UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA

DEPARTMAN ZA ENERGETIKU, ELEKTRONIKU I TELEKOMUNIKACIJE
KATEDRA ZA TELEKOMUNIKACIJE I OBRADU SIGNALA

EDVIN PAKOCI

UTICAJ MORFOLOŠKIH OBELEŽJA NA MODELOVANJE JEZIKA PRIMENOM NEURONSKIH MREŽA U SISTEMIMA ZA PREPOZNAVANJE GOVORA

DOKTORSKA DISERTACIJA

Novi Sad, 2019

УНИВЕРЗИТЕТ У НОВОМ САДУ ● **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**
21000 НОВИ САД, Трг Доситеја Обрадовића 6

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, **РБР**:	
Идентификациони број, **ИБР**:	
Тип документације, **ТД**:	монографска документација
Тип записа, **ТЗ**:	текстуални штампани материјал
Врста рада, **ВР**:	докторска дисертација
Аутор, **АУ**:	Едвин Пакоци, M.Sc.
Ментор, **МН**:	др Бранислав Поповић
Наслов рада, **НР**:	Утицај морфолошких обележја на моделовање језика применом неуронских мрежа у системима за препознавање говора
Језик публикације, **ЈП**:	српски
Језик извода, **ЈИ**:	српски / енглески
Земља публиковања, **ЗП**:	Република Србија
Уже географско подручје, **УГП**:	Аутономна Покрајина Војводина
Година, **ГО**:	2019.
Издавач, **ИЗ**:	ауторски репринт
Место и адреса, **МА**:	Факултет техничких наука, Трг Доситеја Обрадовића 6, Нови Сад
Физички опис рада, **ФО**: (поглавља/страна/референци/табела/слика/прилога)	7 поглавља / 133 стране / 107 референци / 26 табела / 16 слика / 5 прилога
Научна област, **НО**:	електротехничко и рачунарско инжењерство
Научна дисциплина, **НД**:	телекомуникације и обрада сигнала
Предметна одредница/Кључне речи, **ПО**:	аутоматско препознавање говора, моделовање језика, морфолошко таговање, дубоке неуронске мреже, Калди пакет алата
УДК	
Чува се, **ЧУ**:	у библиотеци Факултета техничких наука, Трг Доситеја Обрадовића 6, Нови Сад
Важна напомена, **ВН**:	

УНИВЕРЗИТЕТ У НОВОМ САДУ ● **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**
21000 НОВИ САД, Трг Доситеја Обрадовића 6

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Извод, **ИЗ**:		Аутоматско препознавање говора је технологија која рачунарима омогућава претварање изговорених речи у текст. Она се може применити у многим савременим системима који укључују комуникацију између човека и машине. У овој дисертацији детаљно је описана једна од две главне компоненте система за препознавање говора, а то је језички модел, који специфицира речник система, као и правила према којим се појединачне речи могу повезати у реченицу. Српски језик спада у групу високо инфлективних и морфолошки богатих језика, што значи да користи већи број различитих завршетака речи за изражавање жељене граматичке, синтаксичке или семантичке функције дате речи. Овакво понашање често доводи до великог броја грешака система за препознавање говора код којих због доброг акустичког поклапања препознавач погоди основни облик речи, али погреши њен завршетак. Тај завршетак може да означава другу морфолошку категорију, на пример, падеж, род или број. У раду је представљен нови алат за моделовање језика, који уз идентитет речи у моделу може да користи додатна лексичка и морфолошка обележја речи, чиме је тестирана хипотеза да те додатне информације могу помоћи у превазилажењу значајног броја грешака препознавача које су последица инфлективности српског језика.	
Датум прихватања теме, **ДП**:		05.09.2019	
Датум одбране, **ДО**:			
Чланови комисије, **КО**:	Председник:	др Владо Делић, ФТН Нови Сад	
	Члан:	др Татјана Грбић, ФТН Нови Сад	
	Члан:	др Јелена Николић, ЕФ Ниш	Потпис ментора
	Члан:	др Никша Јаковљевић, ФТН Нови Сад	
	Члан, ментор:	др Бранислав Поповић, ФТН Нови Сад	

UNIVERSITY OF NOVI SAD ● **FACULTY OF TECHNICAL SCIENCES**
21000 NOVI SAD, Trg Dositeja Obradovića 6

KEY WORDS DOCUMENTATION

Accession number, **ANO**:	
Identification number, **INO**:	
Document type, **DT**:	monograph documentation
Type of record, **TR**:	textual printed material
Contents code, **CC**:	Ph.D. thesis
Author, **AU**:	Edvin Pakoci, M.Sc.
Mentor, **MN**:	Branislav Popović, Ph.D.
Title, **TI**:	Influence of Morphological Features on Language Modeling With Neural Networks in Speech Recognition Systems
Language of text, **LT**:	Serbian
Language of abstract, **LA**:	Serbian / English
Country of publication, **CP**:	Republic of Serbia
Locality of publication, **LP**:	Autonomous Province of Vojvodina
Publication year, **PY**:	2019
Publisher, **PB**:	author's reprint
Publication place, **PP**:	Faculty of Technical Sciences, Trg Dositeja Obradovića 6, Novi Sad
Physical description, **PD**: (chapters/pages/ref./tables/figures/appendixes)	7 chapters / 133 pages / 107 references / 26 tables / 16 figures / 5 appendixes
Scientific field, **SF**:	electrical and computer engineering
Scientific discipline, **SD**:	telecommunications and signal processing
Subject/Key words, **S**/**KW**:	automatic speech recongnition, language modeling, morphological tagging, deep neural networks, Kaldi toolkit
UC	
Holding data, **HD**:	Library of the Faculty of Technical Sciences, Trg Dositeja Obradovića 6, Novi Sad
Note, **N**:	

UNIVERSITY OF NOVI SAD ● **FACULTY OF TECHNICAL SCIENCES**
21000 NOVI SAD, Trg Dositeja Obradovića 6

KEY WORDS DOCUMENTATION

Abstract, **AB**:		Automatic speech recognition is a technology that allows computers to convert spoken words into text. It can be applied in various areas which involve communication between humans and machines. This thesis primarily deals with one of two main components of speech recognition systems - the language model, that specifies the vocabulary of the system, as well as the rules by which individual words can be linked into sentences. The Serbian language belongs to a group of highly inflective and morphologically rich languages, which means that it uses a number of different word endings to express the desired grammatical, syntactic, or semantic function of the given word. Such behavior often leads to a significant number of errors in speech recognition systems where due to good acoustic matching the recognizer correctly guesses the basic form of the word, but an error occurs in the word ending. This word ending may indicate a different morphological category, for example, word case, grammatical gender, or grammatical number. The thesis presents a new language modeling tool which, along with the word identity, can also model additional lexical and morphological features of the word, thus testing the hypothesis that this additional information can help overcome a significant number of recognition errors that result from the high inflectivity of the Serbian language.	
Accepted by the Scientific Board on, **ASB**:		05/09/2019	
Defended on, **DE**:			
Defended Board, **DB**:	President:	Vlado Delić, PhD, Faculty of Technical Sciences, Novi Sad	
	Member:	Tatjana Grbić, PhD, Faculty of Technical Sciences, Novi Sad	
	Member:	Jelena Nikolić, PhD, Faculty of Electronic Engineering, Niš	
	Member:	Nikša Jakovljević, PhD, Faculty of Technical Sciences, Novi Sad	Mentor's sign
	Member, mentor:	Branislav Popović, PhD, Faculty of Technical Sciences, Novi Sad	

SAŽETAK

Automatsko prepoznavanje govora je tehnologija koja računarima omogućava pretvaranje izgovorenih reči u tekst, odnosno transkripciju onoga što je rečeno. Ona se može primeniti u mnogim savremenim sistemima koji uključuju komunikaciju između čoveka i mašine. Samo neke od primena su sistemi za diktiranje, govorni asistenti na pametnim mobilnim uređajima, pametne kuće, automatizovani pozivni centri, kao i razni alati za pomoć osobama sa invaliditetom.

U ovoj disertaciji detaljno je predstavljena jedna od dve glavne komponente sistema za automatsko prepoznavanje govora, a to je jezički model. Ova komponenta specificira rečnik sistema – skup svih reči koje sistem može da prepozna, kao i pravila po kojim se pojedinačne reči mogu povezati u rečenicu. Zajedno sa drugom glavnom komponentom sistema, akustičkim modelom, on određuje koja od mogućih sekvenci reči je zapravo izgovorena.

Istorijski, u modelovanju jezika dominirala su dva statistička pristupa. Prvi je pristup preko *n*-grama, odnosno verovatnoća pojavljivanja određenih sekvenci reči dužine najviše *n* u datom jeziku ili konkretnom domenu interakcije, odnosno oblasti primene. On se uspešno koristio veoma dugo, jer je relativno jednostavan, brzo se obučava, i u sistemima sa relativno ograničenom gramatikom (sa ne toliko brojnim mogućnostima za formiranje rečenica) je sasvim odgovarajući i vrlo efikasan. U poslednje vreme, kao jezički modeli su počele da se koriste neuronske mreže, koje su rešile neke od glavnih nedostataka *n*-gram modela, mada su značajno računski složenije.

Pored osnovne, referentne varijante modela jezika kao neuronske mreže sa primenom u srpskom jeziku, u ovoj disertaciji data je i nadogradnja koja u obzir uzima dodatne morfološke informacije. Te informacije imaju potencijal da reše pojedine preostale probleme neuronskih mreža kao modela jezika, specifične za visoko inflektivne jezike kao što je srpski, u kojima isti osnovni oblik reči može dobiti različite nastavke, odnosno sufikse, kojima se određuje gramatička ili sintaksička uloga odgovarajuće reči u rečenici. Neke od morfoloških kategorija o kojima će biti reči su vrsta reči, padež, gramatički broj, gramatički rod, podvrsta reči kod imenica, zamenica ili brojeva, stepen komparacije kod prideva, glagolski oblik i glagolski rod kod glagola. Primenom ovakvog pristupa rešavaju se problemi sistema za automatsko prepoznavanje govora na slobodnijim gramatikama kao što su greške u padežu, rodu ili broju, čak i kada je osnovni oblik reči pogođen.

Na kraju rada su sumirani eksperimentalni rezultati i opisane mogućnosti primene datih tehnika i procedura u sistemima za automatsko prepoznavanje govora na velikim rečnicima na srpskom jeziku, za koje su one prevashodno namenjene.

ABSTRACT

Automatic speech recognition is a technology that allows computers to convert spoken words into text, i.e., to transcribe what has been said. It has a lot of contemporary applications in areas which involve communication between humans and machines. These applications include dictation (automatic transcription) systems, voice assistant applications for smartphones, various smart home uses, automated call centers, as well as an array of tools for aiding people with certain disabilities.

This thesis primarily deals with one of the two main components of speech recognition systems - the language model, that specifies the vocabulary of the system and the rules by which individual words can be linked into sentences. Alongside the other main component, the acoustic model, it determines which one of the possible sequences of words is actually uttered.

Historically, two approaches dominated the field of language modeling. The first one is a statistical approach using *n*-grams, i.e., occurency probabilities of sequences of words of length up to *n* in the given language or domain of interaction (area of usage). This approach was successfully used for a very long time because of its simplicity, extremely fast training, efficiency and more than adequate results for most applications with constrained grammars (with not too many allowed ways for sentence construction). In very recent history, neural networks started to be used as language models too, because they managed to fix several major downsides of *n*-gram models, even though they are a lot more computationally complex.

In addition to the basic, referent variant of language models based on a neural network with application to the Serbian language specifically, this thesis also introduces an upgrade to those models by taking morphological information into account. This additional data has the ability to solve some of the remaining problems with neural network based language models which occur for highly inflective and morphologically rich languages like Serbian, where a number of different word endings can be used to express the desired grammatical, syntactic, or semantic function of the given word. Each word ending may indicate a different morphological category, for example, word case, grammatical gender, or grammatical number. Other categories will be discussed as well – noun, pronoun and number type (subtype), degree of comparison for adjectives, verb form, transitivity and reflexivity, and so on. An approach like this can prevent speech recognition errors such as mistaken case, number or gender, even when the basic form of the word is correct.

In the end, experimental results are summarized, and the application possibilities of suggested methods and procedures in the field of automatic speech recognition on large vocabularies (for which they are primarily developed) are presented.

ZAHVALNICA

Zahvaljujem se profesoru Vladi Deliću koji me je zainteresovao za oblast govornih tehnologija, uveo me u nju i omogućio mi da se uspešno bavim ovom oblašću i nakon studija. Takođe se zahvaljujem Milanu Sečujskom, Nikši Jakovljeviću i ostalim kolegama sa Katedre za telekomunikacije i obradu signala na prijateljskoj saradnji i savetima tokom studija i kasnije tokom istraživanja i pisanja naučnih radova.

Zahvaljujem se i Darku Pekaru i preduzeću AlfaNum na pruženoj prilici da radim u oblasti koju volim, na ustupljenim računarima i softveru, odnosno na prijatnoj radnoj atmosferi i podršci u toku potrebnih istraživanja.

Veliku zahvalnost dugujem i kolegi i mentoru Branislavu Popoviću na uspešnoj saradnji koju imamo godinama, kao i na svoj pomoći tokom pripreme i pisanja ove disertacije.

Konačno, posebno se zahvaljujem i svojoj porodici, a pre svega supruzi Terezi na pruženoj podršci, strpljenju i razumevanju.

SADRŽAJ

SPISAK TABELA

SPISAK SLIKA

LISTA SKRAĆENICA

AM	Acoustic Model	akustički model
ANN	Artificial Neural Network(s)	veštačke neuronske mreže
ASR	Automatic Speech Recognition	automatsko prepoznavanje govora
BNF	Backus-Naur Form	Bakus-Naur format
BOS	Beginning Of Sentence	simbol za početak rečenice
BP	BackPropagation (algorithm)	algoritam propagacije unazad
BPTT	BackPropagation Through Time (algorithm)	algoritam propagacije unazad kroz vreme
CE / XE	Cross Entropy (criterion)	kriterijum unakrsne entropije
CER	Character Error Rate	stopa greške prepoznavanja karaktera (slova)
DBN	Deep Belief Network(s)	neuronske mreže dubokog uverenja
DNN	Deep Neural Network(s)	duboke neuronske mreže
EOS	End Of Sentence (symbol)	simbol za kraj rečenice
FFNN	Feed-Forward Neural Network(s)	neuronske mreže sa propagacijom unapred
FLM	Factored Language Model(s)	faktorisani jezički modeli
GMM	Gaussian Mixture Model(s)	modeli Gausovih smeša
HMM	Hidden Markov Model(s)	skriveni Markovljevi modeli
IWER	Inflective Word Error Rate	inflektivna stopa greške prepoznavanja reči
LDA	Linear Discriminant Analysis	linearna diskriminativna analiza
LM	Language Model	jezički model
LSTM	Long Short-Term Memory (unit)	jedinica duge kratkoročne memorije
LSTMP	LSTM Projected	LSTM sa projekcijom

LVASR	Large Vocabulary Automatic Speech Recognition	automatsko prepoznavanje govora na velikim rečnicima
LVCSR	Large Vocabulary Continuous Speech Recognition	prepoznavanje kontinualnog govora na velikim rečnicima
MDL	Minimum Description Length	minimalna dužina opisa
MFCC	Mel Frequency Cepstral Coefficient(s)	Mel-frekvencijski kepstralni koeficijenti
ML	Maximum Likelihood (estimation)	estimacija maksimizacijom verodostojnosti
NCCF	Normalized Cross-Correlation Function	normalizovana funkcija uzajamne korelacije
NCE	Noise Contrastive Estimation	estimacija poređenjem sa šumom
NFA	Nondeterministic Finite Automaton	nedeterministčan automat sa konačnim brojem stanja
NLP	Natural Language Processing	obrada prirodnog jezika
OOV	Out-Of-Vocabulary (word)	reč izvan rečnika (ASR sistema)
PER	Phoneme Error Rate	stopa greške prepoznavanja fonema
PLP	Perceptual Linear Prediction	perceptivna linearna predikcija
POS	Part-Of-Speech (tagging)	morfološko tagovanje
ReLU	Rectified Linear Unit	-
RMS	Root-Mean-Square	srednja kvadratna vrednost
RNN	Recurrent Neural Network(s)	rekurentne neuronske mreže
RNNLM	Recurrent Neural Network Language Model(s)	jezički modeli za bazi rekurentnih neuronskih mreža
SA	Speaker Adapted	adaptiran na govornika
SAT	Speaker Adaptive Training	obuka sa adaptacijom na govornika
SD	Speaker Dependent	zavisan od govornika
SGD	Stochastic Gradient Descent (algorithm)	stohastički algoritam opadajućeg gradijenta
SI	Speaker Independent	nezavisan od govornika

SNR	Signal-to-Noise Ratio	odnos signal-šum
SP	Speed Perturbation	perturbacija brzine (govora)
TDNN	Time Delay Neural Network	neuronska mreža sa vremenskim kašnjenjem
VTLN	Vocal Tract Length Normalization	normalizacija dužine vokalnog trakta
WER	Word Error Rate	stopa greške prepoznavanja reči
WFST	Weighted Finite-State Transducer	ponderisani transduktori sa konačnim brojem stanja

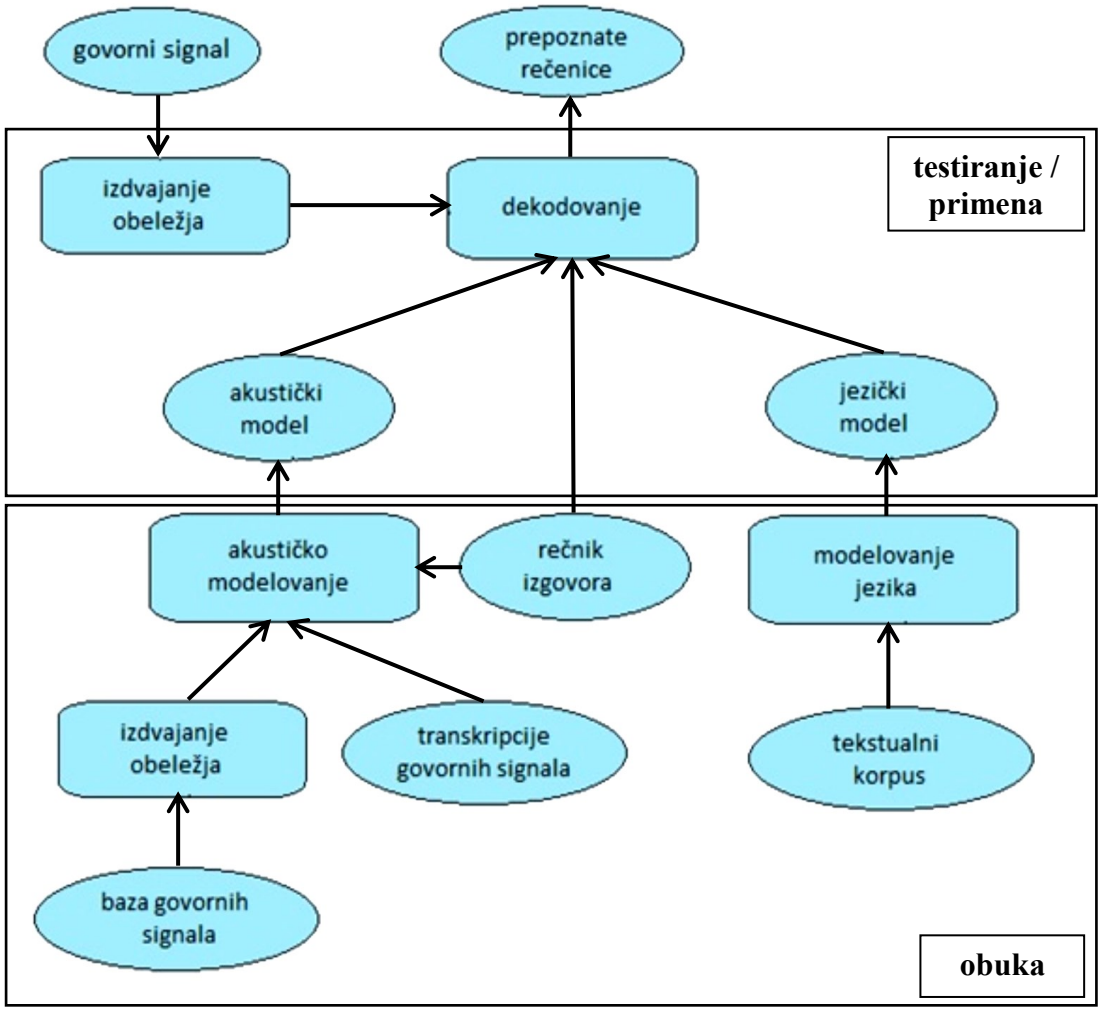
POGLAVLJE I:

UVOD U PROBLEMATIKU PREPOZNAVANJA GOVORA

Automatsko prepoznavanje govora (eng. *Automatic Speech Recognition*, ASR; ponekad se skraćeno naziva prepoznavanje govora) je tehnologija koja računarima omogućava pretvaranje izgovorenih reči u tekst radi dalje obrade i određivanja odgovarajuće akcije. U savremeno doba, postoje mnogi sistemi koji uključuju komunikaciju između čoveka i mašine, u kojima je ASR ključna komponenta, uz razumevanje prirodnog jezika, upravljanje dijalogom, generisanje prirodnog jezika i sintezu govora, to jest pretvaranja pisanog teksta u govor. Neke od brojnih primena ASR-a su sistemi za diktiranje, govorni asistenti na pametnim mobilnim uređajima, govorni automati, automatizovani pozivni centri, pametne kuće i kućni aparati, razni alati za pomoć osobama sa invaliditetom u komunikaciji sa računarima, sistemi za verifikaciju i autorizaciju govornika, sistemi za virtuelnu realnost i moderne video igre, ali ima i mnogih drugih. ASR komponenta često omogućava korisniku neke platforme da se ponaša prirodnije, kao i da lakše obavi ono što želi (na primer, bez šetanja po menijima i/ili kucanja teksta na tastaturi).

Svaki sistem za prepoznavanje govora ima dve glavne komponente, kao što se može videti na slici 1. Prva je akustički model (eng. *Acoustic Model*, AM), koji opisuje akustičke karakteristike odgovarajućih govornih jedinica (najčešće kontekstno-zavisnih fonema) za jednog ili više govornika. Druga je jezički model, (eng. *Language Model*, LM) koji treba da opiše rečnik ASR sistema – skup svih reči koje sistem može da prepozna, kao i pravila za formiranje rečenica u ciljanom jeziku ili domenu interakcije, to jest oblasti primene, i koji zajedno sa akustičkim modelom treba da odredi koja od mogućih rečenica je izgovorena i sa kojom verovatnoćom, odnosno cenom. Na kraju svakog prepoznavanja, rečenica sa najboljom verovatnoćom (najnižom cenom) odnosi pobedu, i postaje rezultat prepoznavanja izgovorene rečenice. Važnu ulogu obavlja i rečnik izgovora, koji modeluje veze

između reči i izabranih govornih jedinica, odnosno, specificira niz govornih jedinica koje čine svaku od reči iz rečnika. Dati rečnik izgovora se koristi i pri obuci ASR sistema i pri prepoznavanju tokom testiranja ili praktične primene, uz obučeni akustički i jezički model.



Slika 1. Dijagram ASR sistema

Akustički modeli se obučavaju na osnovu ulazne baze govornih signala i njihovih transkripcija, dok se jezički modeli obučavaju na osnovu ulaznog tekstualnog korpusa, koji može, ali ne mora da sadrži i pomenute transkripcije. U fazi testiranja ili praktičnog korišćenja, tokom dekodovanja obeležja datog test signala, ovi modeli određuju najverovatniju izgovorenu rečenicu.

STATISTIČKO MODELOVANJE ASR SISTEMA I IZBOR OBELEŽJA

Tokom dekodovanja, akustička komponenta ASR sistema ima zadatak da proceni koliko se dati govorni segment akustički poklapa sa predloženom rečenicom,

dok jezička komponenta treba da odredi verovatnoću da niz reči u predloženoj rečenici uopšte bude na taj način, odnosno tim redom izgovoren. Ove dve komponente se objedinjuju Bajesovim pravilom, koje je u domenu prepoznavanja govora dato sa

$$P(s|O) = \frac{P(O|s) \cdot P(s)}{P(O)} , \tag{1.1}$$

gde je $P(s|O)$ verovatnoća da je rečenica *s* izgovorena u datom govornom segmentu O, $P(O|s)$ verovatnoća da je izgenerisan segment O ako je izgovorena rečenica *s*, $P(s)$ apriori verovatnoća da rečenica *s* bude izgovorena, a $P(O)$ normalizacioni termin koji predstavlja verovatnoću pojave govornog segmenta O (Povey, 2003). Prilikom prepoznavanja neke audio sekvence O, obučeni ASR sistem će dati onu rečenicu *s* za koju je verovatnoća $P(s|O)$ najveća, uzevši u obzir sve statističke informacije koje sistem ima, akustičke i jezičke: u datoj formuli $P(O|s)$ predstavlja akustički model, to jest akustičku komponentu ASR sistema, a $P(s)$ model jezika, odnosno jezičku komponentu sistema. Akustički model je najčešće baziran na skrivenim Markovljevim modelima (eng. *Hidden Markov Models*, HMM) ili veštačkim neuronskim mrežama (eng. *Artificial Neural Networks*, ANN), pri tome posebno dubokim neuronskim mrežama (eng. *Deep Neural Networks*, DNN), a model jezika najčešće na statističkim *n*-gramima ili nekoj varijanti DNN, o čemu će više reči biti u poglavlju 2.

Govorni segment za prepoznavanje O nije predstavljen originalnim odbircima govornog signala – da bi bio omogućen rad ASR sistema, govorni segment se prvo mora obraditi. Prvo šta treba uraditi jeste izdeliti signal na manje, stacionarne segmente (najčešći izbor širine prozora koji se koriste pri ovakvom prozoriranju signala je oko 30 ms), a zatim, koristeći odgovarajuće tehnike iz oblasti digitalne obrade signala, pretvoriti svaki od tih segmenata, koji se nazivaju frejmovi, u vektor obeležja željene dimenzionalnosti, koji opisuje govor koji se nalazi u datom frejmu. Odabir vrste obeležja i dimenzionalnosti vektora obeležja je poseban deo projektovanja ASR sistema. Takođe, da bi se bolje modelovale brze promene obeležja (pre svega na prelazu između fonema), uzastopni frejmovi se međusobno preklapaju, a pomeraj između dva susedna frejma obično iznosi oko 10 ms – da nema preklapanja frejmova, promene na granicama frejmova bi mogle ostati nedetektovane jer je usled efekata prozoriranja smanjen uticaj vrednosti signala na ivicama frejmova. Izdvojeni

vektori obeležja za svaki frejm se obeležavaju sa o_1, o_2, … o_T, gde je T ukupan broj frejmova. Kada se oni nadovežu jedan na drugi, dobijamo ono što je označavano sa O – predstavu govornog signala koja odgovara ASR sistemu za obuku.

Prilikom izbora obeležja koja će opisivati govorne segmente, treba odabrati ona koja će funkcionisati dobro, a da bi se to postiglo, treba se pridržavati najmanje dva pravila: obeležja treba da budu međusobno nezavisna – da budu vrlo malo ili nimalo korelisana jedno sa drugim, i da budu informativna, u smislu da govore nešto korisno o segmentu koji predstavljaju, što će prepoznavač moći da iskoristi za raspoznavanje međusobno različitih segmenata, odnosno govornih jedinica.

Tipična obeležja koja se koriste u prepoznavanju govora su mel-frekvencijski kepstralni koeficijenti, skraćeno MFC koeficijenti, ili MFCC (eng. *Mel Frequency Cepstral Coefficients*) (Das i drugi, 2014). Kepstar se u obradi signala dobija primenom inverzne Furijeove transformacije na logaritam amplitude spektra govornog signala (Bogert i drugi, 1963). Da bi se dobili MFCC, spektar prethodno treba da se transformiše na mel-frekvencijsku skalu, koja povezuje percipiranu visinu glasa, to jest osnovnu učestanost (koja se takođe naziva i pič, od eng. *pitch*), sa stvarnom, na taj način što približava učestanosti onome kako ljudsko uho funkcioniše, a to je da mnogo bolje detektuje male promene piča kada su u pitanju niske učestanosti, nego kada se radi o visokim. Ne postoji jedinstvena transformacija učestanosti na mel skalu, a jedna od najčešće korišćenih data je formulom

$$mel(f[Hz]) = 2595 \cdot \log\left(1 + \frac{f[Hz]}{700}\right). \tag{1.2}$$

Transformacija data sa (1.2) jeste logaritamska za sve učestanosti, ali je približno linearna funkcija za učestanosti do 1000 Hz. Pič od tačno 1 kHz definisan je kao 1000 mela. Pokazuje se da MFCC lepo opisuju obvojnicu (anvelopu) spektra signala, a pošto od obvojnice spektra zavisi boja glasa, oni su vrlo korisni pri međusobnom raspoznavanju pojedinih govornih jedinica. Obično se koristi određen broj nižih MFCC (prvih n), jer viši koeficijenti nešto lošije opisuju govorne jedinice (iako istovremeno bolje opisuju pič). Međutim, MFC koeficijenti se uglavnom ne koriste bez pomoći drugih obeležja (Jakovljević i drugi, 2011). Osim MFCC, često se kao dodatno obeležje koristi energija signala, a ponekad i osnovna učestanost i nekoliko drugih varijanti obeležja vezanih za pič. MFC koeficijenti se ponekad zamenjuju PLP (eng. *Perceptual Linear Prediction*) koeficijentima (Hermansky,

1990), koji koriste Bark skalu umesto mel skale (Strube, 1998), i koji se ponekad pokazuju boljim izborom od MFCC kada postoji velika razlika između skupa za obuku sistema i skupa za testiranje (odnosno, između snimaka u obuci i u praktičnoj primeni) (Woodland i drugi, 1996). Inače, MFC koeficijenti imaju bolje performanse, pa se uglavnom koriste u svim savremenim ASR sistemima.

Osim ovih osnovnih, statičkih obeležja, u ASR sistemima koriste se i dinamička – prvi, odnosno drugi izvodi statičkih obeležja u vremenu, koja opisuju trend promene obeležja: brzinu – gradijent promene vremenski susednih statičkih obeležja, koji nazivamo delta obeležjem; i ubrzanje – gradijent promene susednih delta obeležja, koji nazivamo delta-delta obeležjem (Deng, 1994). Dinamička obeležja se nakon izračunavanja nadovezuju na originalni vektor statičkih obeležja o_i, za svaki frejm *i*. Za njihovo računanje, odnosno procenu vrednosti gradijenta, koriste se simetrični prozori od po nekoliko susednih frejmova pre i posle datog frejma.

Što se tiče nekih dodatnih tehnika koje se povremeno koriste u izdvajanju obeležja za ASR, ponekad se primenjuje linearna diskriminativna analiza (eng. *Linear Discriminant Analysis*, LDA), kojom se dobijena obeležja transformišu tako da se maksimizuje separacija među klasama govornih jedinica (Saon i drugi, 2000), zatim normalizacija srednjih vrednosti i/ili varijansi obeležja na nivou određenog govornog segmenta ili cele rečenice, čime se donekle potiskuju neke varijacije zavisne od govornika i komunikacionog kanala (ako uopšte ima različitih) (Viikki & Laurila, 1998), a takođe i normalizacija dužine vokalnog trakta (eng. *Vocal Tract Length Normalization*, VTLN), kojom se spektar govornog signala transformiše radi ujednačavanja razlika između muških i ženskih glasova (Welling i drugi, 1999). Kasnije tokom obuke se može javiti i transformacija obeležja sa ciljem adaptacije na željenog govornika, kojom se opisuju razlike između datog, pojedinačnog govornika i nekog usrednjenog, prosečnog govornika u datoj bazi podataka za obuku (Povey i drugi, 2008).

IZBOR GOVORNIH JEDINICA U ASR SISTEMU

Osim izbora obeležja koja će se koristiti u sistemu za prepoznavanje govora, mora se odlučiti i šta je tačno to što će se modelovati u akustičkoj komponenti ASR sistema, odnosno kakve govorne jedinice će se koristiti. U većini savremenih sistema,

za svaku reč koja postoji u govornoj bazi podataka za obuku moramo imati odgovarajuću fonetsku transkripciju, odnosno niz fonema koji odgovaraju izgovoru te reči. Fonemi su bazične jedinice govora – najmanji segmenti govora koji imaju sopstveno značenje ili funkciju, i delimično se podudaraju sa slovima u korišćenom alfabetu, odnosno glasovima u korišćenom jeziku, međutim, ima ih obično više nego samih slova, odnosno glasova. Konkretno, u srpskom jeziku osim standardnih slova u azbuci ili abecedi treba formirati posebne modele fonema za nenaglašene i naglašene verzije vokala (samoglasnika), a ponekad se pravi poseban model naglašenog vokala za svaki od četiri osnovna akcenta, kao i za produžen nenaglašen vokal (to jest vokal sa post-akcenatskom dužinom) (Pakoci i drugi, 2018). Osim njih, često se formira poseban model fonema za glas ’R’ kada je on nosilac sloga (taj fonem se obično naziva vokalno ’R’), a postoji i model takozvanog neutralnog vokala (ili „šva“ fonem), koji se javlja prvenstveno u izgovaranju izolovanih konsonanata (suglasnika), kao i posebna govorna jedinica za označavanje tišine. Dodatno, ponekad se koriste i oštećene varijante svih fonema, koje odgovaraju mestima u govoru gde neka reč ili deo reči nije izgovoren jasno ili na očekivan način (na primer, ako postoji značajna pozadinska buka), i služe samo da bi tokom obuke sprečili prljanje modela fonema tim nedovoljno kvalitetnim instancama, dok se u samom prepoznavanju govora ne koriste (Popović i drugi, 2014). Rečnik izgovora, odnosno leksikon, treba da sadrži sve validne izgovore, odnosno nizove fonema za sve reči koje postoje u skupu za obuku, a može sadržati i dodatne reči (sve reči koje želimo da konačni sistem može da prepozna tokom svoje upotrebe).

Osim u inicijalnim fazama obuke akustičkog modela za ASR, fonemi se tipično modeluju u levom i desnom kontekstu, odnosno imamo kontekstno-zavisne foneme (Gauvain & Lee, 1994). Najčešća varijanta takvih fonema su trifoni, kod kojih se posmatra po jedan fonem u levom i desnom kontekstu. Ovo se radi jer pojedini fonemi mogu da zvuče značajno drugačije zavisno od toga koji fonem se nalazi pre njih, odnosno, koji fonem dolazi nakon njih. Ovaj fenomen je poznat pod nazivom koartikulacija.

OBUKA AKUSTIČKIH MODELA U ASR SISTEMIMA

Cilj obuke i akustičkog i jezičkog modela u ASR sistemu jeste određivanje funkcije koja će proizvesti najmanju grešku prepoznavanja. To je vrlo težak zadatak. Takođe treba uzeti u obzir prilagođavanje test skupu – ako je mera kvaliteta datog sistema vezana za ostvaren rezultat na datom test skupu i samo njemu, lako možemo napraviti sistem koji ima lošu moć generalizacije na nešto drugačiji govorni materijal. Zavisno od dostupnog govornog materijala za obuku, sistem se takođe može bolje ili lošije snalaziti u određenom ambijentu (na primer, u prisustvu određenog tipa pozadinske buke), za pojedine govornike (zavisno od pola, akcenta, i slično), a uglavnom je veoma vezan za korišćeni jezik i ne može se lako prilagoditi nekom drugom.

