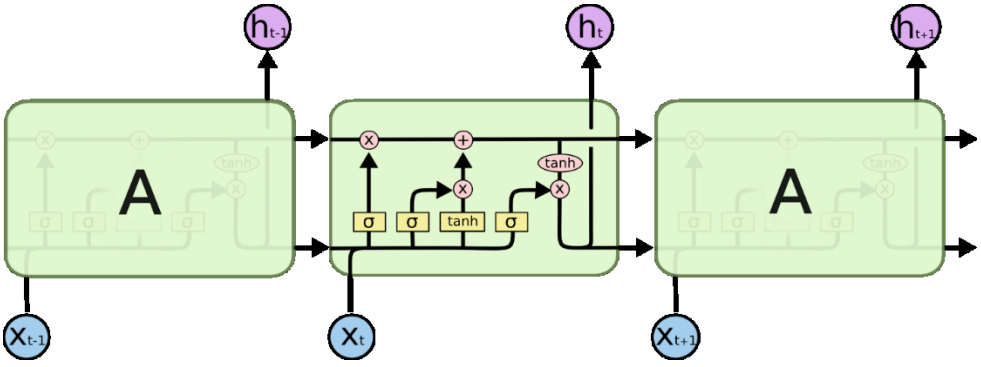
$$h_t^i = \left[\overrightarrow{h_t^i}\, \overleftarrow{h_t^i}\right]. \tag{3.13}$$

f i *b* predstavljaju oznake smjera, naprijed (engl. *forward*) i nazad (engl. *backward*).

3.2.1. LSTM (engl. Long Short-Term Memory)

Kod konvencionalnih povratnih mreža javlja se problem u situacijama kada je potrebno povezati trenutni zadatak s udaljenijom informacijom. Na primjeru predviđanja riječi prema kontekstu, ovaj problem bi se mogao predstaviti na sljedeći način. Recimo da je zadatak mreže predvidjeti posljednju riječ u rečenici. Ako rečenica glasi: „Kiša pada iz tamnih *oblaka*.“, povratna neuronska mreža lako predviđa riječ „*oblaka"* iz konteksta rečenice. Ako rečenica pak glasi: „Rođen sam u Hrvatskoj, …, te tečno pričam *hrvatski*.“, povratna neuronska mreža naslućuje da je posljednja riječ ime jezika, ali ne može predvidjeti o kojem jeziku je riječ jer se Hrvatska spominje na početku duge rečenice. U teoriji bi povratne neuronske mreže trebale uspješno rješavati ovaj problem, ali u praksi to nije slučaj [9].

Posebna vrsta povratnih neuronskih mreža, koju su 1997. osmislili Hochreiter i Schmidhuber, nazvana je LSTM (engl. *Long Short-Term Memory*), odnosno u doslovnom prijevodu, povratna mreža s dugom kratkotrajnom memorijom. LSTM mreže sposobne su učiti udaljene ovisnosti informacija. Imaju lančanu strukturu kao i konvencionalne povratne mreže, no kao ponavljajući dio ne koriste jedan sloj s jednostavnom funkcijom (najčešće *tanh* kod konvencionalnih), već četiri međusobno povezana sloja.

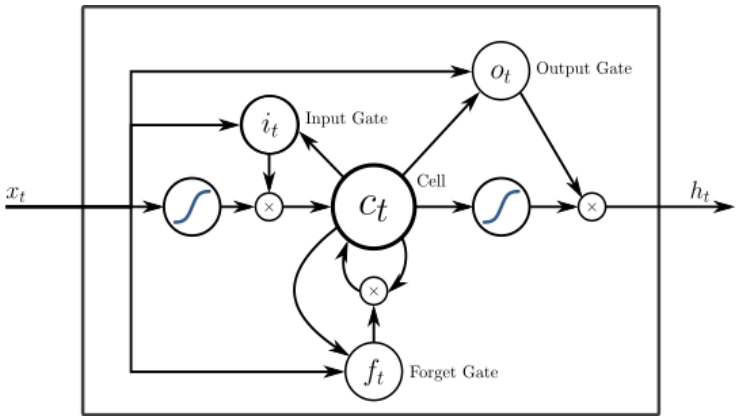


Slika 3.5 Shematski prikaz LSTM mreže [9]

Na slici 3.5 oznaka A predstavlja dio neuronske mreže, žuti pravokutnik predstavlja sloj neuronske mreže, rozi krugovi ili elipse predstavljaju matematičke operacije, a svaka nacrtana linija prenosi cijeli vektor informacija od ulaza do izlaza.

LSTM mreže bazirane su na konceptu ćelija (engl. *cells*) koje čuvaju informacije. Informacija može ostati sačuvana u vremenu ili može biti promijenjena u drugu s obzirom na operacije koje se odvijaju unutar četiri povratna sloja. Ove matematičke operacije nazivaju se vrata (engl. *gates*). Ako je vrijednost na vratima blizu nule, informacija je blokirana, a ako je vrijednost blizu jedan, vrata propuštaju informaciju dalje kroz mrežu. Ulazna vrata (engl. *input gate*) odlučuju hoće li propustiti informaciju iz trenutnog koraka u ćeliju. Vrata zaborava (engl. *forget gate*) odlučuju hoće li trenutna informacija unutar ćelije ostati ili će biti promijenjena. Izlazna vrata (engl. *output gate*) odlučuju hoće li trenutnu informaciju u ćeliji proslijediti dalje ili ne.

Zbog svoje uspješnosti u mnogim zadacima, LSTM povratne mreže su danas najčešće korišteni tip povratnih neuronskih mreža, posebice u sustavima za prepoznavanje govora.



Slika 3.6 Dijagram ćelije i vrata LSTM mreže [10]

3.3. Algoritam učenja povratnim rasprostiranjem pogreške [7]

Učenje neuronske mreže je postupak podešavanja težinskih faktora sa ciljem da izlazi mreže što točnije odgovaraju željenim vrijednostima za određene ulaze. Cilj učenja nije osigurati potpunu točnost, već dovoljno dobru aproksimaciju. Kvaliteta aproksimacije ovisi o više čimbenika kao što su zadatak mreže, struktura mreže i odabrani algoritam učenja. Algoritam učenja povratnim rasprostiranjem pogreške (engl. *backpropagation algorithm*) najpopularnije je metoda nadgledanog učenja umjetnih neuronskih mreža. Najčešće se koristi za učenje unaprijednih mreža, ali s modifikacijama se može primijeniti i na povratne neuronske mreže.

3.3.1. Učenje unaprijedne neuronske mreže

Postupak učenja za svaki korak odvija se u dvije faze, unaprijedna faza i povratna faza. Promjena parametara učenja također se može odvijati na dva načina. Prvi način je da se parametri promjene jednom nakon prolaska mreže kroz čitavi skup za učenje koristeći se srednjom pogreškom u tom skupu. U drugom načinu se parametri mijenjaju za svaki ulazno izlazni par skupa učenja. Drugi način promjene parametra najčešće se koristi, a još se naziva i stohastičkim postupkom.

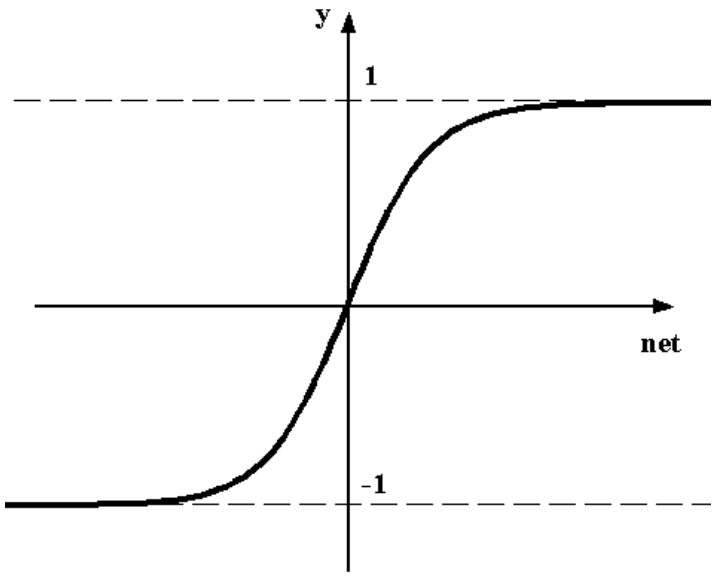
3.3.1.1. Unaprijedna faza učenja

U unaprijednoj fazi učenja uzimaju se vrijednosti svih ulaza mreže **Z** te se na temelju njih izračunava izlaz mreže **O**. Kako bi to bilo moguće potrebno je odrediti početne iznose težinskih faktora **W** i **V**, za što se najčešće koriti generator nasumičnih brojeva. Raspon u kojem se odabiru slučajne vrijednosti odgovara rasponu vrijednosti unutar kojih se kreću ulazi i izlazi. Ako se pojave problemi prilikom učenja, potrebno je smanjiti raspon slučajnih vrijednosti za red veličine naspram raspona ulaznih i izlaznih vrijednosti. Matematički model unaprijedne faze učenja od ulaznog do izlaznog sloja prikazan je u nastavku.

Pomoću ulaza mreže i vrijednosti težinskih faktora, u skrivenom sloju mreže računa se suma *net* neurona skrivenog sloja *H* na sljedeći način:

$$net_{Hj} = \sum_{i=1}^{I} \mathbf{v}_{ji} Z_i\,, \qquad j = 1,2,\dots,J-1, i = 1,2,\dots,I. \tag{3.14}$$

I je broj ulaznih neurona uvećan za jedan, a *J* je broj neurona u skrivenom sloju uvećan za jedan zbog BIASa. Standardno se za aktivacijsku funkciju bira nelinearna bipolarna sigmoidalna funkcija prikazana na slici 3.7.



Slika 3.7 Nelinearna bipolarna sigmoidalna funkcija

Kada se koristi gore prikazana sigmoidalna funkcija kao aktivacijska, izlazi neurona skrivenog sloja mogu se izračunati prema:

$$y_j = \frac{2}{1 + e^{-net_{Hj}}} - 1, \qquad j = 1,2,\dots,J-1, \qquad y_J = 1\ (BIAS). \tag{3.15}$$

Ako mreža ima samo jedan skriveni sloj, izračunate vrijednosti izlaza neurona skrivenog sloja spojene su na ulaz svakog neurona izlaznog sloja preko težinskih faktora w_{kj}. U slučaju da mreža ima više skrivenih slojeva, izlazi jednog povezani su na ulaze sljedećeg skrivenog sloja.

Ulaskom u izlazni sloj, funkcija sume *net* dobiva prvi indeks pripadnog sloja *O*, a drugi prema svakom neuronu izlaznog sloja:

$$net_{Ok} = \sum_{j=1}^{J} \mathbf{w}_{kj} y_j , \qquad k = 1,2, \dots , K. \tag{3.16}$$

K je broj neurona izlaznog sloja, odnosno broj izlaza mreže.

Ako se za aktivacijsku funkciju izlaznog sloja odabere linearna funkcija, onda se izlaz mreže može aproksimirati na sljedeći način:

$$O_k = K_p net_{Ok}, \qquad k = 1,2, \dots , K. \tag{3.17}$$

K_p predstavlja nagib linearne aktivacijske funkcije. Kada se koristi $K_p = 1$, onda izraz (3.17) prelazi u:

$$O_k = net_{Ok}. \tag{3.18}$$

3.3.1.2. Povratna faza učenja

U povratnoj fazi učenja izračunava se pogreška učenja na osnovu ostvarenog i željenog izlaza mreže. Prema pogreški učenja vrši se korekcija vrijednosti težinskih faktora između slojeva. Čitav postupak se ponavlja za svaki ulazno izlazni par seta za učenje sve dok se ne postigne pogreška manja ili jednaka dozvoljenom odstupanju od željenog iznosa izlaza koju određuje učitelj. Jednostavna i uobičajena funkcija cilja koja se koristi kao mjera odstupanja je suma kvadrata pogreške, a glasi:

$$E = \frac{1}{2} \sum_{n=1}^{N} (d_n - O_n)^2, \tag{3.19}$$

gdje je *N* broj elemenata u skupu za učenje.

Postupak podešavanja težinskih faktora odgovara minimiziranju funkcije cilja E izražene izrazom (3.19). Na osnovu odabrane funkcije cilja vrši se promjena težinskih faktora primjenom nekog od algoritama nelinearnog optimiranja. Promjena parametara učenja ϑ izražava se kao:

$$\vartheta(n+1) = \vartheta(n) + \Delta\vartheta(n), \tag{3.20}$$

gdje je n trenutni korak učenja, $\Delta\vartheta(n)$ je veličina promjene parametra ($\vartheta = \mathrm{w}$ za izlazni sloj ili $\vartheta = \mathrm{v}$ za skriveni sloj), a $\vartheta(n+1)$ je nova vrijednost težinskog faktora.

Pogrešku $E(\vartheta)$ moguće je aproksimirati s prva dva člana Taylorovog reda:

$$E(\vartheta + \Delta\vartheta) \approx E(\vartheta) + \Delta E(\vartheta), \tag{3.21}$$

$$\Delta E(\vartheta) = \Delta\vartheta^T \nabla E(\vartheta), \tag{3.22}$$

$$\nabla E(\vartheta) = \frac{\partial E(\vartheta)}{\partial \vartheta}. \tag{3.23}$$

Izraz (3.23) naziva se gradijentom pogreške. Kako bi se pogreška smanjivala najveći mogućim iznosom, treba odrediti $\Delta\vartheta$ za koji promjena pogreške učenja $\Delta E(\vartheta)$ poprima najveći negativni iznos. To se ostvaruje sljedećim uvjetom:

$$\Delta\vartheta = -\eta \nabla E(\vartheta), \tag{3.24}$$

gdje je η mjera te promjena koja se naziva koeficijentom brzine učenja. Koeficijent brzine učenja određuje učitelj, a vrijednost se najčešće kreće između 10^{-3} i 10. Uvrštavanjem izraza (3.24) u (3.20) dobiva se:

$$\vartheta(n+1) = \vartheta(n) - \eta \nabla E(\vartheta(n)), \tag{3.25}$$

što se još naziva i algoritmom najstrmijeg pada ili algoritam povratnog prostiranja pogreške (engl. *error backpropagation algorithm*).