Svi savremeni prepoznavači govora oslanjaju se na neki skup podataka za obuku. Skup podataka za obuku akustičkog modela sastoji se od određenog broja audio datoteka, najčešće sa po jednom izgovorenom rečenicom, i odgovarajućih tekstualnih transkripcija koje su formirali, odnosno pregledali anotatori – ljudi koji su pažljivo preslušali ceo audio materijal i svakoj datoteci pridružili odgovarajući tekst izgovorene rečenice. Govorne baze podataka se uglavnom snimaju tako što govornici čitaju unapred pripremljene rečenice (i takav materijal bi anotatori trebalo da provere, jer su moguće razne greške u snimanju). Svakako je lakša priprema materijala za obuku sistema ako transkripcije već postoje, međutim, često smo prinuđeni da koristimo i drugačije vrste materijala, na primer, slobodno dostupne radio emisije. Razvrstavanje takvog materijala na rečenice i govornike, uz zapisivanje šta je izgovoreno, je vrlo naporan i dugotrajan posao. Kada se pripremi konačan skup za obuku, zadatak je pronalaženje funkcije bazirane na podacima za obuku koja će se koristiti za evaluaciju novih rečenica (onih koje nisu učestvovale u obuci). Očigledno je da ta funkcija mora uspešno da vrši generalizaciju – da naučeni parametri mogu uspešno da se primenjuju i na materijalu van skupa za obuku, odnosno da sistem ne bude preobučen na dati skup za obuku. Generalizacija se može izvršiti na različite načine, a cilj je pronaći odgovarajući, najbolji način.

Današnji prepoznavači govora su najčešće ručno dizajnirani, u smislu da im ljudi zadaju tip, arhitekturu i složenost, bez mnogo elemenata automatskog traženja, na primer, optimalne strukture ili broja parametara. Postoji više razloga za to (Povey,

2003). Prvo, vrlo je teško vršiti pretragu u prostoru svih mogućih funkcija, to jest sistema. Zatim, zbog ograničenosti skupa za obuku (u praksi je on uvek ograničen), automatska pretraga lako može upasti u zamku preobučavanja na dati skup za obuku, jer će verovatno pretražiti jako veliki broj mogućnosti. To posebno važi za manje skupove za obuku (što je tipično za jezike sa malo govornih resursa). Takođe, potrebno vreme je veliki problem – svaka iteracija isprobavanja nove potencijalno optimalne funkcije može uključivati kompletnu obuku ASR sistema, koja sama za sebe može biti prilično dugotrajna (u današnje vreme, i na boljim računarima najčešće traje barem par desetina sati, a na slabijim računarima i dosta duže). Konačno, postojeće tehnike za pronalaženje optimalnog novog algoritma (genetsko programiranje i slično) uglavnom ne rade dovoljno dobro za složenije zadatke. Svi ovi problemi doprinose tome da se favorizuje ručno zadavanje karakteristika ciljanog sistema, iako će se tako teško pronaći baš optimalan sistem (u moru mogućnosti), i iako je ovo rešenje skuplje – jer treba zaposliti programere da to urade. Neki kompromis može biti da programeri zadaju niz određenog broja mogućnosti za konačan sistem, od kojih će automatski biti odabrana najbolja (automatskom optimizacijom pojedinih parametara sistema).

Veoma dugo su u implementacijama akustičkog dela ASR sistema dominirali statistički pristupi koji se oslanjaju na kombinaciju HMM i modela Gausovih smeša (eng. *Gaussian Mixture Models*, GMM) (Rabiner i drugi, 1985; Rabiner, 1989; Delić i drugi, 2010). U poslednje vreme, sa znatnim poboljšanjem performansi računarskih komponenti – pre svega centralnih (CPU) i grafičkih (GPU) procesora, kao i sve pristupačnijim resursima u oblaku (eng. *cloud*), moderni sistemi su počeli da se oslanjaju i na ANN, odnosno DNN (Dahl i drugi, 2013; Popović i drugi, 2015a), što je donelo dalja poboljšanja u rezultatima. Konkretna implementacija akustičkog dela jednog ASR sistema za srpski jezik predstavljena je u poglavlju 6.

Parametri HMM sistema se uobičajeno iterativno estimiraju maksimizacijom verodostojnosti (eng. *Maximum Likelihood*, ML) na rečenicama u okviru datog skupa za akustičku obuku ASR sistema (Rabiner, 1989). Dodatno, postoje različite diskriminativne metode obuke koje se oslanjaju na korišćenje datog skupa za obuku za dalju optimizaciju parametara pojedinih GMM komponenti, to jest njihovih srednjih vrednosti i varijansi, kojima su opisane odgovarajuće govorne jedinice, tako da rezultujući model što tačnije prepozna te rečenice za obuku (Bahl i drugi, 1986;

Povey, 2003). One se oslanjaju na odgovarajuće diskriminativne funkcije cilja, koje opisuju koliko dobro je dati skup podataka za obuku prepoznat.

Duboke neuronske mreže se takođe iterativno optimizuju, imajući u vidu određenu željenu arhitekturu. Često se koristi kriterijum unakrsne entropije (eng. *cross entropy*, CE; često i XE, ili *Xent*) (Bourlard & Morgan, 1994) uz stohastički algoritam opadajućeg gradijenta (eng. *Stochastic Gradient Descent*, SGD) (Bottou, 2010), a postoji nekoliko varijanti implementacije, čiji pojedini detalji su dati u poglavlju 2. Moguće je naravno koristiti i druge algoritme učenja mreže, ali oni nisu tema ove disertacije. Vreme trajanja cele procedure zavisi mnogo od složenosti mreže, paralelizacije i primene raznih načina za pojednostavljenje delova mreže ili aproksimaciju pojedinih funkcija u korišćenim algoritmima (ako je moguće bez bitnog pogoršanja performansi).

Automatsko prepoznavanje govora se dodatno može unaprediti adaptacijom na ciljanog govornika, ako je u pitanju aplikacija koju će koristiti jedan ili konačan broj osoba, i to se primenjuje i na HMM i na DNN modelima za ASR (Leggetter & Woodland, 1995; Dehak i drugi, 2011). Tako dobijamo sisteme za prepoznavanje govora zavisne od govornika (eng. *Speaker Dependent*, SD; ili *Speaker Adapted*, SA). Ako to nije slučaj, koriste se modeli nezavisni od govornika (eng. *Speaker Independent*, SI), koji imaju lošije performanse na zadatom govorniku u odnosu na slučaj sa adaptacijom (pogotovo ako postoji nešto specifično u njegovom govoru), ali se generalno, u proseku, ponašaju bolje na slučajnoj grupi govornika. Pojedini HMM i DNN modeli imaju i mogućnost da se adaptiraju na trenutnog govornika u hodu, odnosno tokom samog prepoznavanja (Povey i drugi, 2008; Pakoci i drugi, 2018).

OBUKA JEZIČKIH MODELA U ASR SISTEMIMA

Kao što je već spomenuto, uz akustički deo sistema za prepoznavanje govora, koji je opisan odgovarajućom HMM ili DNN arhitekturom, vrlo bitan je još jedan deo, a to je jezički. Jezički deo ASR sistema služi da opiše i/ili ograniči spisak i redosled reči koji sistem može da prepozna. On je vrlo bitan, jer koliko god da je sistem akustički dobro obučen, i dalje će teško prepoznavati potpuno proizvoljan niz reči bez dodatnog predznanja, jer često neka reč nije savršeno izgovorena, često postoje razne sitne greške u govoru, a i većina sistema ima tačno definisanu primenu,

pa bi bilo neodgovorno ne iskoristiti lingvističko znanje za predviđanje onoga što može biti rečeno. Naredna reč umnogome zavisi od nekoliko prethodnih, a postoje i druga ograničenja – vrsta reči, padež, gramatički rod, gramatički broj, i tako dalje. Ova ograničenja treba dobro da budu opisana odgovarajućim modelom jezika.

Uglavnom se koriste dve vrste jezičkih modela. Jedna vrsta se oslanja na striktno definisanje svih dozvoljenih sekvenci reči u datoj primeni sistema (takozvane gramatike) (Hopcroft & Ullman, 1979), a takvi modeli su najkorisniji za konkretne aplikacije sa uskom oblasti primene. Drugi tip modela je statistički, a kod njega se opisuju verovatnoće pojave pojedinih reči ako znamo istoriju prethodno izgovorenih reči, odnosno kontekst (Rosenfeld, 2000; Xu & Rudnicky, 2000). Ovaj tip je usko vezan za dati jezik, a dodatno se može adaptirati obukom na skupu tekstova konkretno vezanim za određeni stil govora ili domen upotrebe. Ovakav model jezika može opisati i razne dodatne karakteristike reči (sintaksne, leksičke ili morfološke), što bi moglo da poboljša eliminaciju sitnijih grešaka u prepoznavanju (kao što su, na primer, pogrešni završeci reči).

Nekoliko decenija su jezici širom sveta modelovani statističkim *n*-gram pristupom, u kom se opisuju verovatnoće pojave naredne reči u sekvenci uz aproksimaciju da je za njihovo određivanje od koristi poznavanje samo *n*-1 prethodnih reči u datoj sekvenci (Rosenfeld, 2000). Ovakav pristup je bio prilično uspešan za većinu primena ASR sistema sa malim i srednjim rečnicima. Kao što se desilo i u akustičkom modelovanju, u poslednje vreme (pre svega u poslednjoj deceniji) je počela upotreba veštačkih neuronskih mreža za modelovanje jezika, koje mogu da reše mnoge probleme *n*-grama (Oparin i drugi, 2012). Ti problemi su umnogome vezani za zavisnost performansi *n*-gram modela od veličine tekstualnog korpusa za njegovu obuku, odnosno od potrebe za mnogo većim korpusima za iole veće rečnike da bi se dovoljno precizno odredile sve potrebne statistike, što je za mnoge jezike veoma teško ispuniti. Neuronske mreže su pokazale superiornost, doduše uz neminovno značajno povećanje računske složenosti, kao i produžavanje trajanja obuke (što opet veoma zavisi od dostupnih računskih resursa).

EVALUACIJA ASR SISTEMA

Problem prepoznavanja govora jeste pretvaranje datog govornog materijala (audio datoteke ili sekvence audio odbiraka) u pisani tekst, odnosno niz reči koji se što približnije poklapa sa onim što bi transkribovao čovek koji je preslušao isti materijal. Zadatak jeste pronaći neku funkciju koja ovaj posao radi što je moguće bolje, to jest uspešnije. Performanse ASR sistema se najčešće procenjuju na nekom test skupu govornih sekvenci koji sadrži izvestan broj audio datoteka (odgovarajućeg trajanja), obično određen kao procenat celokupnog dostupnog audio materijala (po broju datoteka ili trajanju). Rezultat testa se najčešće prikazuje u obliku stope greške prepoznavanja reči (eng. *Word Error Rate*, WER), koja se izražava u procentima u odnosu na ukupan broj reči u test skupu i data je sa

$$WER\ [\%]\ = \frac{100 \cdot (\#zamena + \#umetanja + \#brisanja)}{\#referentnih\_reči}, \qquad (1.3)$$

gde je *#referentnih_reči* ukupan broj reči u ispravnim transkripcijama test skupa, a ostale tri vrednosti u izrazu su broj zamenjenih, umetnutih i obrisanih reči respektivno u odnosu na ispravnu transkripciju, kada se međusobno poravnato posmatraju ispravna sekvenca reči i prepoznata hipoteza (a uvek se poravnaju tako da se WER minimizuje). Zavisno od primene, u današnjim ASR sistemima se tolerišu različite vrednosti za WER – naravno da sistemi za specifične namene sa malim brojem reči u opticaju treba manje da greše od nekog sistema opšte namene za spontani govor.

Osim WER, kao mera performanse ASR sistema ponekad se koriste i stopa greške na nivou slova (eng. *Character Error Rate*, CER) i stopa greške na nivou fonema (eng. *Phoneme Error Rate*, PER), a za visoko inflektivne jezike, gde jedan osnovni oblik reči (odnosno lema) može da ima niz različitih završetaka kojima se specificira uloga te reči u rečenici (na primer padež), u koju grupu spada i srpski jezik, ponekad se koristi i takozvana inflektivna stopa greške na nivou reči (eng. *Inflective Word Error Rate*, IWER). Dok se CER i PER računaju na identičan način kao standardni WER, samo što se umesto reči u ispravnom transkriptu i hipotezi koriste slova, odnosno fonemi koji čine odgovarajuće reči, IWER se dobija tako što se u WER računici greške onih zamena koje su posledica pogrešnog oblika reči za ispravno prepoznatu lemu množe nekim koeficijentom manjim od 1, najčešće 0,5

(Bhanuprasad & Svenson, 2008). Poređenjem WER i IWER vrednosti može se donekle doći do zaključka koliko grešaka ASR sistema čine sitne greške u kojima je pogrešan samo konkretan oblik neke reči, dok je osnovni oblik prepoznat kako treba. Međutim, ni takve greške se ne smeju zanemarivati u određenim primenama, na primer u sistemima za diktiranje gde treba zapisati tačno ono što govornik kaže. CER i PER vrednosti za srpski jezik su približno jednake, jer se reči zapisuju manje-više onako kako se i izgovaraju, tako da nema značajnih razlika između niza slova koja čine ortografski oblik reči i niza fonema koji čine njen izgovor.

PREDMET ISTRAŽIVANJA I MOTIVACIJA

Predmet istraživanja ove disertacije je jezički model, tako da je u disertaciji opisivan uglavnom on. Osnovne informacije o jezičkom modelovanju date su već u sekciji „Obuka jezičkih modela u ASR sistemima“. U oblasti modelovanja srpskog jezika, nakon dugog perioda korišćenja *n*-gram modela i gramatika (Popović i drugi, 2015a; Popović i drugi, 2015b; Pakoci i drugi, 2018; Ostrogonac, 2018), prethodnih godina je isprobano i upoređeno nekoliko varijanti modela baziranih na rekurentnim neuronskim mrežama (eng. *Recurrent Neural Networks*, RNN; odnosno, *Recurrent Neural Network Language Models*, RNNLM) (Popović i drugi, 2018). Svi oni su bili značajno bolji u pogledu tačnosti prepoznavanja reči od referentnog *n*-gram modela, ali i dalje su postojali izvesni problemi. Uočeno je da i dalje postoje greške u kojima je pogođena lema, ali je pogrešan završetak reči, koji može da označava određeni padež, rod, broj, ili slično. Kao što je već navedeno, srpski jezik spada u grupu visoko inflektivnih jezika, pa bi stoga obučavanje sistema tako da ume da razlikuje različite morfološke oblike jedne reči potencijalno moglo značajno da doprinese poboljšanju performansi.

Konkretno, u disertaciji je predstavljeno korišćenje pristupa za modelovanje jezika preko RNN koji inkorporira dodatna obeležja reči kao što su njihova učestanost u korpusu za obuku (odnosno, njihova unigram log-verovatnoća), njihova dužina, kao i skup slovnih *n*-grama (grupa uzastopnih slova do dužine *n*) koji se mogu naći u njima. Osim toga, biće predstavljen i modifikovani pristup koji obraća pažnju i na različite morfološke kategorije koje se vezuju za svaku od reči u skupu za obuku, i koristi te informacije kao dodatna obeležja za formiranje konačnog modela.

Ranije obučeni RNN modeli za srpski jezik su, iako puno bolji od *n*-grama po ostvarenoj stopi greške, i dalje bili daleko od savršenstva, a uz to su bili prilično kompleksni, zahtevali instalaciju i korišćenje dodatnih spoljnih alata i obučavali se veoma dugo, i povrh svega zahtevali i prilične računske resurse.

Cilj urađenog istraživanja bio je da ispita mogućnosti razvoja novog RNN modela koristeći kao polaznu osnovu najmodernija postojeća rešenja u svetu, ali sa specifičnostima za srpski jezik i dodatnim morfološkim obeležjima koja će potencijalno omogućiti dalje značajno smanjenje stope greške sistema za prepoznavanje govora. Predloženi pristupi su testirani na najopsežnijoj postojećoj test bazi za srpski jezik i upoređeni sa ranijim pristupima na istoj ili sličnim bazama (Popović i drugi, 2018; Pakoci i drugi, 2019). Testirana je hipoteza da se korišćenjem dodatnih i leksičkih i morfoloških obeležja kao informacija o pojedinim rečima u rečniku sistema za automatsko prepoznavanje govora mogu bitno poboljšati njegove performanse. Pretpostavka je da se uvođenjem pomenutih dodatnih informacija u proceduru obuke modela jezika može pomoći modelu da preciznije opiše pravila za formiranje očekivanih nizova reči u srpskom jeziku.

U poglavlju 2 dat je detaljan opis procedure modelovanja jezika – osnovni pojmovi, istorijat, načini evaluacije i aktuelno stanje u oblasti u svetu i za srpski jezik. Nakon toga opisani su dostupni resursi za obuku i akustičkog i jezičkog modela za srpski jezik u poglavlju 3, najvažniji korišćeni alati u kreiranju ASR sistema za srpski jezik na velikim rečnicima (eng. *Large Vocabulary Continuous Speech Recognition*, LVCSR; ponekad *Large Vocabulary Automatic Speech Recognition*, LVASR), kao i istaknuta dosadašnja dostignuća u poglavlju 4. U poglavlju 5 fokus prelazi na morfološka obeležja u srpskom jeziku, kao i alat koji omogućava ubacivanje dodatnih leksičkih i morfoloških obeležja u model jezika. Konačno, u poglavljima 6 i 7 su prikazani obavljeni eksperimenti sa detaljima postavke, eksperimentalni rezultati i odgovarajući zaključci.

POGLAVLJE II:

MODELOVANJE JEZIKA

Model jezika je, pored akustičkog modela, druga značajna komponenta svakog sistema za automatsko prepoznavanje govora. Njegova funkcija je da zada sistemski rečnik, odnosno spisak svih reči koje mogu biti prepoznate u ulaznom govornom signalu, kao i da opiše pravila po kojima se te reči mogu povezati u rečenice u datom jeziku (u ASR sistemima opšte namene) ili u specifičnom domenu upotrebe. Zajedno sa akustičkim modelom, on estimira verovatnoće, odnosno cene za izgovaranje pojedinih rečenica zavisno od ulaznih akustičkih obeležja izračunatih na osnovu govornog signala, i na taj način omogućuje ASR dekoderu da odredi najverovatniju sekvencu reči, odnosno rečenicu sa najmanjom ukupnom cenom.

Naučna oblast u okviru koje se kao jedan od glavnih zadataka pojavljuje modelovanje jezika jeste obrada prirodnog jezika (eng. *Natural Language Processing*, NLP). Njeni počeci se javljaju sredinom dvadesetog veka. Tada se pojavljuju prvi komunikacioni sistemi na relaciji čovek-mašina sa ciljem da ta komunikacija bude što približnija prirodnom jeziku (Weizenbaum, 1966). Ovi sistemi su se prvenstveno bazirali na skupu ručno zadatih pravila, i naravno, kako zbog samih ograničenih mogućnosti takvog načina određivanja pravila, tako i zbog nemogućnosti preciznog određivanja bilo kakvih parametara takvog sistema, nisu bili ni za kakvu širu namenu. Krajem dvadesetog veka, zajedno sa pomacima u razvoju računarskih tehnologija, pojavili su se i prvi sistemi za modelovanje jezika koji su se obučavali na datom ulaznom skupu tekstova (tekstualnom korpusu), na osnovu kojih su se, automatski ili poluautomatski, nekom statističkom analizom izdvajala pravila za generisanje rečenica (Manning & Schütze, 1999). Vrlo brzo je došlo do rešenja mnogih problema u ovoj naučnoj oblasti. Uz pomoć koncepata i znanja iz drugih srodnih oblasti (pre svega matematike i statistike), obrada prirodnog jezika je nastavila da evoluira, kako na sintaksnom i semantičkom nivou, tako i na fonološkom, morfološkom, i drugim

nivoima. Tek u poslednjoj deceniji u priču su ušle i neuronske mreže sa nekoliko različitih varijanti implementacije. One su se pokazale kao dobro rešenje za modelovanje složenih zavisnosti među različitim segmentima prirodnog jezika (Mikolov i drugi, 2011a).

U teoriji, model jezika treba da prestavlja raspodelu verovatnoća nad skupom svih mogućih nizova reči u datom jeziku, ili željenom domenu upotrebe. U praksi, pošto nije moguće uzeti u obzir apsolutno sve moguće nizove reči, estimiraju se verovatnoće samo pojedinih nizova reči. Ove raspodele verovatnoća se mogu predstaviti preko nekoliko različitih oblika modela jezika – regularnih gramatika, *n*-gram modela jezika i modela baziranih na neuronskim mrežama, koji će detaljno biti predstavljeni u narednim sekcijama. Matematika koja stoji iza svakog od ovih modela je uglavnom nezavisna od konkretnog jezika ili ciljane primene, međutim, sama obuka modela, odnosno njihova upotreba (način upotrebe, odnosno konkretna aplikacija) jako zavisi od vrste materijala, odnosno tekstova za obuku, količine materijala, kao i kvaliteta podataka koji su na raspolaganju. Na primer, za jezike koji su visoko inflektivni i imaju vrlo kompleksnu morfologiju, kao što je srpski jezik, korpus za obuku modela jezika bi trebalo da bude znatno obimniji i da obuhvata što više različitih konteksta koji se mogu naći u rečenicama (a da odgovaraju željenoj primeni), za razliku od jezika sa dosta jednostavnijom morfologijom, kao što je recimo engleski jezik. Svakako je sama procedura prikupljanja i pripreme materijala za obuku izuzetno duga i zahtevna, pa mnogi jezici i dalje nemaju adekvatne resurse za obuku kvalitetnih modela jezika, pre svega za rad sa velikim rečnicima, na primer od stotinu hiljada reči i više.

EVALUACIJA MODELA JEZIKA

Kvalitet datog jezičkog modela se generalno može odrediti na dva načina (Chen i drugi, 1998). Jedan način predstavljaju mere vezane za konkretnu primenu, to jest spoljašnje mere. Ako je ta primena sistem za automatsko prepoznavanje govora, od spoljašnjih mera svakako najrasprostranjenija je stopa greške na nivou reči (u ciljanim test uslovima), o kojoj je bilo reči u uvodu. WER se najčešće smatra objektivnom merom kvaliteta modela jezika, međutim, on zavisi i od akustičkog modela i njegove usaglašenosti sa jezičkim modelom (na primer da li je akustički

model obučavan na sličnom tipu govornog materijala kao što su i tekstovi na kojima se obučava jezički model), a svakako je veoma značajna i reprezentativnost test skupa za određivanje WER, pri čemu je vrlo bitno koristiti nešto što je prilagođeno željenoj aplikaciji. Još jedna mana evaluacije uz pomoć stope greške prepoznavanja je vreme, koje u zavisnosti od test skupa i računskih resursa na raspolaganju može biti dosta dugačko.

Zbog svega navedenog često su u upotrebi i unutrašnje mere, odnosno mere koje su nezavisne od konkretne primene modela jezika. Tipična mera ovakve prirode je perpleksivnost (Jelinek i drugi, 1977). Perpleksivnost, u opštem slučaju, meri koliko dobro model jezika predviđa test podatke. Dobar model jezika je model koji najčešće dodeljuje visoke verovatnoće datim test rečenicama (pretpostavlja se da su test rečenice ispravne). Neki LM koji nije naročito „pametan“ može pretpostaviti da su sve reči u rečniku jednako verovatne i zato ih modelovati diskretnom uniformnom raspodelom verovatnoća (ako je ukupan broj reči u rečniku *N*, sve verovatnoće su jednake $1/N$). Perpleksivnost ovakvog modela jezika bi iznosila *N*. To svakako nije dobro, jer nisu sve reči jednako verovatne, a i određene reči i fraze se često pojavljuju jedne uz druge. Poboljšanjem modela jezika, perpleksivnost modela opada. Na primer, ako imamo rečnik od *N* = 10000 reči, a izračunata perpleksivnost je 100, to znači da je u proseku za svaku narednu reč u rečenicama test skupa broj opcija sa datim LM smanjen sa 10000 na 100 reči.

Matematički, perpleksivnost se za diskretan skup verovatnoća $p(x)$ određuje izrazom

$$PPL = 2^{H(p)} = 2^{-\sum_x p(x) \log_2 p(x)}, \tag{2.1}$$

gde je $H(p)$ entropija datog skupa verovatnoća $p(x)$, a x je vrednost iz skupa svih mogućih vrednosti date slučajne promenljive. Entropija (Shannon, 1948) prestavlja prosečan broj bita potrebnih za kodovanje informacija sadržanih u slučajnoj promenljivoj, tako da se perpleksivnost na osnovu izraza (2.1) može interpretirati kao ukupna količina svih mogućih informacija, ili, preciznije, prosečni broj izbora koji slučajna promenljiva ima. Na primer, ako se prosečna rečenica u test skupu može kodovati sa 10 bita, perpleksivnost modela iznosi $2^{10} = 1024$. Manja entropija (to jest manje neuređen sistem) je svakako povoljnija od veće entropije, pošto su predvidljivi rezultati poželjniji u odnosu na slučajne. Stoga je i mala perpleksivnost dobra, a

velika loša. U idealnom slučaju, poređenje perpleksivnosti modela i prave perpleksivnosti jezika bi bila najbolja mera kvaliteta datog modela.

Perpleksivnost se, u kontekstu modelovanja jezika, još može definisati i kao recipročna vrednost verovatnoće datog niza reči iz skupa podataka za testiranje, normalizovana dužinom tog niza (brojem reči u njemu). Do tog zaključka se može doći matematički. Ako diskretni izvor informacija generiše niz reči $w_1, w_2, \dots, w_n$ iz datog rečnika, entropija ovakvog izvora na nivou reči je data sa

$$H(p) = -\lim_{n\to\infty}\frac{1}{n}\sum_{w_1,\dots,w_n} P(w_1, w_2, \dots, w_n)\log_2 P(w_1, w_2, \dots, w_n)\,. \quad (2.2)$$

Sumiranje treba vršiti po svim mogućim nizovima reči. Međutim, može se pretpostaviti ergodičnost (Walters, 1982), između ostalog, zbog činjenice da se datim jezikom možemo uspešno služiti i kada nismo prethodno čuli ili videli sve moguće reči tog jezika, a i jer značenje reči možemo uglavnom utvrditi na osnovu relativno male količine teksta ili govora koji joj prethodi. Uzevši ergodičnost u obzir, formula (2.2) dobija oblik

$$H(p) = -\lim_{n\to\infty}\frac{1}{n}\log_2 P(w_1, w_2, \dots, w_n)\,. \quad (2.3)$$

Za dovoljno veliku vrednost *n*, entropija se približno može proceniti kao

$$\hat{H}(p) = -\frac{1}{n}\log_2 P(w_1, w_2, \dots, w_n)\,. \quad (2.4)$$

Odnosno, perpleksivnost se može dobiti izrazom

$$PPL = 2^{\hat{H}(p)} = \hat{P}(w_1, w_2, \dots, w_n)^{-\frac{1}{n}}, \quad (2.4)$$

gde je $\hat{P}(w_1, w_2, \cdots, w_n)$ verovatnoća koju je jezički model dodelio datom nizu reči dužine *n* iz test skupa podataka. Uz dodatnu pretpostavku da verovatnoća naredne reči u sekvenci zavisi samo od prethodnih *m* reči (a ne od kompletne istorije reči), primenom lančanog pravila na izraz (2.4) dobija se

$$PPL = \sqrt[n]{\frac{1}{\prod_{i=1}^{n} P(w_i|w_{i-m+1}, \dots, w_{i-1})}}, \quad (2.5)$$

gde *m* kod ovakvog tipa modela predstavlja red modela, a oni se u opštem slučaju nazivaju *n*-gram (a u ovom slučaju *m*-gram) modeli jezika.

Još jedno zapažanje jeste da ako bi dati model jezika određenom skupu test reči dodelio verovatnoću 0, perpleksivnost bi bila beskonačna. Ovo je jedan od razloga što je u NLP uveden koncept ublažavanja (eng. *smoothing*), o čemu će više reči biti u sekciji „Statistički *n*-gram modeli".

Problem sa merom perpleksivnosti jeste i činjenica da model sa niskom vrednošću ove mere ne mora uvek da funkcioniše dobro u praktičnoj primeni, pogotovo ako u toj primeni podaci nemaju isti raspodelu verovatnoća kao što je imao test skup. Zaista, pojedina istraživanja su pokazala da perpleksivnost u određenim primenama nije značajno korelisana sa stvarnim performansama datog LM (Klakow & Peters, 2002). Međutim, u nedostatku drugih efikasnih načina za evaluaciju modela jezika, perpleksivnost i dalje jeste korisna mera za poređenje različitih modela.

VRSTE MODELA JEZIKA

REGULARNE GRAMATIKE

Jedan od jednostavnijih načina za formiranje modela jezika je preko gramatika. Vrlo su korisne u slučajima kada prepoznavač govora treba da funkcioniše u vrlo uskom domenu upotrebe, sa ograničenim rečnikom i strogim pravilima formiranja rečenica. Gramatike omogućavaju direktno navođenje svih mogućih rečeničnih struktura u željenom ASR sistemu.

Pod formalnim jezikom se podrazumeva svaki skup reči nad datim alfabetom, pri čemu se jezik može specificirati navođenjem svih reči koje se nalaze u njemu (Rozenberg & Salomaa, 1997). Ipak, mora se pronaći način da se u konačnoj formi specificiraju i jezici sa beskonačno mnogo elemenata. Upravo to omogućavaju formalne gramatike (Hopcroft & Ullman, 1979). One predstavljaju fiksan skup pravila za generisanje reči formalnog jezika, odnosno validnih nizova simbola nad datim alfabetom. Formalne gramatike se mogu koristiti i za produkciju reči i rečenica iz datog jezika, ali i za proveru da li dati niz reči pripada konkretnom jeziku, što je korisno pri dekodovanju u ASR sistemu.

Svaka gramatika ima početni simbol *S* i niz pravila produkcije simbola *P*. Simboli mogu biti terminalni (Σ) i neterminalni (N); neterminalni simboli predstavljaju apstraktne oznake za gramatičke konstrukcije sastavljene od više simbola, terminalnih i/ili drugih neterminalnih. Primer formalne gramatike je dat u nastavku. Ako su u ponudi simboli $N = \{S\}$ i $\Sigma = \{a, b\}$, možemo imati na primer:

$$\begin{matrix} S \to aSb \\ S \to ba \end{matrix} . \tag{2.6}$$

Primer generisanja sekvence simbola ovom gramatikom može biti recimo

$$S \to \boldsymbol{aSb} \to a\boldsymbol{aSb}b \to aa\boldsymbol{ba}bb\,. \tag{2.7}$$

U sekvencama simbola datim u (2.7) podebljani su novi simboli dobijeni od neterminalnog simbola u prethodnoj iteraciji. Zapravo, za dati primer, gramatika generiše beskonačno mnogo sekvenci simbola koje se mogu zajedno predstaviti izrazom $\{a^n bab^n \mid n \geq 0\}$.

Za formalnu gramatiku se kaže da je desno-linearna ako su sva pravila iz skupa *P* u jednom od sledećih oblika:

$$\begin{matrix} A \to a \\ A \to aB \end{matrix} , \tag{2.8}$$

gde važi $a \in \Sigma$, i $A, B \in N$, odnosno levo-linearna ako su sva pravila tipa:

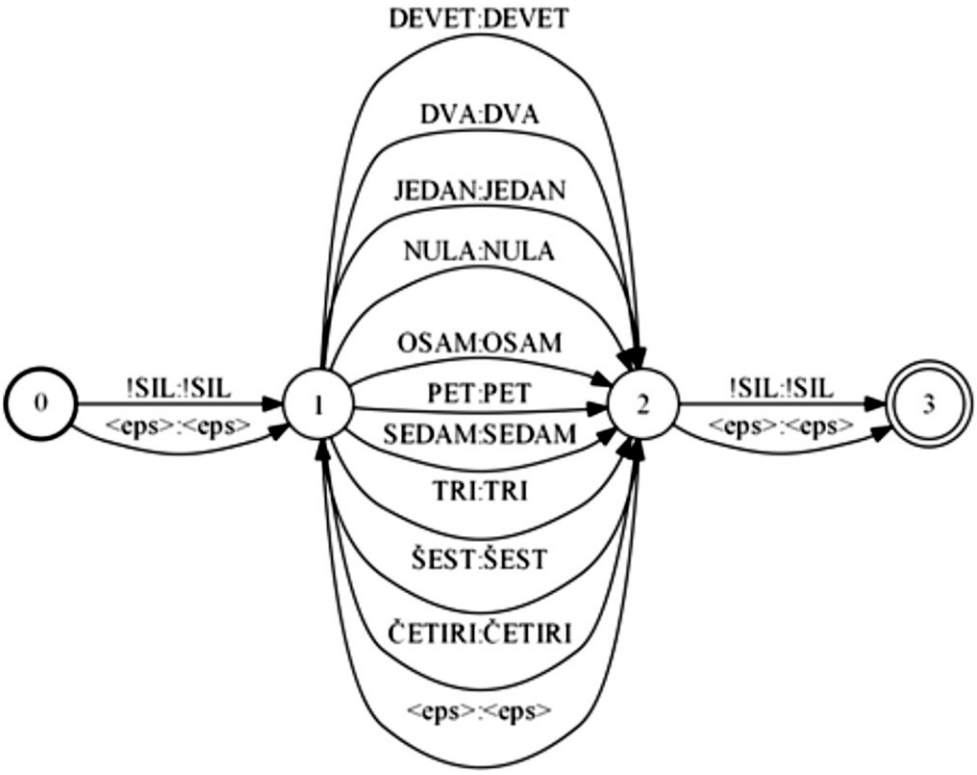
$$\begin{matrix} A \to a \\ A \to Ba \end{matrix} . \tag{2.9}$$

Dodatno implicitno pravilo u oba slučaja je $A \to \varepsilon$, gde ε predstavlja epsilon simbol, odnosno prazan simbol (simbol dužine 0). Ova dva tipa gramatika se zajedno nazivaju regularne gramatike, a jezik koji je generisan datom regularnom gramatikom naziva se regularan jezik. Može se pokazati da postoji 1-1 veza između pravila date desno-linearne regularne gramatike i nedeterminističnog automata sa konačnim brojem stanja (eng. *Nondeterministic Finite Automaton*, NFA), takva da data gramatika generiše upravo onakav jezik koji automat prihvata (ovo je Klinova teorema) (Rozenberg & Salomaa, 1997). Stoga, desno-linearne regularne gramatike generišu sve regularne jezike. Levo-linearne regularne gramatike opisuju direktne suprotnosti takvih jezika, odnosno, takođe regularne jezike. Dok regularne gramatike opisuju isključivo regularne jezike, suprotno ne važi – regularni jezici takođe mogu biti opisani i gramatikama koje nisu regularne.

Jedan primer regularne gramatike dat je na slikama 2 i 3. U pitanju je gramatika koja može da generiše niz cifara (od 0 do 9), od jedne do beskonačno mnogo, a pre i posle tog niza opciono prihvata tišinu (tišina se često posebno denotira u ASR sistemima, na primer specijalnom „rečju“ *!SIL*, i u modelima jezika se beleži na mestima gde se želi naglasiti da mora postojati pauza u govoru). Na slici 2 data je ova gramatika u BNF formatu (eng. *Backus-Naur Form*), koji se često koristi za zapisivanje gramatika (McCracken & Reilly, 2003). U ovom formatu zapisa, mogu se specificirati promenljive koje odgovaraju pojedinim delovima dozvoljenih sekvenci, koje se zatim referenciraju u drugim promenljivama, ili glavnom pravilu (eng. *main*), koje predstavlja krajnje uputstvo za formiranje reči. U datom primeru postoji samo glavno pravilo. Uglaste zagrade [] označavaju da niz reči unutar njih može, ali ne mora da postoji u celokupnoj generisanoj sekvenci. Trouglaste zagrade <> označavaju da se reči unutar njih mogu ponoviti jednom ili više puta uzastopno. Karakteri | označavaju operaciju ILI, to jest mogućnost izbora jedne od ponuđenih opcija.

```
main = [ !SIL ] < NULA | JEDAN | DVA | TRI | ČETIRI | PET | ŠEST | SEDAM | OSAM | DEVET > [ !SIL ];
```

Slika 2. Primer regularne gramatike (BNF format)



Slika 3. Regularna gramatika prikazana kao odgovarajući akceptor

Na slici 3 dat je odgovarajući akceptor, odnosno prihvatač (eng. *acceptor*) – automat koji može da prihvati ili odbaci ulaznu sekvencu simbola (to jest reči), u zavisnosti od toga da li se nakon obrade svih simbola nađe u završnom (finalnom) stanju. U datom primeru, početno stanje je 0, a završno 3. Oznake <*eps*> odgovaraju već spomenutim epsilon simbolima, odnosno situaciji kada nije emitovan nijedan simbol (na primer, kada je opciona tišina preskočena).

STATISTIČKI *N*-GRAM MODELI

Dok se gramatike oduvek, pa tako i danas mogu koristiti u specifičnim aplikacijama, u uskim domenima interakcije (na primer, za zadavanje osnovnih komandi pri upravljanju pametnim mobilnim telefonom, kao što su pozivanje kontakata iz imenika ili upravljanje porukama i evidencijom poziva (Popović i drugi, 2015b)), generalno, u modelovanju jezika za iole opštije namene su do sada dominirala dva pristupa. Prvi od njih je statistički model baziran na *n*-gramima, odnosno verovatnoćama pojavljivanja pojedinačnih sekvenci reči, dužine od jedne do najviše *n* reči. Drugi pristup je baziran na neuronskim mrežama, o čemu će više reči biti u sekciji „Modeli bazirani na neuronskim mrežama“.

Jezički modeli bazirani na *n*-gramima su veoma dugo bili osnova za sve najkvalitetnije tehnologije za prepoznavanje govora (Rosenfeld, 2000). Ovakvi modeli aproksimiraju estimacioni problem tako što posmatraju jezik kao Markovljev izvor reda *n*-1, pri čemu se verovatnoća naredne, *i*-te reči u rečenici u zavisnosti od dosadašnje istorije reči $P(w_i|h_i)$ može prikazati izrazom

$$P(w_i|h_i) \approx P(w_i|w_{i-n+1}, \dots, w_{i-1}) , \tag{2.10}$$

gde je w_i *i*-ta reč u rečenici, a $h_i \stackrel{\text{def}}{=} \{w_1, w_2, \dots, w_{i-1}\}$ istorija, odnosno niz svih prethodnih reči. Drugim rečima, smatra se da verovatnoća pojave naredne reči u rečenici zavisi isključivo od prethodnih *n*-1 reči. Najčešći izbor za vrednost parametra *n* je 3, i pri tome se odgovarajući *n*-grami nazivaju trigrami. U slučaju da u korpusu za obuku nema dovoljno reči da bi se trigrami dobro estimirali (što uglavnom znači da ima manje od nekoliko miliona reči), ili je ciljana upotreba u nekom relativno ograničenom domenu, koriste se i bigrami, $n = 2$. Retko se koristi neka vrednost parametra *n* veća od 3. Pokazuje se da u slučaju veće mogućnosti pojave reči kojih

nema u rečniku ASR sistema, povećanje reda *n*-gram modela čak može dovesti do pogoršanja performansi (Mikolov, 2012; Ostrogonac, 2018).

Uzimajući u obzir izraz (2.10), primenom lančanog pravila može se odrediti verovatnoća svakog niza reči *W*, ukupne dužine *K* reči

$$P(W) = P(w_1, w_2, \dots, w_K)$$
$$= P(w_1)P(w_2|w_1) \dots P(w_{n-1}|w_1, \dots, w_{n-2}) \prod_{i=n}^{K} P(w_i|w_{i-n+1}, \dots, w_{i-1}) \,. \quad (2.11)$$

U slučaju trigrama ($n = 3$), izraz 2.11 glasi

$$P(W) = P(w_1)P(w_2|w_1) \prod_{i=3}^{K} P(w_i|\, w_{i-2}, w_{i-1}) \,. \quad (2.12)$$

Tokom obuke, verovatnoće pojedinih *n*-grama se na osnovu datog, po mogućstvu što opširnijeg, tekstualnog korpusa, prema definiciji uslovne verovatnoće, dobijaju preko izraza

$$P(w_i|w_1, \dots, w_{i-1}) = \frac{\#(w_1, \dots, w_{i-1}, w_i)}{\#(w_1, \dots, w_{i-1})}, \quad (2.13)$$

gde se u brojiocu javlja broj pojava sekvenci reči od w_1 do w_i (uključujući i samu reč w_i), dok je u imeniocu broj pojava niza reči od w_1 do w_{i-1} (bez reči w_i). Ako se, dakle, u korpusu reč w_i ne javlja nikad nakon sekvence $\{w_1, \dots, w_{i-1}\}$, tom *n*-gramu se dodeljuje verovatnoća 0. Ovim bi svim dužim nizovima reči, koje sadrže u sebi taj *n*-gram, verovatnoća takođe bila 0. U suprotnom slučaju, ako nakon niza reči od w_1 do w_{i-1} u korpusu uvek sledi reč w_i, *n*-gramu će se dodeliti verovatnoća 1. Ovo otkriva jedan od najvećih problema koji postoje kod modelovanja jezika preko *n*-grama.

U pitanju je problem retkosti podataka za obuku (eng. *data sparsity*), koji proizilazi iz ograničenosti količine materijala za obuku, čak i za veoma velike tekstualne korpuse. Pošto se povećanjem broja *n* broj kombinacija od toliko reči eksponencijalno uvećava, veliki broj viših *n*-grama se ne pojavljuje nijednom u korpusu, a od ostalih velika većina se često pojavljuje samo jedanput ili dvaput. Stoga, retke kombinacije reči biće neminovno loše modelovane. Ovaj problem je već za trigrame dosta veliki. Problem je još izraženiji kod visoko inflektivnih jezika, u

koju grupu spada i srpski jezik, jer je za dobar model dodatno potrebno da se u korpusu za obuku pojavljuju svi morfološki oblici promenljivih reči, u što većem broju konteksta. U *n*-gram modelima jezika se zbog toga ne koristi estimacija maksimalne verodostojnosti (ML obuka) na osnovu brojnosti pojedinih *n*-grama u korpusu. Umesto toga, koristi se takozvana *back-off* procedura rekurzivnog vraćanja na *n*-grame nižeg reda u slučaju neviđenih viših *n*-grama (uz određenu cenu tog koraka), uz primenu tehnika ublažavanja raspodele verovatnoća.

Postoji nekoliko tehnika ublažavanja. One se razlikuju po složenosti, ali i po tome koliko doprinose kvalitetu modela jezika na kom se primenjuju (Chen & Goodman, 1999). U ranijim eksperimentima, pokazano je da je Kneser-Ney ublažavanje optimalno za najveći broj aplikacija (Kneser & Ney, 1995; Pakoci i drugi, 2017). To je jedna od naprednijih tehnika ublažavanja, koja osim eliminacije nultih verovatnoća pojedinih *n*-grama vodi računa i o distinkciji među njima (ne dodeljuje im iste verovatnoće), koristeći *n*-grame nižeg reda kao izvor informacija za tu namenu (radi se interpolacija sa modelom nižeg reda), a oni se još dodatno prilagođavaju određenim situacijama od značaja, kada bi se određenim nižim *n*-gramima neopravdano dodelila relativno visoka verovatnoća (tipičan primer su sekvence *Gornji Milanovac* i *Donji Milanovac*, koje, ako se jave dovoljno često u korpusu, mogu rezultovati prevelikom verovatnoćom unigrama *Milanovac* u nekom novom kontekstu, iako se ta reč javlja gotovo isključivo nakon reči *Gornji* ili *Donji*).

Još jedan problem sa *n*-gram modelima je nemogućnost modelovanja dužih konteksta, jer smo ograničeni aproksimacijom da je od celokupne istorije bitno samo poslednjih *n*-1 reči, a povećanjem vrednosti *n* samo se eksponencijalno pogoršava prethodno opisani problem dimenzionalnosti podataka, koji nikakva dodatna matematika ne može dovoljno dobro ispraviti.

Na kraju modelovanja se obično vrši i odsecanje (eng. *pruning*), to jest izbacivanje vrlo retkih, uglavnom viših *n*-grama iz modela jezika, jer se pretpostavlja da se njihove verovatnoće ipak ne mogu dovoljno precizno estimirati (za njih preostaje *back-off* procedura, kao i za neviđene *n*-grame). Određivanje odgovarajućeg praga odsecanja zavisi od jezika, korpusa za obuku i donekle željene primene. Opciono se ovaj korak može izbaciti (Pakoci i drugi, 2017).

Modeli jezika bazirani na *n*-gramima se obično čuvaju u tekstualnom ARPA-MIT formatu, koji definiše broj instanci svakog reda *n*-grama (unigrama, bigrama, trigrama…), a nakon toga redom svaki *n*-gram, odgovarajuću verovatnoću, i u pojedinim slučajevima (kada postoji) vrednost *back-off* koeficijenta za taj *n*-gram. Primer modela jezika u ovom formatu prikazan je u dodatku 1.

MODELI BAZIRANI NA NEURONSKIM MREŽAMA

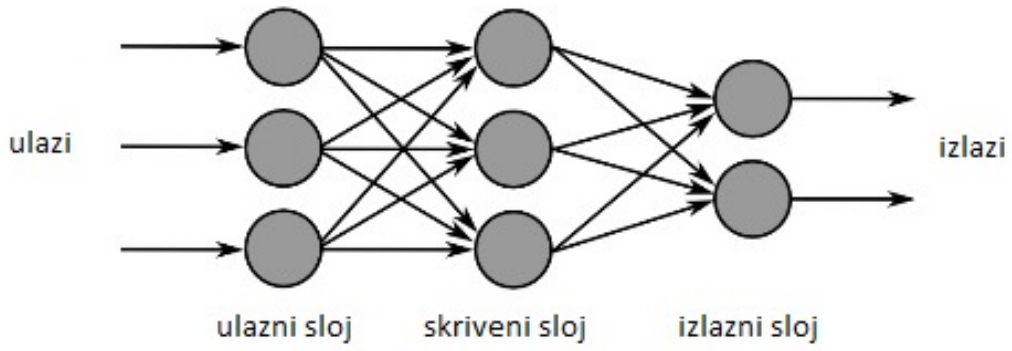
Veštačke neuronske mreže su se pojavile u drugoj polovini dvadesetog veka, ali je njihova primena u NLP čekala još decenijama, iako su se vremenom razvijale i nove strukture, i rešavala različita ograničenja tadašnjih neuronskih mreža. Prvi značajan pokušaj da se jezik modeluje preko ANN pripisuje se Elmanu, koji je rekurentnom mrežom modelovao jezik na osnovu rečenica prethodno generisanih preko gramatičkih pravila (Elman, 1990). Nakon toga, redom je pokazivano da modeli jezika na osnovu neuronskih mreža (NNLM) mogu pružiti informacije koje su donekle komplementarne onima što daje *n*-gram model (Schwenk & Gauvain, 2005), predložene su NNLM sa mrežama sa propagacijom unapred (eng. *Feed-Forward Neural Networks*, FFNN) (Bengio i drugi, 2003), i na kraju savremene rekurentne mreže za modelovanje jezika (RNNLM) (Mikolov i drugi, 2010), kao i modeli bazirani na mrežama dubokog uverenja (eng. *Deep Belief Networks*, DBN) (Arisoy i drugi, 2012).

Uspešna primena ANN u modelovanju jezika vezana je tek za prethodnu deceniju. Ove moderne mreže uspele su zaista da reše neke od glavnih nedostataka *n*-gram modela, mada su značajno računski složenije. Ovakav pristup je u mnogim primenama pokazao superiornost u odnosu na *n*-grame, ali je i pored mnogo truda ostao računski veoma zahtevan i složen, što rezultira pre svega značajno dužim vremenom potrebnim za obuku modela (Oparin i drugi, 2012; Popović i drugi, 2018). Ipak, prednosti odnose prevagu nad ovom glavnom manom, s obzirom na to da je tačnost prepoznavanja uglavnom ono što treba maksimizovati u ASR sistemu.

Neuronske mreže sa propagacijom unapred

Primer neuronske mreže sa propagacijom unapred (Bengio i drugi, 2003) dat je na slici 4. One koriste ograničen kontekst, i zbog toga je LM baziran na njima vrlo sličan *n*-gram modelima jezika. Taj kontekst je implementiran tako što se na ulaz

mreže dovodi čitava sekvenca reči koja čini relevantnu prethodnu istoriju, to jest prethodnih *n*-1 reči. Za predstavljanje svake od reči u rečniku koristi se takozvano 1-od-*N* kodovanje, ako veličinu rečnika označimo sa *N*, što podrazumeva da se svaka reč prikazuje kao vektor dimenzije *N*, u kojem svi elementi osim jednog (koji odgovara datoj reči) imaju vrednost 0, dok taj jedan obično ima vrednost 1.



Slika 4. Primer neuronske mreže sa propagacijom unapred

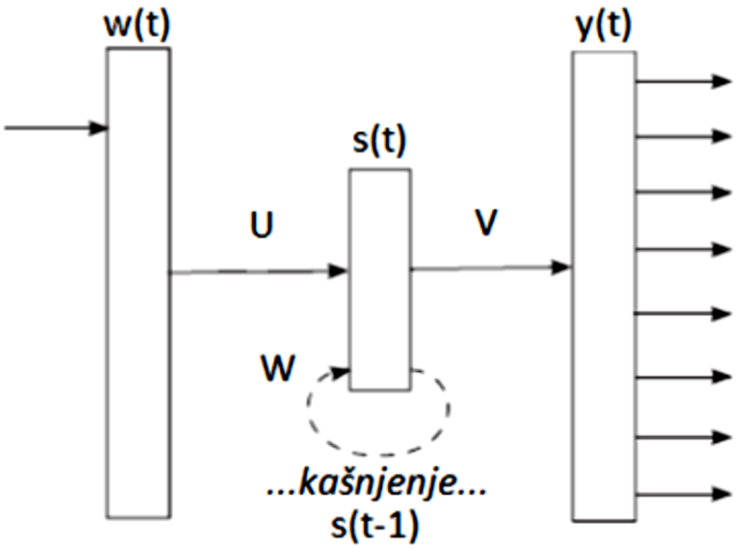
U praksi, za primenu u NLP, broj ulaza (dimenzionalnost ulaznog vektora) je jednak dužini relevantne istorije puta veličina rečnika, što može biti u stotinama hiljada, pa se često prvo vrši linearna projekcija ulaznih podataka na znatno manju dimenzionalnost. Skriveni sloj je tipično dimenzije nekoliko stotina neurona, a izlazni sloj uvek ima onoliko elemenata koliko je i reči u rečniku.

Osim ulaznog sloja, u ovakvoj mreži postoji i skriveni sloj, koji sadrži nelinearnu aktivacionu funkciju, što je u većini slučajeva sigmoidalna ili tangens hiperbolična funkcija. Ovaj sloj uglavnom ima dimenziju od nekoliko stotina neurona. Završni sloj mreže je uvek izlazni sloj, čija dimenzija je jednaka *N*. U obučenoj mreži, izlazni sloj daje raspodelu verovatnoća $P(w|h_j)$, pri čemu važi $w \in \{w_1, \ldots, w_N\}$, a h_j je relevantni kontekst (istorija) od *n*-1 prethodnih reči (koje su bile na ulazu mreže).

Ponekad se između ulaznog sloja i skrivenog sloja dodaje i projekcioni sloj (Mikolov i drugi, 2011a). U tom slučaju se ulazna istorija od *n*-1 1-od-*N* vektora reči linearnom projekcijom preko određene zajedničke projekcione matrice propagira ka tom novom sloju, koji je puno manje dimenzionalnosti od ulaznog. Ovo se radi jer ulazni sloj može da sadrži na stotine hiljada binarnih promenljivih (na primer, sa kontekstom dužine 4 i rečnikom od 25000 reči, ulazni vektor je dimenzije 100000 elemenata, od kojih su samo 4 različita od nule). Projekcioni sloj je u praksi obično dimenzije od nekoliko desetina neurona po svakoj ulaznoj reči.

Rekurentne neuronske mreže

U poslednjoj deceniji, RNN modeli su počeli da se koriste da bi konačno rešili problem konfuzne i složene implementacije, i smanjili računsku kompleksnost prethodnih pristupa (Mikolov i drugi, 2010). Oni rešavaju oba problema *n*-grama: problem retkosti podataka tako što se sve reči iz rečnika projektuju u prostor vektora koji se mogu opisati ograničenim skupom parametara, a istovremeno problem ograničenog konteksta rešavaju svojim rekurentnim vezama, kojima se mogu modelovati duži konteksti, tačnije, sekvence proizvoljne dužine. Sprovedeni su eksperimenti koji potvrđuju da su RNNLM modeli superiorni i u odnosu na *n*-grame, i u odnosu na modele koji koriste standardne mreže sa propagacijom unapred, koji kao i *n*-grami imaju problem limitiranog konteksta (Mikolov, 2012). Međutim, RNNLM modeli su i dalje prilično računski složeni, i zavisno od implementacije mogu da rezultuju prilično dugim vremenima potrebnim za obuku modela.



Slika 5. Primer rekurentne neuronske mreže za obuku LM

Kombinovanje ulazne i skrivene reprezentacije je tipično za RNN. Sa ovakvom strukturom, do reprezentacije istorije dolazi se tokom same obuke, i ona u teoriji može biti neograničeno duga.

Primer strukture jednostavne RNN mreže dat je na slici 5 (Mikolov, 2012). Takva struktura se ponekad naziva i Elmanova mreža. Na ulazu imamo trenutnu reč $w(t)$ u svojoj vektorskoj 1-od-*N* reprezentaciji. Osim tog vektora, kao dodatni ulaz imamo i vektor $s(t-1)$, što je vremenski zakašnjena izlazna vrednost skrivenog

sloja mreže iz prethodnog koraka. Takvo kombinovanje sa skrivenom reprezentacijom je tipično za rekurentne mreže (za razliku od standardnih rekurzivnih neuronskih mreža, gde se kombinovanje umesto toga radi sa reprezentacijom pretka). Skriveni sloj $s(t)$ može biti identičnih karakteristika kao kod mreža sa propagacijom unapred. Obučena rekurentna mreža će na izlaznom sloju $y(t)$ dati raspodelu verovatnoća $P(w(t+1)|w(t), s(t-1))$, ili, jasnije napisano, $P(w_{t+1}|w_t, s_{t-1})$. Matrice *U* i *W* na slici opisuju parametre veze ulaznih podataka i skrivenog sloja (pri čemu matricu *W* čine rekurentni težinski koeficijenti), a matrica *V* vezu skrivenog sloja sa izlaznim slojem. Obuka RNNLM podrazumeva određivanje parametara u okviru ovih matrica.

Obuka RNN tipično se izvodi uz pomoć algoritma propagacije unazad (eng. *BackPropagation*, BP), ili algoritma propagacije unazad kroz vreme (eng. *BackPropagation Through Time*, BPTT). Tokom obuke, izlazi pojedinih slojeva mreže mogu se predstaviti ovako:

$$s_j(t) = f\left(\sum_i w_i(t)\boldsymbol{u}_{ji} + \sum_l s_l(t-1)\boldsymbol{w}_{jl}\right) \tag{2.14}$$

$$y_k(t) = g\left(\sum_j s_j(t)\boldsymbol{v}_{kj}\right). \tag{2.15}$$

U izrazima (2.14) i (2.15), $f(z)$ i $g(z)$ su sigmoidalna, odnosno *softmax* aktivaciona funkcija. *Softmax* funkcija na izlaznom sloju mreže služi da bi se osiguralo da izlazi mreže daju validnu raspodelu verovatnoća, to jest da su svi izlazi pozitivni (ili 0) i da im je zbir jednak 1. Ove funkcije su date izrazima:

$$f(z) = \frac{1}{1+e^{-z}} \tag{2.16}$$

$$g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}}. \tag{2.17}$$

U izrazu (2.17) *softmax* funkcija je data po elementu (*m* je indeks elementa) umesto na nivou celog vektora. Ciljna funkcija, najčešće kriterijum unakrsne entropije (Bourlard & Morgan, 1994), tokom obuke se koristi za dobijanje gradijenta vektora greške na izlaznom sloju mreže, koji se propagira unazad skrivenom sloju. Oba tipa propagacije (BP i BPTT) se tipično oslanjaju na standardni iterativni stohastički

algoritam opadajućeg gradijenta (SGD algoritam) (Bottou, 2010), mada postoje i drugi. Deo podataka za obuku se koristi kao validacioni skup preko kog se kontroliše brzina učenja mreže i određuje najbolja iteracija obuke (što ne mora biti i poslednja).

Algoritam propagacije unazad (BP)

U tipičnoj konfiguraciji, BP algoritam se oslanja na dobijeni gradijent vektora greške na izlaznom sloju mreže, koji se SGD algoritmom propagira unazad ka skrivenom, i zatim ulaznom sloju (Rumelhart i drugi, 1986; Bodén, 2002).

Cela obuka neuronske mreže je podeljena na epohe, a u svakoj se iterativno prolazi kroz celokupan korpus za obuku. Inicijalna brzina učenja mreže se takođe mora definisati, i ona se koristi u narednim iteracijama obuke dokle god se dobija poboljšanje, to jest opadanje entropije na validacionom skupu, i to iznad date granice (i suviše malo poboljšanje inicira naredne korake). Kada poboljšanje više nije dovoljno dobro, ili se dobije pogoršanje, brzina učenja se umanjuje, a najčešće prepolovljava. Obuka traje ili unapred određen broj epoha, ili dok se ne dostigne neka zadata granica u brzini učenja (finalna brzina učenja), ili određeni nivo poboljšanja validacionog rezultata (Mikolov, 2012).

Na početku obuke, za $t = 0$, elementi matrica *U*, *V* i *W* se postavljaju na slučajno određene male vrednosti, na primer na osnovu Gausove raspodele verovatnoća sa srednjom vrednošću 0 i vrlo malom varijansom (na primer 0,1). Takođe, inicijalizuju se stanja svih neurona *j* skrivenog sloja mreže na $s_j(0) = 1$. Za naknadne epohe koriste se obučeni parametri iz prethodne epohe.

Svaka epoha obuke neuronske mreže BP algoritmom obuhvata prolazak kroz sve reči $w(t)$ u rečenicama datog korpusa, i može se opisati u nekoliko koraka (Mikolov, 2012):

- $t = t + 1$;
- na ulazni sloj se dovodi vektor $w(t)$ koji predstavlja odgovarajuću ulaznu reč, i zatim se na njega kopira i stanje skrivenog sloja $s(t-1)$;
- Propagacijom unapred dobijaju se stanja skrivenog sloja $s(t)$ i izlaznog sloja $y(t)$, kako je dato u izrazima (2.14) i (2.15);
- Na izlaznom sloju se izračunava gradijent vektora greške $e_y(t)$;

- $e_y(t)$ se propagira unazad kroz mrežu, uz odgovarajuće menjanje elemenata matrica *U*, *V* i *W*;
- Vraća se na prvi korak dokle god se ne obrade svi ulazni podaci (reči) za obuku.

Gradijent vektora greške se koristi sa ciljem maksimizacije verodostojnosti ispravnih reči, što se može pokazati da je ekvivalentno minimizaciji unakrsne entropije. Naime, ako sa q_i označimo estimirane verovatnoće podataka (reči *i* u određenom kontekstu) iz datog skupa za obuku, a sa p_i relativnu učestanost tih podataka (reči *i* u datom kontekstu) u skupu za obuku (broj pojava u odnosu na ukupan broj reči), i ako u skupu za obuku ukupno imamo *M* uzoraka (reči), funkcija verodostojnosti za dati skup za obuku data je sa

$$\Lambda = \prod_i {q_i}^{Mp_i}\,, \tag{2.18}$$

gde se množenje vrši za svaki uzorak (reč) *i*. Na osnovu toga, prosečna log-verodostojnost iznosi

$$\log \Lambda_{avg} = \frac{1}{M}\log \Lambda = \sum_i p_i \log q_i = -H(p,q). \tag{2.19}$$

Iz izraza (2.19) zaključuje se da je maksimizacija verodostojnosti zaista ekvivalentna minimizaciji unakrsne entropije $H(p,q)$, tako da obuka može maksimizirati log-verodostojnost na skupu za obuku, što je dato sa

$$ML = \sum_{i=1}^{M} \log\, y_{c_i}\,, \tag{2.20}$$

gde je sa y_{c_i} dat element izlaznog vektora mreže $y(t)$ koji odgovara indeksu reči iz rečnika koju je trebalo predvideti u koraku $t = i$. U ovakvoj postavci, gradijent vektora greške, koji maksimizuje izraz (2.20), dat je sa

$$e_y(t) = w(t+1) - y(t)\,, \tag{2.21}$$

gde je $w(t+1)$ 1-od-*N* vektor koji predstavlja reč u narednom koraku, odnosno reč koja je trebalo da bude predviđena. Dalje, koristeći vektor $e_y(t)$, koeficijenti matrice

V, koja opisuje vezu skrivenog i izlaznog sloja, obeleženi sa $\boldsymbol{v}_{kj}$, koriguju se pomoću izraza

$$\boldsymbol{v}_{kj}(t+1) = \boldsymbol{v}_{kj}(t) + s_j(t) \cdot e_{y_k}(t) \cdot \alpha, \tag{2.22}$$

gde je sa α označena trenutna brzina učenja mreže, sa $s_j(t)$ odgovarajući neuroni skrivenog sloja mreže, a sa $e_{y_k}(t)$ gradijent greške *k*-tog neurona izlaznog sloja mreže. Startna vrednost brzine učenja može biti na primer 0,1, i ona se zatim umanjuje, i to najčešće prepolovljava, nakon svake epohe obuke u kojoj ostvareno poboljšanje na validacionom skupu nije dovoljno dobro (Mikolov, 2012). Često se u obukama neuronskih mreža koriste metodi regularizacije u cilju prevencije preobučavanja na date podatke. Od njih je najpoznatija L_2 (ili Tihonova) regularizacija. Ako se ona koristi, izraz (2.22) se modifikuje u

$$\boldsymbol{v}_{kj}(t+1) = \boldsymbol{v}_{kj}(t) + s_j(t) \cdot e_{y_k}(t) \cdot \alpha - \boldsymbol{v}_{kj}(t) \cdot \beta, \tag{2.23}$$

pri čemu je β regularizacioni parametar, čija je vrednost recimo 10^{-6}, i služi da održava vrednosti težina matrice *V* relativno malim (bliskim nuli), jer bi kompaktniji skup težina u teoriji trebalo da ima bolju moć generalizacije (Rissanen, 1978). Zatim sledi propagacija gradijenta greške na skriveni sloj mreže (Rumelhart i drugi, 1986), pri čemu je svaki element novog vektora greške $e_s(t)$ dat sa

$$e_{s_j}(t) = \left(\sum_k e_{y_k}(t) \cdot \boldsymbol{v}_{kj}(t) \right) \cdot s_j(t) \cdot \Big(1 - s_j(t)\Big). \tag{2.24}$$

Pomoću vektora greške $e_s(t)$ koriguju se težine matrice *U* (označene sa $\boldsymbol{u}_{ji}$), koja opisuje vezu ulaznog sloja i skrivenog sloja mreže, što je dato sa

$$\boldsymbol{u}_{ji}(t+1) = \boldsymbol{u}_{ji}(t) + w_i(t) \cdot e_{s_j}(t) \cdot \alpha - \boldsymbol{u}_{ji}(t) \cdot \beta. \tag{2.25}$$

U izrazu (2.25) sa $w_i(t)$ je dat odgovarajući element vektora ulazne reči u koraku *t*. Pošto je u svakom koraku *t* u vektoru ulazne reči $w(t)$ samo jedan element različit od nule (zbog 1-od-*N* reprezentacije reči), ažuriranje težina u matrici *U* se može ograničiti samo na one koje zavise od tog elementa (aktivnog neurona ulaznog sloja), i na taj način se ubrzati (ostale težine se ne diraju).

Na kraju date iteracije procedure, vrši se korekcija elemenata rekurentne matrice *W*, koji su obeleženi sa $\boldsymbol{w}_{jl}$, što je dato sa

$$\boldsymbol{w}_{jl}(t+1) = \boldsymbol{w}_{jl}(t) + s_l(t-1) \cdot e_{s_j}(t) \cdot \alpha - \boldsymbol{w}_{jl}(t) \cdot \beta. \quad (2.26)$$

Procedura se ponavlja dokle god postoji još ulaznih podataka (reči) u skupu za obuku modela jezika.

Algoritam propagacije unazad kroz vreme (BPTT)

BPTT algoritam je nadogradnja osnovnog BP algoritma (Rumelhart i drugi, 1986). Motivacija za njegovo korišćenje je činjenica da BP algoritam, iako koriguje parametre mreže u cilju optimizacije predikcije reči koristeći prethodnu reč i prethodno stanje skrivenog sloja, nije pravljen tako da pamti bilo kakve korisne informacije u skrivenom sloju, koje bi potencijalno mogle dobro doći za buduće korake. Ako se ispostavi da je mreža zapamtila neke informacije o dugim kontekstima u skrivenom sloju, to je gotovo uvek više slučajno nego namerno (Mikolov, 2012). BPTT algoritam ima za cilj da tako nešto postigne namerno. RNN jezički modeli opisani u poglavljima 4, 5 i 6 ove disertacije se uglavnom baziraju na njemu.

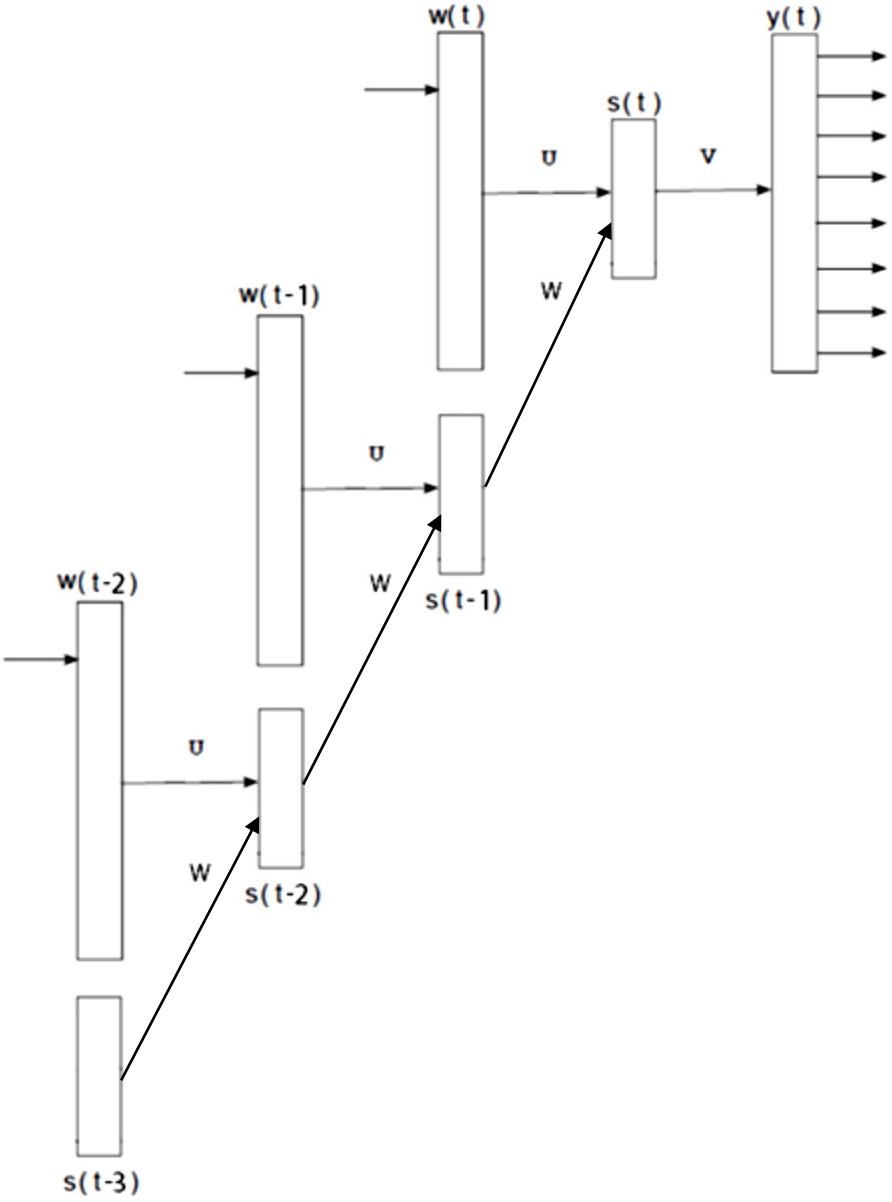
Algoritam polazi od ideje da ako se rekurentna neuronska mreža posmatra u T uzastopnih vremenskih koraka, to je ekvivalentno posmatranju duboke neuronske mreže sa propagacijom unapred koja ima T skrivenih slojeva, koji su svi iste dimenzionalnosti, a „odmotane“ matrice rekurentnih težina W su identične (ovakav proces transformacije RNN u FFNN se često naziva „odmotavanje“ (eng. *unfolding*)). Primer transformirane RNN strukture u FFNN za $T = 3$ trenutka dat je na slici 6 (Mikolov, 2012).

Pri obuci takve mreže uobičajenim SGD metodom, vektor gradijenta greške se propagira sa skrivenog sloja koji odgovara datom koraku $s(t)$ nazad ka skrivenom sloju koji odgovara prethodnom koraku $s(t-1)$, i na osnovu toga se koriguje matrica W. Ovakav metod zahteva čuvanje vrednosti stanja skrivenih slojeva iz svakog od prethodnih koraka. Gradijent greške $e_s(t)$ se propagira rekurzivno (za svaki vremenski korak odmotavanja τ), na način predstavljen izrazom

$$e_{s_j}(t-\tau-1) = \left(\sum_l e_{s_l}(t-\tau) \cdot \boldsymbol{w}_{lj}(t-\tau) \right) \cdot s_j(t-\tau-1) \cdot (1 - s_j(t-\tau-1)). \quad (2.27)$$

Odmotavanje mreže u vremenu se, teoretski, može raditi u onoliko koraka koliko je do datog trenutka obrađeno reči iz skupa za obuku. Međutim, pokazuje se da gradijenti greške brzo nestaju tokom propagacije unazad kroz vreme, a u retkim

slučajima dolazi i do eksplozije greške (Bengio i drugi, 1994). Zbog toga se odmotavanje mreže u praksi radi u svega nekoliko koraka, a za RNN modele jezika obično je dovoljno oko 5 koraka (Mikolov, 2012). Takva procedura se ponekad naziva skraćena propagacija unazad kroz vreme (eng. *truncated BPTT*). Interesantno je napomenuti da bez obzira na ograničenje broja koraka odmotavanja na *T*, to i dalje dozvoljava mreži da nauči, to jest zapamti i kontekstualne obrasce duže od *T* koraka (kao što i standardna RNN, koja je ekvivalentna FFNN sa jednim korakom odmotavanja, ponekad nauči i duže kontekste).



Slika 6. Primer RNN „odmotane“ unazad u vremenu, odnosno ekvivalentna FFNN

U konkretnom primeru, odmotavanje je rađeno u 3 koraka. U praksi, ono se u RNNLM obukama tipično radi u tek nešto više (4-5) koraka.