Kako bi se smanjio broj koraka učenja, odnosno broj potrebnih iteracija, uvodi se momentum. Koeficijent momentuma α određuje učitelj a vrijednost mu se kreće između 0,1 i 0,9. Kada se uključi i momentum, izraz (3.25) prelazi u:

$$\vartheta(n+1) = \vartheta(n) - \eta\nabla E(\vartheta(n)) + \alpha\Delta\vartheta(n-1). \tag{3.26}$$

Promjena parametara učenja odvija se izlaznog ka ulaznom sloju. Promjena težinskih faktora između izlaznog i skrivenog sloja odvija se na sljedeći način:

$$\mathbf{w}_{kj}(n+1) = \mathbf{w}_{kj}(n) - \eta\nabla E(n) + \alpha\Delta\mathbf{w}_{kj}(n-1). \tag{3.27}$$

Gradijent pogreške za težine $\mathbf{w}_{kj}$ računa se kao:

$$\nabla E(n) = \frac{\partial E(n)}{\partial \mathbf{w}_{kj}}. \tag{3.28}$$

Zadatak učenja svodi se na određivanje pripadajućeg gradijenta pogreške, što se lako rješava uporabom parcijalnih derivacija:

$$\frac{\partial E(n)}{\partial \mathbf{w}_{kj}} = \frac{\partial E(n)}{\partial O_k}\,\frac{\partial O_k}{\partial net_{Ok}}\,\frac{\partial net_{Ok}}{\mathbf{w}_{kj}}. \tag{3.29}$$

Pomoću izraza (3.16), (3.18) i (3.19) moguće je izračunati svaku pojedinu parcijalnu derivaciju, a one redom glase:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k), \tag{3.30}$$

$$\frac{\partial O_k}{\partial net_{Ok}} = \gamma_k = 1, \tag{3.31}$$

$$\frac{\partial net_{Ok}}{\mathbf{w}_{kj}} = y_j. \tag{3.32}$$

Karakteristična vrijednost algoritma povratnog rasprostiranja pogreške definira se kao:

$$\delta = -\frac{\partial E(n)}{\partial net}, \tag{3.33}$$

a prema (3.29), (3.30) i (3.31) izraz poprima oblik:

$$\delta_{Ok} = d_k - O_k. \tag{3.34}$$

Konačan izraz za gradijent pogreške glasi:

$$\nabla E(n) = \frac{\partial E(n)}{\partial \mathbf{w}_{kj}} = -(d_k - O_k)y_j = -\delta_{Ok}y_j, \tag{3.35}$$

a konačni algoritam promjene težinskih faktora izlaznog sloja:

$$\mathbf{w}_{kj}(n+1) = \mathbf{w}_{kj}(n) + \eta\delta_{Ok}y_j + \alpha\Delta\mathbf{w}_{kj}(n-1). \tag{3.36}$$

Nakon promjene težinskih faktora izlaznog sloja, slijedi promjena težinskih faktora skrivenog sloja $\mathbf{v}_{ij}$. Temeljna jednadžba analogna je izrazu (3.27):

$$\mathbf{v}_{ji}(n+1) = \mathbf{v}_{ji}(n) - \eta\nabla E(n) + \alpha\Delta\mathbf{v}_{ji}(n-1). \tag{3.37}$$

Zadatak se ponovo svodi na računanje gradijenta pogreške koji se računa parcijalnim derivacijama:

$$\frac{\partial E(n)}{\partial \mathbf{v}_{ji}} = \frac{\partial E(n)}{\partial y_j}\,\frac{\partial y_j}{\partial net_{Hj}}\,\frac{\partial net_{Hj}}{\mathbf{v}_{ji}}. \tag{3.38}$$

Na promjenu svake težine skrivenog sloja utječu svi neuroni izlaznog sloja pa prvi razlomak na desnoj strani izraza (3.38) poprima vrijednost:

$$\frac{\partial E(n)}{\partial y_j} = \frac{\partial E(n)}{\partial O_1}\frac{\partial O_1}{\partial net_{O1}}\frac{\partial net_{O1}}{\partial y_j} + \\ + \frac{\partial E(n)}{\partial O_2}\frac{\partial O_2}{\partial net_{O2}}\frac{\partial net_{O2}}{\partial y_j} + \\ + \cdots + \\ + \frac{\partial E(n)}{\partial O_K}\frac{\partial O_K}{\partial net_{OK}}\frac{\partial net_{OK}}{\partial y_j}. \tag{3.39}$$

Pri tome je:

$$\frac{\partial E(n)}{\partial O_k} = -(d_k - O_k), \qquad k = 1,2,\dots,K, \tag{3.40}$$

$$\frac{\partial O_k}{\partial net_{Ok}} = 1, \qquad k = 1,2,\dots,K, \tag{3.41}$$

$$\frac{\partial net_{Ok}}{\partial y_j} = \mathbf{w}_{kj}, \qquad k = 1,2,\dots,K, \qquad j = 1,2,\dots,J-1. \tag{3.42}$$

Uvrštavanjem izraza (3.40), (3.41), (3.42) te (3.34) u izraz (3.39) dobiva se:

$$\frac{\partial E(n)}{\partial y_j} = -\sum_{k=1}^{K}(d_k - O_k)\mathbf{w}_{kj} = -\sum_{k=1}^{K}\delta_{Ok}\mathbf{w}_{kj}. \tag{3.43}$$

Drugi i treći razlomak na desnoj strani izraza (3.38) određuju se na sljedeći način:

$$\frac{\partial y_j}{\partial net_{Hj}} = \gamma_j = \frac{1}{2}\left(1 - y_j^2\right), \tag{3.45}$$

$$\frac{\partial net_{Hj}}{\mathbf{v}_{ji}} = Z_i. \tag{3.46}$$

Konačni oblik algoritma promjene težinskih faktora skrivenog sloja glasi:

$$\mathbf{v}_{ji}(n+1) = \mathbf{v}_{ji}(n) + \frac{1}{2}\eta\left(1 - y_j^2\right)Z_i\left(\sum_{k=1}^{K}\delta_{Ok}\mathbf{w}_{kj}.\right) + \alpha\Delta\mathbf{v}_{ji}(n-1). \tag{3.47}$$

Kada mreža posjeduje više skrivenih slojeva, numerički postupak ostaje ekvivalentan, samo se izrazi malo više proširuju. Nakon učenja mreža s više skrivenih slojeva ima sporiji odziv od mreže koja ima samo jedan skriveni sloj. Ove krajnje jednadžbe vrijede samo za statičku mrežu prikazanu na slici 3.4. U slučaju promjene aktivacijske funkcije, mijenjaju se jedino pripadajuće parcijalne derivacije po funkciji sume neurona, dok svi ostali izvodi parcijalnih derivacija ostaju isti.

3.3.2. Učenje povratne neuronske mreže

Kao i kod unaprijednih neuronskih mreža, kod učenja povratnih koristi se algoritam učenja rasprostiranjem pogreške kako bi se smanjio gradijent pogreške. Uz težinske faktore, parametrima za učenje se dodaju i dinamičke (vremenske) komponente neurona. Promjena svih parametara učenja odvija se u svakom koraku učenja. Učenje je podijeljeno na unaprijednu i povratnu fazu, a detaljni matematički izvodi postupka učenja prikazani su u [7].

4. MARKOVLJEVI LANCI

U sustavima za prepoznavanje govora skriveni Markovljevi lanci koriste se u akustičnom modeliranju za stohastičko modeliranje govora. Stohastički (slučajni) proces je skup slučajnih varijabli:

$$X = \{X_t, t \in T\}, \tag{4.1}$$

gdje je T parametarski skup stohastičkih procesa, a $t \in T$ je parametar. Stohastički proces možemo shvatiti i kao funkciju dviju varijabli, skupa vremena *T* i skupa stanja *S* (skup unutar kojeg proces poprima vrijednosti):

$$X: T \times \Omega \to S. \tag{4.2}$$

Stohastički procesi razlikuju se po prirodi skupa T u dvije grupe: procesi kontinuirani u vremenu i diskretni procesi. Markovljevi lanci, na kojima su bazirani skriveni Markovljevi lanci, proučavaju nizove slučajnih varijabli kod kojih su skup vremena T i skup stanja S diskretni [11].

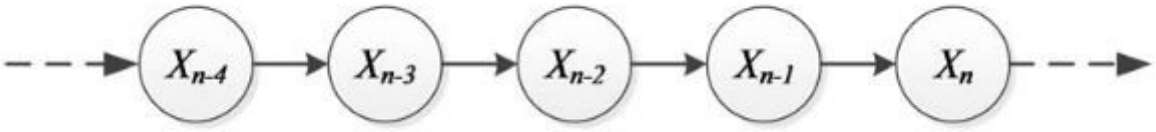
4.1. Markovljevi lanci [11]

Markovljevi lanci predstavljaju niz stanja sustava, a proučavaju stohastičke procese. U svakom trenutku sustav može prijeći u neko novo stanje ili može ostati u istom stanju. Promjene stanja nazivaju se tranzicije. Niz diskretnih slučajnih varijabli X_0, X_1, ... naziva se stohastički lanac. Slučajne varijable uzimaju vrijednosti u konačnom skupu $S = \{s_0, s_1, \dots, s_n\}$.

Lanac $X_0, X_1, \dots$ je Markovljev lanac prvog reda, ako za sve izbore stanja $s_0, s_1, \dots, s_n \in S$ vrijedi:

$$P(X_n = s_n | X_{n-1} = s_{n-1}, \dots, X_0 = s_0) = P(X_n = s_n | X_{n-1} = s_{n-1}). \tag{4.3}$$

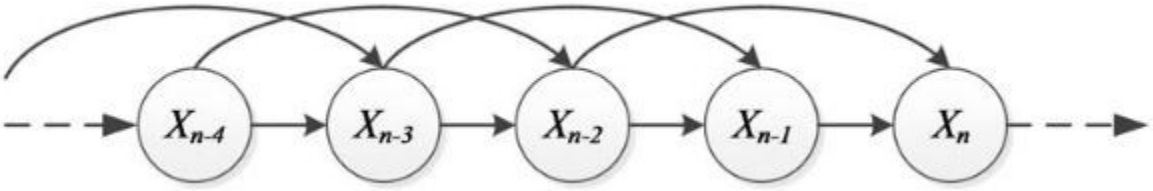
n predstavlja sadašnjost, a $0, \dots, n-1$ prošlost. Sadašnje stanje s_n ovisi samo o prethodnom s_{n-1}, ali ne i o načinu na koji je proces dospio u prethodno stanje tj. vrijednostima procesa u ranijim trenutcima.



Slika 4.2 Markovljev lanac prvog reda [11]

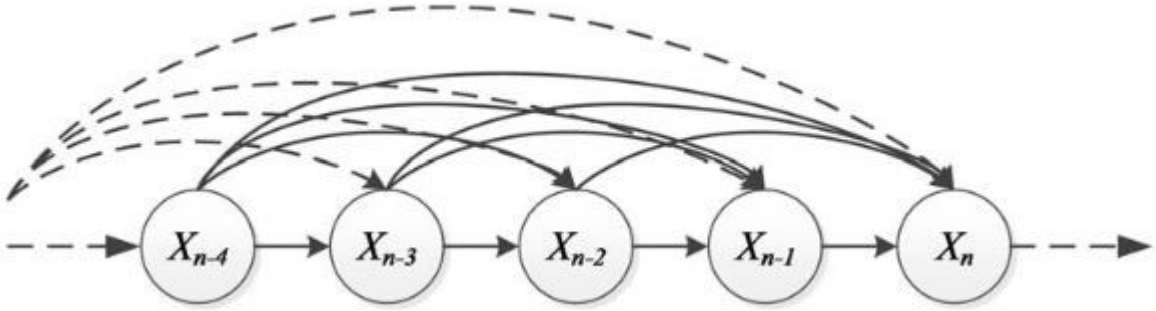
Markovljev lanac drugog reda ovisi o dvama prethodnim stanjima s_{n-1} i s_{n-2} te vrijedi:

$$P(X_n = s_n | X_{n-1} = s_{n-1}, \dots, X_0 = s_0) = P(X_n = s_n | X_{n-1} = s_{n-1}, X_{n-2} = s_{n-2}). \quad (4.4)$$



Slika 4.1 Markovljev lanac drugog reda [11]

Sukladno tome, Markovljev lanac *k*-tog reda ovisi o *k* prethodnih stanja s_{n-1}, s_{n-2}, …, s_{n-k}, a shematski je prikazan na slici 4.3. Viši redovi lanaca daju više informacija o procesu jer ovise o više prethodnih stanja.



Slika 4.3 Markovljev lanac k-tog reda [11]

Markovljeve lance možemo podijeliti na stacionarne i nestacionarne. Za Markovljeve lance koji imaju svojstvo stacionarnosti prijelazne vjerojatnosti ne ovise o koraku, odnosno trenutku.

Veza između slučajnih varijabli X_n i X_{n-1} zadana je prijelaznim vjerojatnostima. Vjerojatnost prijelaza iz stanja s_i u stanje s_j je p_{ij}:

$$p_{ij} = P\left(X_n = s_j \middle| X_{n-1} = s_i\right). \tag{4.5}$$

Matrica s elementima p_{ij} označava se s **P** i naziva se matrica prijelaznih vjerojatnosti:

$$\mathbf{P} = \left(p_{ij}\right) \ i, j \ \in \{1,2, \dots, k\}. \tag{4.6}$$

Matrica prijelaznih vjerojatnosti prvog reda za k - broj stanja:

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1k} \\ p_{21} & p_{22} & \cdots & p_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ p_{k1} & p_{k2} & \cdots & p_{kk} \end{bmatrix}. \tag{4.7}$$

Elementi ove matrice su nenegativni, $p_{ij} \geq 0$, a zbroj elemenata u svakom njezinom retku jednak je jedan:

$$p_{ij} \geq 0, \qquad \sum_{j=1}^{k} p_{ij} = 1 \quad \forall i, j \ \in \{1,2, \dots, k\}. \tag{4.8}$$

4.2. Skriveni Markovljevi modeli

Kod Markovljevih lanaca je proces vidljiv, odnosno stanje sustava je u svakom trenutku poznato. Kada nemamo informacije o stanju sustava tj. kada su neka stanja sustava nevidljiva (skrivena), radi se o skrivenim Markovljevim modelima/lancima (engl. *Hidden Markov Models, HMM*). Promatranjem stanja koje nam je poznato, potrebno je predvidjeti stanje koje je skriveno, a koje nas zanima i nepoznato je. Svako promatrano stanje je određenom vjerojatnošću povezano sa skrivenim procesom tj. skrivenim stanjima, i obrnuto. Drugim riječima, skriveni Markovljevi modeli su dvostruko stohastički proces, sastavljen od skrivenog i vidljivog stohastičkog procesa. U prvom procesu se vrši prijelaz kroz skup stanja, a u drugom se emitiraju (generiraju) vidljiva stanja koja se još nazivaju i opservacije.

Skriveni Markovljevi modeli definiraju se pomoću sljedećih pet parametara.

1) *N* je broj stanja u kojima se proces može nalaziti pa skup svih mogućih stanja glasi:
$$S = \{S_1, S_2, \dots, S_N\}. \tag{4.9}$$
Kod mnogo praktičnih problema postoji jasna veza između interpretacije stanja i njihovog fizičkog značenja. Općenito za stanja u skrivenim Markovljevim modelima vrijedi da se iz svakog stanja može doći u bilo koje drugo stanje. Kod prepoznavanja govora se ograničava tranzicija stanja na model „s lijeva na desno“ (engl. *left-to-right*) što znači da se samo stanje koje je neposredno ispred utječe na trenutno stanje. Ovakav način modeliranja idealan je za prepoznavanje fonema.
2) *M* je broj mogućih opservacija (vidljivih stanja) koje se mogu generirati iz skrivenih stanja. U prepoznavanju govora ovaj parametar najčešće odgovara broju slova u abecedi jezika prema kojem je sustav modeliran. Skup svih opservacija označava se s *V*, a zapisuje se kao:
$$V = \{v_1, v_2, \dots, v_M\}. \tag{4.10}$$
3) **A** je matrica tranzicijskih vjerojatnosti, koja se matematički definira kao:
$$\mathbf{A} = \{a_{ij}\}, a_{ij} = P(q_{t+1} = S_j | q_t = S_i), \ 1 \leq i, j \leq N. \tag{4.11}$$
Stanje modela u nekom vremenskom trenutku *t* označava se s q_t.

4) **B** je matrica emitiranih vjerojatnosti, odnosno vjerojatnosti da se iz nekog skrivenog stanja generira određeno vidljivo stanje. Raspodjela vjerojatnosti generiranja u stanju *j* glasi:

$$\mathbf{B} = \{b_{jk}\} = P(o_t = v_k | q_t = S_j), \qquad 1 \leq j \leq N, \qquad 1 \leq k \leq M. \tag{4.12}$$

5) π je inicijalna raspodjela vjerojatnosti stanja:

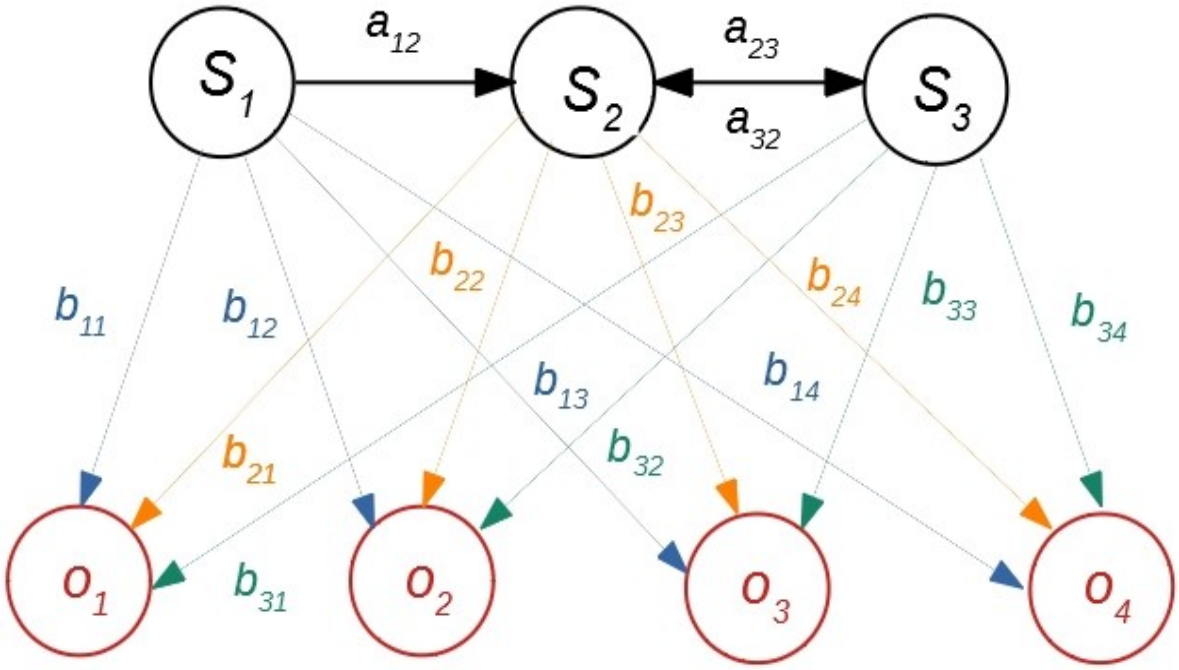
$$\pi = \{\pi_i\} = P(q_1 = S_i), \qquad 1 \leq i \leq N. \tag{4.13}$$

S ovim parametrima je HMM sustav u potpunosti opisan i može generirati niz opservacija duljine T:

$$O = \{o_1, o_2, \dots, o_T\}. \tag{4.14}$$

Postupak generiranja niza opservacija odvija se prema sljedećim koracima [12]:

1) Odabire se početno stanje $q_1 = S_j$ prema inicijalnoj raspodjeli vjerojatnosti π.
2) Vrijeme se postavlja na $t = 1$.
3) Odabire se opservacija $o_t = v_k$ shodno raspodjeli vjerojatnosti pojave simbola V_k u stanju S_j, b_{jk}.
4) Vrši se prelazak u novo stanje $q_{t+1} = S_i$ na osnovu raspodjela tranzicijskih vjerojatnosti a_{ij}.
5) Vrijeme prelazi u $t = t + 1$ te, ako je $t < T$, vraća se na treći korak , a ako je $t = T$, postupak se završava.