Težinski koeficijenti matrice U ($\boldsymbol{u}_{ji}$) se putem BPTT ažuriraju sa

$$\boldsymbol{u}_{ji}(t+1) = \boldsymbol{u}_{ji}(t) + \sum_{\tau=0}^{T} w_i(t-\tau) \cdot e_{s_j}(t-\tau) \cdot \alpha - \boldsymbol{u}_{ji}(t) \cdot \beta. \quad (2.28)$$

U prethodnom izrazu, T je željeni broj koraka za odmotavanje mreže u vremenu. Takođe, izmena matrice U se mora raditi odjednom, a ne inkrementalno (korak po korak) tokom odmotavanja mreže, jer to inače može dovesti do nestabilnosti obuke (Bodén, 2002). Na kraju, težine rekurentne matrice W ($\boldsymbol{w}_{jl}$) se ažuriraju sa

$$\boldsymbol{w}_{jl}(t+1) = \boldsymbol{w}_{jl}(t) + \sum_{\tau=0}^{T} s_l(t-\tau-1) \cdot e_{s_j}(t-\tau) \cdot \alpha - \boldsymbol{w}_{jl}(t) \cdot \beta. \quad (2.29)$$

DRUGA ISTRAŽIVANJA U OBLASTI MODELOVANJA JEZIKA

POREĐENJE *N*-GRAM PRISTUPA SA PRISTUPIMA NA BAZI NEURONSKIH MREŽA

Prilikom poređenja *n*-gram i NNLM pristupa treba obratiti pažnju na pojedine detalje da bi poređenje bilo fer. Najveći problem pri tome mogu biti veličine modela. Kod *n*-gram modela, na njihovu veličinu utiče svakako red modela, ali i primenjeni parametar odsecanja za retko viđene *n*-grame. Sa druge strane, kod neuronskih mreža, na veličinu modela pre svega utiču parametri arhitekture mreže, počevši od toga šta se konkretno koristi (koja vrste mreže), a onda i broj skrivenih slojeva i broj neurona po određenim slojevima. Pošto rezultujući *n*-gram model zavisi od primenjenih tehnika ublažavanja i odsecanja, poređenje sa datom neuronskom mrežom ne može biti potpuno korektno. Ipak, pojedina istraživanja su došla do određenih zaključaka o tome u kakvim situacijama se koji tip modela bolje snalazi.

Jedno od istraživanja (Oparin i drugi, 2012) je do rezultata poređenja dva tipa modela došlo primenom pretpostavke da je bolji onaj model koji validnim nizovima reči dodeli veće verovatnoće. Očekivano je bilo da se neuronska mreža prvenstveno bolje snalazi u slučaju kada *n*-gram model jezika mora da iskoristi *back-off* proceduru za dobijanje verovatnoće reči. Na datom test skupu može se posmatrati i svaki *back-off* nivo zasebno (nivo trigrama, nivo bigrama, i tako dalje), odnosno svaki od nivoa

se može posebno porediti sa ponašanjem neuronske mreže u istoj situaciji. Da bi se došlo do rezultata poređenja, može se, na primer, izračunati procenat situacija u kojima neuronska mreža daje bolji rezultat od *n*-grama, ili se odrediti perpleksivnost na celom test skupu, ili se mogu odrediti prosečne razlike u verovatnoćama koje poređeni modeli daju istim rečima u odgovarajućim kontekstima.

Na dve specifične situacije je posebno obraćena pažnja. Prva je slučaj *n*-grama koji se pojavljuju samo jedanput u celom korpusu za obuku, i kojima se zbog primenjene tehnike ublažavanja često dodeljuje vrlo mala verovatnoća (skoro nula). Drugu situaciju čine *n*-grami kod kojih je poslednja reč jedina koja se u korpusu pojavljuje nakon (*n*-1)-grama koji se sastoji od preostalih, *n*-1 prethodnih reči (broj pojava datog *n*-grama je identičan broju pojava njegove istorije, to jest konteksta).

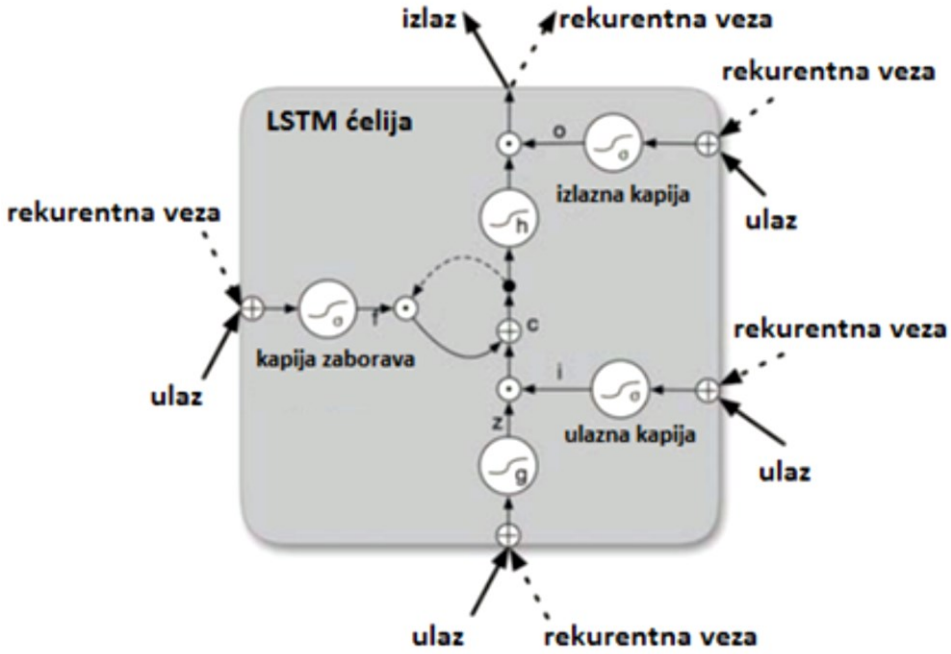
Autori istraživanja (Oparin i drugi, 2012) su došli do nekolicine glavnih zaključaka. Prvo, modeli bazirani na RNN i FFNN davali su bitno veće verovatnoće od *n*-gram modela u približno polovini slučajeva *n*-grama koji postoje u korpusu za obuku, s tim što procenat raste sa porastom *back-off* nivoa *n*-gram modela. Zatim, za slučaj *n*-grama koji postoje samo na jednom mestu u korpusu, nije bilo bitnih razlika u ponašanju dve vrste modela jezika. Statistički *n*-grami su čak davali bolje rezultate u situaciji kada je broj pojava datog *n*-grama i njegove istorije u korpusu jednak. Konačno, u poređenju RNN i FFNN, rekurentni modeli su bili značajno bolji od FFNN u poredivim konfiguracijama (ako se posmatraju slične arhitekture).

JEZIČKI MODELI BAZIRANI NA LSTM NEURONSKIM MREŽAMA

Već je spomenuto da tokom BPTT obuke neuronske mreže lako može da dođe do takozvanog problema nestajućeg gradijenta greške. Postojali su razni pokušaji rešavanja tog problema, međutim, oni su gotovo uvek dovodili do značajnog povećanja složenosti procesa obuke (na primer: Martens & Sutskever, 2011).

Jedan od načina da se problem prevaziđe jeste primena jedinica duge kratkoročne memorije (eng. *Long Short-Term Memory*, LSTM) u okviru neuronske mreže. Mogućnost primene LSTM u modelovanju jezika pominje se u pojedinim radovima tek u vrlo bliskoj prošlosti (Sundermeyer i drugi, 2012). Problem u proceduri propagacije gradijenta greške je zapravo to što se njegove vrednosti skaliraju faktorom koji je praktično uvek neka vrednost različita od 1 (što može

dovesti do nestajanja, a ponekad i do eksplozije vrednosti gradijenta, već u nekoliko koraka). Da bi se to izbeglo, napravljena je nova neuronska jedinica, takva da je faktor skaliranja fiksan i uvek jednak 1. Ovakva jedinica je očigledno ograničena po mogućnosti učenja, tako da se u daljem razvoju LSTM blokova pojavljuju takozvane kapije (eng. *gates*, ili *gating* jedinice). Konačna struktura je data na slici 7.



Slika 7. Šema strukture LSTM jedinice u rekurentnoj neuronskoj mreži

U odnosu na standardan neuron, gde je izlazna vrednost neurona sa ulaznom vrednošću povezana direktno preko date aktivacione funkcije, na primer tangens hiperbolične funkcije ($izlaz = \tanh ulaz$), na prethodnoj slici se vidi da se kod LSTM neurona, nakon primene aktivacione funkcije na ulaznu vrednost, rezultat množi prvim skalirajućim faktorom (i). Nakon toga, preko rekurentne veze unutar jedinice, na skaliranu vrednost se dodaje odgovarajuća vrednost iz prethodnog koraka, koja je prethodno pomnožena drugim faktorom skaliranja (f). Zatim, na dobijeni rezultat se (uglavnom) primenjuje još jedna tangens hiperbolična funkcija, a taj rezultat se dodatno skalira poslednjim, trećim faktorom (o), čime se dobija izlazna vrednost neurona. Sva tri pomenuta faktora skaliranja su pozitivne vrednosti manje od 1, a kontrolišu ih tri kapije – ulazna (eng. *input gate*), izlazna (eng. *output gate*) i kapija zaborava (eng. *forget gate*), pomoću svojih, najčešće sigmoidalnih funkcija koje primenjuju na svoje ulaze. Od tri kapije je posebno bitna kapija zaborava, jer ona

direktno kontroliše u kojoj meri se trenutni podaci u neuronu pamte (Hochreiter & Schmidhuber, 1997).

LSTM jedinice se mogu posmatrati i kao diferencijabilne verzije računarske memorije, pa se često nazivaju i LSTM memorijske ćelije. Ovakva arhitektura može rešiti problem nestajućeg gradijenta. U dubokim rekurentnim neuronskim mrežama, LSTM jedinice se mogu postaviti i samo u rekurentni sloj (dok ostali slojevi mogu koristiti standardne aktivacione funkcije). Pokazuje se da uz LSTM u rekurentnom sloju, dodavanje novih skrivenih regularnih slojeva mreže često ne doprinosi mnogo rezultatu (konkretno, perpleksivnosti) (Sundermeyer i drugi, 2012).

FAKTORISANI JEZIČKI MODELI

U faktorisanim jezičkim modelima (eng. *Factored Language Models*, FLM) reči su predstavljene vektorima obeležja, preko kojih je, osim samog identiteta reči (što je slučaj u 1-od-*N* reprezentaciji reči u uobičajenim jezičkim modelima), moguće ubaciti dodatne informacije, kao što su, na primer, morfološke ili semantičke informacije za datu reč, i sve to eksplicitno integrisati u jedinstven jezički model (Kirchhoff & Yang, 2005; Wu i drugi, 2012).

Ako pretpostavimo da se za svaku reč w_i može odrediti K obeležja, $w_i \equiv f_i^{1:K}$, verovatnoća date rečenice W u trigram modelu jezika se može definisati sa

$$\begin{aligned} P(W) &= P(f_1^{1:K}, f_2^{1:K}, \cdots, f_T^{1:K}) \\ &\approx P(f_1^{1:K}) P(f_2^{1:K} \mid f_1^{1:K}) \prod_{t=3}^{T} P(f_t^{1:K} \mid f_{t-2}^{1:K}, f_{t-1}^{1:K}) , \end{aligned} \tag{2.30}$$

gde su $P(f_i^{1:K})$ estimirane verovatnoće pojave reči, odnosno odgovarajućih K obeležja na datom mestu u rečenici.

Svaka reč zavisi ne samo od niza prethodnih reči, već i od dodatnih, paralelnih nizova obeležja, to jest faktora. Drugim rečima, verovatnoće reči dodatno zavise eksplicitno od skupa od K faktora prethodne reči i implicitno od faktora vezanih za određen broj reči pre nje (u teoriji to mogu biti sve prethodne reči). Ovakva reprezentacija omogućava robustniju estimaciju verovatnoća reči, pogotovo u slučaju kada se određeni *n*-gram ne pojavljuje u korpusu za obuku, ali odgovarajuća kombinacija obeležja (na primer, niz odgovarajućih korena reči ili morfoloških

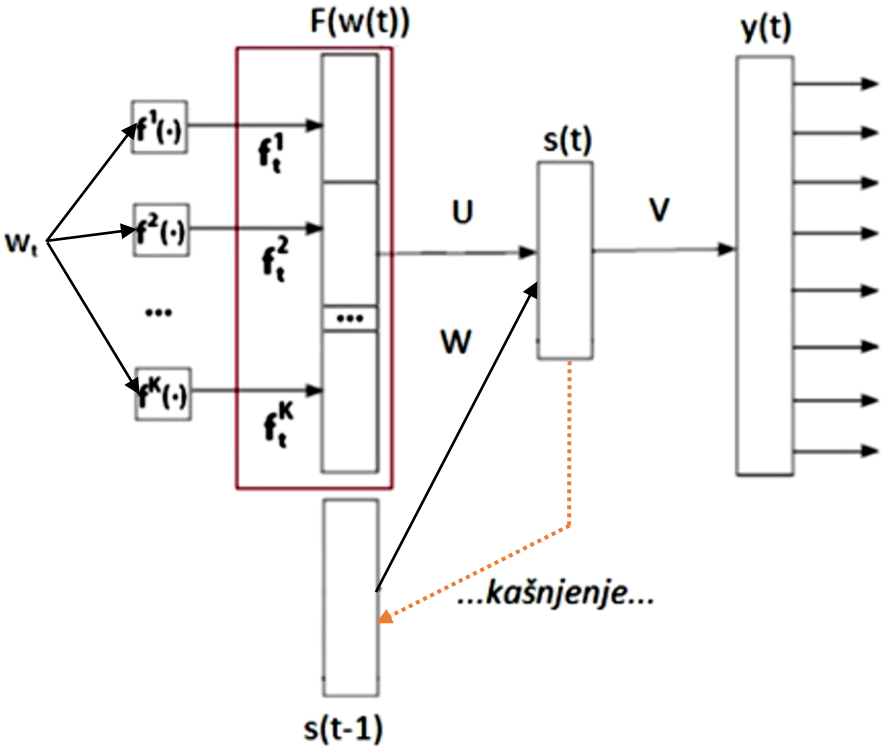
kategorija) postoji. Na taj način, FLM je dizajniran tako da efikasnije iskoristi materijal za obuku, koji neminovno ima problem retkosti podataka. Međutim, čak i kada postoji dovoljna količina materijala za obuku, FLM obuka može dovesti do toga da model bude pristrasan prema pojedinim kombinacijama obeležja, na primer prema čestim kombinacijama morfoloških kategorija.

U okviru faktorisanih *n*-gram modela, obeležja reči se integrišu koristeći generalizovanu varijantu paralelne *back-off* tehnike. Za razliku od standardne tehnike, može postojati više kombinacija odgovarajućih obeležja na koje se može „vratiti“ preko *back-off* metoda (dok se kod običnih *n*-grama *back-off* uvek radi sa trigrama na bigrame i sa bigrama na unigrame, bez alternativnih mogućnosti). Određivanje optimalne kombinacije obeležja za *back-off* se može odraditi unapred na osnovu ekspertskog, lingvističkog znanja, ili tokom obuke na osnovu određenih statistika u korpusu. Dodatno, moguće je odabrati i više *back-off* putanja i iskombinovati njihove estimacije verovatnoća (preko odabrane funkcije, koja može poprimiti razne oblike, što može biti, na primer, srednja vrednost, ponderisana srednja vrednost, proizvod, minimum ili maksimum funkcije raspodele verovatnoća na podskupovima pojedinih obeležja, i tako dalje). Zato su pri obuci FLM vrlo bitna pitanja izbora dodatnih obeležja reči, načina odabira optimalne *back-off* putanje, i određivanja parametara ublažavanja verovatnoća. Optimalan izbor obeležja je teško odrediti, ali se mogu izvršiti određene vođene procedure pretrage, recimo na bazi genetskih algoritama, koje bi optimizovale FLM prema željenom kriterijumu, što je tipično perpleksivnost modela na datom validacionom ili test skupu (Duh & Kirchhoff, 2004).

Ako su u pitanju rekurentne mreže, u FLM pristupu kodovanje obeležja je slično kodovanju reči u standardnom RNNLM – 1-od-*N*, ali sa dopunskim vektorima za svaki odabrani faktor, gde svaki od faktora ima zasebnu, različitu vrednost *N* (ukupan broj različitih mogućih vrednosti obeležja). Ulazni sloj mreže se u ovom slučaju formira od vektora za trenutnu reč, koji se sastoji od izdvojenih vektora obeležja reči za sve odabrane faktore, i od kopije stanja skrivenog sloja iz prethodnog koraka (kao i u standardnoj konfiguraciji). Ostatak mreže (skriveni sloj, izlazni sloj, matrice težinskih koeficijenata) je identičan standardnom RNN modelu.

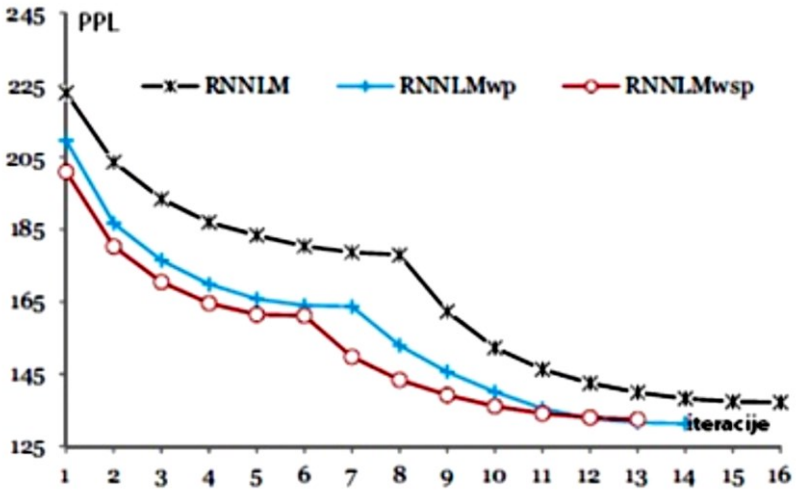
Primer faktorisanog RNN modela dat je na slici 8. U poslednje vreme, FLM su počeli da se primenjuju i na morfološki bogatijim jezicima (češki, arapski, ruski), gde

se može očekivati dobijanje i boljih rezultata nego recimo na engleskom jeziku, zbog potencijalno većeg značaja određenih morfoloških kategorija.



Slika 8. Prikaz faktorisane rekurentne neuronske mreže

Ulazni sloj koristi faktore reči u vektorskoj reprezentaciji f_t^k, $k \in \{1, \cdots, K\}$, koji se izdvajaju u odgovarajućim blokovima, uz zakašnjenu vrednost stanja skrivenog sloja. Ostatak arhitekture je identičan standardnoj RNN.



Slika 9. Konvergencija perpleksivnosti RNNLM u odnosu na dve fRNNLM varijante

U datom primeru (Wu i drugi, 2012), u pitanju je engleski jezik sa rečnikom od oko 150000 reči. Faktorisani model sa oznakom RNNLMwp je uključio i vrste reči kao dodatna obeležja, a RNNLMwsp je uključio i vrste i korene reči.

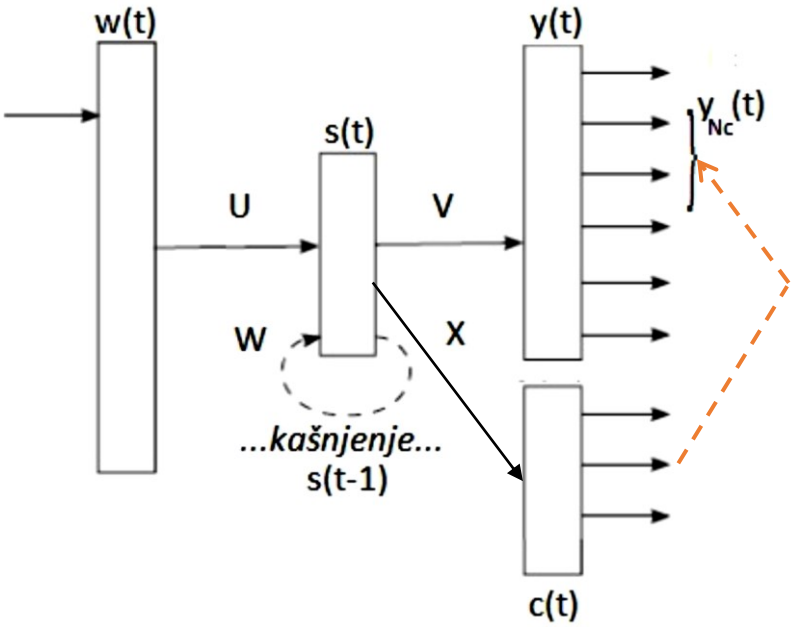
FLM varijanta RNNLM (fRNNLM) ima više slobodnih parametara nego klasični RNNLM, međutim, pokazuje se da u realnim primenama sa vrlo ograničenim brojem koraka propagacije gradijenta unazad (4-5) razlika u vremenskoj kompleksnosti RNNLM i fRNNLM postaje zanemarljiva. Štaviše, fRNNLM može čak i brže da iskonvergira, kao u primeru na slici 9 (Wu i drugi, 2012).

MODELI SA FAKTORISANIM IZLAZNIM SLOJEM

Ova vrsta modela na osnovu neuronskih mreža nema veze sa FLM, iako su sličnog naziva. Ideja kod modela jezika sa faktorisanim izlaznim slojem je da se bitno smanji složenost obuke, tako što se neće računati verovatnoće svih *N* reči iz rečnika na izlaznom sloju mreže, nego će se verovatnoće estimirati prvo za grupe, to jest klase reči, a tek onda za konkretne reči iz odgovarajuće grupe (Goodman, 2001).

Na slici 10 prikazan je primer RNN sa faktorisanim izlaznim slojem (Mikolov, 2012). Prvo se izračunava raspodela verovatnoća na klasama, kojih ima ukupno *C*. Zatim, računa se raspodela verovatnoća samo na rečima koje pripadaju određenoj klasi, to jest za N_c elemenata izlaznog sloja. Dakle, ukupno se izračunava samo $C + N_c$ izlaza, a *softmax* izlazna funkcija, definisana u (2.17), primenjuje se zasebno na *C* elemenata, odnosno na N_c elemenata. Vrednost *C* je konstantna, i uvek relativno mala, dok vrednost N_c može varirati, i zavisi od načina podele reči u klase.

Jedan od načina određivanja zasnovan je na učestanosti reči (to jest, unigram verovatnoći reči) u korpusu za obuku (Mikolov i drugi, 2011b). Reči koje su vrlo česte u korpusu nalaze se u sopstvenim klasama, koje ne sadrže mnogo reči ukupno. Sa druge strane, retke reči se nalaze u zasebnim klasama, i ima ih mnogo više po klasi u odnosu na češće reči. To dovodi do toga da je i N_c vrednost uglavnom mala (u najvećem broju slučajeva). Alternativno, određivanje klasa se može raditi istim algoritmom, ali sa kvadratnim korenima učestanosti reči umesto samim učestanostima. Pokazano je da se na taj način postiže ubrzanje obuke koje je još veće nego u osnovnoj varijanti, a u oba slučaja se može postići ubrzanje za red veličine u odnosu na referentne RNN modele, pa čak i više (Mikolov, 2012).



Slika 10. Prikaz rekurentne neuronske mreže sa faktorizacijom izlaznog sloja

Nakon određivanja raspodele verovatnoća na izlaznom sloju za klase c(t), računaju se verovatnoće samo na podskupu izlaznog sloja za reči y(t) koji sadrži onoliko elemenata koliko reči ima u datoj klasi (N_c), što je na slici označeno strelicom.

Ovakav pristup praktično razbija standardni izlazni sloj neuronske mreže na dva dela – na klasni sloj $c(t)$ i na sloj reči (novi izlazni sloj $y(t)$), pri čemu se uvek pristupa samo podskupu sloja reči. Umesto izraza (2.15), sada je veza skrivenog sloja mreže i novog klasnog izlaznog sloja data preko

$$c_m(t) = g\left(\sum_j s_j(t)\boldsymbol{x}_{mj}\right), \tag{2.31}$$

gde su $\boldsymbol{x}_{mj}$ težinski koeficijenti matrice X koja opisuje vezu skrivenog sloja i klasnog sloja. Veza skrivenog sloja sa podskupom izlaznog sloja $y_{N_c}(t)$ je data sa

$$y_{N_c k}(t) = g\left(\sum_j s_j(t)\boldsymbol{v}_{N_c kj}\right), \tag{2.32}$$

gde su $\boldsymbol{v}_{N_{c_{kj}}}$ elementi matrice težina V_{N_c}, koja predstavlja odgovarajući deo matrice težina V. Verovatnoća naredne reči $w(t+1)$ se izračunava preko

$$P\left(w_{t+1}|\, s(t)\right) = P\left(c_i|\, s(t)\right) \cdot P\left(w_i|\, c_i, s(t)\right). \tag{2.33}$$

U prethodnom izrazu, w_i je predviđena reč (u okviru svoje klase), a c_i klasa kojoj pripada. Tokom obuke, gradijenti greške se izračunavaju i za klasu i za reč, i zatim se propagiraju unazad skrivenom sloju, gde se potom sabiraju. Na taj način, skriveni sloj se obučava da predviđa i raspodelu verovatnoća nad rečima, i raspodelu verovatnoća nad klasama.

KOMBINACIJA VIŠE JEZIČKIH MODELA

Rečeno je već da se težinski koeficijenti neuronske mreže tipično inicijalizuju malim, slučajnim vrednostima. To za posledicu ima da svaka obuka može dovesti do nešto drugačijeg finalnog modela, u zavisnosti od odabranih početnih težina, čak i ako je skup podataka za obuku identičan. Zbog toga, jedan način da se dobiju potencijalno bolje performanse datog ASR sistema je uprosečavanje izlaza nekoliko modela jezika.

Linearna interpolacija je uobičajeni metod za kombinovanje izlaza datih modela. U tom slučaju, verovatnoća date reči *w* u kontekstu *h*, ako na raspolaganju imamo ukupno *N* modela jezika, je

$$p(w|h) = \sum_{i=1}^{N} \frac{p_i(w|h)}{N}. \tag{2.34}$$

U prethodnom izrazu, $p_i(w|h)$ je estimirana verovatnoća pojave date reči *w* od strane *i*-tog modela jezika. Svaki od modela može biti identične arhitekture, samo inicijalizovane drugačijim vrednostima težina, ili to mogu biti potpuno različite arhitekture ili tipovi modela.

Interpolaciju bi idealno bilo raditi na skupu svih mogućih arhitektura, a dodatno bi trebalo ponderisati verovatnoće $p_i(w|h)$ dobijene od pojedinih modela različitim koeficijentima umesto jednakim, koji zavise od njihove takozvane dužine opisa (odnosno, minimalne potrebne količine informacija da bi se opisao dati model – eng. *Minimum Description Length*, MDL) (Rissanen, 1978). Izračunavanje dužine opisa modela je za bilo koji netrivijalan slučaj težak problem – iako se može pretpostaviti da je ta vrednost čvrsto povezana sa brojem parametara modela, pokazuje se da je to pogrešno, pošto je, između ostalog, jasno da različiti parametri

zahtevaju različit broj bita da bi se čuvali, kao i da su mnogi parametri često redundantni (Mikolov, 2012). Najbolja alternativa je estimacija pondera modela na nekom validacionom skupu. U praksi se pokazuje da je najbolje obučiti što veće modele, kao i da modele sa identičnom arhitekturom treba interpolirati koristeći jednake pondere (Mikolov, 2012). Međutim, makar za slučaj dubokih neuronskih mreža, može se očekivati da velika većina rezultujućih modela iskonvergira ka istom, ili makar približno istom, lokalnom maksimumu. Ako se to prihvati, individualna rešenja bi trebalo da su vrlo verovatno prilično slična jedna drugom. Sa druge strane, bilo bi zanimljivo isprobati tehnike koje mogu dati mnogo raznolikije modele (na primer, neke evolutivne tehnike) u budućim istraživanjima.

Moguće je koristiti i druge metode kombinovanja izlaza modela osim linearne kombinacije, na primer, log-linearnu kombinaciju, ili nelinearno kombinovanje preko još jedne, dodatne neuronske mreže, recimo nekog dodatnog RNN modela (Mikolov, 2012). Eksperimenti na tu temu su van opsega ove disertacije. U praksi se uglavnom koriste linearne kombinacije, opciono ponderisane, a često se upravo RNNLM estimacija verovatnoće reči kombinuje sa *n*-gram estimacijom. Određivanje optimalnih pondera u tom slučaju se najčešće radi na osnovu rezultata na datom validacionom ili test skupu, i bazira se na minimizaciji perpleksivnosti ili stope greške rezultujuće kombinacije.

MODELI JEZIKA BAZIRANI NA MORFOLOGIJI

Nekoliko pristupa ubacivanju morfoloških informacija u model jezika je izučavano u istraživanjima (Kirchhoff i drugi, 2006; Matthews i drugi, 2018; Pakoci i drugi, 2019). Većina autora koristi neku vrstu parsera, odnosno modula za dekompoziciju reči, koji utvrdi značajne morfološke jedinice – morfeme, koren reči, afikse, i tako dalje – za određivanje odgovarajućih leksičkih obeležja i klasa reči, i zatim se te informacije koriste kao dodatna ograničenja (preko odgovarajućih težina) za dekoder, u kombinaciji, ili umesto samih reči, koje se u tu svrhu koriste u standardnom pristupu. Pokazalo se da mnogi morfološki bogati jezici dele slične probleme u oblasti modelovanja jezika (Kwon & Park, 2003; Sarikaya i drugi, 2008; Sak i drugi, 2010; Müller i drugi, 2012).

Jedan od standardnih pristupa su klasni jezički modeli (Jardino, 1996; Whittaker & Woodland, 2001) bazirani na morfologiji. Dobijaju se morfološkom anotacijom tekstualnog korpusa za obuku, i nakon toga klasifikacijom reči u grupe uz pomoć informacija dobijenih iz tagovanog korpusa, što je praćeno obukom na transformisanom korpusu, gde umesto samih reči figurišu odgovarajuće klase. Sama obuka može biti praktično identična kao u regularnom pristupu. Ono što je još različito u odnosu na standardni pristup jeste činjenica da se moraju odrediti i verovatnoće pojedinih reči u okviru samih klasa. Označimo li niz prethodnih reči sa $w_1, w_2, \dots, w_{n-1}$, a njihove odgovarajuće klase sa $c_1, c_2, \dots, c_{n-1}$, verovatnoća naredne reči w_n koja odgovara klasi c_n data je izrazom

$$P(w_n | w_1, w_2, \cdots, w_{n-1}) = P(w_n | c_n) \cdot P(c_n | c_1, c_2, \cdots, c_{n-1}) . \quad (2.35)$$

Obučeni klasni jezički model se može koristiti bilo samostalno, bilo kao pomoćni model. Češće se koristi kao jedan od modela, čijom interpolacijom se dobijaju konačne verovatnoće. Takva interpolacija se, na primer, može raditi sa osnovnim modelom (baziranim samo na rečima), klasnim morfološkim modelom i modelom baziranim na lemama.

Nekoliko praktičnih problema postoji vezano za primenu klasnih modela jezika baziranih na morfologiji. Jedan problem jeste sam skup mogućih morfoloških klasa, koji je ograničen. Grupisanje reči u klase u većini slučajeva nije korisno u praksi ako je broj klasa premali, to jest, ako imamo na desetine hiljada reči koje pripadaju istoj klasi (čest slučaj kod morfoloških klasa). Moguće je kreirati proizvoljan broj klasa nekim automatskim algoritmom, na primer, Braunovim algoritmom klasterizacije, koji se bazira na iterativnom spajanju klasa do željenog broja, sa tim da se polazi od situacije u kojoj je svaka reč klasa za sebe (Brown i drugi, 1992), ali je tu teško odrediti optimalan broj, pošto on uvek zavisi i od veličine i sadržaja dostupnog korpusa za obuku, a ne samo od ciljane primene modela. Drugi problem je korišćenje klasnog modela u aplikacijama za koje je bitno da rade u realnom vremenu, pošto se zahteva automatska morfološka anotacija i klasifikacija reči u toku korišćenja, a taj zadatak je prilično vremenski zahtevan. Jedan od načina da se pomenuti problem reši jeste da se formira rečnik u kom se svaka reč uvek preslikava u istu morfološku klasu, što eliminiše potrebu za anotacijom i klasifikacijom tokom rada sistema, ali se, sa druge strane, na taj način potencijalno

gubi značajna količina informacija. Zbog toga ne bi smelo da se za svaku reč prosto izbaci svaka mogućnost morfološke kategorizacije osim jedne (čak i ako je to najčešća), već bi trebalo da se u ulaznom korpusu omogući distinkcija pojave iste reči u različitim morfološkim kategorijama (Pakoci i drugi, 2019).

Za predstavljanje morfoloških informacija pogodni su svakako i faktorisani modeli jezika. U njima, reči se mogu predstaviti vektorima obeležja, čije delove mogu činiti informacije vezane za identitet same reči (tog pojavnog oblika), koren reči, vrstu reči, određenu morfološku kategoriju, i tako dalje. Naravno, kao i kod svakog FLM, potrebno je odrediti odgovarajuću *back-off* proceduru na osnovu informacija o sekvencama svih obeležja za svaku od reči u datom kontekstu. Ako u modelu ne postoji sekvenca kompletnih skupova obeležja za date reči, obično se odbacuje jedno po jedno obeležje dok se ne pronađe postojeća kombinacija u modelu. Redosled odbacivanja obeležja može se zadati unapred, ili se može paralelno ići svim mogućim *back-off* putanjama, a konačan rezultat da se dobije uprosečavanjem.

Što se tiče modela srpskog jezika, nedavno je kreiran takozvani hibridni model, kod kojeg se osim identiteta samih reči u obzir uzimaju i osnovni oblik reči i određena morfološka klasa (Ostrogonac, 2018). Kod njega se, na primer, u slučaju trigrama, ako se dati niz reči $w_1w_2w_3$ ne nalazi u modelu, umesto prelaska na bigram w_2w_3, pokušava inicijalno sa pronalaženjem trigrama $l_1w_2w_3$, gde l_1 odgovara lemi od reči w_1 (koja je najudaljeniji deo relevantne istorije). Slično je definisana i celokupna putanja pri *back-off* proceduri – ako ne postoji ni trigram $l_1w_2w_3$, prelazi se na trigram $c_1w_2w_3$, gde c_1 predstavlja odgovarajuću morfološku klasu reči w_1, pa ako ni to ne postoji, prelazi se na sledeću reč, i tako dok se ne istroše sve reči u istoriji (u slučaju trigrama, imamo samo sledeću reč w_2 i to je sve). Ako nijedan od ponuđenih trigrama nije pronađen u modelu, tek onda se prelazi na bigrame. Eksperimenti su pokazali da se to vrlo retko dešava (Ostrogonac, 2018).