Slika 4.4 Shematski prikaz skrivenog Markovljevog modela s tri skrivena stanja i četiri moguće opservacije

Skriveni Markovljevi modeli koriste se u većini modernih sustava zajedno s povratnim neuronskim mrežama, tvoreći tako hibridne sustave. Posebice su poznati za korištenje u zadacima privremenog prepoznavanja uzorka (engl. *temporal pattern recognition*), što im je i zadatak u sustavima za prepoznavanje govora. Šezdesetih i sedamdesetih godina prošlog stoljeća razrađena je matematička teorija iza primjene HMM, a već kasnih sedamdesetih koriste se u ranim sustavima za prepoznavanje govora. Usprkos poznavanju tehnike skrivenih Markovljevih modela, šira uporaba omogućena je tek razvitkom računala s većim procesorskim mogućnostima.

5. PREPOZNAVANJE GOVORA

Računala su sastavni dio svakodnevnog života današnjice, a dolaze u različitim oblicima, od stolnih računala i laptopa do mobitela i kućnih pomagala, kao i industrijskih robota i navigacije na brodovima. Ljudi su stvorili računala kao pomagala, stoga mora postojati interakcija između čovjeka i računala kako bi se zadatak uspješno izvršio. Danas postoji puno načina interakcije čovjeka i računala, bilo dodirom ekrana, tipkovnicom, specijaliziranim upravljačem ili pak govorom. Kao što je rečeno u drugom poglavlju, govor je ljudima instinktivan način komunikacije pa je logično da se razvijaju sustavi bazirani na govoru kao sredstvu komunikacije.

Prepoznavanje govora se jednostavno može objasniti kao pretvorba ljudskog govora u tekst pomoću računala. Ovako jednostavno objašnjenje više odgovara algoritmima čiji je zadatak u osnovi transkripcija. Oni određene zvučne valove povezuju s određenim slovima „bez ikakvog razmišljanja“. Umjetne neuronske mreže mijenjaju problematiku prepoznavanja govora. Uporabom umjetnih neuronskih mreža pri prepoznavanju govora teži se da računala s jednakom lakoćom i točnošću kao čovjek, ako ne i većom, mogu raspoznavati govor. Ondje gdje bi jednostavni algoritmi činili greške, primjerice kod različitih govornika ili velike pozadinske buke, odnosno kod različitih mogućih smetnji, idealno bi umjetna neuronska mreža morala skoro pa besprijekorno izvršavati svoju funkciju. Iz ovoga se vidi da je potrebno proširiti prethodno pojednostavljenje problematike prepoznavanja govora. Prepoznavanje govora pomoću umjetnih neuronskih mreža svodi se na zadatak prepoznavanja istog fonema, bez obzira na bilo kakve moguće smetnje, te povezivanje s njegovim grafemom. Odnosno, osnovni cilj sustava za prepoznavanje govora je točno i efikasno pretvaranje zvučnog signala u tekstualnu poruku, neovisno o uređaju kojim je govor snimljen, naglasku govornika, njegovom psiho-fizičkom stanju, utjecaju telekomunikacijskog kanala, akustičkog okruženja itd.[7] U teoriji, idealni sustav za prepoznavanje govora, koji je osmišljen da oponaša ljudsku percepciju govora, trebao i u potpunosti parirati ljudskim sposobnostima, ako ne biti i bolji. Ovaj cilj još uvijek nije u potpunosti ostvaren te se u razvitku svih budućih sustava radi na njegovom ostvarenju.

Ovisno o svrsi, postoji nekoliko različitih izvedba sustava za prepoznavanje govora što se najbolje može vidjeti kroz povijesni pregled razvitka istih.

5.1. Povijesni pregled sustava za prepoznavanje govora

Ljudska želja za automatiziranjem svakodnevnih zadataka seže više od stotinu godina u prošlost. Isto vrijedi i za ljudsku težnju da svoje primarno sredstvo komunikacije, govor, spoji sa svojim izumima. Već 1881. Alexander Graham Bell sa svojim rođacima stvara prototip uređaja za snimanje zvuka. Razvitak uređaja za snimanje zvuka nastavlja se i u idućem stoljeću kada početkom 20. stoljeća nastaju gramofoni i diktafoni. Uskoro se razvijaju i telefonske linije diljem svijeta, kao i radio veze. Do tada nespojiva mjesta na planetu postaju bliža zahvaljujući tehnologiji koja prenosi zvuk na nevjerojatne udaljenosti. Sada kada je čovječanstvo uspjelo snimiti ljudski govor i tako ga ovjekovječiti, javila se i težnja za mogućim novim uporabama ovih tehnologija.

Kao i kod mnogih znanstvenih otkrića i tehnoloških izuma, početna ideja nastala je u znanstvenoj fantastici. Ideja računala koje prepoznaje ljudski govor i uspješno održava komunikaciju s ljudima popularizirala se romanom Arthura C. Clarkea „2001: Odiseja u svemiru“ te filmskom adaptacijom istog u režiji Stanleyja Kubricka. U romanu inteligentno računalo imena „HAL“ komunicira s putnicima u svemirskom brodu. Prepoznaje i razumije njihov govor i odgovara im smisleno računalno stvorenim glasom koji zvuči skoro pa ljudski. 60ih i 70ih godina prošlog stoljeća, kada su sustavi za prepoznavanje govora bili tek u povojima, postojanje ovakvog „inteligentnog“ računala bila je samo znanstvena fantastika, ali danas postoje sustavi koji su jako blizu ostvarenju te ideje.

Iako su ljudi i prije sanjali o sustavima koji će prepoznavati ljudski govor, intenzivniji razvoj istih počinje tek drugom polovicom prošlog stoljeća. Povod ovom bio je razvitak tehnologije koja je znanstvenicima stvorila više mogućnosti. Ovdje će biti navedeni neki od značajnijih izuma koji su doveli do današnjih sustava za prepoznavanje govora, ali ne i svi.

Rani pokušaji stvaranja sustava za prepoznavanje govora temeljili su se na teoriji akustične fonetike. 1952. Davis, Biddulph i Balashek iz Bell Laboratories u SAD-u razvijaju sustav imena „Audrey“ koji se danas smatra prvim uspješnim sustavom za prepoznavanje govora. Sustav „Audrey“ bio je u potpunosti analogan, a služio je za izolirano prepoznavanje znamenki za jednog govornika. Prepoznavanje se baziralo na mjerenju frekvencije određenog fonema u izgovoru svake znamenke. Točnost sustava bila je i do 99% ako je sustav bio prilagođen govorniku.

Kako je bila riječ o predodređenoj konstantnoj vrijednosti frekvencije za svaku znamenku, sustav je ovakvu točnost imao samo kada je govornik bio muškarac određene dobi i kada je znamenke izgovarao s naglašenom pauzom između.

60ih godina prošlog stoljeća, Sakai i Doshita na Kyoto Sveučilištu u Japanu osmislili su prvi sustav koji je koristio segmente govora za analizu i prepoznavanje govora iz različitih dijelova ulaznih snimki. Za usporedbu, sustav „Audrey" je pretpostavljao da ulazna snimka govora sadrži u potpunosti izgovorenu znamenku te stoga nije bilo potrebno podijeliti zapis u segmente. Rad Sakaija i Doshita može se smatrati pretečom sustava za prepoznavanje kontinuiranog govora.

U isto vrijeme IBM predstavlja „Shoebox" sustav koji je prepoznavao 16 izgovorenih riječi engleskog jezika te znamenke od 0 do 9. Takav rječnik omogućavao je govorniku da riječima zadaje matematičke operacije sustavu. „Shoebox" je kao i „Audrey" bio u potpunosti analogan, a za prikaz prepoznatih riječi koristili su se svjetleći signali.

Kasnih 60ih Atal i Ikatura neovisno stvaraju osnovne koncepte linearno prediktivnog kodiranja (engl. *Linear Predictive Coding, LPC*) koji je iznimno pojednostavio procjenu fonema prema valnom zapisu snimljenog govora. Do sredine 70ih godina Itakura, Rabiner, Levinson i drugi predložili su implementaciju ideje prepoznavanja uzorka bazirane na LPC-u u sustavima za prepoznavanje govora.

Istovremeno dolazi do porasta broja istraživanja na području prepoznavanja govora zbog ulaganja Ministarstva obrane SAD-a, skraćeno DARPA. Program DARPA-e bio je jedan od najvećih programa istraživanja prepoznavanja govora u povijesti, a trajao je do kraja 70ih godina prošlog stoljeća. Konačni produkt višegodišnjeg istraživanja bio je sustav imena „Harpy" s rječnikom 1011 riječi. Doprinos „Harpy" sustava bila je uporaba efikasnog algoritma za pretraživanje govora, engl. *beam search algorithm*. Ulazna snimka govora se nakon analize značajki dijelila na segmente koji su se uspoređivali s uzorcima fonema.

80ih godina javljaju se dva smjera razvoja sustava za prepoznavanje govora, jedan od strane Freda Jelineka iz IBM-a, a drugi u AT&T Bell Laboratories. IBM je želio stvoriti sustav koji će pisati transkripte snimljenog govora (engl. *Voice Activated Typewriter, VAT*).

Ovaj sustav, nazvan „Tangora“, ovisio je o govorniku i morao je biti prilagođen za njega. Naglasak je bio na veličini rječnika, cilj je bio da bude što je veći mogući te da mu primarna svrha bude uredska korespondencija. U sustavu „Tangora“ po prvi put je korišten jezični model koji je koristio *N*-grame koji su definirali vjerojatnost pojave određenog niza riječi duljine *N*. Od tada pa sve do danas, ovaj tip jezičnih modela nezamjenjiv je te se pojavljuje u velikoj većini sustava za prepoznavanje govora.

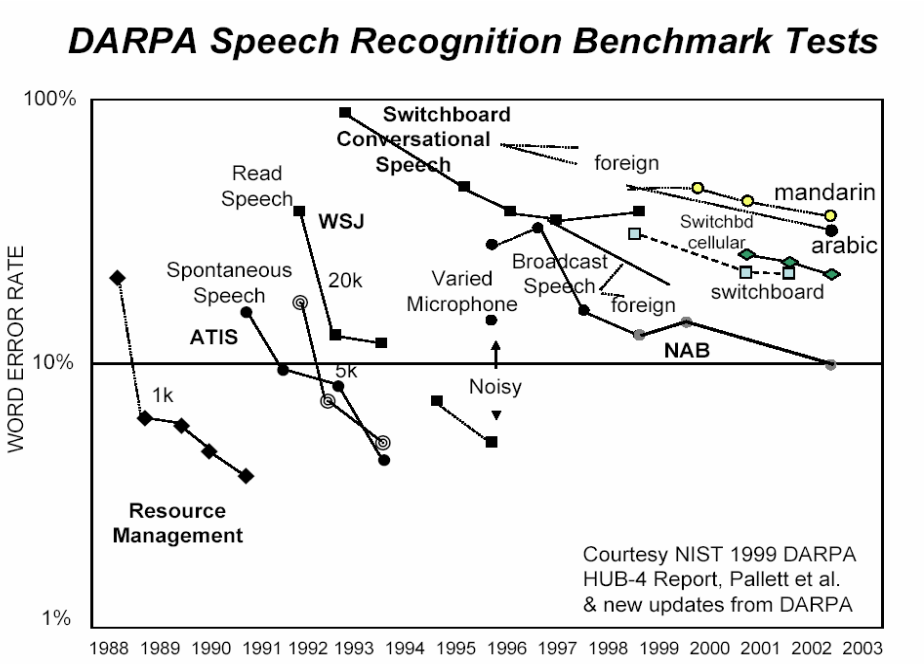
U AT&T Bell Laboratories cilj je bio stvoriti automatske telekomunikacije koje će korisniku omogućiti upravljanje telefonskim uređajem pomoću glasovnih naredbi. Težnja je bila stvoriti sustav koji će biti neovisan o govorniku i koji će jednako precizno raditi na desecima milijuna ljudi bez obzira na dijalekte. Stvoren je akustični model koji je koristio algoritme za spektralnu i statističku analizu. Također, po prvi put se koristilo zapažanje ključnih riječi (engl. *keyword spotting*) kao primitivni oblik razumijevanja govora.

Istraživanja sustava za prepoznavanje govora u IBM-u i u AT&T Bell Laboratories dovela su do velikih pomaka u tom području. Iako su krenuli s različitim ciljevima i razvili različite tehnologije, one su se na kraju spajale u jedno zbog istovremenog brzog razvoja statističkih metoda, posebice skrivenih Markovljevih modela. Dogodio se prijelaz sa metode običnog prepoznavanja uzorka na statističko modeliranje. Ideja za korištenje skrivenih Markovljevih modela javila se već kasnih 60ih godina, no tek sredinom i krajem 80ih oni postaju sastavni dio sustava za prepoznavanje govora.

Krajem 80ih još jedna tehnologija ulazi na područje prepoznavanja govora, umjetne neuronske mreže. Prvi pokušaj korištenja umjetnih neuronskih mreža u prepoznavanju govora dogodio se već na početku, 50ih godina prošlog stoljeća, ali rezultati ovih pokusa nisu ostvarili nikakve bitne rezultate. Uporaba višeslojnih perceptrona i algoritma učenja s povratnim rasprostiranjem pogreške oživjela je ideju oponašanja rada ljudskog mozga. Isprva su umjetne neuronske mreže korištene za prepoznavanje par izoliranih fonema ili par izoliranih riječi i tu su pokazale iznimne rezultate, no nisu uspjele svladati problem vremenskih varijacija koji je neizbježan kod prepoznavanja govora.

Kao reakcija na ovaj problem, javila su se istraživanja koja su kombinirala uspješnost umjetnih neuronskih mreža s mogućnošću skrivenih Markovljevih modela da dobro rukuju s vremenskim varijacijama.