Kod ovog hibridnog modela problem je svakako sam obim modela, odnosno broj parametara. Prvi korak u pripremi njegove obuke bila je modifikacija tekstualnog korpusa, tako da se svaka reč zamenila uređenom trojkom koju čine reč, njena lema i morfološka klasa. Međutim, ne modeluju se verovatnoće uređenih trojki, već se formiraju *n*-grami od svih mogućih kombinacija informacija iz uređenih trojki. Na primer, za bigram *„Nikolu je“*, formiraju se svi sledeći bigrami: *„Nikolu je“*, *„Nikolu (biti)“*, *„Nikolu [glag_pmć_biti]“*, *„(Nikola) je“*, *„(Nikola) (biti)“*, *„(Nikola)*

[glag_pmć_biti]“, *„[imen_vl_aku_mr_jd] je“*, *„[imen_vl_aku_mr_jd] (biti)“*, i *„[imen_vl_aku_mr_jd] [glag_pmć_biti]“*. U prethodnim primerima, običnim zagradama obeleženi su osnovni oblici reči, a uglastim zagradama morfološke klase (pomoćni glagol *biti*, vlastita imenica u akuzativu jednine muškog roda). Broj parametara modela se na ovaj način za svaki *n*-gram sa *n* podigao na 3^n, što rezultujući model čini mnogo većim. Takođe, potrebno je bilo skalirati pojedine verovatnoće u zavisnosti od broja reči koje se preslikavaju u datu lemu ili klasu, a zatim procedurom odsecanja retkih *n*-grama eliminisati one kombinacije reči, lema i klasa za koje se ne može sa dovoljno velikom sigurnošću tvrditi da je estimirana verovatnoća njihove pojave prihvatljivo tačno određena. Odgovarajuće leme i morfološke klase morale bi, naravno, biti određivane i tokom praktične primene ovakvog sistema. Iako je eksperimentisano sa varijantama klasterizacije reči u klase (automatsko, ili na osnovu morfoloških klasa) i u okviru RNN arhitekture sa faktorisanim izlaznim slojem (Ostrogonac i drugi, 2019), eksplicitno modelovanje morfologije u okviru modela jezika kao što su FLM nije do sada rađeno za srpski.

POGLAVLJE III:

DOSTUPNI RESURSI ZA OBUKU SRPSKIH ASR SISTEMA

Za razvoj ASR sistema na velikim rečnicima, sa performansama koje omogućavaju upotrebu u praksi u okviru iole širih domena upotrebe, od izuzetnog je značaja postojanje i korišćenje kvalitetnih i dovoljno obimnih baza podataka i sličnih resursa. U prethodnom periodu, uz kontinuirani razvoj alata za obuku akustičkih i jezičkih modela za srpski jezik, prikupljeno je i obrađeno nekoliko različitih audio baza podataka, a stalno se radilo i na proširivanju skupa tekstualnih sadržaja koji je na raspolaganju. Što se tiče audio baza, za razne primene formirane su baza telefonskih snimaka, baza oglasa, baza audio knjiga, baza radio emisija, mobilna baza (interakcije ljudi sa pametnim mobilnim telefonima), ali i neke druge baze za specifične namene. Što se tiče tekstova, osim transkripcija audio baza, radilo se na proširivanju prvenstveno srpskog novinskog korpusa, ali i na prikupljanju materijala što više drugih funkcionalnih stilova.

U ovom poglavlju biće detaljno predstavljeni trenutno dostupni audio i tekstualni resursi za srpski jezik, koji su se koristili u eksperimentima opisanim u kasnijim poglavljima.

AUDIO RESURSI

SRPSKE AUDIO BAZE PODATAKA

Za sve aktuelne obuke srpskih akustičkih modela koristi se nedavno proširena govorna baza podataka na srpskom jeziku. Ovaj materijal se sastoji iz tri velike celine, kao što je prikazano u tabeli 1.

Prvu celinu čine audio knjige, koje su snimljene u studio okruženju, i čitane od strane profesionalnih govornika. Snimci su uglavnom veoma visokog kvaliteta, bez mnogo pozadinske buke ili grešaka u izgovoru reči. Baza audio knjiga je pominjana u većem broju radova vezanih za srpske ASR sisteme, kao i kompletan proces njenog sakupljanja i pripreme za potrebe automatskog prepoznavanja govora, što osim transkribovanja materijala i provere potencijalnih grešaka u snimcima pre svega

uključuje podelu na kraće snimke koji sadrže po jednu rečenicu (Suzić i drugi, 2014; Pakoci i drugi, 2017; Pakoci i drugi, 2018). U skorije vreme, ova audio baza je značajno proširena dodavanjem još materijala (Pakoci i drugi, 2019). Trenutno, ukupno je dostupno oko 168 sati audio materijala, od kojih čist govor čini oko 140 sati, dok ostatak čine tišina i drugi negovorni segmenti (buka, mljackanje, disanje govornika i razni artefakti). U toj količini materijala, identifikovano je 32 različita muška i 64 različita ženska govornika, sa tim da muški govornici imaju mnogo više materijala po govorniku u proseku. Generalno, neki govornici, prvenstveno muški, su dominirali po količini materijala (neki i sa više od 5 sati), dok su neki imali značajno manje (pola sata, pa i manje). Da bi se što više ujednačila količina materijala po govorniku, čime bi se sprečila pristrasnost (eng. *bias*) rezultujućih modela govornicima sa značajno više materijala, a istovremeno veštački uvećao broj govornika, odnosno raznolikost glasova, materijal svakog od govornika je, ako ga je bilo dovoljno, bio izdeljen na manje delove od po najviše 30-35 minuta, a zatim je svaki od tih delova, osim prvog, modifikovan pažljivo odabranim kombinacijama promene brzine govora (tempa) i visine glasa (piča). Time je praktično dobijen veliki broj novih, međusobno distinktnih veštačkih govornika za obuku akustičkog modela. Opisana procedura je rezultovala sa 398 novih govornika (uključujući stare, nepromenjene) sa relativno bliskom količinom materijala, od toga 208 muških i 190 ženskih.

Drugu celinu čine snimci iz nekolicine radio emisija, koje su se takođe primenjivale u ranijim istraživanjima (Suzić i drugi, 2014; Pakoci i drugi, 2017; Pakoci i drugi, 2018). I ovi snimci su svi izdeljeni tako da obuhvataju po jednu rečenicu, i transkribovani uz obeležavanje oštećenih reči, kod kojih postoji nejasan ili pogrešan izgovor reči, postoji značajna buka ili muzika, ili više ljudi priča u isto vreme. Ovi snimci su nešto lošijeg kvaliteta, imaju značajno manji spektralni opseg od audio knjiga, a dodatno ima i prethodno pomenutih grešaka u govoru. Međutim, oni su od ključnog značaja za modelovanje spontanijeg govora, kakvog nema u čitanom tekstu, a transkripcije mogu poslužiti kao dodatak korpusu u razgovornom funkcionalnom stilu. U odnosu na neke ranije radove, ovaj deo audio baze podataka je značajno proširen (Pakoci i drugi, 2019). Trenutno, on obuhvata oko 179 sati materijala (više čak i od knjiga), od čega je 150 sati govora. Ima puno više muških govornika, ukupno 21, u odnosu na 14 ženskih. Ukupan broj govornika nije veliki, jer

su se uglavnom snimale iste emisije (voditelj je uvek isti), a i svaka emisija pojedinačno može dugo da traje. Zbog toga je i ovde dobrodošla procedura ujednačavanja govorne baze. Nakon modifikacija tempa i piča, ukupno je dobijeno čak 350 novih veštačkih muških govornika, ali i samo 70 ženskih (ukupno 420 ujednačenih govornika). Vredi napomenuti da je određenim promenama visine glasa moguće dobiti i od originalnih muških glasova nove glasove koje se makar približavaju ženskim (i obratno).

Poslednja celina je takozvana mobilna baza. U pitanju je skup snimaka interakcija ljudi sa pametnim mobilnim telefonima. Takođe se koristio već u pojedinim radovima (Pakoci i drugi, 2017; Pakoci i drugi, 2018; Pakoci i drugi, 2019), a za razliku od prethodnih delova govorne baze, nije se menjao. Svi govornici u okviru ovog segmenta su izgovarali neke od datih govornih komandi mobilnom telefonu, simulirajući interakciju sa aplikacijom tipa govornog asistenta, na srpskom jeziku. Svaki snimak sadrži eksplicitnu komandu (otvaranje imenika, pozivanje nekog kontakta po imenu, prikaz liste poziva, promena podešavanja, i tako dalje), pitanje (na primer, traženje informacija o voznom redu ili sportskom rezultatu), niz brojeva, datum, ime i/ili prezime, toponim, spelovano ime, ili sličnu, uglavnom kratku rečenicu, najčešće baziranu na nekom upitu (tip rečenica koji se može očekivati u interakciji sa govornim asistentom na mobilnom telefonu). Sve rečenice su izgovarane nešto spontanije od čitanih audio knjiga, ali ne toliko spontano kao u radio emisijama. Kvalitet snimaka je uglavnom dobar (sa nekoliko izuzetaka), iako zavisi od mikrofona korišćenog telefona, a u odnosu na preostale dve celine snimci su u proseku puno kraći, sa veoma limitiranim rečnikom (svega 3500-4000 različitih reči ukupno), pa se stoga i mnoge rečenice ponavljaju kod različitih govornika. Količina materijala je već približno usklađena po govorniku, jer je od svih zahtevan jednak broj rečenica svakog tipa. Mobilna baza sadrži ukupno 61 sat materijala, od čega je 41 sat govor. Ukupno ima 169 distinktnih muških i 181 ženskih govornika.

Svi pomenuti snimci su jednokanalni PCM zapisi i imaju učestanost odabiranja 16 kHz, sa 16 bita po odbirku. Od celokupne govorne baze podataka, između 5% i 10% svake od celina je izdvojeno za potrebe testiranja (oko 10% knjiga i emisija, odnosno oko 5% znatno ujednačenije mobilne baze). Ova test baza se ukupno sastoji od oko 29 sati materijala, od čega je oko 23 sata govora, sa ukupno 81 na slučaj odabranih govornika (50 muških i 31 ženskih). Nakon što je svaki test govornik

određen, njegov celokupni materijal je prešao u test skup, to jest, nije uopšte učestvovao u obuci akustičkog modela. Takođe, njegove transkripcije se nisu koristile za obuku jezičkih modela. Na taj način su svi eksperimenti potpuno fer. Opisani test skup je trenutno najopsežnija postojeća ASR test baza na srpskom jeziku.

Tabela 1. Pregled audio baze podataka po celinama

Deo govorne baze	Količina materijala (u satima)	Količina čistog govora (u satima)	Broj muških govornika	Broj ženskih govornika
audio knjige	168	140	208	190
radio emisije	179	150	350	70
mobilna baza	61	41	169	181
ukupno	408	331	727	441
za obuku	379	308	677	410
za test	29	23	50	31

Sve vrednosti su date nakon ujednačavanja količine materijala po govorniku. Svaki govornik u bazama audio knjiga i radio emisija ima najviše 30-35 minuta materijala. U mobilnoj bazi nije bilo potrebe za ujednačavanjem.

HRVATSKE AUDIO BAZE PODATAKA

Uz sve prethodno navedene srpske audio baze podataka, u pojedinim obukama korišćene su i ekvivalentne hrvatske baze, radi poboljšanja prepoznavanja ASR sistema i u širem spektru govornika. Osim toga, duboke neuronske mreže gotovo uvek imaju koristi od značajnog proširivanja skupa za obuku. Ove baze su korišćene isključivo za obuku akustičkog modela (nisu ulazile u test skup). Hrvatske audio knjige sadrže oko 196 sati materijala, raspoređenih na 414 govornika (272 ženska i 142 muška), usklađenih po trajanju govora. Hrvatske radio emisije sadrže oko 184 sata materijala i ukupno čak 673 govornika (428 muških i 245 ženskih). Konačno, hrvatsku mobilnu bazu čini oko 155 sati materijala od 655 različitih govornika (328 ženskih i 327 muških). Dodavanjem hrvatskih baza količina podataka za obuku modela je i više nego duplirana. I ovi snimci su, naravno, u istom formatu kao i srpska audio baza.

Zajedno, srpske i hrvatske audio baze podataka za ASR obuke sadrže čak oko 914 sati materijala, raspoređenog na 2829 govornika, od čega je čak 525 sati i 1742 govornika dodato preko hrvatskih baza. To je svakako najveća audio baza koja se koristi za obuku srpskih akustičkih modela.

ZAŠUMLJENE AUDIO BAZE PODATAKA

U cilju daljeg veštačkog povećanja količine materijala za ASR obuku, jedan od mogućih pristupa je dodavanje šuma na postojeće audio snimke, i zatim korišćenje rezultujućih zašumljenih snimaka kao dodatka postojećem materijalu.

Da bi se ovoj proceduri pristupilo, pre svega je neophodno imati dovoljno veliku bazu snimaka šumova. Ona treba da sadrži što više različitih tipova buke koje mogu da utiču na slabije performanse ASR sistema u praksi. Neke od mogućih vrsta šuma su muzika, saobraćajna buka, žamor, pozadinski govor, vetar, lavež pasa, kašljanje, građevinski radovi, i tako dalje. U okviru ASR obuka, koristi se baza šuma koja trenutno sadrži preko 7 sati snimaka različite buke. Još je bitno napomenuti da je ponekad poželjno imati stereo snimke šuma, pogotovo ako je planirano koristiti ASR sistem u okruženju sa više mikrofona (na primer, u aplikacijama za mobilne telefone). U tom slučaju, naravno, očekuje se da i originalna audio baza bude u stereo formatu.

Da bi dodavanje šuma bilo još efikasnije, postojeća baza šuma je dalje izdeljena na snimke trajanja od oko 4 sekunde. To je urađeno iz dva razloga. Prvi razlog je omogućavanje iskorišćenja celokupnih snimaka, jer ako su originalni audio snimci kraći od snimaka šuma, veliki delovi tih snimaka neće nikad biti iskorišćeni za zašumljenje (samo početni delovi bi bili korišćeni). Drugi razlog je povećanje raznolikosti dodatog šuma – pošto se snimak šuma za dodavanje nekom originalnom snimku bira na slučaj, povećanjem broja snimaka šuma smanjuje se šansa da ćemo izabrati neki koji je već ranije korišćen. U slučaju da je snimak šuma kraći od snimka na koji se dodaje (najčešći slučaj), šum se ponovi onoliko puta koliko treba dok ne pokrije ceo originalni signal.

Kao što je već spomenuto, biranje snimka šuma za dodavanje datoj audio datoteci se radi na slučaj (recimo preko generatora pseudoslučajnog niza brojeva). Još jedan bitan parametar pri dodavanju šuma je željeni odnos signal-šum (eng.

Signal-to-Noise Ratio, SNR). Njegova vrednost se tipično izražava u decibelima, a dobija se formulom

$$SNR\ [dB] = 10\log_{10}\frac{P_s}{P_n} = 10\log_{10}\left(\frac{A_s}{A_n}\right)^2. \tag{3.1}$$

U izrazu (3.1) P_s i P_n su snage korisnog signala i šuma, a A_s i A_n srednje kvadratne vrednosti (eng. *Root-Mean-Square*, RMS) njihovih amplituda. Empirijski je utvrđeno da vrednosti SNR mnogo manje od 10 dB ne koriste ASR sistemima na veoma velikim rečnicima za srpski jezik, jer značajno povećavaju varijabilnosti u materijalu za obuku, pa zbog toga i zahtevaju bitno povećan broj parametara sistema da bi modeli bili dovoljno dobri, što usporava ASR sistem tokom praktičnog korišćenja. U eksperimentima sa postojećom bazom šuma, koristile su se SNR vrednosti od 11, 13, 15 i 17 decibela. Ako je planirana namena ASR sistema u relativno tihim uslovima, SNR od 17 dB se pokazao kao sasvim dovoljan. Takođe, ukoliko želimo višestruko umnožiti postojeću audio bazu podataka, možemo odabrati nekoliko različitih SNR vrednosti umesto jedne i ponoviti proceduru dodavanja šuma toliko puta, svaki put dobijajući novu zašumljenu bazu sa različitim nivoom šuma (a zbog slučajnosti, neće uvek isti snimak šuma biti dodat na isti originalni snimak, što je takođe dobro). Međutim, ukoliko u celosti koristimo pomenutu srpsku i hrvatsku audio bazu, za tako nečim nema potrebe.

TEKSTUALNI RESURSI

Kako je srpski visoko inflektivan i morfološki bogat jezik, za obuku dobrog modela srpskog jezika potrebna je veoma velika količina podataka. Zato je proces prikupljanja i adekvatne pripreme tekstualnog korpusa vrlo značajan korak i za formiranje kvalitetnog jezičkog modela i, kasnije, za performanse celog ASR sistema.

Postojeća baza tekstova za srpski jezik jedva može da se smatra bazom srednje veličine, ako je poredimo sa bazama za druge jezike (Mikolov i drugi, 2011a). Njen najveći deo čine ranije prikupljeni i obrađeni tekstovi za potrebe obuke jezičkih modela (Pakoci i drugi, 2017; Popović i drugi, 2018). Oni su podeljeni prema funkcionalnom stilu koji u njima dominira u nekoliko grupa. U pitanju su novinski (ili žurnalistički, publicistički) korpus, koji je najveći, zatim literarni (ili književno-umetnički), administrativni (ili birokratski), naučni, naučno-popularni i razgovorni

korpusi. Sve njih čine tekstovi prikupljeni iz mnoštva izvora. Tekstove novinskog korpusa uglavnom čine novinski članci raznih tematika. Literarni korpus čine odlomci pojedinih romana i kratkih priča. Sadržaj administrativnog korpusa čine delovi Ustava Republike Srbije i pojedini zakoni, kao i tekstovi raznih Ugovora, sudskih dokumenata, molbi, žalbi i slično. Tekstovi koji pripadaju naučnom stilu su preuzeti iz doktorskih disertacija, diplomskih radova i drugih naučnih i stručnih radova i publikacija. Naučno-popularni tekstovi su izdvojeni iz pojedinih naučno-popularnih časopisa. Konačno, kao razgovorni stil poslužili su titlovi filmova. Primeri rečenica za svaki od ovih funkcionalnih stilova dati su u dodatku 4.

Pojedini od nabrojanih stilova nisu dovoljno veliki da bi samostalno učestvovali u modelovanju jezika (pre svega naučno-popularni i razgovorni korpusi), ali se njima mogu dopuniti ostali. U cilju pokrivanja što više varijabilnosti, mogu se koristiti svi korpusi zajedno, pošto je pokazano da različiti funkcionalni stilovi mogu značajno da se razlikuju međusobno (Ostrogonac i drugi, 2012). Alternativno, moguće je obučiti posebne modele jezika za svaki od stilova (makar za stilove sa dovoljno materijala), pa u praksi raditi interpolaciju rezultata pojedinih modela sa različitim ponderima. Nažalost, takav pristup bi zahtevao još dosta više tekstova nego što trenutno postoji, za većinu stilova (Ostrogonac, 2018).

Nakon prikupljanja, sve tekstove je potrebno dodatno obraditi i pripremiti za upotrebu u LM obukama. Ta procedura obuhvata niz operacija, od kojih su samo neke usaglašavanje raznih konvencija (na primer, tretman brojeva i skraćenica), eliminacija nebitnih delova teksta (na primer, crtice ili brojevi tokom nabrajanja, pojedini znaci interpunkcije, i tako dalje), podela na rečenice, formatiranje teksta i slično. Za tu namenu razvijeno je nekoliko alata koji pojedine zadatke mogu da obave automatski ili poluautomatski (Ostrogonac, 2018), dok bi ipak ponešto ostalo i za ručnu korekciju (na primer, tabele sportskih rezultata, koje su uglavnom bile potpuno izbačene, jer bi inače narušavale statistike, zatim matematičke formule i izrazi u naučnim tekstovima, i slično). Neki od zadataka koji su razvijeni alati za obradu teksta izvršavali automatski su konverzija pisma (iz ćirilice u latinicu, gde je bilo potrebe), uklanjanje nepoželjnih simbola, segmentacija teksta na rečenice, izdvajanje spiska svih različitih reči za morfološku anotaciju, zamena reči odgovarajućim klasama za određene zadatke, kreiranje raznih statistika (recimo učestanosti pojava reči i grafema, to jest karaktera), pronalaženje dupliranih rečenica, i još mnogo toga drugog.

Dodatno, na postojeće tekstove za razne funkcionalne stilove mogu se dodati i transkripcije srpskih govornih baza podataka za obuku akustičkog modela. Ta dodatna količina podataka svakako nije zanemarljiva. Prema sadržaju tih audio baza, može se lako zaključiti da u njima dominiraju literarni (audio knjige), razgovorni (pojedine radio emisije, potencijalno i mobilna baza) i novinski stilovi (ostatak radio emisija).

U tabeli 2 dat je pregled kompletnog korpusa za obuku jezičkog modela za srpski jezik. Ukupno, na raspolaganju ima oko 1,4 miliona rečenica, odnosno oko 26 miliona reči. Treba napomenuti još da je tokom RNNLM obuka, od ovih rečenica njih 20000 uvek izdvajano kao validacioni skup, koji se nije koristio u samoj obuci modela. U tabeli 3 su date najčešće reči u korpusu po stilovima, sa brojem pojavljivanja.

Tabela 2. Pregled tekstualnog korpusa

Deo korpusa (stil)	Broj rečenica	Broj reči	Broj različitih reči	Broj karaktera (grafema)
novinski	737k	17M	313k	94M
literarni	303k	3,9M	184k	18M
naučni	23k	503k	48k	3M
administrativni	15k	378k	19k	2M
naučno-popularni	18k	357k	30k	2M
razgovorni	38k	128k	15k	530k
transkripcije	251k	3,2M	158k	15M
ukupno	1,4M	26M	458k	135M
validacioni skup	20k	470k	55k	2,6M

Primećuje se velika razlika u dužinama rečenica u novinskom, administrativnom i naučnom stilu (22-26 reči), u odnosu na literarni (13) i svakako razgovorni stil (manje od 4 reči u proseku). Transkripcije audio baza su takođe velikim delom u literarnom stilu.

Tabela 3. Najčešće reči u korpusu i njihov broj pojava

Deo korpusa (stil)	Najčešće reči (broj pojava)	Najčešće reči sa >4 slova (broj pojava)
novinski	je (732k), u (604k), i (583k), da (495k), na (264k), za (232k), se (219k), su (170k)	Srbije (67k), rekao (54k), godine (52k), danas (49k), hiljadu (48k)
literarni	je (144k), i (139k), da (135k), se (95k), u (90k), na (54k), su (40k), ne (36k)	nešto (6k), jedan (6k), nisam (5k), svoje (5k), vreme (5k), ništa (4k)
naučni	i (19k), u (16k), je (11k), se (11k), na (9k), da (7k), za (6k), su (6k)	posto (1k), jedan (1k), godine (1k), hiljadu (1k), odnosno (1k)
administrativni	i (12k), u (12k), se (10k), za (6k), ili (6k), je (6k), na (6k), da (6k), od (4k), ovog (4k)	člana (3k), stava (2k), odnosno (2k), prvog (2k), zakona (1k), društva (1k)
naučno-popularni	u (12k), se (10k), i (9k), je (9k), da (7k), na (6k), za (3k), sa (3k)	signala (1k), sistema (1k), zvuka (1k), jedan (1k)
razgovorni	je (6k), da (5k), ne (3k), to (3k), se (2k), šta (2k), ti (2k), u (2k)	dobro (700), zašto (600), hvala (500), nisam (400)
transkripcije	i (117k), da (117k), je (115k), u (73k), se (63k), na (41k), to (37k), su (35k)	jedan (7k), nešto (5k), treba (4k), godine (4k), dvadeset (4k), nisam (4k)

U svim stilovima dominiraju iste kratke reči – pomoćni glagoli „je“ i „su“, veznici „i“ i „da“, predlozi „u“, „na“ i „za“, povratna zamenica „se“ i slično. Broj pojava dužih reči, sa druge strane, veoma zavisi od funkcionalnog stila.

POGLAVLJE IV:

DOSADAŠNJA OSTVARENJA U RAZVOJU SRPSKIH LM

U modelovanju srpskog jezika su do pre nekoliko godina dominirali *n*-gram modeli. Prvenstveno, koristili su se trigram modeli, za sve primene na više od nekoliko hiljada reči, a za vrlo konkretne primene na manjim rečnicima upotrebljavani su i bigram modeli, kao i regularne gramatike. Objavljeni su rezultati za rečnike do oko 120000 reči, dok je za sve šire primene utvrđeno da trenutni metodi nisu dovoljno dobri (Pakoci i drugi, 2018).

Prethodnih par godina ispitane su mogućnosti primene postojećih javno dostupnih RNNLM rešenja i za srpski. Pojedini poznati alati, kao što su Mikolov RNNLM, CUED-RNNLM i TensorFlow paket alata, prilagođeni su srpskom jeziku i testirani na postojećim bazama podataka. U ovom poglavlju su ukratko opisani do sada korišćeni alati (i za akustičko i za jezičko modelovanje), a zatim je dat pregled dobijenih rezultata.

PREGLED KORIŠĆENIH ALATA ZA OBUKU AKUSTIČKIH I JEZIČKIH MODELA

U trenutku testiranja jezičkih modela koji su prikazani u ovom poglavlju, akustički modeli su obučavani na nešto manjoj audio bazi podataka nego što je to opisano u poglavlju 3, kao i bez ujednačavanja količine materijala po govorniku (Pakoci i drugi, 2018). Shodno tome, i test skup je bio nešto manji i drugačiji, ali izdvajan po istom principu. Najveći korišćeni rečnik je imao oko 121000 reči, a korpus za obuku modela jezika sastojao se samo od transkripcija audio baze za obuku akustičkog modela i dela novinskog korpusa, koji je služio samo za bolju estimaciju verovatnoća *n*-grama, u smislu da se u rečnik nisu dodavale nove reči koje su se nalazile u njemu, jer su same transkripcije sadržale preko 120000 reči, a više od toga bi bilo previše za tada korišćene metode obuke LM.

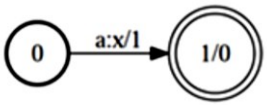
U narednim sekcijama su opisani korišćeni alati za pripremu ovih ASR sistema. Za akustičku obuku korišćen je paket alata Kaldi (Povey i drugi, 2011), koji

se oslanja na automate sa konačnim brojem stanja, odnosno paket alata OpenFst preko kojeg su oni implementirani (Allauzen i drugi, 2007). Trigram modeli jezika su uvek obučavani preko alata SRILM (Stolcke i drugi, 2011).

TRANSDUKTORI I OPENFST PAKET ALATA

Dekodovanje ulaznog signala u ASR sistemu predstavlja proces rekonstrukcije izgovorenih reči na osnovu akustičkih informacija u govoru (verovatnoće odgovarajućih govornih jedinica na osnovu izračunatih akustičkih obeležja) i ograničenja koja nameće jezički model. Da bi se to omogućilo, ASR dekoderu je potrebna neka arhitektura, a moderna rešenja često za tu namenu koriste transduktore, odnosno pretvarače (eng. *transducers*) – oni su varijanta akceptora, to jest automata sa konačnim brojem stanja koji na osnovu date ulazne sekvence simbola proizvode binarni izlaz koji određuje da li je ulazna sekvenca simbola prihvaćena ili ne (Mohri i drugi, 2002). Kao i kod akceptora, svako stanje transduktora može biti prihvatajuće (to jest finalno) ili neprihvatajuće, a ulazna sekvenca se prihvata ako se nakon obrade svih ulaznih simbola transduktor nalazi u prihvatajućem stanju. Za razliku od akceptora, za sve moguće prelaze između svojih stanja transduktori imaju dve pridružene labele, ulaznu i izlaznu (dok akceptori imaju jednu labelu, što je ekvivalentno transduktoru kojem su ulazne i izlazne labele uvek iste). Pri prepoznavanju govora uz pomoć transduktora, na primer u transduktoru koji opisuje rečnik izgovora, ulazne labele mogu biti fonemi, a izlazne reči. Umesto konkretne labele, prelazu može biti pridružen i specijalni epsilon simbol (ε), koji prosto označava da se pri konkretnom prelazu ne emituje ništa (u prethodno pomenutom primeru, reč kao izlazna labela može da se emituje samo u prelazu kada je ulazna labela početni fonem te reči, a preostalim ulaznim simbolima (fonemima) može da se pridruži izlazna labela ε). Transduktori mogu biti i ponderisani, i kod njih se svakom prelazu pridružuje i odgovarajuća težina, odnosno cena (eng. *weight*). Dodatno, cena se može pridružiti i svakom finalnom stanju. Tada, ako za jedan niz ulaznih simbola postoji više prihvaćenih putanja kroz transduktor, kao rezultat se uzima ona sa najnižom ukupnom cenom. Na slici 11 dat je vrlo jednostavan ponderisani transduktor – on ima početno stanje 0, finalno stanje 1 sa cenom 0, i prelaz između njih sa ulaznom labelom *a* i izlaznom labelom *x*, i cenom 1. Velika prednost transduktora u odnosu na neke druge arhitekture dekodera je to što se nekoliko njih može lako

uvezati primenom operacije kompozicije, a to se može iskoristiti za međusobno uvezivanje transduktora koji predstavljaju model jezika, rečnik izgovora i akustički model.



Slika 11. Primer jednostavnog transduktora

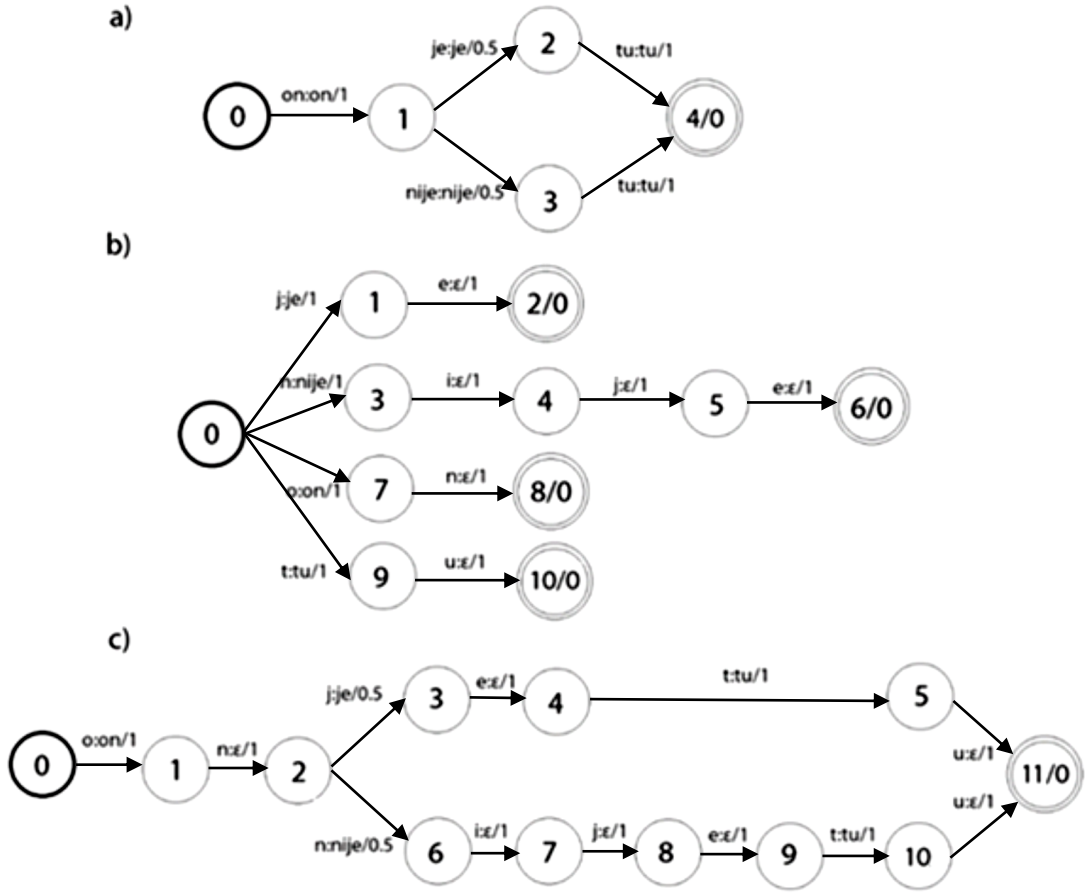
OpenFst (Allauzen i drugi, 2007) je javno dostupan paket alata koji sadrži efikasnu implementaciju ponderisanih transduktora sa konačnim brojem stanja (eng. *Weighted Finite-State Transducer*, WFST). Sastoji se od C++ biblioteke šablonskih funkcija za reprezentaciju i operacije sa WFST. Omogućava naravno i čuvanje WFST u odgovarajućem formatu na disku računara, kao i njegovo kasnije učitavanje. Sadrži i pripremljene programe za izvršavanje raznih operacija nad WFST. Dizajniran je tako da bude vrlo efikasan, i vremenski, a i po pitanju resursa, kao i da se lako skalira na velike probleme.

Najznačajnije operacije koje OpetFst omogućava, a od velike su koristi u ASR, jesu kompozicija, determinizacija i minimizacija. Postoje i mnoge druge, ali one nisu u fokusu ove disertacije.

Kompozicija dva WFST-a je operacija uvezivanja, odnosno kombinovanja dva srodna WFST-a u pogledu labela (odnosno reprezentacije). Na primer, transduktor rečnika izgovora, koji povezuje reči sa odgovarajućim nizovima fonema, kompozicijom se može uvezati sa transduktorom gramatike ili statističkog modela jezika, koji određuje dozvoljene nizove reči, da bi se napravio transduktor koji mapira foneme na reči, ali čiji nizovi reči su ograničeni datom gramatikom ili datim statističkim jezičkim modelom. Pri kompoziciji je bitno da izlazne labele jednog WFST-a odgovaraju ulaznim labelama drugog. Prikaz kompozicije dva WFST-a, koji predstavljaju jednostavan model jezika i rečnik izgovora, dat je na slici 12.

Operacija determinizacije nad datim transduktorom formira ekvivalentan WFST koji u svakom stanju za datu ulaznu labelu ima tačno jedan mogući prelaz u naredno stanje (nikako više od 1), čime sprečava dvosmislenosti u obradi ulaznih podataka, i uz to ne sadrži ni jedan epsilon simbol među ulaznim labelama. Deterministčan WFST za svaki niz ulaznih simbola ima najviše jednu moguću

putanju kroz njegova stanja. U ASR-u je ova operacija vrlo bitna pre svega zbog redundantnosti u transduktoru rečnika izgovora u prepoznavanju na velikim rečnicima.



Slika 12. Kompozicija dva WFST-a

a) Primer prostog jezičkog modela u obliku WFST, omogućava rečenice „on je tu“ i „on nije tu“; b) Primer rečnika izgovora u obliku WFST; c) Rezultat kompozicije – WFST koji mapira foneme na nizove reči ograničene modelom jezika.

Operacija minimizacije od datog determinističnog transduktora formira ekvivalentan manji transduktor, sa minimalnim brojem stanja i prelaza od svih mogućih ekvivalentnih determinističnih WFST. Pokazuje se da se svaki deterministčan WFST može minimizovati korišćenim klasičnim algoritmom (Mohri, 2000). Ova operacija je korisna zbog uštede i u pogledu resursa (potrebne memorije), ali i u pogledu vremena (minimizovani WFST je efikasniji u naknadnim operacijama, kao i u dekoderu).

Za dekodovanje uz pomoć ponderisanih transduktora najčešće se koristi varijanta takozvanog *token-passing* algoritma sa Viterbijevim kriterijumom (Young i drugi, 1989). Pri tome se tokeni, koji u sebi sadrže informaciju o istoriji dekodovanja do tog trenutka (prethodna stanja, cene i tako dalje), propagiraju mogućim prelazima kroz stanja datog transduktora u svakom kvantu vremena, odnosno frejmu, to jest za svaki sledeći ulazni simbol. Takođe, u svakom frejmu vrši se i odsecanje (eng. *pruning*) aktivnih tokena prema trenutno akumulisanoj ceni (određeni broj tokena sa najlošijom cenom se odbacuje), i na kraju dekodovanja, kada se obrade svi podaci, bira se najbolja moguća putanja prema ukupnoj ceni, odnosno na osnovu najboljeg prihvaćenog tokena, što je token sa najmanjom ukupnom cenom. Na osnovu najbolje putanje se rekonstruišu akustička stanja, fonemi i konačno odgovarajuće reči kroz koje je taj token prošao.