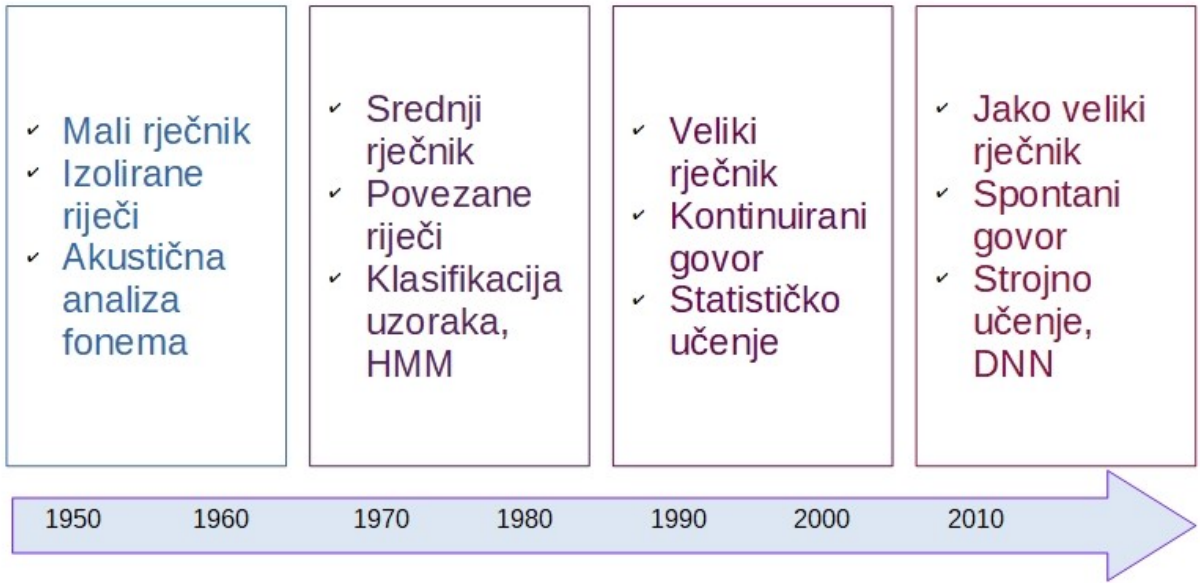
Uspješnost statističkih metoda prepoznavanja govora je ponovo probudila interes DARPA-e za ovo područje, što je dovelo do razvoja novih sustava za prepoznavanje govora na prijelazu 80ih na 90e godine. Neki od tih sustava bili su: BBN-ov „BYBLOS“, SRI-jev „DECIPHER“ te CMU-ov „Sphinx“. „Sphinx“ je uspješno integrirao statističke metode skrivenih Markovljevih modela s algoritmom pretraživanja grafova sustava „Harpy“ te je bio sposoban raspoznati foneme prema kontekstu što je rezultiralo vrhunskim rezultatima u prepoznavanju kontinuiranog govora uporabom velikih rječnika. Istraživanja DARPA-e su također razvila metode evaluacije sustava za prepoznavanje govora, kao što su mjerenje pogreške riječi i rečenica.



Slika 5.1 Grafički prikaz ključnih napredaka u istraživanjima DARPA-*e na više različitih područja zanimanja prepoznavanja govora [14]*

90ih godina napretkom softvera, razvilo se puno individualnih sustava za prepoznavanje govora diljem svijeta. Sustavi su postali sofisticiraniji s velikim rječnicima i milijunima parametara za prepoznavanje fonema. Tim vođen Steve Youngom je na Sveučilištu Cambridge razvio softver koji je koristio skrivene Markovljeve modele (engl. *Hidden Markov Model Tool Kit, HTK*) koji je i danas u širokoj upotrebi.

Ranih 90ih predstavljeni su sustavi koji su težili imitiranju prave ljudske komunikacije. Sustavi „Pegasus“ i „Jupiter“ nastali na Massachusetts Institute of Technology pod vodstvom Victor Zuea primjer su ranih pokušaja takvih sustava. „Pegasus“ je bio sustav govorne komunikacije koji je izdavao informacije o zračnim letovima preko telefonske veze, a „Jupiter“ je bio sličan sustav koji je radio isto za vremenske uvjete. Koristili su prepoznavanje određenih fraza u izgovorenom govoru, primjerice „Vrijeme u New Yorku danas navečer“. Kasnih 90ih sustavi za upravljanje aplikacijama i izdavanje informacija korisnicima napredovali su i ušli u sve širu primjenu.



Slika 5.2 Vremenska linija razvitka sustava za prepoznavanje govora

2000ih godina DARPA nastavlja s istraživanjima, a Agencija za nacionalnu sigurnost SAD-a (engl. *National Security Agency, NSA*) počinje koristiti zapažanje ključnih riječi prilikom pretraživanja velikih količina snimaka govora što im omoguće indeksiranje i lako pronalaženje određenih razgovora. Područjem prepoznavanja govora i dalje dominiraju skriveni Markovljevi modeli u kombinaciji s unaprijednim neuronskim mrežama. Oko 2007. dominantnu ulogu preuzimaju povratne LSTM neuronske mreže. 2009. je uporaba dubokih unaprijednih neuronskih mreža dovela do velikog porasta točnosti sustava.

Posljednjeg desetljeća došlo je do porasta komercijalnih sustava za prepoznavanje govora. 2011. Apple je pustio u komercijalnu upotrebu jedan od najpoznatijih sustava virtualnog asistenta, „Siri“, koji koristi prepoznavanje govora. 2014. Microsoft je predstavio svoju inačicu virtualnog asistenta imena „Cortana“. Iste godine nastala je i „Amazon Alexa“. Google je ostvario veliki napredak u prepoznavanju govora korištenjem CTC-LSTM neuronskih mreža (engl. *Connectionist Temporal Classification, CTC*) koje se i trenutno koriste u „Google Voice“ sustavu. Skoro svi aktualni sustavi su sustavi za automatsko prepoznavanje govora (engl. *automatic speech recognition, ASR*) koji vrše svoj zadatak u stvarnom vremenu. Sustavi za automatsko prepoznavanje govora i dalje se kontinuirano poboljšavaju u čemu im puno pomaže sve veći broj korisnika koji komuniciranjem s uređajem stvaraju sve veće baze snimljenog govora prema kojem se nove inačice sustava mogu učiti.

5.2. Kreiranje sustava za prepoznavanje govora

5.2.1. Definiranje zadatka

Prvo je prilikom kreiranja sustava za prepoznavanje govora potrebno odabrati odgovarajuća obilježja govora koja će se analizirati. Pritom se koristi kombinacija akustičnih, govornih i slušnih obilježja kako bi se značajke govora spektralno modelirale uzimajući u obzir ljudsku percepciju govora. Potom je potrebno definirati zadatak prepoznavanja. Prepoznavanje govora može biti relativno jednostavno, primjerice kada sustav mora prepoznati svega nekolicinu riječi. Takvi sustavi koriste se u okruženjima gdje govornik koristi određene fraze, odnosno naredbe. Također, sustavi za prepoznavanje govora mogu biti i jako kompleksni, posebice kada im je cilj u potpunosti imitirati ljudsku komunikaciju. Kompleksni sustavi moraju biti robusni, odnosno otporni na veliki broj mogućih smetnji. Prilikom definiranja zadatka sustava ključno je odrediti veličinu rječnika. S obzirom na broj riječi u rječnicima koje prepoznaju, sustavi za prepoznavanje govora dijele se na male, srednje i velike. Mali raspolažu rječnicima do nekoliko stotina riječi, srednji do nekoliko tisuća, a veliki do nekoliko stotina tisuća. Iako su sustavi s malim rječnicima najjednostavniji za modeliranje, u slučaju da se u podacima za učenje nalazi puno sličnih riječi, loši su u prepoznavanju novih riječi kada završi učenje.

5.2.2. Odnos govornika/govora i sustava

Nakon definiranja veličine rječnika sustava, potrebno je odrediti odnos govornika i sustava te govora i sustava. Sustav može biti ovisan ili neovisan o govorniku. Sustavi ovisni o govorniku su stvoreni i prilagođeni govoru pojedinca, dok neovisni sustavi ne ovise o osobi koja govori. Uspoređujući uspješnost ovisnih i neovisnih sustava, sustavi ovisni o govorniku su točniji, odnosno imaju veći uspjeh u prepoznavanju riječi. Ali mana im je što ta točnost vrijedi samo za govornika prema kojem je sustav prilagođen. Postoje i sustavi koji podržavaju više govornika (engl. *multi-speaker systems*). Danas se intenzivno radi na sustavima s više istovremenih govornika, ali još nisu u potpunosti usavršeni.

S obzirom na odnos samog govora i sustava, snimljeni govor može biti izoliran, diskontinuiran i kontinuiran. Izolirani govor znači da je izgovor svake pojedine riječi snimljen odvojeno. Snimka diskontinuiranog govora predstavlja izgovorene riječi s pravilnim pauzama između riječi i interpunkcija. Kod kontinuiranog govora brzina izgovora, pauze i naglasci ovise o govorniku i trenutku u kojem je izgovoren. Izolirani i diskontinuirani govor lakši su za modeliranje zbog jasno izraženih granica između riječi. Prilikom učenja sustava koriste se snimke čitanog ili spontanog govora. Snimke čitanog govora u pravilu nastaju u studijima gdje određeni tekst čitaju profesionalci koji pravilno naglašuju riječi i pauze u danim tekstovima. Spontani govor pak sadrži prekide, uzdahe, iznenadne pauze, nedovršene rečenice, kašalj, smijeh itd. Drugim riječima, to je normalan svakodnevni govor. Sve te smetnje čine ga kompleksnim za modeliranje i kompliciranim za prepoznavanje. Stoga je često potrebna velika i kvalitetna baza snimljenog spontanog govora za učenje sustava, što nije lako sakupiti. Najčešće se u tu svrhu koriste zvučne snimke s radija i televizije. Bez obzira na kompleksnost, većina današnjih sustava bazirana je na spontanom govoru jer im je i krajnji cilj prepoznavanje istog.

5.2.3. Jezično i akustično modeliranje

Sljedeći korak je jezično modeliranje govornih jedinica. Gleda se složenost, odnosno veličina govornih jedinica te njihova osjetljivost s obzirom na kontekst. Prema složenosti, govorne jedinice se dijele na: riječi, slogove, polu-slogove, trifone, generalizirane trifone, difone, monofone, senone i podglasove. Složene govorne jedinice poput riječi i slogova su osjetljivije na kontekst. S porastom složenosti govornih jedinica, učenje sustava je teže, a prepoznavanje lošije, prvenstveno zbog manjeg broja uzoraka koji se koriste prilikom učenja. Općenito vrijedi da su sustavi osjetljiviji na kontekst točniji u prepoznavanju, ali jedino ako su pravilno učeni. Iz ovoga se može zaključiti da modeliranje na bazi riječi i slogova samo ima smisla u sustavima s malim rječnicima, čime im se osigurava velike točnost prepoznavanja. U slučaju velikih rječnika koriste se jednostavnije govorne jedinice poput trifona i monofona.

Uz jezično modeliranje, potrebno je i akustično modeliranje. Postoje dva tipa, modeliranje obrazaca (engl. *Template-based models*) i statističko modeliranje (engl. *Statistical-based models*). Kod sustava čiji su akustični modeli bazirani na obrascima, prepoznavanje se svodi na uspoređivanje nepoznate snimke govora s prethodno snimljenim riječima s ciljem pronalaženja najsličnijih parova riječi. Korištenje savršenih modela riječi je prednost i mana ovih sustava. Obrasci su nepromjenjivi i loše prepoznaju bilo kakve varijacije u govoru. Jedno od mogućih rješenja ovog problema je korištenje velikog broja snimki govora u kojima se pojavljuje puno varijacija izgovora riječi. Sustavi s akustičnim modelom baziranim na statističkom modeliranju koriste statističke varijacije u govoru. Prilikom učenja, za svaki se model stvara statistička raspodjela na osnovi prethodnih snimaka pomoću koje se opisuju različita stanja sustava. Primjer ovakvih sustava su sustavi koji koriste skrivene Markovljeve modele u akustičnom modelu, te su takvi sustavi danas najzastupljeniji.

I jezični i akustični model moraju proći odgovarajući proces učenja. Učenje se provodi iteracijama dok sustav ne postigne željene performanse, nakon čega se sustav testira. Učenje jezičnih modela zahtjeva veliku količinu teksta prema kojoj sustav uči pravila jezika, gramatiku i sintagmu.

5.2.4. Ocjenjivanje sustava

Posljednji korak stvaranja sustava za prepoznavanje govora je njegovo testiranje i evaluacija. Sustavi za prepoznavanje govora ocjenjuju se s obzirom na uspjeh prepoznavanja riječi ili s obzirom na značaj. S obzirom na uspjeh prepoznavanja riječi, gleda se postotak pogrešno prepoznatih riječi (engl. *word error rate, WER*). Postoje tri vrste grešaka:

1. greške supstitucije (engl. *substitution*) – kada sustav pogrešno prepozna jednu riječ kao neku drugu riječ,
2. greške brisanja (engl. *deletion*) – kada sustav ne naslućuje nikakvu riječ, a ona postoji na snimci i u transkriptu govora te
3. greške umetanja (engl. *insertion*) – kada sustav doda riječ koja ne postoji na snimci i transkriptu govora.

Izraz za postotak pogrešno prepoznatih riječi glasi:

$$\text{WER} = \frac{N_{sub} + N_{del} + N_{ins}}{N_{ref}}, \tag{5.1}$$

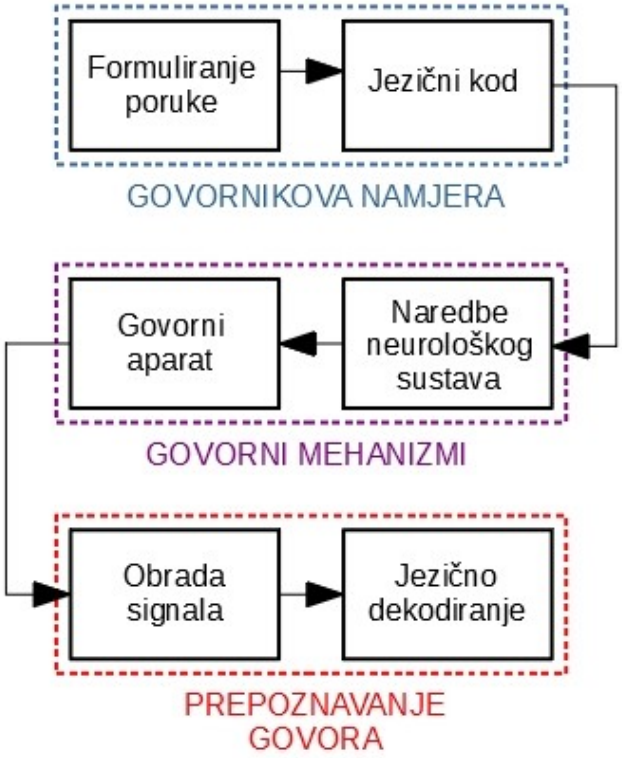
gdje je N_{sub} broj grešaka supstitucije, N_{del} broj grešaka brisanja, N_{ins} broj grešaka umetanja, N_{ref} broj riječi u referentnom transkriptu govora. Ponekad se sve tri vrste grešaka ne smatraju jednako bitnima pa se izraz (5.1) modificira određenim faktorima. U pravilu se WER računa za svaku rečenicu pojedinačno, a za ukupni WER cijelog govora zbrajaju se pojedinačni za rečenice i dijele s brojem rečenica. Uz postotak pogrešnih riječi, može se gledati i postotak pogrešnih rečenica (engl. *sentence error rate, SER*). SER metoda gleda rečenicu po rečenicu te je ocjenjuje s točna ili netočna. Ova metoda se rijetko upotrebljava zbog svoje nepreciznosti.

Druga metoda ocjenjivanja performansi sustava za prepoznavanje govora je testiranje statističkog značaja (engl. *significance testing*). Prilikom ovog testiranja mjeri se razlika između dva eksperimenta ili dva algoritma te se ispituje je li pogreška posljedica samog algoritma, varijabilnih podataka, eksperimentalnog okruženja ili nekih drugih faktora. Najčešće se koristi metoda MAPSSWE (engl. *Matched Pairs Sentence-Segment Word Error*). Prilikom testiranja MAPSSWE metodom setovi za testiranje se odvajaju u segmente pod pretpostavkom da su segmenti međusobno statistički neovisni, što je pogodno za standardna testiranja u kojima se testne izjave daju sustavu za prepoznavanje jedna po jedna.

Osim točnosti, sustavi za prepoznavanje govora ocjenjuju se i prema brzini i latentnosti. Brzina prepoznavanja se mjeri s obzirom na stvarno vrijeme, a označava se s faktorom stvarnog vremena (engl. *real-time factor, RTF*). Ako je RTF = 1, onda se radi o prepoznavanju u stvarnom vremenu, odnosno za 10 sekundi govora potrebno je 10 sekundi procesiranja. Ako je RTF > 1, onda sustavu treba više vremena za procesiranje no što je duljina snimljenog govora. Ovakvi sustavi su korisni kada je točnost bitnija od brzine. Ako je RTF < 1, onda predviđa riječi brže no što ulazni podaci dolaze. To je korisno kada je pokrenuto više sustava na jednom računalu jer je onda moguće procesirati više zvučnih zapisa istovremeno. Ovakvi sustavi mogu „uhvatiti“ stvarno vrijeme, odnosno sakriti svoju latentnost iza svoje brzine.

5.3. Građa sustava za prepoznavanje govora

Jednostavniji model sustava za prepoznavanje govora sastoji se od tri osnovna dijela, govornikove namjere, govornih mehanizama i prepoznavanja govora. Prikazan je na slici 5.3. Govornikove namjere i govorni mehanizmi objašnjeni su s lingvističkog i biološkog aspekta u drugom poglavlju. Prepoznavanje govora je glavni zadatak algoritama, a sastoji se od obrade zvučnog signala i jezičnog dekodiranja. Govor putuje od govornika u obliku zvučnog signala koji je potrebno obraditi i izdvojiti značajke govora. Jezično dekodiranje u pravilu koristi statističku procjenu vjerojatnosti niza riječi od kojih bi se mogao sastojati snimljeni govor. Većina sustava za prepoznavanje govora osmišljena je za određeni jezik te u pravilu neće raditi s jednakom točnošću ako je snimljeni govor na nekom drugom jeziku. Jezično dekodiranje je dio sustava za prepoznavanje govora gdje se prvenstveno koriste umjetne neuronske mreže.



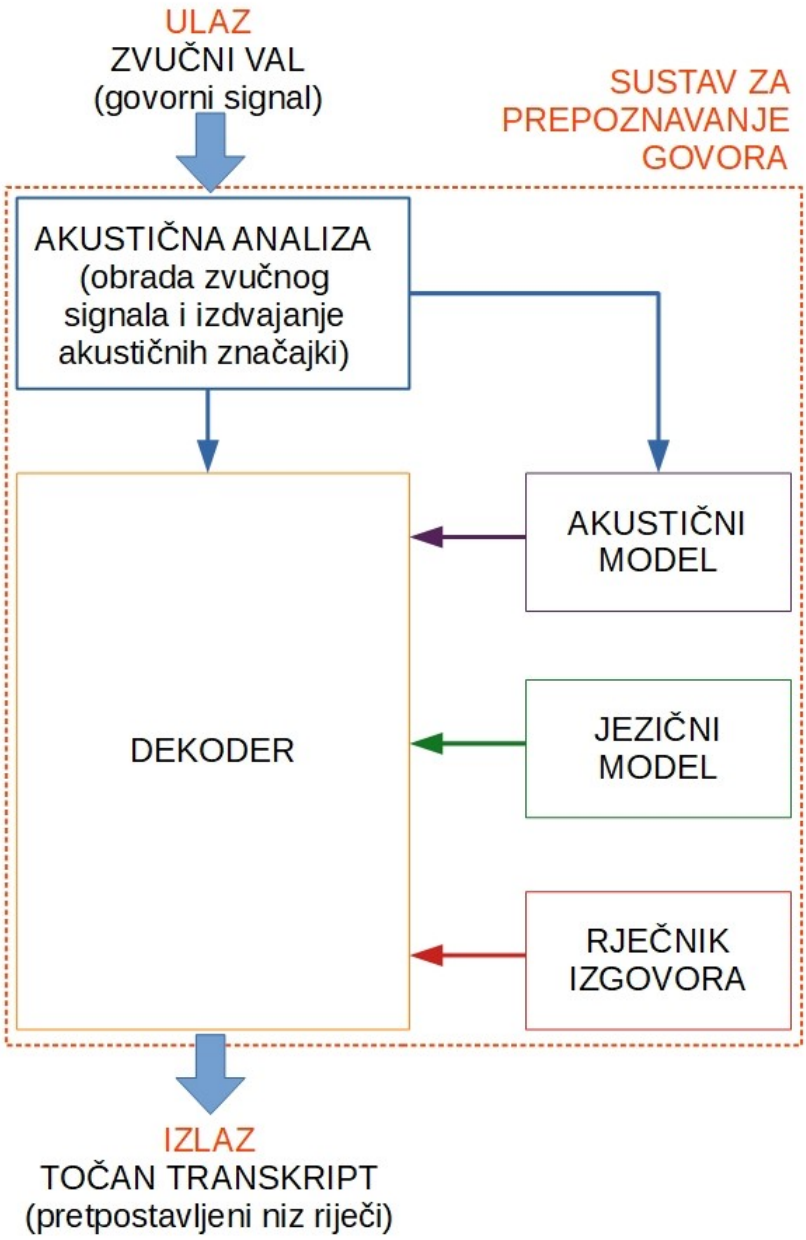
Slika 5.3 Osnovni model prepoznavanja govora

Detaljan prikaz konvencionalnog sustava za prepoznavanje govora prikazan je na slici 5.4. Zvučni signal govora ulazi u sustav te se prvo obrađuje. Obrada zvučnog signala sastoji se od segmentiranja signala i izdvajanja akustičkih značajki pomoću spektralne analize. Akustične značajke označuju se s *x* pa se onda zvučni zapis govora može zapisati kao niz značajki:

$$X = \{x_1, x_2, \dots, x_t\}, \tag{5.1}$$

gdje je *t* broj značajki. Ovako zapisani govorni signal dolazi do bloka za klasifikaciju obrazaca gdje se vrši uspoređivanje ulaznog signala s postojećim obrascima kako bi se pronašao najvjerojatniji par. Prilikom uspoređivanja, sustav se koristi akustičnim modelom, jezičnim modelom i rječnikom izgovora. Akustični model služi za usporedbu fonema, jezični model koristi se za provjeru pravila jezika (gramatika i sintagme), a rječnik izgovora daje vezu između niza fonema i pisane riječi.

Nakon što dekoder sustava odabere najvjerojatniji niz riječi s obzirom na sve modele (akustični, jezični i rječnik izgovora), dobiveni niz riječi se ocjenjuje pa potvrđuje ili ispravlja po potrebi. Izlaz sustava je niz potvrđenih prepoznatih riječi.



Slika 5.4 Konvencionalan model sustava za prepoznavanje govora

5.3.1. Osnova jednadžba sustava za prepoznavanje govora

Matematički se problem prepoznavanja govora može formulirati kao problem statističkog donošenja odluka. Ovaj problem se svodi na traženje Bayesove maksimalne *a posteriori* vjerojatnosti (engl. *Maximum A posteriori Probability, MAP*) koja se odnosi na niz riječi $\hat{R}$ koji najviše odgovara nizu akustičnih značajki *X*. Niz riječi nepoznate duljine *N* zapisuje se kao:

$$R = \{r_1, r_2, \dots, r_N\}. \tag{5.2}$$

Potrebno je pronaći maksimum *a posteriori* vjerojatnosti P(*R*|*X*), odnosno niz riječi koji najvjerojatnije odgovara nizu akustičnih značajki dobivenih prilikom akustične analize:

$$\hat{R} = \underset{R}{\operatorname{argmax}}\, \mathrm{P}(R|X). \tag{5.3}$$

Koristi se Bayesovo pravilo za rješavanje pa vrijedi:

$$\mathrm{P}(R|X) = \frac{\mathrm{P}(X|R)\mathrm{P}(R)}{\mathrm{P}(X)}. \tag{5.4}$$

Sve komponente izraza (5.4) definiraju se odgovarajućim raspodjelama vjerojatnosti koje se dobivaju učenjem akustičnog i jezičnog modela. P(*X*|*R*) je *a posterori* vjerojatnost koja opisuje akustični model sustava za prepoznavanje govora. P(*R*) je *a priori* vjerojatnost koja predstavlja vjerojatnost da se određene riječi nađu zajedno u uređenom nizu *R,* a opisuje jezični model sustava za prepoznavanje govora. *A priori* vjerojatnost P(*X*) ne utječe na niz riječi *R* pa se može zanemariti. Tada izraz (5.3) prelazi u:

$$\hat{R} = \underset{R}{\operatorname{argmax}}\, \mathrm{P}(X|R)\mathrm{P}(R), \tag{5.5}$$

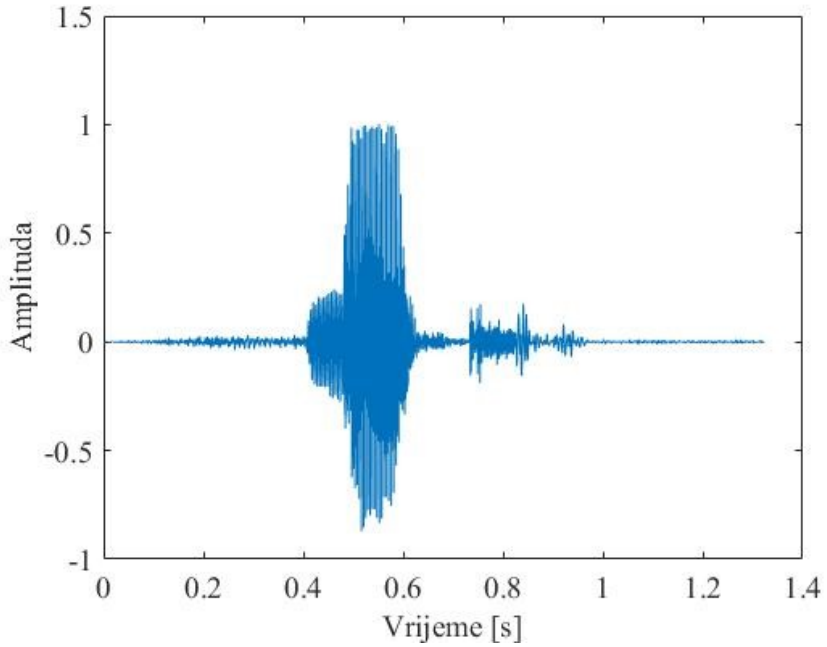
što se naziva osnovnom jednadžbom prepoznavanja govora.

5.3.2. Obrada zvučnog signala i izdvajanje akustičnih značajki

Prvi korak u prepoznavanju govora je pretvorba govora u oblik s kojim računalo može raditi, odnosno A/D pretvorba zvučnog signala. U trenutku kada je govor izgovoren, on postaje zvučni val i kao takav dolazi u kontakt sa sustavom za prepoznavanje govora. Zvučni signal potrebno je obraditi i izdvojiti značajke govora, odnosno provesti akustičnu analizu. Akustična analiza sastoji se od sljedećih koraka:

1. predobrada signala;
2. uzorkovanje signala u niz preklapajućih segmenata;
3. za svaki segment potrebno je:
    a. upotrijebiti Hammingovu funkciju na rubove signala,
    b. izračunati Fourierovu transformaciju,
    c. izračunati magnitudu (apsolutnu amplitudu) spektra,
    d. upotrijebiti Mel filtraciju;
4. primjena logaritamske operacije;
5. te ako je potrebno, normaliziranje značajki nakon filtriranja.

Govor nošen zvučnim valovima putuje kroz zrak sve do mikrofona. Svrha mikrofona je da pretvori akustične vibracije u obliku tlaka zraka u električnu energiju koja se može analizirati. Ulazni govorni zvučni signal se digitalizira određenom frekvencijom uzorkovanja. Govorni zvučni valovi imaju frekvenciju do 8000 Hz pa se najčešće koristi uzorkovanje od 16000 Hz, ali moguće je koristiti frekvencije u rasponu od 8000 Hz do 22050 Hz. Primjerice, govor preko tipičnih telefonskih linija ograničen je na 3400 Hz pa se u tom slučaju koristi uzorkovanje od 8000 Hz. Na slici 5.5 prikazan je valni zapis riječi „bok“ u mom izgovoru, a snimljen je pomoću aplikacije za snimanje govora na smartphone uređaju Samsung A5 te potom obrađen u programskom jeziku MATLAB.



Slika 5.5 Valni zapis riječi „bok“ prikazan pomoću programskog jezika MATLAB

Govor nije stacionaran signal kao što se vidi iz njegovog zapisa u obliku zvučnih valova. To znači da se njegova svojstva mijenjaju u vremenu. Stoga, kako bi se govorni signal točno analizirao, potrebno ga je podijeliti na manje dijelove, segmente signala, u kojima se može pretpostaviti da je signal stacionaran. U pravilu se za prepoznavanje govora koriste segmenti signala u trajanju od 25 ms s međusobnim preklapanjem od 10 ms, što znači da je jedna sekunda signala prikazana sa 100 segmenata. Zbog toga što su segmenti dio signala, potrebno je na njihove rubove primijeniti neku prozorsku funkciju (engl. *window function*). Prozorska funkcija osigurava da ne dođe do diskontinuiteta signala između susjednih segmenata. Najčešće se koristi Hamming funkcija koja glasi:

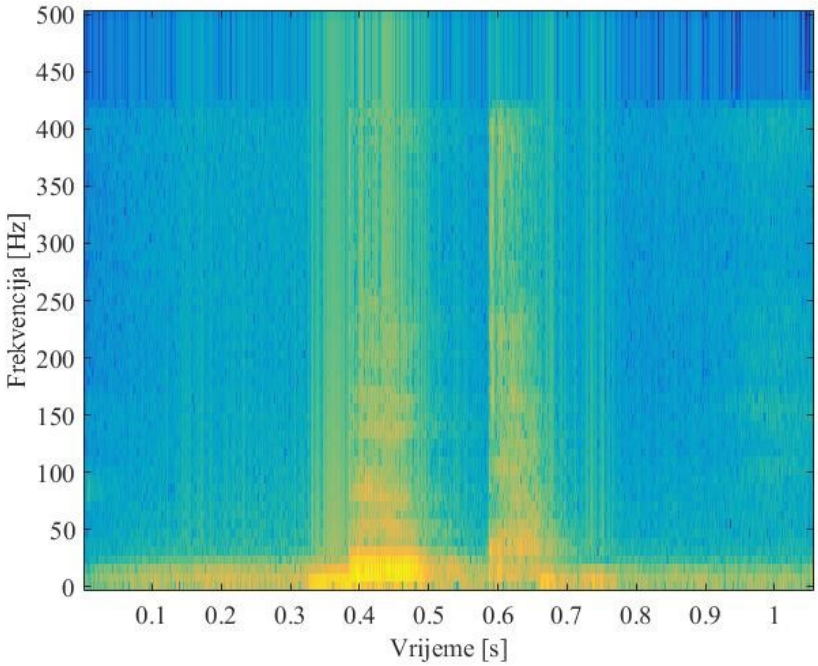
$$w(n) = 0{,}54 - 0{,}46\cos\left(2\pi\frac{n}{N}\right), \qquad 0 \leq n \leq N, \tag{5.6}$$

gdje je N broj segmenata signala.

Za svaki segment potrebno je izračunati Fourierovu transformaciju. Koristi se vremenski kratka Fourierova transformacija (engl. *short time Fourier transform, STFT*) koja je u većini današnjih sustava za prepoznavanje govora algoritamski implementirana kao brza Fourierova transformacija (engl. *fast Fourier transform*, *FFT*). Svaki uzorak signala se prebacuje u frekvencijsku domenu vremenski kratkom Fourierovom transformacijom na sljedeći način:

$$X_m(f) = \sum_{n=-\infty}^{\infty} x(n)w(n-m)\, e^{-j2\pi fn}, \tag{5.7}$$

gdje je $x(n)$ predstavlja zvučni signal, *w(n)* neku prozorsku funkciju, a *m* je indeks segmenta. Apsolutna vrijednost Fourierove transformacije $X_m(f)$ koristi se za spektralni prikaz signala u obliku 2D spektrograma. Horizontalna os predstavlja indeks segmenta signala izražen u 10 ms, a vertikalna os frekvenciju od 0 Hz do Nyquistove frekvencije, odnosno polovične vrijednosti signala frekvencije uzorkovanja. Na slici 5.6 vidi se primjer spektrograma za riječ „bok“. Dijelovi s visokom energijom prikazani su narančasto i crveno, a dijelovi s niskom energijom zeleno i plavo.



Slika 5.6 Spektrogram za zvučni zapis riječi „bok“ prikazan pomoću programskog jezika MATLAB

Kako bi se maknuo utjecaj harmonične strukture u govoru, kao i utjecaj nasumične buke u dijelovima bez govora, koriste se posebne operacije izglađivanja prema spektru magnituda. Najčešće se koristi Mel skup filtera (engl. *mel filterbank*). Primijenjeni u otprilike logaritamskoj skali na os s frekvencijama, ti filtri postaju širi i međusobno udaljeniji s povećanjem frekvencije. Mel filtri mogu se zapisati u obliku matrice, gdje svaki red predstavlja jedan filter.

Posljednji korak izdvajanja značajki signala je primjena logaritamske operacije. Ona sažima dinamički raspon signala, a kao izlaz daje Mel frekvencijske kepstralne koeficijente (engl. *Mel-frequency cepstral coefficients - MFCC*), koji pomoću nelinearne Mel-frekvencijske skale aproksimiraju karakteristike ljudskog slušnog sustava. Kako se najčešće koriste MFCC raspona 40, izlazni spektrogram nakon logaritmiranja prikazuje koeficijente filtriranja u obliku 40-dimenzionalnog skupa filtera. U usporedbi s originalnim spektrogramom, spektrogram koeficijenata filtriranja je puno glađi duž frekvencijske osi jer su uklonjeni šumovi kao i harmoničnost. Uz MFCC postoje još dva tipa kepstralnih koeficijenata koji se mogu koristiti u akustičnoj analizi, TECC (engl. *Teager-Energy Cepstral Coefficients*) i TEMFCC (engl. *Teager-based Mel-Frequency Cepstral Coefficients*).