Jedan od najboljih paketa alata za razvoj sistema za prepoznavanje govora koji inkorporira korišćenje WFST i *token-passing* algoritam za dekodovanje jeste Kaldi (Povey i drugi, 2011). Kaldi je javno dostupan skup alata namenjen prvenstveno za ASR, napisan u C++ programskom jeziku, a sadrži i odgovarajuće skriptove preko kojih je moguće pokrenuti obuke različitih varijanti akustičkih modela (i HMM i DNN, sa mnogo podvarijanti), nezavisnih od govornika ili adaptiranih na govornika. Takođe implementira i nekoliko algoritama za izdvajanje obeležja iz govornog signala (MFCC, PLP, osnovne učestanosti, i drugih), na osnovu kojih se estimiraju parametri modela. Kaldi upotrebljava OpenFst funkcije u okviru svojih alata koji pretvaraju i akustički model i *n*-gram jezički model u odgovarajuće transduktore, a omogućava i sve potrebne funkcionalnosti za pripremu i korišćenje dekodera na osnovu njih. Postoji nekoliko verzija dekodera, u zavisnosti od vrste akustičkih modela.

WFST nad kojim operišu Kaldi dekoderi se obično naziva HCLG FST. Ime potiče od pojedinačnih transduktora koji su ukomponovani u taj krajnji, odnosno od onoga šta oni predstavljaju. Prvi WFST predstavlja gramatiku ili statistički model jezika (G). Drugi predstavlja rečnik izgovora, to jest leksikon (L). Primer rečnika izgovora u formatu koji koristi Kaldi dat je u dodatku 2. U proceduri sličnoj datom primeru na slici 12, ova dva WFST-a se uvežu u jedan (LG = L ○ G, gde je ○ simbol operacije kompozicije). Dobijeni WFST se zatim determinizuje i minimizuje pre daljih operacija. Potom se vrši kompozicija sa WFST koji mapira kontekstno-zavisne

foneme (tipično trifone) na standardne, kontekstno-nezavisne foneme (C). U okviru ovog WFST-a uzima se u obzir i kontekstno stablo (eng. *context-dependency tree*), na osnovu kog je tokom obuke akustičkog modela definisano koji trifoni se modeluju istim parametrima (Young i drugi, 1994), što se radi jer nije moguće precizno modelovati svaki različiti trifon jednostavno zato što ih ima jako mnogo, a podaci za obuku naravno nisu neograničeni. Dobijeni WFST (CLG = C ∘ LG) se potom uvezuje sa poslednjim WFST, koji predstavlja akustički model (H), odnosno njegovu topologiju (na primer, HMM stanja sa svim prelazima između njih), koja se u Kaldi-ju naziva model prelaza, to jest tranzicija (eng. *transition model*). Ta topologija zavisi od vrste akustičkog modela, ali u svakom slučaju odgovarajući WFST mapira identifikatore prelaza u akustičkom modelu (za koje su vezani pojedini parametri modela estimirani tokom obuke, na primer konkretna komponenta Gausove smeše u HMM-GMM sistemu) na kontekstno-zavisne foneme. WFST dobijen poslednjom kompozicijom se opet determinizuje i minimizuje, i uz još nekoliko sitnijih dodatnih operacija dobija se konačni HCLG FST.

SRILM PAKET ALATA

Najpoznatiji skup alata za istraživanja vezana za *n*-gram modele jezika je SRILM (*Stanford Research Institute Language Modeling toolkit*) (Stolcke i drugi, 2011). On je takođe javno dostupan, i sastoji se od brojnih C++ biblioteka, na osnovu kojih je razvijena nekolicina programa za obuku, interpolaciju, modifikaciju i testiranje jezičkih modela.

Najznačajniji program je *ngram-count*, koji služi za obuku *n*-gram modela jezika na osnovu datog korpusa i željenog rečnika. On izdvaja potrebne statističke podatke iz korpusa i uz tehnike ublažavanja raspodela verovatnoća i odsecanja viših *n*-grama formira model željene veličine i karakteristika. Dat je izbor nekoliko procedura ublažavanja, kao naravno i izbor željene vrednosti parametra *n*. Opciono, rečnik se može automatski generisati na osnovu svih reči u ponuđenom korpusu. Sa druge strane, ako se rečnik zada kao ulazni argument, sve reči koje postoje u korpusu, ali ne i u rečniku, se ignorišu, ili posmatraju kao dodatna, nepoznata reč (često se obeležava sa *UNK*, od engleskog *unknown*). Identifikator nepoznate reči se može i dodati rečniku, ili može prosto da predstavlja nepoznat kontekst.

Drugi programi u okviru SRILM paketa su *ngram-merge*, koji služi za interpolaciju više modela uz određene težine, zatim program *ngram*, koji služi za evaluaciju modela jezika (izračunava perpleksivnost na zadatom test skupu rečenica), a može da se koristi i za generisanje rečenica na osnovu datog modela (što je korisno pri analizi kvaliteta modela), *ngram-class*, koji vrši automatsku klasterizaciju reči u grupe na osnovu bigram statistika korpusa, *nbest-lattice*, koji može da posluži za izbor najverovatnije hipoteze procenom njihovih verovatnoća na osnovu datog modela, i *segment*, koji vrši automatsku segmentaciju datog teksta na rečenice uz pomoć obučenog modela.

PREGLED DOSADAŠNJIH REZULTATA U MODELOVANJU SRPSKOG JEZIKA

U ovoj sekciji su opisani rezultati primene trigram i RNN jezičkih modela na problemu prepoznavanja srpskog govora na velikim rečnicima (Popović i drugi, 2018). Korišćeni su svi prethodno predstavljeni alati. U svim navedenim testovima u pitanju je rečnik od oko 121000 reči, a baze podataka za obuku akustičkog i jezičkog modela su verzije pre poslednjeg proširenja i ujednačavanja po govorniku za audio bazu, odnosno bez dodatih funkcionalnih stilova za tekstualni korpus. Korpus za obuku LM je činilo oko 150000 rečenica iz transkripcija audio materijala, uz dodatih oko 442000 rečenica iz novinskog korpusa. Akustički model baziran na dubokoj neuronskoj mreži je fiksan, ranije određen kao optimalan od svih testiranih arhitektura (Pakoci i drugi, 2018).

TRIGRAM PRISTUP

Trigram model jezika je obučen preko SRILM alata, uz Kneser-Ney algoritam ublažavanja, koji je prethodno određen kao optimalan, i prag odsecanja viših *n*-grama od 10^{-7}, što se u ranijim eksperimentima pokazalo kao dobar kompromis između veličine rezultujućeg modela i njegove tačnosti (Pakoci i drugi, 2017). Takvom obukom dobijeno je, uz 121000 unigrama, oko 1,3 miliona različitih bigrama i 358000 trigrama u konačnom modelu. Izračunata perpleksivnost ovog modela jezika na datom test skupu bila je oko 768,8. Obučeni jezički model je korišćen u Kaldi dekoderu. Rezultat ovakvog modela jezika po pitanju stope greške prepoznavanja reči bio je 9,06%. Sa druge strane, stopa greške na nivou karaktera, odnosno slova,

iznosila je svega 2,37%, što nagoveštava potencijalne probleme vezane za različite oblike istih reči – sistem je relativno često grešio tako što je pogrešio samo oblik reči, a pogodio lemu. Tim povodom je procenjena i inflektivna stopa greške prepoznavanja reči (IWER), pri čemu se greška računala sa koeficijentom 0,5 u slučaju da je lema pogođena (ako nije, koeficijent ostaje 1). Ova izmenjena stopa greške je iznosila 7,23%, što ukazuje da je makar petina grešaka posledica pogrešnog oblika ispravne reči (Pakoci i drugi, 2018). Leme su za ovu priliku određene uz pomoć jednog alata o kom će više reči biti u poglavlju 5.

MIKOLOV RNNLM PRISTUP

Mikolov RNNLM pristup koristi standardnu rekurentnu neuronsku mrežu koja se sastoji od po jednog ulaznog, skrivenog i izlaznog sloja (Mikolov i drugi, 2011c). Na ulazu mreže reči su date u 1-od-*N* reprezentaciji, a taj vektor se zatim povezuje sa prethodnim stanjem skrivenog sloja. Za neurone skrivenog sloja koristi se sigmoidalna aktivaciona funkcija. Izlazni sloj predstavlja verovatnoću trenutne reči, uzimajući u obzir prethodnu reč i stanje skrivenog sloja u prethodnom koraku. Rekurentni težinski faktori se izračunavaju koristeći skraćenu verziju BPTT algoritma, uz zadavanje željenog broja koraka odmotavanja mreže. Takođe, izlazni sloj mreže je faktorisan – imamo podelu svih reči iz rečnika u željeni broj klasa na osnovu učestanosti pojave u korpusu za obuku, pa se na izlazu mreže modeluju verovatnoće klasa, a te verovatnoće se onda kombinuju sa uslovnom verovatnoćom konkretne reči ako je u pitanju data klasa. U ovom RNNLM pristupu, kao i u mnogim drugim, dodatno ubrzanje se postiže direktnim modelovanjem verovatnoća samo za skraćenu listu, to jest podskup svih reči (eng. *shortlist*), dok se ostatak ravnomerno raspoređuje na preostale reči. Ovaj uobičajeni postupak, uz podelu reči u klase, služi da bi se umanjio problem dimenzionalnosti podataka, a uz to doprinosi značajno i smanjenju izlaznog sloja mreže i ubrzavanju obuke, odnosno smanjenju potrebnih računskih resursa. Oko 5% rečenica (njih oko 30000) korišćeno je kao validacioni skup i one su bile izuzete iz obuke.

Obučena neuronska mreža modela jezika se u okviru Kaldi dekodera koristi tako što se učita graf dekodovanja (eng. *lattice*), koji predstavlja skup svih potencijalnih putanja kroz odgovarajući HCLG FST dobijen dekodovanjem trigram modelom jezika, i zatim se reskoruje. Procedura reskorovanja (eng. *rescoring*)

podrazumeva oduzimanje skorova (to jest cena, ili težina) koji dolaze od trigram jezičkog modela iz dobijenog grafa (na svim prelazima), i zatim dodavanje na ista mesta skorova koji bi se dobili obučenom neuronskom mrežom. Najčešće se, u stvari, koristi neki interpolacioni faktor za kombinovanje trigram i RNNLM skorova umesto čiste zamene – u konkretnom slučaju, u pitanju je bila RNNLM težina od 0,75 (kod proste zamene bi bilo 1,0). Postoji dva načina da se ovakvo reskorovanje obavi. Jedan način je direktan rad na grafovima (Liu i drugi, 2016), a drugi je formiranje *N-best* lista od grafova (Mikolov i drugi, 2010), što su spiskovi najverovatnijih *N* rečenica na osnovu trigram dekodovanja, i zatim skorovanje tih rečenica sa RNNLM da bi se odredile krajnje verovatnoće svake od mogućnosti sa liste (uz moguću interpolaciju sa trigram skorom). U slučaju direktnog rada na grafu, pri reskorovanju se vrši aproksimacija *n*-gramima, i to tipično trigramima ili 4-gramima (kvadrigramima), čime se u grafu objedinjuju sve istorije koje su jednake na nivou trigrama ili 4-grama, jer se na taj način sprečava eksponencijalna eksplozija mogućih putanja kroz graf tokom rada. U okviru Mikolov RNNLM pristupa koriste se razvijeni pomoćni alati za određivanje skorova na *N-best* listama, koje se prethodno dobijaju standardnim Kaldi programima i skriptovima.

Koristeći preporučenih 300 neurona u skrivenom sloju mreže i 40000 reči u skraćenoj listi reči (dakle, oko trećine rečnika) raspoređenih u 400 klasa, uz $N = 1000$ unosa u *N-best* listama, dobijen je najbolji WER rezultat sa Mikolov RNNLM alatima i on je iznosio 7,41%, a odgovarajući CER rezultat bio je tačno 2,00%. Relativno poboljšanje u odnosu na trigrame iznosilo je oko 18%. Razne druge kombinacije parametara nisu poboljšale rezultat. Perpleksivnost modela jezika dobijena na validacionom skupu iznosila je oko 116,1 (Popović i drugi, 2018).

FASTER-RNNLM PRISTUP

Faster-RNNLM pristup (takođe poznat i kao Yandex RNNLM po kompaniji Yandex Technologies čiji istraživači su razvili metod) se od Mikolov pristupa razlikuje prvenstveno po efikasnosti, odnosno brzini obuke, i donekle testa (Bakhtin & Edrenkin). Topologija mreže se sastoji od ulaznog sloja koji prihvata kompletnu relevantnu istoriju reči do datog trenutka u vidu jedinstvenog vektora (reprezentacije prethodnih reči spojene u jedan vektor), uz 1-od-*N* reprezentaciju date reči, zatim jednog skrivenog sloja koji koristi sigmoidalnu aktivacionu funkciju za formiranje

nove reprezentacije i izlaznog sloja koji daje krajnje RNNLM verovatnoće. U izlaznom sloju (koji je faktorisan, kao i kod Mikolov RNNLM pristupa), umesto standardne *softmax* aktivacione funkcije, koristi se ili takozvani hijerarhijski *softmax* uz primenu Hafmanovog binarnog stabla (Mikolov i drugi, 2013) koje kreira kratke binarne kodove za najčešće reči u korpusu, ili se aktivaciona funkcija aproksimira NCE (eng. *Noise Contrastive Estimation*) metodom koja koristi nelinearnu regresiju da diskriminiše među ulaznim opservacijama i datom raspodelom šuma, što se pokazuje kao bolje rešenje za veoma velike rečnike (Chen i drugi, 2015a). Ovom metodom je ostvareno ubrzanje obuke od oko 50% u odnosu na Mikolov RNNLM.

Što se tiče rezultata prepoznavanja, najbolji dobijen WER bio je za varijantu sa 300 neurona u skrivenom sloju i oko 78000 reči u skraćenoj listi (tu su ušle sve reči koje su se pojavile barem 3 puta u korpusu za obuku) podeljenih u 500 klasa, i iznosio je 7,46%, dok je CER bio 2,06%, što je blago lošije od Mikolov modela, što je bilo i očekivano, ali uz značajno kraću obuku. Perpleksivnost ovog RNNLM modela iznosila je oko 130,0 na validacionom skupu, koji je odabran na identičan način kao kod Mikolov RNNLM pristupa (Popović i drugi, 2018).

CUED-RNNLM PRISTUP

CUED-RNNLM pristup se od prethodna dva pristupa razlikuje prvenstveno po izlaznom sloju mreže, koji u ovom slučaju nije faktorisan (ne koriste se klase), već je u pitanju pun izlazni sloj (Chen i drugi, 2015b). Ovim je dobijena puno zahtevnija procedura obuke, koja zbog toga koristi grafički procesor računara (GPU). Takođe, umesto standardnog kriterijuma unakrsne entropije za određivanje gradijenta vektora greške, koristi se unapređen kriterijum – regularizacija varijanse – kod koga se eksplicitno dodaje varijansa normalizacionog faktora standardnoj CE funkciji cilja (Chen i drugi, 2015c), a dodatno se koristi i prethodno pomenuti NCE pristup umesto standardne *softmax* aktivacione funkcije (Chen i drugi, 2015a); naime, pretpostavlja se da se svaka reč zajednički generiše i iz raspodele podataka za obuku (RNNLM raspodele) i iz raspodele šuma (na slučaj odabrane grupe uzoraka šuma, procedurom koja se tipično oslanja na unigrame), a funkcija cilja treba da diskriminiše između ove dve raspodele. Primenjenim metodima se dobija da procedura obuke uvek gleda ciljnu reč i k uzoraka u izlaznom sloju, gde je k broj uzoraka šuma koji se koristi, umesto celog izlaznog sloja, čime se on efektivno značajno smanjuje i prestaje da zavisi od

veličine rečnika, i na taj način se upotreba mreže ubrzava. Reskorovanje se radi direktno na grafu, uz odgovarajuće razvijene pomoćne alate koji umeju da rade sa Kaldi dekoderom.

Najbolji rezultat prepoznavanja dobijen je sa oko 78000 reči u skraćenoj listi, sa svega 200 neurona u skrivenom sloju mreže, uz aproksimaciju 4-gramima pri reskorovanju, i on je iznosio 7,53% (dok je CER bio 2,03%), odnosno još malo lošije od Mikolov i Faster-RNNLM pristupa. Perpleksivnost obučenog RNNLM modela iznosila je 147,2 na izdvojenom validacionom skupu (Popović i drugi, 2018).

TensorFlow pristupi

Korišćeni RNNLM pristupi bazirani na TensorFlow sistemu koriste LSTM blokove da bi se sprečili problemi nestajućeg i eksplodirajućeg gradijenta tokom standardne SGD obuke uz algoritam propagacije unazad (Abadi i drugi, 2016). U odnosu na klasični TensorFlow pristup sa običnom neuronskom mrežom sa jednim skrivenim slojem koja se obučava standardnim BP algoritmom, TensorFlow LSTM pristup je robustniji, sekvencijalni pristup („sekvenca-po-sekvenca“) (Sutskever i drugi, 2014), koji koristi skrivenu vektorsku reprezentaciju ulazne sekvence (rečenice) za određivanje verovatnoća svake od reči u rečenici. Omogućeno je obučavanje i više od jednog skrivenog sloja neuronske mreže. Takođe, kao i kod CUED-RNNLM, nema podele reči u klase, i model se obučava na grafičkom procesoru. Reskorovanje se radi direktno na trigram grafu, uz razvijene pomoćne alate koji umeju da barataju i sa TensorFlow formatom modela, i sa Kaldi dekoderom (Liu i drugi, 2016).

Druga varijanta TensorFlow LSTM pristupa se razlikuje po postojanju algoritma odsecanja prilikom aproksimacije *n*-gramima tokom reskorovanja, koji dodatno smanjuje prostor pretrage u odnosu na normalnu aproksimaciju, i koristi izvesne heuristike za određivanje boljih istorija tokom ekspanzije grafa, čime se osim kraćeg vremena reskorovanja mogu postići i bolji ASR rezultati (Xu i drugi, 2018a). Osim toga, ova varijanta ima izmenjenu *softmax* aktivacionu funkciju – za novu „*softmax*“ funkciju se može reći da automatski normalizuje neuronsku mrežu tako što pazi da zbir izlaza mreže bude uvek blizu vrednosti 1, čime dodatno značajno olakšava proceduru reskorovanja.

Najbolji rezultati su dobijeni ipak za prvu varijantu pristupa, već sa 2 skrivena sloja sa po 200 neurona, za oko 57000 reči u skraćenoj listi (to su reči koje se javljaju makar 5 puta u korpusu za obuku). WER je za ovu konfiguraciju iznosio 7,25%, dok je CER bio 1,99%, što je najbolji rezultat od svih modela jezika do sada (oko 20% relativnog unapređenja u odnosu na trigrame). Perpleksivnost je iznosila 124,9 na izdvojenom validacionom skupu (Popović i drugi, 2018). To nije najbolji rezultat što se ovog parametra tiče, ali pokazano je da manja perpleksivnost ne mora da istovremeno označava i bolji model jezika u praktičnoj primeni (po stopi greške), i obratno (Klakow & Peters, 2002). Uz to, broj reči u skraćenoj listi veoma utiče na perpleksivnost, tako da treba naglasiti da taj broj nije bio isti u svim navedenim eksperimentima (tačnije, u najboljim varijantama svakog pristupa), pa rezultate po pitanju perpleksivnosti treba uzeti sa rezervom.

U tabeli 4 dat je pregled svih rezultata predstavljenih u ovoj sekciji, po pitanju WER, CER, kao i perpleksivnosti na test skupu (kod trigrama) ili automatski izdvojenom validacionom skupu (kod RNNLM obuka).

Tabela 4. Pregled dosadašnjih rezultata srpskih modela jezika

Pristup	Broj slojeva	Broj neurona po sloju	Broj reči u skraćenoj listi	Broj klasa	WER [%]	CER [%]	PPL
trigram	-	-	-	-	9,06	2,37	768,8
Mikolov	1	300	40k	400	7,41	2,00	116,1
Faster	1	300	78k	500	7,46	2,06	130,0
CUED	1	200	78k	-	7,53	2,03	147,2
TF var.1	2	200	57k	-	7,25	1,99	124,9
TF var.2	2	200	57k	-	7,44	2,01	135,3

Interesantno bi bilo napomenuti i trajanja pojedinih obuka. Mikolov i CUED obuke su bile najduže (trajanje se merilo u danima). Faster-RNNLM obuka je pružila veliko ubrzanje u odnosu na njih. Međutim, sekvencijalne TensorFlow (TF) LSTM obuke su po brzini bez premca (trajanje je u satima). Kod druge varijante TensorFlow obuke (TF var.2), sa odsecanjem pri reskorovanju, sam proces reskorovanja je dodatno ubrzan, ali je to uticalo na blago pogoršanje stope greške.

POGLAVLJE V:

MORFOLOŠKI MODELI SRPSKOG JEZIKA

Kao što je opisano u poglavlju 4, svi testirani RNNLM pristupi su ostvarili značajna poboljšanja u odnosu na referentni trigram model jezika, i prema WER parametru, i prema CER vrednosti, i prema perpleksivnosti. Međutim, ono što je primećeno čak i u najboljem RNNLM modelu (TensorFlow pristup na bazi LSTM), jeste da je velika većina tipova problema iz trigram sistema preneta, samo ih je generalno bilo dosta manje. Jedna od čestih vrsti grešaka bila je ona kada je lema, to jest osnovni oblik reči, pogođena, ali je završetak reči pogrešan, što rezultuje vrlo niskom CER vrednošću u odnosu na WER. Uzrok ove vrste problema je najverovatnije visok stepen inflektivnosti srpskog jezika – ista reč se može naći u različitim padežima, gramatičkim brojevima, gramatičkim rodovima, licima, i tako dalje. Koje sve izvedene oblike može da poprimi jedan osnovni oblik reči zavisi prvenstveno od vrste reči.

Morfološke informacije nije bilo moguće direktno ubaciti u model jezika ni u jednom od prethodno opisanih metoda. Zbog toga su obuke prebačene na novi alat i prateće procedure – Kaldi-RNNLM (Xu i drugi, 2018b). Detaljniji opis ovog alata i njegovo prilagođavanje za upotrebu u srpskim ASR sistemima dati su u sekciji „Kaldi-RNNLM alat“. Nakon toga, opisane su postojeće i korišćene morfološke kategorije u srpskom jeziku, a zatim i novi, predloženi načini da se dodatne morfološke informacije inkorporiraju u ovakav model jezika.

KALDI-RNNLM ALAT

Kaldi-RNNLM je skup programa i procedura razvijenih u okviru samog Kaldi-ja, koji, shodno tome, ne zahtevaju instalaciju i korišćenje spoljnih alata i pratećih metoda, kao što je to bio slučaj pre svega za TensorFlow modele jezika, ali donekle i za ostale dosadašnje RNNLM. Takođe je potpuno konfigurabilan, tako da se može relativno lako modifikovati po potrebi za svaki jezik i razne pristupe,

uključujući potpunu zamenu delova koda za izdvajanje skupa obeležja za reči i njihovo pripisivanje individualnim rečima iz željenog rečnika.

Kaldi-RNNLM dozvoljava korišćenje obeležja koja su ispod nivoa reči (eng. *subword features*). U poslednje vreme, pokazalo se da modeli jezika koji uzimaju u obzir obeležja niža od nivoa reči – na primer, obeležja koja se zasnivaju na slovima (karakterima), slogovima ili afiksima, nadmašuju u performansama standardne modele koji su na nivou reči, i po stopi greške i po perpleksivnosti (Mikolov i drugi, 2012). Kombinovanje modela jezika na nivou karaktera sa modelom na nivou reči uz odgovarajuće pondere je takođe dovelo do poboljšanja perpleksivnosti na raznim test korpusima (Kwon & Park, 2003; Sak i drugi, 2010). Konkretno, za predstavljanje informacija ispod nivoa reči, i istovremenu borbu sa rečima van rečnika (eng. *Out-Of-Vocabulary*, OOV) na kom je sistem obučen, Kaldi-RNNLM koristi obeležja koja opisuju postojanje i brojnost pojedinih slovnih *n*-grama u okviru reči, odnosno nizova od određenog broja slova. Tipično se koriste slovni bigrami i trigrami, a ponekad i kvadrigrami. Posebno se obraća pažnja na ivične slovne *n*-grame – početne i završne. Na primer, za reč *sam*, dobijaju se sledeći slovni bigrami i trigrami: *^s*, *^sa* (to su početni *n*-grami; simbol ^ označava početak reči), *sa*, *sam*, *am*, *am$* i *m$* (ovo su završni *n*-grami; simbol *$* označava kraj reči). Kao obeležja se koriste samo oni slovni *n*-grami koji se javljaju dovoljno često na nivou celog korpusa, što se definiše minimalnom zahtevanom učestanošću *n*-grama (Xu i drugi, 2018b).

Osim slovnih *n*-grama, Kaldi-RNNLM omogućava korišćenje i nekoliko augmentovanih obeležja, kao što su unigram log-verovatnoća reči u korpusu za obuku i dužina reči (u slovima), koja se koriste za dobijanje boljih rezultata u drugačijim domenima u odnosu na one koji su sretani tokom obuke. Dodatno, svaka od V najčešćih reči u korpusu za obuku dobija odgovarajuću uobičajenu *one-hot* reprezentaciju (1-od-V) u okviru svog vektora obeležja, što znači da sadrži podvektor dimenzije V u kom je samo jedan element, onaj koji odgovara konkretnoj reči, različit od nule (dok reči koje nisu u V najčešćih imaju sve nule u ovom podvektoru). Sve u svemu, u standardnom Kaldi-RNNLM modelu, osim *one-hot* reprezentacije za najčešće reči, svaka reč ima dodatnu reprezentaciju u vidu vektora broja pojavljivanja određenih slovnih *n*-grama i pomenuta dva augmentovana obeležja. Rezultat preslikavanja vektora odgovarajuće reči na ulaznom sloju RNN, takozvani *embedding* vektor, dobija se sumiranjem termina koji odgovaraju svakom od definisanih obeležja

(to je prvi zbir u izrazu (2.14)). Mnogi savremeni modeli jezika za velike rečnike koriste ovakve *embedding* vektore čiji elementi su realni brojevi. Na ovaj način, umesto uobičajene 1-od-*N* reprezentacije (gde je *N* broj reči u rečniku), svaka reč se povezuje sa tačkom u prostoru vektora, gde je dimenzionalnost ovih vektora mnogo manja od *N*. Ako posmatramo ceo rečnik, vektorske reprezentacije svih reči mogu se predstaviti takozvanom *embedding* matricom koja je na nivou nižem od reči, a čija svaka kolona odgovara određenoj reči iz rečnika, odnosno njenoj nižedimenzionalnoj vektorskoj reprezentaciji. *Embedding* matrica se obučava u paraleli sa parametrima samog modela jezika, to jest težinama skrivenih slojeva neuronske mreže, koja istovremeno uči raspodele verovatnoća nizova reči izražene na osnovu ovih reprezentacija (Bengio i drugi, 2003).

Uz pomenute slovne *n*-grame, u cilju prevazilaženja problema retkosti podataka za veće rečnike, Kaldi-RNNLM koristi i metod deljenja ulazne i izlazne *embedding* matrice neuronske mreže (eng. *input/output embedding sharing*), pri čemu se tokom obuke forsira da ove dve matrice budu međusobno jednake (Press & Wolf, 2017). Ovaj metod koristi činjenicu da, iako se u literaturi gotovo uvek samo ulazna *embedding* matrica RNN (matrica U u (2.14)) koristi kao reprezentacija reči iz rečnika, tu ulogu može da ima i izlazna *embedding* matrica (matrica V u (2.15)) – u obe matrice redovi koji pripadaju rečima koje su slične (po značenju, funkciji, i tako dalje) treba takođe da budu slični. Na primer, što se tiče ulaznog preslikavanja, od mreže se očekuje da slično reaguje na sinonime, dok se kod izlaznog preslikavanja očekuje da dobijeni skorovi za reči koje su međusobno zamenjive u rečenici budu slični. Pokazuje se da forsiranjem jednakosti *embedding* matrica ($U = V$) rezultujuća deljena *embedding* matrica, uz smanjenje broja parametara mreže, dovodi i do poboljšanja perpleksivnosti modela jezika. U kombinaciji sa slovnim *n*-gramima, primena ovog metoda dovodi do toga da, čak i u slučaju veoma velikog rečnika, nema potrebe da se koristi skraćena lista reči za direktno modelovanje verovatnoća samo dela rečnika, što je bio slučaj kod dosadašnjih metoda RNNLM obuka za srpski jezik (Xu i drugi, 2018b).

Primer potpunog vektora obeležja za jednu reč iz rečnika dat je u tabeli 5. U primeru iz tabele 5 uzeto je $V = 9970$ najčešćih reči, što su sve reči koje su se pojavile makar 252 puta u korpusu za obuku, a pri tome je korišćen novi, prošireni korpus

opisan u tabeli 2; sa rečima koje su se pojavile po 251 put, vrednost *V* bi prešla 10000, što je bilo zadato ograničenje.

Tabela 5. Primer vektora obeležja za reč *sam* u okviru Kaldi-RNNLM obuke

Indeks obeležja	Vrsta obeležja	Obeležje	Vrednost	Napomena
0	konstanta	konstanta	0,01	-
6	unigram	unigram log-verovatnoća	0,00733	sa ofsetom, skalirano
7	dužina	dužina reči	0,0051	skalirano
33	reč	česta reč *sam*	0,2	na bazi unigram log-verovatnoće, skalirano
10038	završni *n*-gram	3-gram *–am$*	0,12	skalirano
10480	završni *n*-gram	2-gram *–m$*	0,047	skalirano
10756	završni *n*-gram	4-gram *–sam$*	0,16	skalirano
11890	početni *n*-gram	2-gram ^*s*-	0,03	skalirano
11891	početni *n*-gram	3-gram ^*sa*-	0,069	skalirano
11898	početni *n*-gram	4-gram ^*sam*-	0,14	skalirano
12646	*n*-gram	2-gram -*am*-	0,065	skalirano
18321	*n*-gram	2-gram -*sa*-	0,057	skalirano
18339	*n*-gram	3-gram -*sam*-	0,11	skalirano

Navedena su samo obeležja različita od nule. Sva koja nisu navedena, jednaka su 0. U konkretnom slučaju, za slovne n-grame korišćeni su bigrami, trigrami i kvadrigrami. Ukupno je bilo 20476 obeležja (9970 za najčešće reči, 8178 slovnih n-grama uz 1229 početnih i 1091 završnih, 2 augmentovana, konstanta i 5 specijalnih), što je svakako puno manje od broja reči u rečniku u slučaju svakog sistema za velike rečnike.

Većina obeležja reči se dodatno skalira normalizacionim faktorom tako da se ograniči kvadratna sredina (RMS) za dato obeležje na celom korpusu na zadatu maksimalnu vrednost, i to se radi za sva dovoljno česta obeležja, za koje je ograničavanje RMS neophodno. Takođe, postoji još nekoliko specijalnih obeležja koja do sada nisu navedena – konstanta kao prvo obeležje, koje imaju sve reči, i koja je jednaka definisanoj maksimalnoj RMS (max_rms, tipično 0,01) i postoji iz čisto matematičkih razloga, kao i jedan mali *one-hot* vektor za takozvane „specijalne reči", u koju grupu spadaju definisani simboli za početak rečenice (eng. *Beginning Of Sentence*, BOS) i kraj rečenice (eng. *End Of Sentence*, EOS), simbol za nepoznate reči (reči iz korpusa kojih nema u definisanom rečniku, UNK), simbol koji označava završetak rečenice ili rečenične celine, koji se uglavnom koristi u slučaju da jedna linija teksta ne odgovara tačno jednoj rečenici (eng. *break*, BRK) i simbol za reč eksplicitne tišine (!SIL). Reči iz ove grupe imaju samo konstantu i ovu malu *one-hot* reprezentaciju kao svoja obeležja, dok se ostala obeležja ne definišu (ostaju jednaka nuli). Simbol BRK se ne koristi u korpusima koji su unapred segmentirani po rečenicama. Sama procedura obuke i validacije prilagođena je radu na grafičkom procesoru računara, što značajno ubrzava celu proceduru.

Pojedina obeležja se izračunavaju na način opisan u nastavku. Pre svega, formula po kojoj se ograničava RMS za željena obeležja, odnosno, pomoću koje se izračunavaju faktori skaliranja obeležja, data je sa

$$f_{RMS}(rms) = \begin{cases} \dfrac{max\_rms}{rms}, & ako\ rms > max\_rms \\ 1, & ako\ rms \leq max\_rms \end{cases}. \tag{5.1}$$

Koristeći formulu (5.1), obeležja svake reči *w* iz rečnika data su sledećim izrazima:

$$F_{const}(w) = max\_rms, \tag{5.2}$$

gde je F_{const} konstantno obeležje;

$$F_{spec_w}(w) = f_{RMS}\left(\sqrt{P_w}\right), \tag{5.3}$$

gde je F_{spec_w} odgovarajući element *one-hot* vektora za specijalne reči (BOS, EOS, UNK, BRK i !SIL), i definiše se samo ako *w* spada u tu grupu reči, dok je P_w relativna učestanost reči *w* u celom korpusu za obuku;

$$F_{unigram}(w) = offset + \log P_w \cdot scale = \\ -mean \cdot scale + \log P_w \cdot scale, \tag{5.4}$$

gde je $F_{unigram}$ unigram obeležje date reči, i kod kojeg se određuju ofset (odstupanje, eng. *offset*) i faktor skaliranja (eng. *scale*) tako da je srednja vrednost ovog obeležja na nivou celog rečnika 0, a RMS jednaka *max_rms*; vrednost *mean* predstavlja srednju vrednost unigram log-verovatnoća reči u rečniku, a *scale* je količnik *max_rms* i standardne devijacije unigram log-verovatnoća, ograničen sa gornje strane na 1 (Xu i drugi, 2018b), i ove vrednosti su date sa:

$$mean = \frac{\sum_i P_{w_i} \cdot \log P_{w_i}}{\sum_i P_{w_i}} \tag{5.5}$$

$$scale = \min\left(\frac{max\_rms}{\sqrt{\sum_i P_{w_i} \cdot \left(\frac{\sum_i P_{w_i} \cdot \left(\log P_{w_i}\right)^2}{\sum_i P_{w_i}} - mean^2\right)}}, 1\right); \tag{5.6}$$

$$F_{length}(w) = length(w) \cdot f_{RMS}\left(\sqrt{\sum\nolimits_i P_{w_i} \cdot \left(length(w_i)\right)^2}\right), \tag{5.7}$$

gde je F_{length} obeležje dužine reči, a funkcija $length(w)$ vraća dužinu reči *w* u karakterima;

$$F_{word_w}(w) = f_{RMS}\left(\sqrt{P_w}\right), \tag{5.8}$$

gde je F_{word_w} odgovarajući element *one-hot* vektora za *V* najčešćih reči u korpusu i definiše se samo ako *w* spada u tu grupu reči;

$$F_{ngram_x}(w) = count(x\ in\ w) \cdot f_{RMS}\left(\sqrt{\sum\nolimits_i P_{w_i} \cdot \left(count(x\ in\ w_i)\right)^2}\right), \tag{5.9}$$

gde je F_{ngram_x} obeležje koje se pripisuje slovnom *n*-gramu *x* i definiše se za sve slovne *n*-grame koji postoje u reči *w*, a koji su u grupi svih *n*-grama za koje obeležje postoji (odnosno, koji su dovoljno česti u korpusu za obuku), dok funkcija

$count(x\ in\ w)$ vraća broj pojavljivanja slovnog *n*-grama *x* u reči *w* (pri čemu se posebno posmatraju početni, završni i ostali slovni *n*-grami).