Postoje još neke metode predprocesiranja zvuka koje se mogu dodatno koristiti pri izdvajanju značajki kako bi se dobili bolji rezultati. Neke od njih su:

a) (engl. *dithering*): dodavanje vrlo male količine buke signalu kako bi se spriječili matematički problemi tijekom izdvajanja značajki; posebice računanje logaritma od 0;
b) DC uklanjanje (engl. *DC removal*): uklanjanje bilo kakvog konstantnog odstupanja od zvučnog vala;
c) prednaglašavanje (engl. *pre-emphasis*): primjena visokofrekvencijskog filtera na signal prije izdvajanju značajki kako bi se ispravilo svojstvo da naglašen govor u pravilu ima veću energiju pri nižim frekvencijama, no što je ima nenaglašen govor pri visokim frekvencijama.

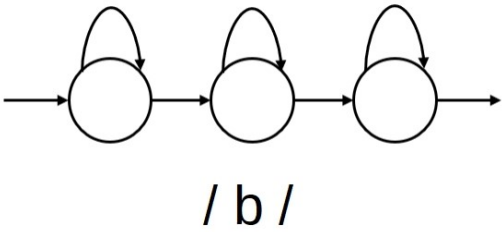
U slučaju da u signalu postoje smetnje, postoji i dodatan korak u obradi zvučnog signala, normalizacija značajki. Moguće je da se u komunikacijskom kanalu pojavi nekakvo konstantno ometanje. Primjerice, postoji puno različitih vrsta mikrofona, a neki modeli nemaju ravan frekvencijski odziv što utječe na kvalitetu snimljenog zvuka. Varijacije u signalu koji predstavlja isti govor utječu na računanje MFCC. Ovakve smetnje mogu se predstaviti konvolucijom, matematičkim operatorom koji računa preklapanja između dviju funkcija, a može se predstaviti množenjem u frekvencijskoj domeni. Normalizacija se provodi izravno na značajkama nakon logaritmiranja i filtriranja.

5.3.3. Akustični model

U većini današnjih sustava je akustični model (engl. *acoustic model*) hibrid umjetnih neuronskih mreža i skrivenih Markovljevih modela. Duboke neuronske mreže koriste se za pretpostavljanje na razini fonema, a skriveni Markovljevi modeli pretpostavljene foneme pretvaraju u pretpostavljeni niz koji čini riječ. Uz naziv hibridi, koristi se i naziv DNN-HMM sustavi.

5.3.3.1. Skriveni Markovljevi modeli u akustičnom modelu

Skriveni Markovljevi lanci kao prvo moraju iz vektora akustičkih značajki odrediti koliko segmenata odgovara jednom fonemu. Ovaj broj jako može varirati s obzirom na položaj fonema u riječi, kao i na karakteristike govora kao što su brzina izgovora i melodičnost. Tipično se svaki fonem modelira s tri stanja kako bi se odvojilo modeliranje početka, sredine i kraja izgovora fonema. Svako stanje ima mogućnost prijelaza u samo sebe ili u iduće stanje, kao što se vidi na slici 5.7.



Slika 5.7 Shematski prikaz modeliranja fonema / b / *s tri stanja uporabom HMM-a*

Povijesno se za raspodjelu vjerojatnosti po stanjima koristio Gaussov miješani model (engl. *Gaussian Mixture Model, GMM*). Današnji sustavi više ne koriste Gaussov miješani model, već jednu duboku neuronsku mrežu kojoj je izlaz niz vrijednosti koji predstavlja sva stanja skrivenog Markovljevog modela za sve moguće foneme. Na primjeru hrvatskog jezika koji ima 32 fonema, izlazni vektor neuronske mreže sadržavao bi 96 (32 x 3) vrijednosti.

Kako fonemi jako ovise o položaju u riječi, često se modeliraju fonemi u ovisnosti o kontekstu (engl. *contex dependent phones*). Gleda se prethodni fonem, trenutni i sljedeći, odnosno radi se o tri uzastopna fonema koji se jednom riječju nazivaju trifoni. Postoje sustavi koji modeliraju i duži niz uzastopnih fonema, ali oni su rijetkost. Kada se koriste fonemi u kontekstu dolazi do velikog porasta stanja s obzirom na broj fonema u jeziku. Broj trifona je N^3, gdje N predstavlja broj fonema. Kada se koristi po tri stanja za svaki trifon, za hrvatski jezik je broj stanja 98304. Ovakav porast broja stanja dovodi do dva velika problema. Prvo, postoji puno manje podataka prema kojima bi se učili trifoni te drugo, neki trifoni se neće pojaviti pri učenju mreže, ali pojavit će se prilikom testiranja ili upotrebe. Ovi problemi rješavaju se upotrebom senona. Senoni povezuju stanja s međusobno sličnim trifonima i stvaraju od njih novo spojeno stanje čime se smanjuje broj potrebnih stanja.

Postupak stvaranja senona odvija se pomoću drva odluka. Prvo se pronalaze svi trifoni sa zajedničkim središnjim fonemom i oni predstavljaju korijen drva. Drvo se širi prema pitanjima o lijevim i desnim, odnosno prethodnim i sljedećim fonemima. Primjerice, koliko su naglašeni ili kojoj skupini fonema pripadaju. Drvo se širi po ovom modelu sve dok vjerojatnost pojavljivanja bilo kojeg trifona nije iznad preddefinirane granice. Po završetku rasta drva, krajevi drva predstavljaju senone. Skoro svi aktualni sustavi za prepoznavanje govora koriste senone u ovisnosti o kontekstu.

5.3.3.2. DNN akustični model

Za akustično modeliranje mogu se koristiti unaprijedne neuronske mreže i povratne neuronske mreže. Najjednostavnija te stoga i najčešće korištena umjetna neuronska mreža u akustičnom modelu je standardna unaprijedna neuronska mreža koja je u potpunosti povezana. U prethodnim poglavljima su objašnjeni općeniti principi rada unaprijednih mreža pa će u ovom poglavlju biti naglasak na ključne aspekte akustičnog modeliranja pomoću dubokih neuronskih mreža.

Korištenje dubokih neuronskih mreža je jedan od velikih napredaka u sustavima za prepoznavanje govora proteklih godina. Kao što je već spomenuto, hibridni sustavi umjesto Gaussovog miješanog modela koriste duboke neuronske mreže čiji izlaz odgovara senonima. U slučaju korištenja unaprijedne neuronske mreže, mreža se uči kako bi klasificirala svaki segment ulaznog signala. Prilikom klasifikacije korisno je stvoriti kontekstualni prozor (engl. *context window*) za svaki segment koji će poslužiti kao ulaz u mrežu. Za segment u trenutku t, ulaz u mrežu je simetričan prozor od N segmenata poslije i N segmenata prije. Ako je x_t vektor značajki u trenutku t, ulazni vektor mreže glasi:

$$X_t = [x_{t-N}, x_{t-N-1}, \dots, x_t, \dots, x_{t+N-1}, x_{t+N}]. \tag{5.8}$$

U pravilu se N kreće između vrijednosti 5 i 10, ovisno o količini dostupnih podataka za učenje. Veći kontekstualni prozor osigurava više informacija, ali zahtjeva veću ulaznu matricu značajki, što može biti nezgodno u situacijama s malom količinom podataka za učenje. Često se i vektor značajki proširuje sa svojim vremenskim derivacijama koje se ponekad nazivaju i delta značajkama (engl. *delta features*). Ove značajke mogu se računati na jednostavan način kao obične razlike ili korištenjem kompliciranijih izraza. Jednostavan primjer računanja delta značajki glasi:

$$\Delta x_t = x_{t+2} - x_{t-2} \tag{5.9}$$

i

$$\Delta^2 x_t = \Delta x_{t+2} - \Delta x_{t-2}, \tag{5.10}$$

a ulaz u mrežu je kontekstualni prozor koji se sastoji od originalnog vektora značajki, delta značajki i delta-delta značajki ($x_t, \Delta x_t, \Delta^2 x_t$). Učenje se provodi po algoritmu učenja povratnog prostiranja pogreške koji je prethodno opisan.

Najčešća funkcija cilja koja se koristi za učenje ovih mreža je funkcija prijelazne entropije (engl. *cross entropy*) koja se i inače koristi pri učenju mreža čiji je zadatak klasifikacija. Funkcija za svaki segment zvučnog zapisa glasi:

$$E = -\sum_{i=1}^{M} t_m \log(y_m), \tag{5.11}$$

gdje je *M* broj klasa senona, t_m je oznaka (1 ako pripada klasi *m* i 0 ako ne), a y_m je izlaz mreže. Odnosno, za svaki segment potrebno je generirati vektor dimenzija *M* x 1 koji se sastoji od nula, osim jedne jedinice koja odgovara točnom senonu. Funkcija prijelazne entropije zahtjeva da svaki segment bude klasificiran, što može dovesti do problema jer se u govoru javljaju pauze i prekidi. Ovo se rješava uporabom povezane vremenske klasifikacije (engl. *Connectionist Temporal Classification, CTC*) koja omogućuje dodavanje oznaka skupu fonema, kao što je simbol „prazno“.

Za klasificiranje segmenata zvučnog zapisa općenito se koristi metoda prisilnog poravnavanja (engl. *forced alignment*) koja generira najvjerojatniji senon za dani segment. Ova metoda zahtjeva već postojeći sustav za prepoznavanje govora kako bi funkcionirala. To može biti sustav s Gaussovim miješanim modelom ili sustav s dubokim neuronskim mrežama koje su već naučene. Izlaz ove metode je datoteka u kojoj je zapisano sljedeće: početni segment i vrijeme početka izgovorene riječi unutar njega, krajnji segment i vrijeme završetka izgovorene riječi te odgovarajuća oznaka senona. Primjer ovakve izlazne datoteke je MLF datoteka koju koristi već spomenuti softver za prepoznavanje govora, HTK. Zapis MLF datoteke prikazan je na slici 5.8 za izraz „dobro jutro“ na engleskom jeziku.

```
#!MLF!#
"test utterance.lab"
0        17700000 sil[2]      0.031854    sil 719.849243 <s>
17700000 27100000 sil[3]      140.979706
27100000 34800000 sil[4]      152.653412
34800000 35500000 g_s2_6      8.032190    sil-g+uh 27.682287 good
35500000 35800000 g_s3_25     6.538751
35800000 36100000 g_s4_11     13.111345
36100000 36200000 uh_s2_7     3.356279    g-uh+d 11.706375
36200000 36300000 uh_s3_15    4.878324
36300000 36400000 uh_s4_15    3.471773
36400000 36500000 d_s2_51     1.766836    uh-d+m 16.887865
36500000 36800000 d_s3_44     11.397981
36800000 36900000 d_s4_26     3.723048
36900000 37200000 m_s2_7      13.519948   d-m+ao 38.862099 morning
37200000 37500000 m_s3_21     20.174339
37500000 37600000 m_s4_79     5.167810
37600000 37700000 ao_s2_51    3.014942    m-ao+r 11.643064
37700000 37900000 ao_s3_56    5.589801
37900000 38000000 ao_s4_46    3.038320
38000000 38100000 r_s2_178    5.339163    ao-r+n 26.518557
38100000 38300000 r_s3_100    14.487432
38300000 38400000 r_s4_61     6.691961
38400000 38600000 n_s2_28     17.327641   r-n+ih 35.669243
38600000 38700000 n_s3_145    9.536380
38700000 38800000 n_s4_112    8.805222
38800000 38900000 ih_s2_15    5.309596    n-ih+ng 25.129721
38900000 39000000 ih_s3_160   5.600624
39000000 39200000 ih_s4_123   14.219500
39200000 39300000 ng_s2_8     4.993470    ih-ng+m 7.568606
39300000 39400000 ng_s3_14    2.377748
39400000 39500000 ng_s4_4     0.197388
```

Slika 5.8 Zapis MLF *datoteke za „good morning“ [10]*

Stupci MLF datoteke interpretiraju se kao:

1. vrijeme početka u 100 ns,
2. vrijeme završetka u 100 ns,
3. oznaka senona,
4. ocjena akustičnog modela za taj segment senona,
5. HMM trifon model,
6. ocjena akustičnog modela za trifon,
7. transkript izgovorene riječi (pojavljuje se na početku izgovora).

Povratne neuronske mreže iznimno su pogodne za sustave za prepoznavanje govora zbog svoje vremenske ovisnosti. U akustičnom modeliranju povratne neuronske mreže mogu naučiti vremenske uzorke značajki fonema zapisanih u obliku vektorskog niza. Kontekstualni prozor se u pravilu ne koristi jer nije potreban zbog mogućnosti mreže da uči korelaciju vremena i podataka. Kod jednosmjernih povratnih mreža korisno je stvoriti nekoliko prozora zbog budućeg konteksta, ali su kontekstualni prozori za tu svrhu daleko manji od onih korištenih kod unaprijednih neuronskih mreža. Kod dvosmjernih povratnih mreža nema koristi od kontekstualnih prozora jer prilikom obrade signala, mreža je već vidjela cijeli zapis izgovorene riječi bilo u unaprijednom ili povratnom smjeru.

Učenje povratnih mreža se isto može odvijati pomoću funkcije cilja prijelazne entropije s malo modificiranom metodom izračuna gradijenta. Koristi se varijacija učenja s povratnim rasprostiranjem pogreške, algoritam učenja povratnog rasprostiranja pogreške u vremenu (engl. *back-propagation through time, BPTT*). Kao i standardni algoritam učenja s povratnim rasprostiranjem pogreške, BPTT optimizira parametre mreže korištenjem algoritma najstrmijeg pada primjenjujući pritom gradijent pogreške. Prilikom učenja dolazi do množenja velike količine gradijenata što se može odraziti na vrijednost izraza. Moguća su dva krajnja slučaja koje treba izbjegavati prilikom učenja. Prvo, izraz može poprimiti vrijednost blizu nule, tzv. nestajući gradijent (engl. *vanishing gradient*), pri kojemu se ne događa učenje. Drugo, izraz može poprimiti iznimno velike vrijednosti, tzv. eksplodirajući gradijent (engl. *exploding gradient*), koji dovodi do nestabilnosti i divergencije. Za rješavanje problema nestajućeg gradijenta postoje dvije metode:

1. korištenje specifične povratne strukture, poput LSTM-a ili
2. ograničiti BPTT algoritam da gleda samo određeni broj prethodnih vrijednosti.

Problem eksplodirajućeg gradijenta rješava se metodom podrezivanja (engl. *gradient clipping*) kojom se određuje gornja granica apsolutne vrijednosti gradijenta za bilo koji parametar mreže. Svi gradijenti koji su veći od gornje granice postavljaju se na vrijednost gornje granice i učenje se nastavlja.

5.3.4. Jezični model

U osnovnoj jednadžbi sustava za prepoznavanje govora, prikazanoj izrazom (5.5), $\mathrm{P}(R)$ predstavlja jezični model koji ovisi isključivo o nizu riječi R. Zadatak jezičnog modela je da predviđa sljedeću riječ prije no što je i izgovorena. Jezični model dodjeljuje veliku vjerojatnost najvjerojatnijim nizovima riječi, a jako malu onim netipičnima, no pritom niti jedna moguća kombinacija riječi ne ostaje nemoguća jer tko zna što će govornik reći. Dodjeljivanje vjerojatnosti vrši se prema pravilima jezika, kao što su gramatika i semantika, te prema podacima za učenje. Ne koriste se nefleksibilno programirana pravila gramatike i semantike zbog nepredvidivosti spontanog govora govornika. Prolaskom kroz podatke za učenje jezični model statistički raspoređuje vrijednost po mogućim nizovima riječi.

Pri stvaranju jezičnog modela prvo je potrebno dodijeliti vjerojatnost svakom mogućem nizu riječi R duljine N. N predstavlja broj riječi koji je u osnovi bezbrojan, no zbog pojednostavljenja problema ograničuje se na konačnu vrijednost. Taj konačan skup riječi naziva se rječnikom sustava. Sustav za prepoznavanje govora ne može prepoznati riječ za koju jezični model smatra da ima vjerojatnost pojave jednaku nuli. Riječi koje se ne pojavljuju u rječniku, a pojave se u govoru na ulazu sustava, dovest će do pogreške prilikom prepoznavanja. Zato je potrebno složiti rječnik sustava koji će minimizirati tu pogrešku. Često postoji optimalna veličina rječnika koja balansira brzinu sustava s njegovom točnošću. Veliki rječnici povećavaju točnost, ali smanjuju brzinu dekodiranja riječi, katkad čak i ometaju rad akustičnog sustava što rezultira neželjenim pogreškama.