NOVA FUNKCIJA CILJA ZA ESTIMACIJU NENORMALIZOVANIH VEROVATNOĆA

U uobičajenoj CE obuci, ako sa *z* obeležimo sloj neuronske mreže neposredno pre završne *softmax* operacije, na osnovu izraza (2.17) i primenom operacije logaritmovanja, funkcija cilja za dati ulazni podatak (reč) dobija oblik

$$f_{obj}(z) = z_j - \log \sum_i \mathrm{e}^{z_i} \ , \tag{5.10}$$

gde je *j* indeks ispravne reči, koju je trebalo predvideti. Ako sada uzmemo u obzir da uvek važi $\log x \leq x - 1$, može se definisati izmenjena funkcija cilja

$$f_{obj}(z) = z_j + 1 - \sum_i \mathrm{e}^{z_i}. \tag{5.11}$$

U odnosu na funkciju datu sa (5.10), funkcija u (5.11) je uvek manja od nje ili jednaka njoj, sa jednakošću samo za slučaj $\sum_i e^{z_i} = 1$. Drugim rečima, nova funkcija cilja je ograničenje sa donje strane uobičajene CE funkcije cilja, sa jednakošću u slučaju normalizovane raspodele verovatnoća. Zbog toga je maksimizacija ove funkcije analogna CE obuci uz penalizacioni termin koji čini da zbir izlaza mreže bude vrednost bliska 1, odnosno $\sum_i e^{z_i} \simeq 1$ (Xu i drugi, 2018b).

Tokom testiranja i praktične primene obučene mreže, umesto računanja izraza (5.10) ili (5.11), u Kaldi-RNNLM se prosto koristi z_j kao aproksimacija izračunate verovatnoće, pošto znamo da je matematičko očekivanje izraza $1 - \sum_i e^{z_i}$ u obučenoj neuronskoj mreži jednako 0. Ovakva aproksimacija omogućava značajno ubrzanje računanja u svim zadacima kod kojih je labela (to jest reč) za koju želimo da odredimo verovatnoću poznata, što uključuje obe varijante reskorovanja pri automatskom prepoznavanju govora (i preko grafa i preko *N-best* liste).

STABILNOST OBUKE

Potencijalni problem sa korišćenjem funkcije cilja date izrazom (5.11) jeste moguća nestabilnost obuke, pogotovo u početnim iteracijama. Nestabilnosti su

moguće zbog eksponencijalnih termina u izrazu funkcije cilja, koji mogu da dovedu do veoma visokih izračunatih vrednosti njenih izvoda. Zbog toga se u Kaldi-RNNLM procedurama obuke koriste dva metoda za obezbeđivanje konvergencije (Xu i drugi, 2018b).

Prvi metod je pažljiva inicijalizacija težina neuronske mreže. Umesto uobičajene inicijalizacije težina u izlaznoj *embedding* matrici tako da imaju srednju vrednost 0, postavlja se srednja vrednost od oko $-\log N$, gde je N broj reči u rečniku. Empirijski je utvrđeno da bi takav odabir težina trebalo da spreči potencijalne nestabilnosti (Xu i drugi, 2018a).

Drugi metod je računanje funkcije cilja preko transformisane vrednosti z umesto direktno preko z. Transformacija se vrši funkcijom

$$f(z) = \begin{cases} z, & ako\ z \leq 0 \\ \log(z+1), & ako\ z > 0 \end{cases}. \tag{5.12}$$

Ovo je ekvivalentno postojanju odgovarajućeg dodatnog sloja mreže neposredno pre dobijanja izlaznih verovatnoća, međutim, realizovano je u okviru implementacije samog računanja vrednosti funkcije cilja. Korišćenjem transformacije $f(z)$ bi izlazi mreže trebalo da budu sprečeni da postanu suviše veliki, čime bi i odgovarajući izvodi bili u razumnim granicama, a time i obuka stabilna. Kaldi-RNNLM se uglavnom bazira na drugom metodu, dok je prvi rezervisan samo za specifične obuke na bazi i Kaldi-ja i TensorFlow alata, kojima se ova disertacija ne bavi.

ALGORITMI UZORKOVANJA ZA RAČUNANJE FUNKCIJE CILJA

Za računanje vrednosti funkcije cilja date sa (5.11) tokom obuke i dalje je potrebno prolaženje kroz sve reči iz rečnika da bi se izračunao termin $\sum_i e^{z_i}$. Kaldi-RNNLM koristi metod koji se bazira na uzorkovanju da bi dobio nepristrasnu estimaciju ove sume. Treba navesti da u standardnim CE sistemima primena ovakvog uzorkovanja na osnovu važnosti nije dopuštena zbog nelinearne operacije logaritmovanja (uvek bi se unosila pristrasnost u estimaciju), dok je u funkciji cilja datoj sa (5.11) to izbegnuto (sama operacija sumiranja ne unosi pristrasnost).

Uzorkovanje se tokom obuke modela koristi na sledeći način. Pretpostavimo da u datom podskupu za obuku (eng. *minibatch*) imamo ukupno *k* reči, a želimo da koristimo uzorak veličine *m* reči (na primer, $m = 1024$). Tada, u svakom podskupu za obuku imaćemo *k* podataka (eng. *data-points*), što uključuje *k* ili manje različitih tačnih reči (skup *Y*). Prvo treba za svaku reč *w* iz rečnika generisati verovatnoću $p(w)$ uključivanja te reči u pomenuti uzorak veličine *m*. Za sve reči iz skupa *Y* mora važiti $p(w) = 1$, a inače uvek važi $p(w) \leq 1$. Zbir verovatnoća uključivanja na nivou celog rečnika mora biti jednak veličini uzorka *m* (ne dozvoljavaju se duplikati u uzorku). Određivanje tih verovatnoća opisano je u narednom pasusu. Nakon što su verovatnoće uključivanja određene, za svaki podskup reči za obuku se na slučaj odabere uzorak *S* od *m* reči. Tokom obuke se onda sumiranje $\sum_i e^{z_i}$ ograničava samo na reči iz skupa *S* (umesto da se radi nad svim rečima iz rečnika), s tim što se vrši i ponderisanje recipročnim vrednostima verovatnoća $p(w)$. Ovo je uobičajeni pristup korišćen u metodu uzorkovanja na osnovu važnosti – on obezbeđuje da raspodele verovatnoća koje daju funkcija cilja, odnosno njeni izvodi, imaju srednje vrednosti jednake onim koje bi imale da uzorkovanje nije rađeno (Xu i drugi, 2018b).

Raspodele verovatnoća iz kojih se na slučaj biraju uzorci u Kaldi-RNNLM za svaki podskup za obuku izračunavaju se uprosečavanjem *n*-gram raspodela za sve istorije (to jest kontekste) u tom podskupu, a te *n*-gram raspodele dolaze iz *back-off n*-gram modela jezika prethodno obučenog specifično za ovu namenu na postojećem korpusu, estimiranog i skraćenog tehnikama odsecanja tako da uzorkovanje iz njega bude efikasno (Xu i drugi, 2018b). Samo uzorkovanje se vrši takozvanim algoritmom sistematskog uzorkovanja sa nejednakim verovatnoćama (Deville & Tille, 1998). Dodatno ubrzanje osnovnog algoritma postiže se uzorkovanjem u dva koraka (Xu i drugi, 2018b) – pre svega, sve reči iz rečnika se podele u grupe u kojima je zbir verovatnoća uključivanja u uzorak manji od 1; u prvom koraku uzorkovanja se u ovakvoj postavci odabere *m* grupa reči od svih ponuđenih grupa, a u drugom koraku se iz svake grupe bira tačno jedna reč na osnovu pojedinačnih verovatnoća. Ovakvom procedurom se znatno smanjuje broj reči koje se uopšte razmatraju, što donosi veliko ubrzanje (u originalnom algoritmu se mora proći kroz ceo rečnik).

Morfološke kategorije reči u srpskom jeziku

Kao što je već rečeno, česte greške prepoznavača govora za srpski jezik, pogotovo na većim rečnicima, jesu zamene određene morfološke kategorije reči, dok je osnovni oblik prepoznate reči ispravan. U ranijim radovima su pokazani upravo takvi rezultati prepoznavanja, a kao moguće rešenje problema autori su označili korišćenje morfoloških klasa ili kategorija u okviru obuke modela jezika (Pakoci i drugi, 2018; Popović i drugi, 2018).

Alat Kaldi-RNNLM omogućava ubacivanje i novih obeležja u model jezika, osim onih koji već postoje u originalnim procedurama. To otvara mogućnost da se modifikovanjem odgovarajućih skriptova za izdvajanje i pridruživanje obeležja rečima iz rečnika sistema u model direktno inkorporiraju i morfološka obeležja, odnosno morfološke informacije. Da bi se to obavilo, potreban je alat koji za dati ulazni tekst na srpskom jeziku vraća potrebne informacije – osnovne oblike reči, odnosno sve moguće morfološke podatke za pojedine reči iz datog teksta. Taj alat je kratko opisan u podsekciji „Alat AnTagger i akcenatsko-morfološki rečnik“, a nakon toga opisana je procedura formiranja odgovarajućih ulaznih datoteka za rad u Kaldi-RNNLM okruženju.

Alat AnTagger i akcenatsko-morfološki rečnik

U istraživanjima vezanim prvenstveno za automatsku sintezu govora iz teksta na srpskom jeziku, razvijeno je nekoliko sistema za automatsku morfološku anotaciju datog ulaznog teksta, od kojih je poslednja verzija poslužila je kao baza za formiranje alata za morfološko, odnosno *Part-Of-Speech* (POS) tagovanje srpskih tekstova – AnTagger (Ostrogonac, 2015). Ovaj alat koristi ažuriranu verziju srpskog akcenatsko-morfološkog rečnika (Sečujski, 2009) da bi vratio sve željene morfološke informacije o svakoj reči iz teksta. Osim samog rečnika, koristi se i dodatnim ekspertskim znanjem, analizom konteksta i raznim heuristikama. AnTagger je ranije već korišćen u okviru pojedinih eksperimenata sa trigram modelima jezika da bi se odredila inflektivna stopa greške ASR sistema, što zahteva poznavanje ispravne leme u datim test rečenicama i prepoznatim hipotezama (Pakoci i drugi, 2017; Pakoci i drugi, 2018). Osim toga, alat AnTagger se upotrebljavao i u okviru postupka klasterizacije reči iz korpusa za obuku modela jezika u klase, u cilju formiranja klasnog i hibridnog

n-gram modela preko SRILM alata, gde je pokazano da takve tehnike potencijalno mogu biti od pomoći u rešavanju, ili makar smanjenju problema dimenzionalnosti u *n*-gram sistemima (Ostrogonac i drugi, 2019).

Glavne funkcionalnosti AnTagger-a su svakako lematizacija, odnosno zamena pojavnih oblika reči u ulaznom tekstu njihovim osnovnim oblikom, to jest lemom, i pretvaranje reči u morfološke klase, određene nizom odgovarajućih morfoloških kategorija. Ove procedure zahtevaju analizu konteksta, te se preporučuje zadržavanje znakova interpunkcije u ulaznom tekstu.

Dodatne funkcionalnosti alata AnTagger uključuju segmentaciju teksta na rečenice na osnovu ekspertskog znanja, analize konteksta i ručno implementiranih pravila, uklanjanje interpunkcije iz teksta (što se radi za potrebe obuke modela jezika za ASR sisteme), konverziju brojeva u odgovarajuće reči (opet, analizom konteksta) i formiranje hibridnog korpusa (kombinacije osnovnog korpusa, lematizovanog korpusa i morfoloških klasa).

KORIŠĆENE MORFOLOŠKE KATEGORIJE REČI

Inicijalno su u istraživanju datom u ovoj disertaciji iskorišćene samo neke od osnovnih morfoloških kategorija – vrsta reči, padež, gramatički broj i gramatički rod. Intuitivno je određeno da su te 4 kategorije najopsežnije i da nose najviše korisnih morfoloških informacija. U nekoliko ranijih testova sa klasnim i hibridnim *n*-gram modelima jezika je eksperimentisano i sa još nekoliko kategorija, kao što su na primer vrsta imenice – vlastita, zajednička, gradivna, zbirna ili apstraktna, stepen komparacije za prideve – pozitiv, komparativ ili superlativ, i slično (Ostrogonac i drugi, 2019). Naknadne obuke i testovi su obuhvatili sve morfološke kategorije koje alat AnTagger može da odredi, kako prethodno navedene, tako i mnoge druge. U tabeli 6 prikazane su osnovne morfološke kategorije, kao i moguće vrednosti za svaku od njih, dok su u tabelama od 7 do 12 prikazane dodatne korišćene kategorije po vrsti reči – za imenice, zamenice, prideve, brojeve, glagole i nepromenljive reči.

U odnosu na standardne vrste reči u srpskom jeziku, postoje dve dodatne – akronim i izolovano slovo. Ove dve dodatne vrste reči se označavaju samo u slučaju kada AnTagger vrati informaciju da za neku reč nije uspeo da odredi morfološke kategorije, što se dešava jedino ako ta reč ne postoji u akcenatsko-morfološkom

rečniku, a uz to nije uspeo ni metod analogije, ni metod rimovanja sa drugim rečima (što su neki od metoda koje AnTagger koristi pri radu) – tada, ako reč ima samo jedno slovo, vraća se vrsta reči „slovo“, a inače, ako se reč sastoji samo od konsonanata (isključujući ’R’, koje potencijalno može biti vokalno), vraća se vrsta reči „akronim“. Uvidom u izlaz AnTagger-a utvrđeno je da je u većini slučajeva kada je morfološka kategorija nepoznata u pitanju neki akronim koji nije bio unet u akcenatsko-morfološki rečnik. Ukoliko su pak uneti, akronimi se tipično smatraju vlastitom imenicom, jer se uglavnom na taj način i ponašaju. Što se tiče izolovanih slova, to su uglavnom lokacije gde je nešto eksplicitno napisano slovo po slovo (sa razmakom između slova), što takođe može biti situacija koja se može sresti u praksi (kada se, na primer, zahteva od korisnika da izgovori kako se nešto piše).

Tabela 6. Osnovne morfološke kategorije reči sa mogućim vrednostima

Kategorija	Spisak mogućih vrednosti
vrsta reči	imenica, zamenica, pridev, broj, glagol (promenljive), predlog, prilog, veznik, rečca, uzvik (nepromenljive), akronim, slovo (specijalne)
padež	nominativ, genitiv, dativ (= lokativ), akuzativ, vokativ, instrumental, genitiv ili akuzativ, dativ ili instrumental
gramatički broj	jednina, množina
gramatički rod	muški, ženski, srednji, muški ili srednji

Kombinacije padeža (genitiv ili akuzativ, dativ ili instrumental) ili gramatičkih rodova (muški ili srednji rod) se vrlo retko koriste, kada nikako nije moguće odrediti koji od dva padeža ili roda odgovara datoj reči.

Konačno, postoji i vrsta reči „nepoznato“, ako nijedan od prethodnih uslova nije ispunjen, i tada se za reč svakako ne označava nijedna morfološka kategorija. Što se tiče padeža, dativ i lokativ su spojeni u istu vrednost kategorije (vode se kao dativ), jer su ta dva oblika reči identična u srpskom jeziku. Naravno, nisu sve kategorije validne za svaku reč, na primer, nijedna nepromenljiva reč ne može imati padež, rod, broj ili lice (delimični izuzetak su predlozi koji su uvek praćeni rečju u određenom padežu, i njima se ipak pripisuje kategorija padeža).

Tabela 7. Dodatne morfološke kategorije imenica sa mogućim vrednostima

Kategorija	Spisak mogućih vrednosti
vlastita imenica	da, ne
vrsta imenice	ime, prezime, toponim ili etnička, organizacija (vlastite), zajednička, gradivna, zbirna, apstraktna

Osim ove dve kategorije, sve imenice imaju i sve četiri osnovne.

Tabela 8. Dodatne morfološke kategorije zamenica sa mogućim vrednostima

Kategorija	Spisak mogućih vrednosti
vrsta zamenice	lična, sa imeničkom promenom, sa pridevskom promenom
enklitika	da, ne
povratna zamenica	da, ne
lice	prvo, drugo, treće
specijalna pridevska zamenica	*čija*, *koja*, *kakva*, *kolika*

Povratna lična zamenica ima kategorije povratna zamenica, enklitika i padež, a ostale lične imaju enklitika, padež, broj, rod i lice. Zamenice sa imeničkom promenom imaju samo padež. Zamenice sa pridevskom promenom imaju padež, broj i rod, a navedene specijalne pridevske zamenice imaju i to dodatno obeležje. Obeležje vrsta reči se podrazumeva za sve.

Tabela 9. Dodatne morfološke kategorije prideva sa mogućim vrednostima

Kategorija	Spisak mogućih vrednosti
specijalni pridev	nepromenljivi pridevi, *nalik*
stepen komparacije	pozitiv, komparativ, superlativ

Sem stepena komparacije, ne-specijalni pridevi imaju i padež, broj i rod.

Tabela 10. Dodatne morfološke kategorije brojeva sa mogućim vrednostima

Kategorija	Spisak mogućih vrednosti
vrsta brojeva	osnovni, redni, zbirni, imenički, nepromenljivi
grupa brojeva	grupa 1 (*trima*, *četirima*), grupa 2 (*triju*, *četiriju*)
specijalni broj	*jedan*, *nijedan*, *dva*, *dvama*, *dvaju*, *dvema*, *dveju*, *tri*, *četiri*

Sve grupe i specijalni brojevi spadaju u osnovne. Osnovni brojevi mogu imati i padež, rod ili broj (ne svi). Redni, zbirni i imenički uvek imaju sve osnovne kategorije.

Tabela 11. Dodatne morfološke kategorije glagola sa mogućim vrednostima

Kategorija	Spisak mogućih vrednosti
enklitika	da, ne
pomoćni glagol	da, ne
specijalni pomoćni glagol	*jesam*, *hteti*, *biti*, *nisam*, *neću*
specijalni glagol	*nemoj*, *hajde*, *treba*
modalnost	modalni (ili fazni), nemodalni
tranzitivnost	prelazni, neprelazni
refleksivnost	povratni, nekad povratni, nepovratni
glagolski oblik	infinitiv, prezent, futur, aorist ili imperfekat, imperativ, radni glagolski pridev, glagolski prilog sadašnji, glagolski prilog prošli
lice	prvo, drugo, treće

Glagolski rod je razbijen u dve kategorije – tranzitivnost i refleksivnost. Aorist i imperfekat često dele isti oblik reči, pa su spojeni. Velika većina glagola ima definisane sve navedene šire kategorije (od modalnosti naniže). Enklitika se vezuje samo za pomoćne glagole (koji su svi navedeni i kao specijalni pomoćni glagoli).

Tabela 12. Dodatne morfološke kategorije nepromenljivih reči sa vrednostima

Kategorija	Spisak mogućih vrednosti
grupa priloga	grupa 1 (*vrlo*, *veoma*), grupa 2 (*kad(a)*), grupa 3 (*kud(a)*), grupa 4 (*otkad(a)*), grupa 5 (*dokad(a)*)
specijalni prilog	*gde*, *kako*, *zašto*, *odakle*, *dokle*
prilog uz šta	prilog uz imenicu, prilog uz pridev, prilog uz glagol
grupa veznika	grupa 1 (*da*), grupa 3 (*nek(a)*), grupa 4 (*nit(i)*), grupa 5 (*ili(ti)*), grupa 6 (*umesto*, *osim*, *sem*, *izuzev*), grupa 7 (*no*, *nego(li)*), grupa 8 (*jer(bo)*), grupa 9 (*iako*, *mada*, *premda*), grupa 10 (*gde*, *kad(a)*), grupa 11 (*ako*, *dok*, *čim*, *pošto*, *već*…)
specijalni veznik	*što*, *kao*, *i*, *ni*, *pa*, *te*, *bilo*, *em*, *ali*
grupa rečci	grupa 1 (*možda*, *valjda*, *uglavnom*, *zapravo*, *ustvari*, *baš*…), grupa 2 (*pogotovo*, *naročito*), grupa 3 (*bar(em)*, *čak*), grupa 4 (*dakle*, *štaviše*, *naime*, *bogami*, *recimo*, *nažalost*…), grupa 5 (*jedino*), grupa 6 (*naravno*, *dabome*, *jašta*…), grupa 7 (*naprotiv*)
specijalna rečca	*l(i)*, *god*, *makar*, *taman*, *takođe*, *inače*, *svejedno*, *odnosno*, *(i)pak*, *nipošto*, *kamoli*, *ma*, *uopšte*, *zar*, *nešto*, *okej*…

Osim navedenog, većina predloga ima kategoriju padeža, kao što je ranije objašnjeno. Pojedine vrste reči su grupisane prema značenju, funkciji ili načinu upotrebe. Kao što se može primetiti, uzvici, akronimi i slova nemaju dodatne definisane kategorije.

ALAT PRERNNLMPROC I PRIPREMA RNNLM OBUKE

Nakon obrade korpusa alatom AnTagger, na raspolaganju stoje tri ulazne datoteke – osnovni tekstualni korpus, lematizovani korpus, i korpus morfoloških klasa. Da bi se oni pretvorili u oblik pogodan za obuku u Kaldi-RNNLM okruženju, napravljen je alat PreRnnlmProc.

Zadatak ovog alata je prvenstveno formiranje korpusa sa morfološkim, to jest POS sufiksima. Da bi se to postiglo, određeni su jedinstveni troslovni sufiksi, od kojih svaki jednoznačno odgovara jednoj vrednosti neke od morfoloških kategorija. Na primer, sufiks *gla* odgovara vrsti reči glagol. Alat svakoj reči iz ulaznog, čistog

korpusa dodaje sve sufikse koji joj odgovaraju, i na kraju vraća modifikovani, POS korpus. Sufiksi su međusobno, a i od pripadajuće reči, odvojeni donjim crtama ('_'). Primeri nekih reči sa POS sufiksima dati su u tabeli 13. Treba napomenuti da PreRnnlmProc alat očekuje da je ulazni korpus već sređen, u smislu brisanja nepotrebne interpunkcije, konvertovanja brojeva u odgovarajuće reči i slično, kao i da lematizovani korpus i korpus sa morfološkim klasama odgovaraju osnovnom korpusu po broju rečenica i broju reči u svakoj.

Tabela 13. Primeri nekih čestih reči iz korpusa, pre i posle dodavanja POS sufiksa

Reč bez POS sufiksa	Reč sa POS sufiksima	Napomene
je	je_gla_enk_pom_gje_nmf_ npz_npv_prz_3lc_jed	**gla**gol, **enk**litika, **pom**oćni, specijalni **g**lagol (**je**sam), **nem**odalni, **nep**rela**z**ni, **nepov**ratni, **prez**ent, **3**. **lic**e, **jed**nina
i	i_vez_vii	**vez**nik, specijalni **v**eznik (**i**)
u	u_pre_dat	**pre**dlog, uz **dat**iv
da	da_vez_v01	grupa **v**eznika **1**
se	se_zam_lic_enk_pov_aku	**zam**enica, **lič**na, **pov**ratna, **aku**zativ
na	na_pre_dat	-
koji	koji_zam_zpr_zkj_nom_mur_jed	**z**amenica sa **pr**idevskom promenom, specijalna **z**amenica (**koj**a), **nom**inativ, **mu**ški **r**od
bi	bi_gla_enk_pom_gbi_mmf_ plz_npv_aor_3lc_jed	specijalni pomoćni **g**lagol (**bi**ti), **m**odalni, **p**rel**az**ni, **aor**ist
Srbije	Srbije_imn_zer_vla_top_gen_jed	**imen**ica, **že**nski **r**od, **vla**stita, **top**onim ili etnička, **gen**itiv

Sve vrednosti svake od morfoloških kategorija imaju jedinstveni sufiks.

Osim POS korpusa, alat PreRnnlmProc vraća i dve dodatne datoteke potrebne za RNNLM obuku. Jedna je datoteka koja sadrži mapiranje sa reči na leme, u kojoj je definisano koja lema (ili potencijalno više njih, što je doduše vrlo retko) odgovara

svakoj od reči u POS korpusu. Ova datoteka će postati značajna pri određivanju dodatnih morfoloških obeležja u Kaldi-RNNLM. Poslednja datoteka koju PreRnnlmProc vraća je tabela brojnosti svake od reči u POS korpusu (ukupan broj instanci svake različite reči sa sufiksima, uz sortiranje prema broju pojava). Ova datoteka je značajna za određivanje željenog rečnika nad kojim se obučava model jezika, jer se rečnik tipično bira tako što se uzme određeni broj najčešćih reči iz korpusa, uz dodatak nekih reči koje moraju da se nađu u rečniku, što na primer mogu biti tipične reči za neki domen upotrebe u svim mogućim oblicima.

UBACIVANJE MORFOLOŠKIH INFORMACIJA U RNNLM

Hipoteza koja je ispitana u ovoj disertaciji jeste da bi korišćenje morfoloških informacija tokom obuke modela jezika moglo da poboljša kvalitet tog modela, odnosno rezultate na testovima prepoznavanja reči. Prva pretpostavka je da bi već samo dodavanje POS sufiksa u reči za obuku RNNLM moglo da donese neko poboljšanje rezultata tako što će napraviti distinkciju među do sada identičnim rečima koje su u stvari određeni oblici različitih lema koji se igrom slučaja međusobno poklapaju, ili čak i različiti oblici iste leme, sa drugačijom kombinacijom morfoloških kategorija, koji se, opet, slučajno poklapaju, iako imaju drugačiju gramatičku ili sintaksičku funkciju. Ovakav pristup bi, preko unošenja dodatnih informacija na ulazu, trebalo da pomogne neuronskoj mreži da malo preciznije modeluje pojedina pravila u jeziku, ili bolje nauči odnose između pojedinih reči (Pakoci i drugi, 2019).

Na ovaj način očigledno se modifikuje i rečnik, pošto su sve reči u korpusu promenjene (sada imaju morfološke sufikse). Međutim, novi rečnik se može formirati na identičan način kao stari, ovaj put brojanjem reči u korpusu sa POS sufiksima, uz dodavanje obaveznih, željenih domenskih reči, takođe nakon dodavanja sufiksa. Radi fer poređenja sa rezultatima obuke na originalnom korpusu, treba ciljati što približniji broj reči u oba rečnika.

Osim samog dodavanja POS sufiksa na reči u korpusu za obuku i u rečniku, morfološke informacije se mogu i direktno modelovati preko neuronske mreže. U ovoj disertaciji predloženo je nekoliko načina da se to uradi, pri čemu se svi načini zasnivaju na određivanju i ubacivanju novih obeležja u vektor obeležja reči. Sva tri predložena metoda data su u nastavku.

ONE-HOT VEKTORI ZA MORFOLOŠKE KATEGORIJE

Prvi pristup je sličan proceduri formiranja *one-hot* vektora za 1-od-V reprezentaciju najčešćih V reči iz datog rečnika u postojećem korpusu za obuku. Razlika u odnosu na tu proceduru jeste u tome što se umesto reči u rečniku posmatraju različite vrednosti definisanih morfoloških kategorija. Pre svega, za svaku kategoriju, kojih ima nešto više od 30 (tabele 6-12), analizira se ceo POS korpus i određuju se verovatnoće pojavljivanja svake od mogućih vrednosti date kategorije na nivou kompletnog teksta. Zatim, tokom procedure određivanja skupa obeležja reči za RNNLM, za svaku morfološku kategoriju se formira po jedan dodatni *one-hot* vektor. Za svaku reč, u svakom od ovih dodatnih podvektora, samo onaj element koji odgovara vrednosti date POS kategorije koja je vezana za datu reč biće različit od nule – imamo 1-od-N_c reprezentaciju kategorije, gde je N_c broj mogućih vrednosti za datu kategoriju. Alternativno, ako POS kategorija nije primenljiva na datu reč, odgovarajući podvektor će sadržati sve nule.

Analogno izračunavanju vrednosti ne-nula elementa u *one-hot* vektoru najčešćih reči, odgovarajući element podvektora za datu kategoriju c u konkretnoj reči w se dobija preko

$$F_{POS_{c_j}}(w) = f_{RMS}\left(\sqrt{P_{c_j}}\right), \tag{5.13}$$

gde je j indeks vrednosti morfološke kategorije c koja odgovara datoj reči w, P_{c_j} relativna učestanost pojavljivanja te vrednosti kategorije c u nekoj reči na nivou celog POS korpusa, a $F_{POS_{c_j}}$ dobijena vrednost odgovarajućeg elementa *one-hot* vektora za kategoriju c. Faktor skaliranja je i ovde, kao i kod ostalih obeležja reči u Kaldi-RNNLM okruženju, odabran tako da se ograniči RMS obeležja u datom korpusu.

Ideja dodavanja neke *one-hot* reprezentacije na ulazu neuronske mreže se često koristi i u drugim primenama mreža, ako se mreži želi proslediti informacija da ulazni podatak pripada nekoj klasi ili kategoriji. Pretpostavlja se da će ova dodatna informacija u obuci modela jezika doprineti boljem uočavanju i učenju pravila formiranja rečenica u srpskom jeziku od strane neuronske mreže.

SLOVNI *N*-GRAMI NA MORFOLOŠKIM SUFIKSIMA

Još jedan način da se morfološke informacije ubace direktno u neuronsku mrežu jeste da se posmatraju morfološki sufiksi koji su dodati rečima. U pitanju su, dakle, troslovne, jedinstvene oznake svih mogućih vrednosti svake od POS kategorija, koje istovremeno možemo posmatrati i kao slovne trigrame. Formiranjem vektora mogućih POS slovnih trigrama, i zatim njihovim detektovanjem u svakoj datoj reči iz rečnika i dodelom odgovarajuće brojne vrednosti tim elementima vektora obeležja, možemo dobiti još jednu reprezentaciju morfoloških podataka.

U eksperimentima će biti pokazano koji od dva opisana pristupa je uspešniji. Pretpostavka je da i POS slovni *n*-grami mogu doprineti učenju mreže na sličan način kao i *one-hot* vektori POS kategorija. Međutim, pošto se ovde vrednosti dodatih obeležja određuju na nivou celog korpusa, a ne posebno na nivou svake kategorije, moguće je i da će zajedno dati još bolje rezultate nego pojedinačno.

Kao i za standardne slovne *n*-grame, vrednosti obeležja se skaliraju tako da srednja kvadratna vrednost na nivou korpusa bude ograničena. Nema potrebe za prebrojavanjem svakog od POS trigrama u okviru reči – pošto su jedinstveni, svaka reč ima maksimalno po jedan od svakog. Stoga, termin $count(x\ in\ w_i)$ u referentnom izrazu (5.9) je uvek jednak ili 0 ili 1. Vrednost ovih obeležja za reč *w* data je preko

$$F_{pos\_ngram_x}(w) = f_{RMS}\left(\sqrt{\sum_{i,\, x \in w_i} P_{w_i}}\right), \tag{5.14}$$

gde se sumiranje vrši za sve reči w_i koje sadrže POS trigram *x* (odnosno, $x \in w_i$), a $F_{pos\_ngram_x}$ je obeležje koje se pripisuje POS trigramu *x* i definiše se za sve POS trigrame koji postoje u datoj reči *w*.

ONE-HOT VEKTOR ZA NAJČEŠĆE LEME

Konačno, slično kao za *one-hot* vektor najčešćih reči, u vektor obeležja se može dodati i *one-hot* vektor najčešćih lema. Broj najčešćih lema *L* u ovoj 1-od-*L* reprezentaciji može, ali ne mora biti jednak broju najčešćih reči *V*. Ideja iza ovog pristupa jeste da se odgovarajuća reprezentacija dodeli svim pojavnim oblicima čestih lema, a ne samo pojedinim (najčešćim). Takođe, neuronska mreža bi tokom obuke na

ovaj način mogla da nauči kako da se ponaša ako se u kontekstu nađe bilo koji oblik date leme, a ne samo jedan.

Ubacivanjem dodatnog *one-hot* vektora lema među obeležja ukupan broj obeležja se značajno povećava, međutim, i dalje je manji nego veličina bilo kakvog rečnika koji može da se smatra velikim. I pored toga, dodatne informacije koje vektor lema može da donese mogu biti veoma značajne za obuku mreže. Pretpostavka je da on može bitno doprineti modelovanju konteksta, a uz to dati i reprezentaciju rađim pojavnim oblicima pojedinih reči. Odgovarajuće obeležje za datu reč *w* se dobija sa

$$F_{lemma_{l_w}}(w) = f_{RMS}\left(\sqrt{P_{l_w}}\right), \tag{5.15}$$

gde je $F_{lemma_{l_w}}$ odgovarajući element *one-hot* vektora za *L* najčešćih lema u korpusu i definiše se samo ako datoj reči *w* odgovara lema l_w koja spada u tu grupu lema, a P_{l_w} predstavlja verovatnoću pojave odgovarajuće leme na nivou lematizovanog korpusa. U vrlo retkim slučajevima, više lema može odgovarati istoj reči (sa istom kombinacijom POS sufiksa), i tada ista reč može imati više od jednog unosa u ovom podvektoru. Ovo se dešava kada na različitim mestima u korpusu, za isti pojavni oblik reči i iste pridružene morfološke klase, alat AnTagger vrati različite leme. Jedan primer ovakvog slučaja je glagol *dodaju*, nemodalni, prelazni, nekad povratni, u prezentu, treće lice množine; osnovni oblik ove reči može istovremeno biti i *dodati* i *dodavati*. Alternativno bi mogao da se napravi poseban unos u rečniku za obe različite leme, međutim, nije se radilo na tome, pre svega zbog retkosti ovakve situacije (dešava se za između 0,04% i 0,05% različitih reči u korpusu, i to ne za one najčešće). Dobijanje leme, ili spiska lema za datu reč, radi se na osnovu mapiranja reči u leme koje je ranije dobijeno primenom alata PreRnnlmProc.

POGLAVLJE VI:

EKSPERIMENTALNI REZULTATI I DISKUSIJA

U svim eksperimentima sprovedenim u okviru ove disertacije (koji su opisani u poglavlju 6), akustički model je bio fiksiran. Kao audio baza podataka za njegovu obuku, korišćene su u celosti postojeća srpska i hrvatska baza (detaljno opisane u poglavlju 3), uz dodatno veštačko proširivanje metodom dodavanja šuma, tačnije dodavanjem zašumljene baze sa SNR vrednošću od 17 dB. Nakon svega toga, baza podataka za obuku je dodatno proširena dodavanjem baza sa usporenim i ubrzanim govorom (eng. *Speed Perturbation*, SP) – ovaj metod uključuje korišćenje programa za promenu tempa u ulaznom govornom signalu, koji pri tome čuva spektralne karakteristike signala (odnosno, datog govornika) (Ko i drugi, 2017). U konkretnom slučaju, koristili su se koeficijenti promene tempa od 0,85 i 1,15, odnosno, relativno usporavanje i ubrzavanje govora od 15% u odnosu na osnovnu govornu bazu podataka. Sve u svemu, uz veštačko proširivanje na oba načina, ukupno dostupan materijal za obuku je činilo preko 5400 sati audio snimaka, uključujući i govor i tišine (dok originalni materijal sadrži tek nešto više od 900 sati).

Što se tiče tekstualnog korpusa, uvek je korišćen kompletan materijal dat u tabeli 2, i za obuku trigram modela jezika, i za obuku referentnih RNNLM, kao i jezičkih modela sa morfološkim obeležjima.