5.3.4.1. N-gram model

Čak i s konačnim skupom riječi unutar rječnika, postoji bezbrojno mnogo mogućnosti nizova riječi, odnosno rečenica. Iz tog razloga se jezični model ne bazira na dodjeljivanju vjerojatnosti svakoj mogućoj rečenice, već se koristi lančano pravilo vjerojatnosti za niz riječi određene duljine. Koristeći se pravilima Markovljevih lanaca, moguće je zapisati rečenicu kao umnožak uvjetnih vjerojatnosti riječi, odnosno trenutnu riječ povezati s vjerojatnostima riječi koje su joj prethodile:

$$\mathrm{P}(R) = \mathrm{P}(r_1)\mathrm{P}(r_2|r_1)\mathrm{P}(r_3|r_1 r_2) \dots \mathrm{P}(r_N|r_1 \dots r_{N-1}). \tag{5.12}$$

Iako ovakav zapis odgovara Markovljevim lancima, u modeliranju jezika se koristi izraz *N*-gram model. Primjerice, bigram model predviđa sljedeću riječ samo prema prvoj prethodnoj, trigram prema dvije prethodne i tako dalje. Generalno, *N*-gram model predviđa riječ prema *N*-1 prethodnoj riječi. U praksi se rijetko koriste *N*-grami veći od 4-grama ili 5-grama jer nisu pokazali značajan rast u točnosti.

Kako *N*-grami pridodaju vrijednost nizu riječi konačne duljine *N*, potrebno je uvesti posebnu oznaku koja će predstavljati kraj niza. U tu svrhu dodaje se u rječnik posebna oznaka za kraj rečenice, odnosno niza (engl. e*nd-of-sentence tag*), koja se označava s „< /*s* >“. Također je potrebno uvesti i oznaku za početak niza (engl. *start-of-sentence tag*), „< *s* >“. Umeće se prije prve riječi r_1 te predstavlja kontekst za nju. Ovo je korisno jer postoje riječi koje se učestalo pojavljuju na početku rečenica; primjerice upitne rečenice u hrvatskom najčešće će početi prilogom, upitnom zamjenicom ili upitnom česticom.

Uvjetna vjerojatnost *N*-grama se jednostavno može procijeniti prema učestalosti ponavljanja riječi. Općenito, za *k*-gram model vrijedi:

$$\mathrm{P}(r_k|r_1 \dots r_{k-1}) = \frac{\mathrm{c}(r_1 \dots r_k)}{\mathrm{c}(r_1 \dots r_{k-1})}, \tag{5.13}$$

gdje je $\mathrm{c}(r_1 \dots r_k)$ broj ponavljanja niza duljine *k*.

Relativna učestalost ponavljanja kao procjena vjerojatnosti ima jedan veliki nedostatak; svaki *N*-gram koji se ne pojavi u podacima za učenje ima vjerojatnost nula. Kako su podaci za učenje konačne veličine, ali same mogućnosti govora nisu, niti jedna kombinacija niza riječi ne bi smjela biti nemoguća. Postupkom uglađivanja jezičnog modela (engl. *language model smoothing*) svakom neopaženom *N*-gramu pridodaje se vjerojatnost veća od nule. Postoji puno metoda uglađivanja jezičnog modela, a neke od njih su[15]:

a) uglađivanje interpolacijom (engl. *interpolation smoothing*) – smanji se vjerojatnost viđenih nizova riječi te se ta oduzeta vjerojatnost interpolacijski podijeli između viđenih i neviđenih nizova;
b) Laplaceovo ili aditivno uglađivanje (engl. *additive smoothing*) – prema određenom izrazu se svim viđenim izrazima oduzima dio vjerojatnosti te se svakom neviđenom nizu pridodaje jednaki dio te vjerojatnosti;
c) kolekcijsko uglađivanje (engl. *collection smoothing*) – također se oduzima dio vjerojatnosti viđenim nizovima riječi, a uz to se računa i učestalost pojedinih riječi unutar podataka za učenje te se prema toj učestalosti raspodjeljuje vjerojatnost na neviđene nizove koji možda sadrže te riječi;
d) Jelinek-Mercer uglađivanje ili interpolacija s konstantnim koeficijentima (engl. *fixed coefficient interpolation*) – bazirano na kolekcijskom uglađivanju, pri interpolaciji koristi određeni faktor β;
e) Witten-Bell uglađivanje – jezični model broji svaki put kada pri učenju susretne novu do tada neviđenu riječ, te se s obzirom na broj tih prvi put viđenih riječi smanjuje vrijednost viđenih nizova riječi i raspodjeljuje po neviđenim nizovima;
f) *back-off* metoda – smanjuje duljinu *N*-grama za jedan i vjerojatnost pojave veže uz novonastali (*N-1*)-gram (najčešće bigram) s ciljem da raspodijelili ostatak vjerojatnosti po drugim mogućim nizovima; sve to pod pretpostavkom da među neviđenim nizovima postoje više ili manje vjerojatni nizovi u obliku (*N-1*)-grama.

5.3.4.2. Ocjenjivanje jezičnog modela

Pri evaluaciji jezičnog modela gledaju se tri karakteristike: vjerojatnost (engl. *likelihood*), entropija ili neodređenost (engl. *entropy*) te kompleksnost (engl. *perplexity*). Model se obavezno testira na podacima koji su neovisni o podacima za učenje. Prilikom ispitivanja vjerojatnosti gleda se kako je vjerojatnost raspoređena po viđenim i neviđenim riječima te nizovima riječi. Entropija modela računa se prema izrazu:

$$-\frac{1}{N}\log \mathrm{P}(r_1 \dots r_N), \tag{5.14}$$

gdje je *N* broj riječi u rječniku sustava. Entropija predstavlja mjeru informativnosti niza riječi, odnosno daje prosječan broj bitova potrebnih za rad s određenim nizom riječi. Kompleksnost je recipročna vrijednost srednje vjerojatnosti raspodijeljene po riječima tj. broj riječi u rječniku s istom vjerojatnošću pojavljivanja. Za računanje kompleksnosti koristi se geometrijska sredina jer su vjerojatnosti povezane množenjem, a ne zbrajanjem. Prema tome kompleksnost je suprotna entropiji:

$$\mathrm{P}(r_1 \dots r_N)^{-\frac{1}{N}}. \tag{5.15}$$

Katkada je potrebno ograničiti veličinu jezičnog modela, posebice zato što model raste gotovo linearno s brojem riječi uporabom N-grama. Pri ograničavanju veličine eliminiraju se parametri koji su redundantni. Redundantnost parametra računa se s obzirom na entropiju ili kompleksnost. Za svaki N-gram računa se koliko on utječe na ukupnu entropiju i kompleksnost sustava te ako je razlika ispod nekog predodređenog praga, parametar se odbacuje. Nakon što se iz modela izbace redundantni parametri, potrebno je ponovo normalizirati vjerojatnosti preostalih parametara. Odnos karakteristika i kvalitete jezičnog modela prikazan je u tablici 5.1.

Tablica 5.1 Karakteristike dobrog i lošeg jezičnog modela

DOBAR MODEL	velika vjerojatnost mala entropija mala kompleksnost
LOŠ MODEL	mala vjerojatnost velika entropija velika kompleksnost

5.3.4.3. Jezični model baziran na klasama riječi

Nedostatak modela baziranih na *N*-gramima je što riječi tretiraju kao u potpunosti odvojive pojave. Tek nakon što se sustav dovoljno puta susretne s određenom riječi, može naučiti u kojim *N*-gramima se pojavljuje. Ljudi ne koriste jezik na taj način, već povezuju riječi ne samo prema sintaksama, nego i prema značenjima. Sličnost među riječima može se iskoristiti za generaliziranje jezičnih modela. Na tom principu su građeni jezični modeli bazirani na klasama riječi koji određene skupine riječi klasificiraju. Očiti primjer riječi koje se mogu klasificirati su dani u tjednu, mjeseci u godini ili pak doba dana. Korištenjem klasa riječi smanjuje se broj *N*-grama. Umjesto *N*-grama kao „Utakmica je u četvrtak navečer“ i „Utakmica je u nedjelju popodne“ te svih mogućih vremenskih kombinacija tog tipa, dobiva se jedan *N*-gram koji obuhvaća sve moguće kombinacija a koji glasi: „Utakmica je u *'dan u tjednu' 'doba dana'*“, pri čemu se *'dan u tjednu'* i *'doba dana'* klase riječi. Unutar klasa riječi računa se vjerojatnost pojave određene riječi u klasi.

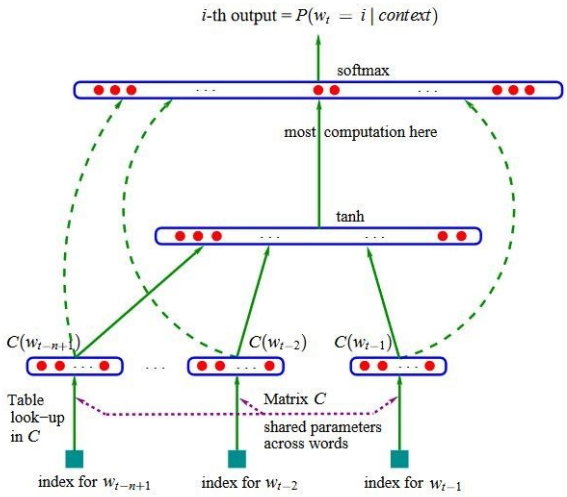
Prethodno navedeni primjer predstavlja jednu metodu klasificiranja riječi, prema području primjene. Ova metoda zahtjeva prethodno znanje o rječniku sustava, odnosno namjeni. Primjerice, ako se sustav za prepoznavanje govora planira koristiti unutar aplikacije za putovanja, moguće je napraviti klase mogućih destinacija, datuma, zračnih linija itd. S istim znanjem moguće je i rasporediti vjerojatnosti unutar klasa, na primjer koliko je koja destinacija popularna u zadnje vrijeme.

Druga metoda klasificiranja je u potpunosti bazirana na podacima, bez ljudskog poznavanja. Ova metoda koristi kompleksne algoritme kako bi prošla kroz velike količine podatka i na određeni način klasificirala riječi.

5.3.4.4. Jezični modeli s umjetnim neuronskim mrežama

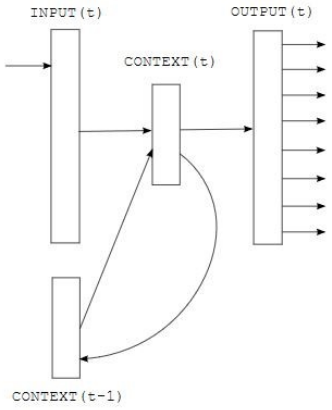
Eksperimenti su pokazali da umjetne neuronske mreže osmišljene za jezične modele daju daleko bolje rezultate od klasičnih *N*-gram modela, ako imaju dovoljno podataka za učenje. Za razliku od N-gram modela, jezični modeli s umjetnim neuronskim mrežama sposobni su generalizirati riječi. Prethodno spomenuta klasifikacija riječi djelomično rješava taj problem u *N*-gram modelima, ali stvara novi problem parametara klasifikacije.

Prva korištena umjetna neuronska mreža u jezičnom modelu bila je unaprijedna neuronska mreža koja je uspješno rješavala problem generalizacije pomoću smještaja riječi (engl. *word embedding*). Ulaz u mrežu je vektorski zapis *N*-1 riječi koje tvore *N*-gram, a izlaz je vektor raspoređenih vjerojatnosti iduće riječi. Ulazni i izlazni vektor su duljine koja odgovara broju riječi u rječniku sustava te stoga lako mogu implementirati u sustave koji su do tada koristili jezični model baziran samo na *N*-gramima. Unutar mreže se ulazni vektor pretvara u višedimenzionalnu matricu. Kontekstualno slične riječi koje na isti način utječu na kontekst rečenice spremaju se blizu jedna drugoj unutar prostora matrice. Na taj način kontekstualno slične riječ imaju slične vjerojatnosti u izlaznom vektoru, te se lakše prepoznaju u novim neviđenim nizovima riječi. Ova mreža prikazana je na slici 5.9.



Slika 5.9 Shematski prikaz unaprijedne neuronske mreže korištene u jezičnom modelu [16]

Umjetne neuronske mreže riješile su još jedan problem *N*-gram modela, skraćivanje konteksta. U *N*-gram modelima riječ ovisi isključivo o *N*-1 prethodnih riječi, a već je spomenuto da kontekstualna povezanost riječi može sezati jako daleko unutar rečenice. Niti jedna smislena vrijednost *N* ne bi mogla predvidjeti sve moguće riječi u kontekstualnom odnosu. Ovaj nedostatak ispravljen je uporabom povratnih neuronskih mreža koje u trenutku *t*-1 aktiviraju funkcije skrivenog sloja. Ovime je omogućen prijenos informacije od jedne riječi do sljedeće i tako dalje bez granice koliko u prošlost ta informacija ide. Postoje određeni matematički problemi koji se javljaju pri učenju ovih mreža, no i oni se uspješno rješavaju uporabom LSTM neuronskih mreža.



Slika 5.12 Shematski prikaz rada povratne neuronske mreže u jezičnom modelu [17]

5.3.5. Dekodiranje govora

Akustični model na ulaz dekodera sustava za prepoznavanje govora šalje zapis o vjerojatnostima mogućih fonema unutar jednog segmenta. Jezični model šalje na ulaz dekodera zapis o vjerojatnostima mogućih nizova riječi. Zadatak dekodera je da usporedi dobivene podatke i odredi najvjerojatniju riječ, odnosno niz riječi i proslijedi ih na izlaz sustava.

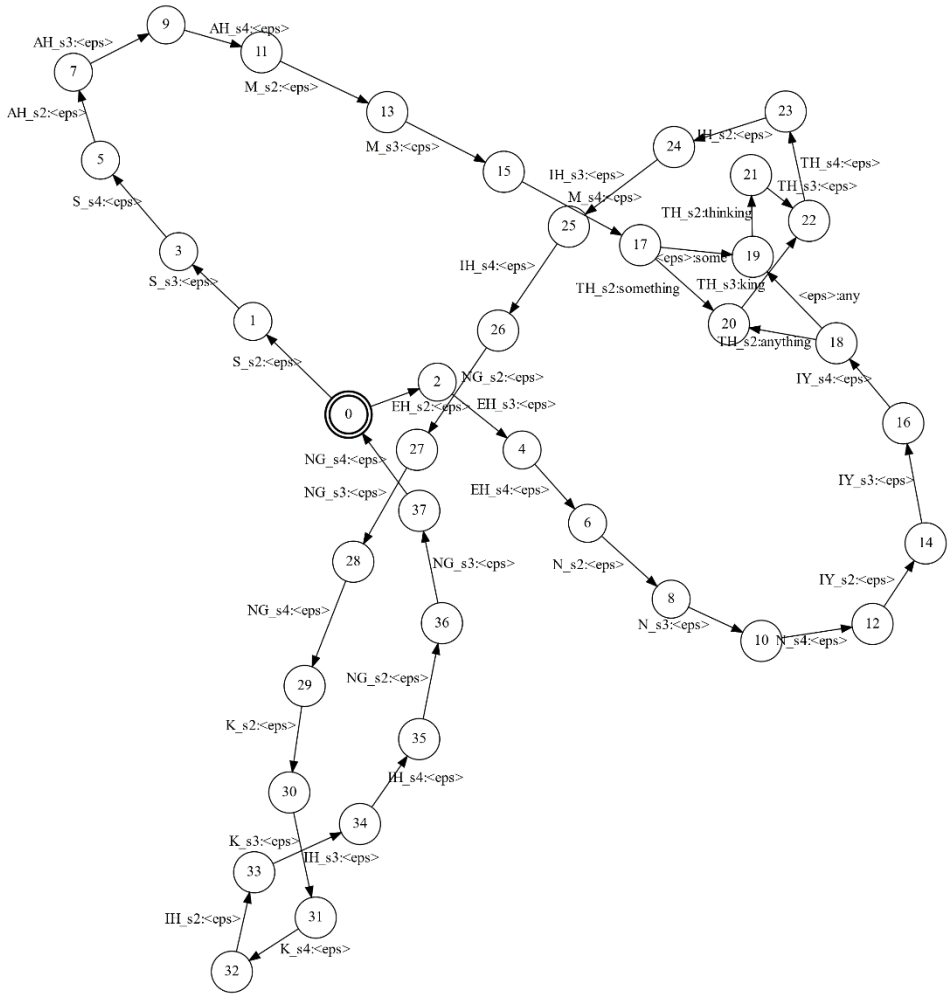
Rječnik izgovora (engl. *pronunciation lexicon*) može se promatrati kao dio dekodera ili kao odvojeni dio sustava. To je baza fonetskih zapisa svih riječi u rječniku sustava koja povezuje izgovor s napisanim oblikom riječi. Riječ je predstavljena nizom slova, koja su predstavljena nizom fonema, a niz fonema je predstavljen nizom akustičnih svojstva. Ulaz u rječnik izgovora je niz provjerenih fonema, a izlaz je napisana riječ. Određena riječ može imati veliki broj izgovora ovisno o govorniku. Uzrok tome može biti naglasak ili brzina govora. Nerijetko prilikom brzog pričanja, ljudi izostave fonem, ali sam smisao riječi i ona sama se ne mijenjaju zbog toga. Iz tog razloga rječnik izgovora ima nekoliko mogućih izgovora koji predstavljaju istu napisanu riječ. Također, ti izgovori se ne moraju biti identični fonetskom zapisu riječi u klasičnom rječniku određenog jezika. U pravilu, u rječniku izgovora sustava za prepoznavanje govora postoji oko 4 verzije izgovara pojedine riječi. Rječnik izgovora za određeni jezik stvaraju lingvistički eksperti u istom jeziku.

Dekodiranje je proces maksimiziranja vjerojatnosti dobivenih iz akustičnog i jezičnog modela. Ovaj postupak najlakše je modelirati pomoću grafa te najvjerojatniji niz slova, odnosno riječi, pronaći pretraživanjem tog grafa. Za pretvorbu ulaznih podataka u graf najčešće se koriste konačni automati (engl. *finite state automata, FSA*) i konačni pretvornici (engl. *finite state transducers, FST*) te njihove težinske verzije. Konačni automati grafički preoblikuju nizove te ih je potom lakše pretraživati. Sastoje se od konačnog broja stanja, prijelaza između stanja i mogućih radnji između stanja. Konačni pretvornici mogu se predstaviti kao konačni automati s dvije trake između kojih se vrši preslikavanje iz jednog skupa znakova u drugi.

Većina modernih sustava za prepoznavanje govora koristi četiri konačna automata i pretvornika:

1. HMM konačni pretvornik *H* koji preslikava niz HMM stanja s oznakama senona u HMM modele s oznakama trifona;
2. kontekstualni težinski konačni pretvornik *C* koji nizove tifona preslikava u nizove fonema;
3. težinski konačni pretvornik izgovora *L* koji preslikava nizove fonema u napisane riječi;
4. gramatički konačni automat *G* može imati predprogramirana pravila ili može biti težinski gramatički automat koji pridodaje težine gramatici i sintagmi iz jezičnog modela.

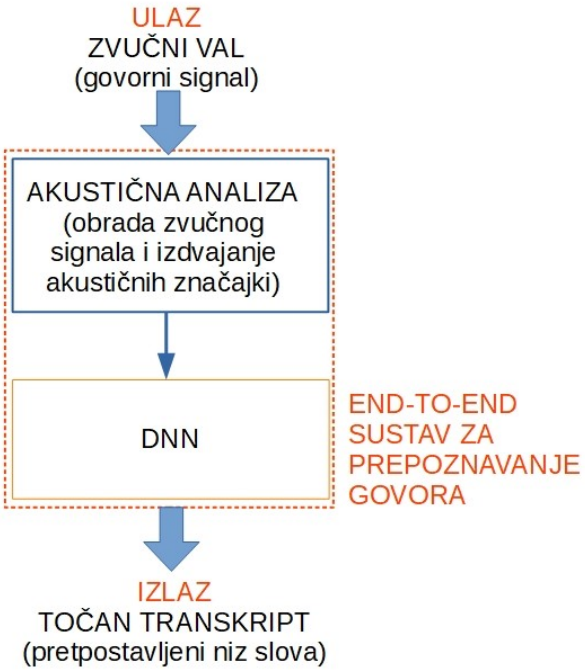
Pretvornici se moraju koristiti redom kojim su prethodno navedeni tvoreći tako HCLG graf. Na slici 5.11 simbol „< *eps* >“ predstavlja dio niza gdje nema simbola. Na primjer, ako govornik izgovori dugo „*a*“ u nekoj riječi, zapis će izgledati kao „*a*< *eps* >“, umjesto „*aaaaa…*“. Pretraživanje grafa svodi se na pronalaženje najkraćeg puta za što se koristi algoritam širinskog pretraživanja (engl. *beam search algorithm*).



Slika 5.15 Primjer HCLG grafa [10]

5.4. *End-to-end* sustavi automatskog prepoznavanja govora

Cilj *end-to-end* sustava je pojednostaviti proces prepoznavanje govora. Ideja je zamijeniti akustični i jezični model te rječnik izgovora s jednom dubokom povratnom neuronskom mrežom. Na taj način sustavi ne bi trebali biti prilagođeni određenom jeziku, već bi univerzalni sustav direktno iz zvučnog zapisa prepoznavao nizove slova bez potrebe za fonetskim preslikavanjem. Time bi se uklonila potreba za lingvističkim ekspertima prilikom stvaranja sustava za prepoznavanje govora. Shema *end-to-end* sustava prikazana je na slici 5.12.

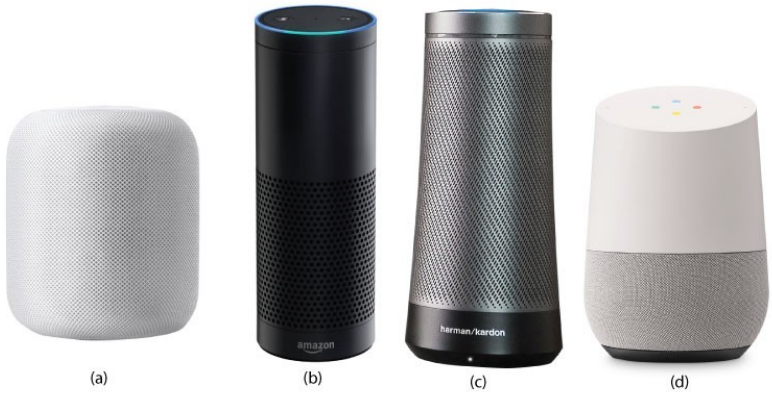


Slika 5.18 Model end-to-end sustava za prepoznavanje govora

Za učenje ovakvih sustava potrebna je ogromna količina podataka jer njihovo prepoznavanje isključivo ovisi o zvučnom zapisu govora i odgovarajućem transkriptu. *End-to-end* sustavi aktualna su tema istraživanja i na njima se intenzivno radi. Zasada još uvijek ne mogu konkurirati sustavima s jezičnim modelom i rječnikom izgovora.

5.5. Najpoznatiji sustavi za prepoznavanje govora današnjice

Sustavi za prepoznavanje govora danas se najviše razvijaju kao sastavni dio sustava koji se bave obradom ljudskog govora. Zbog velike raširenosti smartphone uređaja, ne iznenađuje da su upravo oni glavi uređaji za koje se stvaraju sustavi za prepoznavanje govora. Mnoge tehnološke tvrtke stvorile su vlastite virtualne osobne asistente (engl. *virtual personal assistants, VPA*) kojima su obogatile svoju već postojeću ponudu na tržištu. Virtualni osobni asistenti koriste sustav za



Slika 5.21 Redom s lijeva na desno [18]: Apple HomePod (Siri), Amazon Echo (Alexa), Microsoft Cortana, Google Home (Google Assistant)

prepoznavanje govora kao neizostavan dio svog algoritma. Uz prepoznavanje govora, ovi sustavi koriste i prepoznavanje gesta pomoću slike ili videa, generiranje ljudskog govora, kao i algoritme koji im omogućuju uspješnu govornu komunikaciju s korisnikom. U povijesnom pregledu sustava za prepoznavanje govora spomenuti su neki od najboljih sustava današnjice, kao što su *Appleova* Siri, *Microsoftova* Cortana, *Amazova* Alexa i Google Assistant. Svi ovi sustavi mogu biti implementirani u računala i smartphone uređaje, ali također postoje i posebni uređaji koji samo sadrže te sustave i imaju mogućnost povezati se i upravljati drugim uređajima u svojoj okolini. Uređaji za prethodno spomenute sustave prikazani su na slici 5.13.

5.6. Aktualni problemi

Zbog mogućnosti upotrebe u velikom broju jako različitih područja (interakcija s ljudima, medicina, auto industrija, vojska, telekomunikacije itd.), većina modernih sustava radi na povećanju robusnosti u više pogleda. Kao prvo, sustav mora biti robustan s obzirom na dob, spol, naglasak i sposobnost govornika. Danas je to poprilično ostvareno, no samo kada je u pitanju engleski jezik i njegove moguće varijacije. Google čak razlikuje nekolika verzija engleskog s obzirom na naglasak govornika. Tako prema Googleu postoje: britanski engleski, sjevernoamerički engleski, južnjački engleski, australski engleski, kanadski engleski, indijski engleski i europski engleski. Za svaki od njih postoji velika baza podataka pa njihovo učenje nije problem, kao što je s drugim svjetskim jezicima.

To nas dovodi do drugog područja robusnosti, robusnost s obzirom na jezik. Krajnji cilj istraživača u području prepoznavanja govora jest stvoriti univerzalni sustav koji će s jednakom lakoćom raspoznavati bilo koji svjetski jezik. Ovo je teško ostvariti zbog mnogih razlika između jezika i velikog broja pravila unutar svakog. Čak i ako se sustav ograniči na određenu skupinu jezika, primjerice indoeuropsku skupinu kojoj pripadaju materinji jezici skoro pa polovice čovječanstva, i dalje je riječ o preko 400 jezika koji su iznimno različiti, iako imaju isto podrijetlo. Razlikuju se u pravilima, pismu i morfologiji. Jedan od velikih razlog zašto se engleski toliko proširio svijetom, uz engleska osvajanja, jest što je relativno morfološki oskudan te ga je stoga neizvornom govorniku lako svladati do razine sporazumijevanja. Morfološka oskudnost ga također čini lakšim za modeliranje u sustavima za prepoznavanje govora. Razvitkom *end-to-end* sustava, znanstvenici se nadaju da će se i morfološki bogati jezici poput hrvatskog lakše prepoznavati. Veliki problem kod proširenja sustava za prepoznavanje govora na nove jezike je manjak podataka po kojima bi sustavi mogli učiti. U većini aktualnih sustava prvenstveno je zastupljen engleski, a potom više od većine čine ostali europski jezici, dok su azijski i afrički jezici u zanemarivom postotku, s značajnom iznimkom kineskog.

Treće područje robusnosti na kojem se još radi je otpornost na smetnje iz okoline. Glasne situacije, veliki broj govornika istovremeno, bilo koji oblik pozadinske buke, sve to ne bi smjelo ometati sustav za prepoznavanje govora. Već postoje neki sustavi koji mogu raspoznati nekoliko govornika i čak prepoznavati govor dva do tri govornika istovremeno, no efikasnost tih sustava još uvijek nije ni blizu razine sustava koji prepoznaje govor samo jednog govornika.

Osim robusnosti sustava, kontinuirano se radi na poboljšavanju efikasnosti sustava za prepoznavanje govora. Ovo područje ima koristi od općeg poboljšanja performansi računala i unaprjeđivanja metodi učenja umjetnih neuronskih mreža. Poželjno je smanjiti vrijeme učenja te količinu podataka potrebnih za učenje, posebice za sustave bazirane na jezicima koji ne raspolažu velikom količinom podataka. Također, teži se osigurati da se prepoznavanje govora odvija uistinu u stvarnom vremenu bez prikrivene latentnosti.

6. ZAKLJUČAK

U ovom završnom radu dan je pregled osnova sustava za prepoznavanje govora od lingvističkog aspekta do matematičkog. Navedene su tehnologije koje su unaprijedile prepoznavanje govora i ukratko je opisan njihov način rada. Naglasak je bio na umjetnim neuronskim mrežama u prepoznavanju govora koje se mogu implementirati u svaki dio konvencionalnog sustava za prepoznavanje govora, a u većini današnjih sustava i jesu. Iz navedene literature se može zaključiti da neovisno o mjestu implementacije, bio to jezični model, akustični model ili dekoder, umjetne neuronske mreže daju bolje rezultate, brže su i točnije od prijašnjih sustava, ali isključivo pod uvjetom da je učenje provedeno s dovoljnom količinom kvalitetnih podataka. Uporaba umjetnih neuronskih mreža u sustavima za prepoznavanje govora otvara vrata novim mogućnostima te olakšava eksperimentiranje i testiranje novih ideja. Usprkos velikim napredcima koji su ostvareni proteklih dvadesetak godina, još se puno toga može unaprijediti, ponajviše po pitanju prepoznavanja većeg broja neeuropskih jezika. Prepoznavanje govora kompleksan je problem na kojem se intenzivno radi, posebice na sustavima koji su u potpunosti građeni od umjetnih neuronskih mreža, kao što su *end-to-end* sustavi. Napredak računalnih tehnologija kao i sve veća potražnja tržišta, pozitivno utječu na razvijanje novih i sve boljih sustava.

Uzevši u obzir navedenu literaturu može se zaključiti da su *end-to-end* sustavi budućnost prepoznavanja govora. Prvenstveno zbog jednostavnosti izvedbe u obliku jedne duboke neuronske mreže, što bi rezultiralo poboljšanjima u brzini i smanjenju cijena samih sustava. Također, veliki potencijal *end-to-end* sustava leži u mogućnosti ostvarenja univerzalne primjene.

7. LITERATURA

1. *Leksikografski zavod Miroslav Krleža, http://www.enciklopedija.hr,* (datum pristupa: 21.7.2019.)
2. Branko Vuletić: *Lingvistika govora,* Filozofski fakultet Sveučilišta u Zagrebu, Odsjek za fonetiku, Zagreb, 2007.
3. Elenmari Pletikos: *Kultura govora, Glas*, https://fonet.ffzg.unizg.hr/pletikos/predav-kultura_gov/2_Glas-fonacija.pdf, (datum pristupa: 21.7.2019)
4. Ferdinand de Saussure: *Tečaj opće lingvistike*, Institut za hrvatski jezik i jezikoslovlje, Zagreb, 2000.
5. *Glasovi hrvatskog standardnog jezika*, http://hrvatskijezik.eu/glasovi-hrvatskoga-standardnog-jezika, (datum pristupa: 21.7.2019.)
6. B. Novaković, D. Majetić, M. Široki: *Umjetne neuronske mreže*, Fakultet strojarstva i brodogradnje, Sveučilište u Zagrebu, Zagreb, 2011.
7. Đorđe T. Grozdić: *Primena neuralnih mreža u prepoznavanju šapata*, doktorska disertacija, Elektrotehnički fakultet, Univerzitet u Beogradu, Beograd, 2017.
8. Raul Rojas: *Neural Networks – A Systematic Introduction*, Springer-Verlag, Berlin, 1996.
9. *Understanding LSTM Networks*, http://colah.github.io/posts/2015-08-Understanding-LSTMs/, (datum pristupa: 9.8.2019.)
10. *Speech recognition systems*, Microsoft, https://courses.edx.org/courses/course-v1:Microsoft+DEV287x+2T2019/course/, (datum pristupa: 9.8. 2019.)
11. Bojan Šekronja: *Markovljevi lanci*, Podloge za vježbe iz kolegija Umjetna inteligencija, Fakultet strojarstva i brodogradnje, Sveučilište u Zagrebu, Zagreb
12. Lawrence R. Rabiner: *A tutorial on Hidden Markov Models and selected applications in speech recognition*, IEEE, 1989.
13. S.K. Gaikwad, B.W. Gawali, P. Yannawar: *A Review on Speech Recognition Technique*, International Journal of Computer Applications, Volume 10, No. 3, studeni 2010.
14. B.H. Juang, L.R. Rabiner: *Automatic Speech Recognition – A Brief History of the Technology Development*, 2005.

15. *Smoothing for Language Models*, ML Wiki, http://mlwiki.org/index.php/Smoothing_for_Language_Models, (datum pristupa: 14.8.2019.)
16. Y. Bengio, R. Ducharme, P. Vincent, C. Jauvin: *A Neural Probabilistic Language Model*, Journal of Machine Learning Research, ožujak 2003.
17. T. Mikolov, M. Karafiat, L. Burget, J. Černocky, S. Khudanpur: *Recurrent neural network based language model*, Proc. Interspeech, 2010.
18. V. Këpuska, G. Bohouta: *Next-Generation of Virtual Personal Assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home),* IEEE 8th Annual Computing and Communication Workshop and Conference, Las Vegas, USA, 2018.