OBUKA AKUSTIČKOG MODELA

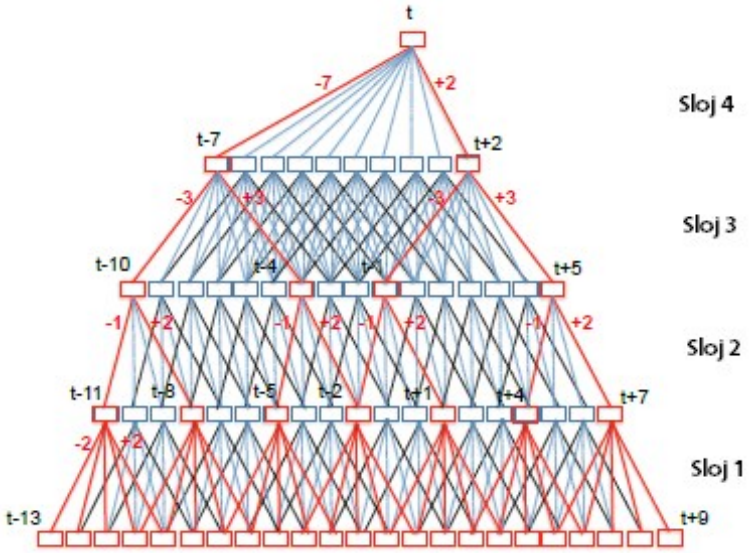
Kao akustički model poslužila je neuronska mreža sa vremenskim kašnjenjem (eng. *Time Delay Neural Network*, TDNN) (Waibel, 1989) sa poduzorkovanjem (eng. *subsampling*) (Peddinti i drugi, 2015). Ova mreža je obučavana takozvanim *chain* sekvencijalnim metodom obuke, koji koristi kriterijum unakrsne entropije (Pakoci i drugi, 2018).

TDNN je višeslojna mreža koja u odnosu na uobičajene DNN sa propagacijom unapred klasifikaciju ulaznih podataka vrše ne posmatrajući eksplicitno datu segmentaciju (u *chain* obuci se specificira vremenski prozor u okviru kog se ulazne

labele, to jest fonemi, mogu naći). To ih čini manje zavisnim od prethodno obučenih, tipično HMM akustičkih modela koji vrše segmentaciju podataka za obuku, to jest određivanje početaka i završetaka svih fonema, pre same obuke neuronske mreže (eng. *shift-invariance*). Takođe, TDNN modeluje vremenski kontekst u svakom sloju mreže, što znači da svaki neuron u svakom sloju na svom ulazu dobija ne samo aktivacije neurona, odnosno obeležja iz sloja ispod, već čitavu šemu aktivacija kroz vreme prethodnih neurona. U predloženoj konfiguraciji, inicijalni slojevi neuronske mreže se obučavaju na užem, a dublji slojevi na širem vremenskom kontekstu aktivacija neurona, odnosno obeležja, čime dublji slojevi mogu bolje da nauče šire vremenske povezanosti. Svaki sloj mreže, dakle, operiše sa različitom vremenskom rezolucijom. Tokom obuke, pojedini slojevi mreže se ažuriraju uz pomoć gradijenta koji se akumulira kroz vreme na celom definisanom vremenskom kontekstu, čime se niži slojevi mreže „teraju“ da nauče transformacije obeležja koje su nezavisne od translacije.

Osim uobičajenih parametara za definisanje DNN, kao što su broj slojeva i neurona po sloju, ili vrsta aktivacione funkcije, kod TDNN se moraju specificirati i ulazni konteksti za neurone svakog od slojeva koji se koriste za izračunavanje aktivacije tih neurona u datom trenutku. Šema rada tipične mreže sa vremenskim kašnjenjem za modelovanje akustike u ASR sistemima data je na slici 13. Ono što je specifično u *chain* metodu obuke jeste poduzorkovanje. Kod standardnih TDNN, aktivacije skrivenih slojeva se računaju na osnovu svih vremenskih koraka u definisanom kontekstu. To dovodi do velikog preklapanja konteksta koji se za dati neuron računaju u susednim trenucima. Pod pretpostavkom da su vremenski susedne aktivacije korelisane, one se mogu poduzorkovati – umesto spajanja (eng. *splicing*) niza vremenski susednih frejmova u svakom sloju, dozvoljavaju se praznine između njih. Na primer, umesto korišćenja neprekidnog konteksta $[t-3, t+3]$ (od trenutnog frejma minus 3, do trenutnog frejma plus 3), može se koristiti na primer $\{t-3, t+3\}$, odnosno mogu se spojiti samo ta 2 frejma (odnosno ulazi u neuron u tim frejmovima) umesto svih 7. U praksi se često spajaju upravo samo dva frejma za svaki skriveni neuron. Na slici 13 ovo je prikazano crvenom bojom. Činjenica da su na slici 13, ali i u praksi, razlike u frejmovima između korišćenih ofseta umnošci broja 3 (ili nekog drugog malog prirodnog broja) nije slučajnost – mreža se tako dizajnira da bi za svaki izlazni frejm bilo potrebno određivanje relativno malog broja

aktivacija skrivenih slojeva. Time se značajno umanjuje potreban broj izračunavanja i prilikom prolaza unapred, i prilikom propagacije unazad tokom obuke, čime se TDNN obuka višestruko ubrzava (traje i po nekoliko puta kraće u odnosu na referentne (Peddinti i drugi, 2015)). To se značajno odražava i na brzinu primene obučene mreže u ASR sistemu. Povrh svega, u *chain* obuci radi se i intenzivna paralelizacija podelom skupa za obuku na podskupove koji se mogu istovremeno obrađivati, što je omogućeno specifičnim izmenama funkcije cilja, i čime se obuka još dodatno ubrzava.



Slika 13. Šema rada TDNN sa poduzorkovanjem i bez njega

Crvenom bojom je dat način rada sa poduzorkovanjem. Na ulaznom sloju spajaju se svi frejmovi od t-2 do t+2, a na tri skrivena sloja frejmovi {t-1,t+2}, {t-3,t+3} i {t-7,t+2}. Ovakvi asimetrični konteksti se često pokazuju pogodnim u pogledu WER. Plavom i crvenom bojom predstavljen je rad mreže bez poduzorkovanja.

Konkretno, u eksperimentima sa različitim modelima jezika iskorišćena je prethodno optimizovana TDNN arhitektura sa 10 skrivenih slojeva i 1024 neurona po sloju. U svim TDNN slojevima korišćena je ReLU (eng. *Rectified Linear Unit*) aktivaciona funkcija $f(x) = \max(0, x)$, za koju se pokazalo da za duboke mreže sa puno skrivenih slojeva, i pogotovo za primene u prepoznavanju govora, poseduje nekoliko bitnih prednosti u odnosu na sigmoidalnu i druge aktivacione funkcije – umanjuje problem nestajućeg gradijenta, računski je puno jednostavnija, i čak brže iskonvergira (Dahl i drugi, 2013). Svaka ReLU komponenta je praćena blokom za renormalizaciju, koji postoji sa ciljem sprečavanja preobučavanja (zajedno se ponekad nazivaju *ReLU-renorm* komponentom). U inicijalnim slojevima (do petog sloja)

mreže vrši se spajanje konteksta $\{t-1, t, t+1\}$ – neuroni u ovim slojevima „vide“ tri uzastopna frejma, a u dubljim slojevima (nakon petog sloja) spajaju se širi konteksti $\{t-3, t, t+3\}$ – ovi neuroni „vide“ isto tri frejma, ali vremenski razmaknuta za po tri frejma jedan od drugog. Ovakvom konfiguracijom mreže omogućeno je da se za njene najdublje slojeve evaluacija vrši na svaka tri frejma (Pakoci i drugi, 2018).

Obuka je rađena u 5 epoha, što je na osnovu ogromne količine podataka za obuku rezultiralo sa ukupno čak 2235 iteracija (u svakoj iteraciji se obrađuje podskup podataka, sve dok se svi podaci ne obrade onoliko puta koliko ima epoha). Ulazna segmentacija skupa za obuku obezbeđena je od strane HMM-GMM sistema sa adaptacijom na govornike (eng. *Speaker Adaptive Training*, SAT) (Povey i drugi, 2008), koji je obuhvatao oko 3500 HMM stanja i 35000 gausijana (komponenti Gausovih smeša). Ulazna akustička obeležja su 40 statičkih MFC koeficijenata visoke rezolucije, kao i 3 dodatna obeležja vezana za pič – ponderisani logaritam osnovne učestanosti, delta vrednost (prvi izvod) istog logaritma, i modifikovana (eng. *warped*) vrednost takozvane normalizovane funkcije uzajamne korelacije (eng. *Normalized Cross-Correlation Function*, NCCF), koja predstavlja jedan od najboljih načina za detekciju i praćenje toka promene piča u glasu, i karakteriše je to što ima vrednost između -1 i 1, pri čemu za zvučne frejmove (u kojima su izgovoreni zvučni fonemi, to jest fonemi za koje se pič može dobro definisati) ima vrednosti bliže 1 (Verteletskaya & Šimák, 2009). Za potrebe modelovanja govornika i komunikacionog kanala, odnosno adaptacije na njih, koristi se vektor identiteta, skraćeno *i*-vektor (Dehak i drugi, 2011), dimenzionalnosti 100, koji se dodaje na ulazni vektor obeležja dimenzije 43. Vektori identiteta se već dugo uspešno primenjuju u raznim zadacima u oblasti prepoznavanja govora i govornika. U ovom slučaju, oni se računaju i pamte po govorniku, uz ažuriranje zapamćenih vrednosti na svakih 10 novih poduzorkovanih frejmova. Dodatno, pošto se za inicijalni sloj neuronske mreže takođe vrši spajanje konteksta $\{t-1, t, t+1\}$, ukupna dimenzionalnost ulaznih obeležja je $3 \cdot 43 + 100 = 229$. Što se tiče ostatka mreže, pošto imamo 1024 neurona po skrivenom sloju mreže, a neuroni uvek „vide“ po tri frejma, svi slojevi osim inicijalnog na ulazu svojih neurona imaju vektor dimenzije $3 \cdot 1024 = 3072$. Ukupan broj mogućih akustičkih stanja na izlazu neuronske mreže je iznosio oko 2000 – pri tome je bilo potrebno uzeti u obzir specifičnu *chain* arhitekturu sa samo po jednim emitujućim stanjem za svaki

kontekstno-zavisni fonem koji se modeluje (Pakoci i drugi, 2018). Efektivna brzina učenja mreže bila je u opsegu od 0,001 (inicijalno, na početku obuke) do 0,0001 (u završnim iteracijama obuke). Preobučavanje na podatke za obuku se, uz postojanje komponenti za renormalizaciju u svim skrivenim slojevima, sprečava i uz pomoć nekoliko metoda inkorporiranih u *chain* proceduru obuke (Povey i drugi, 2016). Prvi od ovih metoda je regularizacija unakrsne entropije na izlaznom sloju pri obuci (takozvanom *Xent* izlaznom sloju) – mreži se tokom obuke specificira posebna verzija izlaznog sloja, čiji se izlaz skalira specificiranom konstantom (na primer 0,1). Drugi metod je regularizacija glavnog izlaznog sloja (ne *Xent* izlaznog sloja), koja se sastoji od dodavanja neke vrednosti na izlaz mreže, na primer $-0.00025 \cdot y \cdot y$, ako je y izlazna vrednost neuronske mreže za dati frejm. Poslednji metod regularizacije u *chain* proceduri obuke jeste upotreba takozvanih „propustljivih“ HMM-ova (eng. *leaky HMMs*), koji eksplicitnim dozvoljavanjem prelaza iz svakog stanja u svako drugo stanje sa nekim malim koeficijentom čine da sistem postepeno zaboravlja kontekst.

REFERENTNI TRIGRAM REZULTATI

Referentni trigram model jezika na pomenutom korpusu obučen je, kao i raniji trigram model opisan u poglavlju 4, preko alata SRILM, uz Kneser-Ney metod ublažavanja raspodele verovatnoća, i prag odsecanja viših *n*-grama (bigrama i trigrama) od 10^{-7}, koji je takođe optimizovan u ranijim istraživanjima (Pakoci i drugi, 2017). Rečnik je odabran na osnovu spiska reči koje se u celom tekstualnom korpusu za obuku nalaze makar tri puta, sa dodatkom i ređih reči ako su se one našle u transkripcijama srpskih audio baza podataka – praktično, uzet je skup svih različitih reči iz srpskih transkripcija kao bazni (njih ima oko 158000, kao što se vidi u tabeli 3), i zatim je on dopunjen rečima iz ostatka tekstova, i to onim koje se pojavljuju barem triput. Naknadno, prethodno neviđene (OOV) reči iz test skupa rečenica su takođe dodate u rečnik sistema, iako se ti tekstovi naravno nisu koristili za obuku modela jezika. Međutim, bez obzira na to što se test rečenice nisu koristile za obuku, moralo bi se napomenuti da ubacivanje OOV reči u rečnik može uticati na tačnost prepoznavanja ASR sistema, što bi moglo dovesti do preoptimističnih rezultata. Odluka da se eksperimenti postave na ovaj način doneta je zbog planiranog načina upotrebe ASR sistema čija su komponenta opisani jezički modeli, a to su relativno

ograničeni domeni sa manje-više unapred poznatim rečnikom, pa je bilo korisno saznati WER mogućnosti sistema u takvim uslovima rada. Osim toga, i u dosadašnjim istraživanjima za srpski jezik (Popović i drugi, 2018) su rečnici formirani na ovakav način. Ipak, korisno je imati i WER rezultate u slučaju kada OOV reči nisu eksplicitno dodate u rečnik. Oni su opisani u sekciji „Eksperimenti sa OOV rečima“. Budući eksperimenti bi mogli da ispitaju i mogućnosti formiranja modela jezika sa otvorenim rečnikom (eng. *open vocabulary LM*), koji je sposoban da nauči, odnosno automatski ubaci nove reči u postojeći model (Qin, 2013; Matthews i drugi, 2018).

U ovom eksperimentu, rezultujući trigram jezički model sadržao je nešto više od 250000 unigrama, a osim toga i oko 1,87 miliona bigrama i oko 551000 trigrama. Na datom test skupu opisanom u poglavlju 2, izračunata je perpleksivnost od 631,5.

Ovaj referentni trigram model jezika u kombinaciji sa postojećim akustičkim modelom proizveo je stopu greške na nivou reči od 8,47%. Istovremeno, izračunata greška po karakteru, CER, bila je svega 2,48%. Uticaj inflektivnosti srpskog jezika se osim u samom odnosu WER i CER može videti i u pregledu najčešćih grešaka. Ako izuzmemo najčešću grešku – mešanje veoma slično zvučećih reči *i* i *je*, koje su istovremeno na samom vrhu najčešćih reči u korpusu – pronaći ćemo dosta pogrešnih padeža, rodova i brojeva, na primer *koju* umesto *koji*, *koje* umesto *koja*, *bila* umesto *bilo*, i tako dalje. Sem ovih grešaka, vidimo i nekoliko instanci u kojima su zamenjene reči koje se akustički vrlo malo razlikuju, a uz to imaju i identično značenje i ulogu u rečenicama, kao što su na primer *kad* i *kada*, odnosno *s* i *sa*, koje se ipak računaju kao i svaka druga greška. Takođe, postoje greške na rečima koje se u spontanom govoru često malo skrate ili neispravno izgovore, recimo *znači* i *’nači* (početni fonem se često ne izgovori ili je skoro nečujan), ili *rekao* i *rek’o*. Ove poslednje greške su pre svega posledica trenutnog stanja korpusa za obuku, tačnije transkripcija audio baza sa spontanijim govorom (što su pre svega radio emisije), u kojima su reči beležene tačno onako kako su izgovorene, čak iako to nije potpuno pravilno i očekivano, a ako se takva situacija ponovila dovoljan broj puta, atipična reč je ušla u rečnik. Slična situacija je i sa rečima koje nekad budu napisane odvojeno, a nekad zajedno, bez obzira da li je neki od ta dva oblika gramatički neispravan (osobe koje pišu tekstove mogu da pogreše) – na primer, „*u stvari*“ i *ustvari*, odnosno „*bi smo*“ i *bismo*. Konačno, uočen je i izvestan broj štamparskih grešaka. Neke greške se možda mogu popraviti i automatskim zamenjivanjem (svakako uz oprez da se na taj način ne

naprave nove greške), ali većina bi zahtevala detaljno prolaženje kroz ceo tekstualni korpus. Od ukupno oko 254000 reči u test skupu, oko 235000 je dobro prepoznato, a locirano je nešto više od 14000 zamena, 5200 brisanja, i 2200 umetanja. Može se primetiti da zbir broja detektovanih grešaka i broja ispravno prepoznatih reči prelazi ukupan broj reči – to se dešava jer se ponekad za istu reč može vezati nekoliko grešaka; na primer, ako su umesto date duže reči prepoznate dve kraće reči, od kojih naravno nijedna nije ispravna, to će se obračunati kao jedno umetanje i jedna zamena.

Trebalo bi spomenuti i da je najveći WER dobijen na delu test baze koji se odnosi na radio emisije (12,37%), dok je stopa greške na audio knjigama puno manja (svega 5,6%), a na mobilnoj bazi (koju, radi podsećanja, čine relativno jednostavne i kratke rečenice iz veoma ograničenog skupa reči) greška je svega 0,98%, što je dosledno ranijim eksperimentima (Pakoci i drugi, 2017; Popović i drugi, 2018), i što se može objasniti prilično malim rečnikom koji odgovara specifično tom delu baze (sadrži tek oko 4000 različitih reči), kao i sintagmama i rečeničnim formulacijama koje se stalno ponavljaju, dok se ponekad čak cele rečenice podudaraju, za praktično sve govornike, što je dozvolilo modelu jezika da se lako prilagodi i veoma dobro nauči da očekuje takav tip rečenica.

POS TRIGRAM REZULTATI

Trigram jezički model obučen je i na POS tagovanom korpusu, koji uključuje reči sa morfološkim sufiksima umesto standardnih rečenica, pripremljenom preko alata PreRnnlmProc. Ovaj model jezika je bitan pre svega za proizvodnju grafova dekodovanja na kojima će kasnije RNNLM obučen na istom korpusu da primenjuje metod reskorovanja. Njegova obuka nije menjana u odnosu na referentni trigram model.

Rečnik POS tagovanih reči je formiran na sličan način kao i referentni. Prvo je napravljen spisak svih reči (sa sufiksima) koje se nalaze u tagovanim transkripcijama srpskih audio baza podataka (i za obuku, i za test), a zatim je spisak dopunjavan rečima iz tagovanog tekstualnog korpusa prema broju pojavljivanja (od najčešćih ka najređim) sve dok se veličina rečnika nije izjednačila sa brojem reči u referentnom rečniku (odnosno, sa oko 250000 reči).

Rezultujući POS trigram jezički model je osim tih 250000 unigrama sadržao i oko 2,03 miliona bigrama i oko 526000 trigrama, što je dovoljno blizu referentnim brojkama da bi poređenje dobijenih rezultata bilo fer. Na POS tagovanom test skupu, određena je perpleksivnost od 718,1. To je više od referentne perpleksivnosti (631,5), i može se donekle opravdati značajnim povećanjem broja različitih reči u opticaju nakon dodavanja sufiksa – naime, sada se na mestima na kojima se nekad nalazila ista reč može naći dva, ili čak i više različitih oblika te reči. Jednostavan primer je reč *radio*, koja sada može biti i imenica *radio*, i to i u nominativu i u akuzativu, i glagol *radio* – umesto nekada jedne reči sada imamo tri. Takvi primeri su brojni, što dovodi do toga da umesto 458000 različitih reči na raspolaganju imamo 600000, od kojih u oba slučaja isti broj, njih 250000, ulazi u rečnik. Veći ukupan fond reči, uz istu veličinu rečnika, može dovesti do veće nesigurnosti u proceni modela jezika koja je sledeća reč. Ova pretpostavka će detaljnije biti ispitana u budućim istraživanjima. Međutim, lošija perpleksivnost ne znači i da će WER rezultat biti gori (Klakow & Peters, 2002). U tabeli 14 dat je uporedni prikaz karakteristika dva trigram modela jezika.

Tabela 14. Karakteristike trigram modela jezika

LM	Broj unigrama	Broj bigrama	Broj trigrama	PPL
trigram ref.	250059	1871060	550318	631,5
trigram POS	250059	2026862	525507	718,1

Osim 250057 reči, u unigrame ulaze i BOS i EOS simboli.

U samom testiranju, POS trigram model jezika je proizveo WER od 8,3% i CER od 2,49%, što je veoma blisko referentnom modelu. Uzevši u obzir da su jedine morfološke informacije u ovom slučaju sufiksi reči, kao i da je fond reči povećan, a perpleksivnost modela nešto lošija, rezultat je i više nego prihvatljiv. Ostvareno blago poboljšanje može biti posledica toga što je razlikovanje nekada istih pojavnih oblika reči u slučaju kada to u stvari i jesu različite reči (ili po lemi ili po morfologiji) malo pomoglo razlikovanju odgovarajućih konteksta.

U tabeli 15 dat je uporedni prikaz rezultata dva trigram modela. Iz nje se može videti da je nešto veće poboljšanje ostvareno kod audio knjiga, a pretpostavka je da je

to slučaj zato što se u delu baze sa audio knjigama nalaze tekstovi pročitani od strane profesionalnih govornika, sa relativno malo loše ili pogrešno izgovorenih reči, odnosno neočekivanih rečeničnih struktura. Nešto manje poboljšanje ostvareno je za radio emisije, a za mobilnu bazu dobijeno je čak blago pogoršanje, iako je rezultat i dalje vrlo dobar – samo oko 1% grešaka. Pretpostavlja se da je uzrok za ovakvu raspodelu grešaka spontanije izgovaranje rečenica i kod emisija i kod mobilne baze, kao i veća mogućnost greške u snimanju ili transkribovanju rečenica u tim bazama podataka (u odnosu na profesionalno čitane knjige), odnosno u POS tagovanju – na primer, mobilna baza sadrži dosta raznih vlastitih imenica u više oblika, koje AnTagger ponekad ne detektuje ispravno, verovatno jer akcenatsko-morfološki rečnik sa kojim radi ne sadrži unose za sve te reči).

Tabela 15. Rezultati testova sa trigram modelima jezika

Rezultat	Ukupno [%]	Audio knjige [%]	Radio emisije [%]	Mobilna baza [%]
WER trigram ref.	8,47	5,60	12,37	**0,98**
WER trigram POS	**8,30**	**5,40**	**12,19**	1,04
CER trigram ref.	**2,48**	**1,27**	**3,97**	**0,36**
CER trigram POS	2,49	1,28	4,00	0,38

Primetno je blago poboljšanje WER, uz skoro identičan CER, što je posledica manjeg broja zamena usled morfoloških grešaka, uz izvestan broj novih grešaka drugog tipa zbog većeg fonda reči.

Što se tiče karakterističnih grešaka, ukupan broj zamena je blago opao (bilo ih je par stotina manje od 14000), dok se broj brisanja i umetanja neznatno razlikuje. Takođe se može primetiti da je broj grešaka koje se odnose na morfologiju (prvenstveno među zamenama) procentualno nešto manji. Što se tiče umetanja i brisanja, u listama tih tipova grešaka i inače stoje pre svega veoma kratke nepromenljive reči (veznici, predlozi, rečce, uzvici), dok su duže reči relativno retke, a u POS rezultatima blago potisnute još niže. U tabeli 16 uporedno su prikazane neke od najčešćih grešaka u sistemima sa trigram modelima jezika, pre i posle dodavanja

morfoloških sufiksa. Očekuje se da će reskorovanjem sa RNNLM na grafu dobijenim trigram POS modelom moći da se dobije bitnije poboljšanje.

Tabela 16. Primeri nekih čestih grešaka sa brojem pojavljivanja u trigram testovima

Zamene: ref / POS	Umetanja: ref / POS	Brisanja: ref / POS
je → i: 89/88	i: 264/255	je: 765/788
reko → rekao: 50/48	je: 231/218	i: 641/650
nači → znači: 47/46	u: 118/99	u: 322/327
i → je: 41/36	da: 110/109	da: 187/183
koji → koju: 40/34	a: 66/65	to: 116/119
koja → koje: 30/28	na: 43/41	on: 114/113
koje → koji: 28/22	su: 17/16	su: 41/40
bila → bilo: 24/23	kaže: 9/8	sam: 33/31
sa → sam: 17/16	koji: 8/6	koji: 19/21

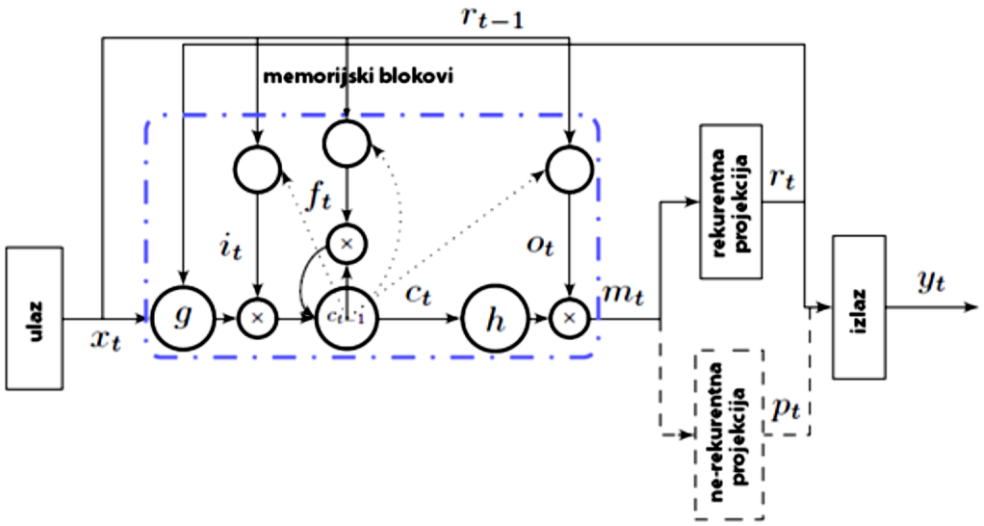
Najviše poboljšanja je viđeno u zamenama, pogotovo za neke promenljive reči za koje je definisano više morfoloških kategorija.

REFERENTNI RNNLM REZULTATI

Referentni RNNLM, kao i svi RNN modeli koji su testirani u nastavku disertacije, obučavan je uz pomoć alata Kaldi-RNNLM. Za ovu referentnu obuku korišćen je, naravno, originalni tekstualni korpus. Vektori obeležja koji se koriste za pojedine reči definisani su na sličan način kao što je dato u tabeli 5. Svaka reč je opisana *one-hot* vektorom za $V = 9970$ najčešćih reči u korpusu (sve reči koje se pojavljuju makar 252 puta) i odgovarajućim vektorom broja pojavljivanja slovnih bigrama, trigrama i kvadrigrama koji se nalaze u njoj (ukupno 1091 početnih, 1229 završnih i 8178 ostalih slovnih *n*-grama, u koju grupu spadaju svi oni sa učestanošću pojavljivanja u korpusu od barem 0,0001), kao i sa dva augmentovana obeležja – unigram log-verovatnoćom i dužinom reči, a tu su i konstantno obeležje za sve reči (sa vrednošću 0,01), kao i *one-hot* vektor za 5 specijalnih reči – „BOS“, „EOS“,

„BRK“, „UNK“ i „!SIL“. Ukupno, bilo je 20476 različitih obeležja, što je svakako puno manje od broja reči (250057 + 2 za BOS i EOS), kolika bi bila dimenzija ulaznog sloja RNN da se koristi 1-od-*N* reprezentacija.

Što se tiče arhitekture neuronske mreže, koristi se kombinovana TDNN+LSTMP konfiguracija. Uopštena TDNN arhitektura je opisana u odeljku „Obuka akustičkog modela“, a ovde se, u osnovnom eksperimentu, konkretno koriste četiri skrivena TDNN sloja sa po 1024 neurona, pri čemu se na ulaznom i izlaznom koristi spajanje vremenskih konteksta $\{t-1, t\}$, a na srednja dva TDNN sloja se vrši spajanje tipa $\{t-2, t\}$. U ovim slojevima koristi se ReLU aktivaciona funkcija, praćena komponentom za renormalizaciju (dakle, *ReLU-renorm*). Između svaka dva TDNN sloja ubačen je po jedan LSTMP (LSTM sa projekcijom, eng. *LSTM Projected*) sloj, dakle, ukupno još 3. U standardnom LSTM neuronu, rekurentne veze sa izlaza neurona vode direktno prema njegovom ulazu i odgovarajućim kapijama (slika 7), a izlaz vodi ka sledećem sloju. LSTMP je jedna od predloženih alternativa toj osnovnoj LSTM arhitekturi (Sak i drugi, 2014). Šematski prikaz neurona u LSTMP sloju dat je na slici 14.



Slika 14. Šema LSTMP arhitekture

LSTMP arhitektura sadrži dodatni sloj rekurentne projekcije i opcioni sloj ne-rekurentne projekcije. Radi jednostavnosti, prikazan je samo jedan memorijski blok. U stvarnosti, memorijskih blokova ima više.

Može se primetiti da izlazi neurona vode prvo na takozvani rekurentni projekcioni sloj (tačnije podsloj), a tek zatim se podaci prosleđuju dalje – nazad na ulaze i kapije, odnosno prema narednim slojevima. Opciono, može postojati i dodatni,

ne-rekurentni projekcioni sloj pre formiranja izlaza neurona, koji je direktno povezan sa narednim slojem (njegovi izlazi se ne šalju nazad prema ulazu i kapijama). Ako rekurentni projekcioni sloj ima dimenziju n_r, a postoji i ne-rekurentni projekcioni sloj dimenzije n_p, oni se mogu posmatrati zajedno kao jedinstveni projekcioni sloj sa $n_r + n_p$ jedinica. Projekcioni slojevi se dodaju da bi omogućili smanjenje ukupnog broja parametara LSTM sloja mreže za faktor $(n_r + n_p)/n_c$, gde je n_c broj LSTM memorijskih jedinica, odnosno, da bi dozvolili povećanje memorijskog kapaciteta mreže (n_c) uz istovremenu kontrolu složenosti. Pokazalo se da duboke LSTMP mreže mogu da naprave modele sa većom moći generalizacije u odnosu na ostale arhitekture slične kompleksnosti, pa je stoga počela njihova primena uglavnom u akustičkom modelovanju za sisteme za automatsko prepoznavanje govora, ali korišćeni su i u kontekstu obuke jezičkih modela (Sak i drugi, 2014). U konkretnom slučaju, u osnovnoj postavci eksperimenta, LSTM slojevi su imali po 1024 neurona u sloju ($n_c = 1024$), sa dimenzijom rekurentnog projekcionog podsloja od 256 ($n_r = 256$) i dimenzijom ne-rekurentnog projekcionog podsloja od isto 256 ($n_p = 256$). To daje ukupnu dimenzionalnost izlaza LSTMP sloja od 512.

Dimenzionalnost ulaznog *embedding* vektora (takozvana *embedding* dimenzionalnost) je takođe 1024 elementa. Uzevši u obzir opisani način spajanja konteksta na ulazu svakog od TDNN slojeva, to znači da na ulazu prvog TDNN sloja imamo dimenzionalnost podataka od $2 \cdot 1024 = 2048$, dok na ulazima svih ostalih TDNN slojeva imamo dimenzionalnost od $2 \cdot 512 = 1024$. Kao što je već rečeno, LSTMP slojevi koji se nalaze između njih prihvataju podatke dimenzionalnosti 1024 (što je ujedno i dimenzionalnost izlaza TDNN slojeva), a dalje šalju podatke dimenzionalnosti 512.

Od ostalih parametara obuke, bitno je spomenuti i broj uzoraka koji se koriste u algoritmu uzorkovanja pri računanju funkcije cilja – 1024, kao i broj epoha obuke – 30, što je sa datom količinom podataka za obuku rezultovalo sa 180 iteracija. Najbolja iteracija se utvrđuje na osnovu log-verodostojnosti na validacionom skupu od 20000 rečenica, koji je unapred izdvojen. Brzina učenja mreže se kretala od 0,001 na početku, do 0,0001 na kraju obuke, dok je brzina učenja za *embedding* matricu ograničena na deseti deo brzine učenja mreže u datoj iteraciji. Obuka je ukupno trajala oko 19 sati, na jednom grafičkom procesoru GeForce GTX 1080 Ti.

Pri reskorovanju u testiranju, koristi se metod direktnog operisanja na grafu dekodovanja, sa aproksimacijom istorija 4-gramima, i to uz primenu odsecanja (kao kod ranijih TensorFlow RNNLM eksperimenata) radi ubrzanja procedure, kao i potencijalnog poboljšanja rezultata (Xu i drugi, 2018a). Interpolacioni faktor pri kombinovanju LM skorova sa referentnim trigram skorovima iznosio je 0,8 (što je ranije optimizovano). Reskorovanje je ukupno trajalo tek oko 39 minuta na procesoru Intel Core i7 4790, uz do 8 paralelnih procesa, što je svakako daleko brže od realnog vremena. Parametri RNNLM obuke i reskorovanja rezimirani su u dodatku 3, zajedno sa konfiguracionom datotekom za opisanu RNN arhitekturu.

Na datom test skupu, sa referentnim RNNLM dobijena je perpleksivnost od oko 236,5, što je mnogo bolje od perpleksivnosti bilo kog od predstavljenih trigram modela. Što se tiče testova prepoznavanja, ukupna stopa greške na nivou reči je iznosila 6,37%, što je relativno poboljšanje od oko 25% u odnosu na referentni trigram jezički model, odnosno 23% u odnosu na POS trigram model. Stopa greške na nivou karaktera spala je na 1,93%, što je bolje za oko 22% u odnosu na oba trigram modela. Najveće poboljšanje je ostvareno, očekivano, na audio bazi podataka sa najčistijim i najtipičnijim izgovorima – audio knjigama, gde je ostvaren WER od 3,84% (bolje za oko 31% i 29% u odnosu na trigram modele). To je praćeno relativnim poboljšanjem od oko 22% i 21% na radio emisijama, za ukupan WER od 9,63%. Konačno, značajno poboljšanje je ostvareno i na mobilnoj bazi – WER je iznosio 0,83%, što je bolje za oko 15%, odnosno 20% u odnosu na trigram rezultate. Opisani rezultati su rezimirani u tabeli 17.

Tabela 17. Poređenje rezultata referentnih trigram i RNN modela jezika

LM	WER ukupno [%]	WER audio knjige [%]	WER radio emisije [%]	WER mobilna baza [%]	PPL
trigram ref.	8,47	5,60	12,37	0,98	631,5
RNNLM ref.	**6,37**	**3,84**	**9,63**	**0,83**	**236,5**

Ostvareno je prosečno relativno WER poboljšanje od preko 31%, uz više nego prepolovljenu perpleksivnost.

Što se tiče karakterističnih grešaka, primećuje se slična raspodela kao i kod referentnog trigram modela jezika, samo što ih generalno ima značajno manje. Na vrhu liste grešaka su opet reči *i* i *je*, a tu su i pojedine greške u padežu/rodu/broju (*koji* umesto *koje*, i slično), zamene tipa *kad* umesto *kada*, atipično izgovorene reči kao što su *rek'o* i *'nači*, pogrešno napisane sekvence reči tipa *ustvari*, kao i druge štamparske greške. Ukupan broj zamena je bio oko 10500, dok je umetanja bilo oko 2000, a brisanja oko 3800. Od oko 254000 test reči ukupno, oko 240000 je prepoznato ispravno.

Osim ovog osnovnog eksperimenta sa referentnom RNNLM postavkom, odrađena je čitava baterija testova u kojima su menjani arhitektura neuronske mreže, izgled vektora obeležja reči, kao i podešavanja vezana za reskorovanje, u cilju pronalaženja optimalne kombinacije parametara. Uporedni rezultati su dati u tabeli 18. Vidi se da je sa pojedinim podešavanjima obuke postignuto neznatno poboljšanje, ali je ono bilo u granicama greške, pa je stoga sasvim moguće da je ostvareno poboljšanje najvećim delom rezultat drugačije inicijalizacije neuronske mreže (koja se inicijalizuje na slučajan način), a ne promenjenih parametara. Najbolji rezultat je postignut sa značajnim povećanjem *one-hot* vektora najčešćih reči (vrednosti V) – WER od tačno 6,3% je dobijen za $V \approx 75000$ i $V \approx 100000$, međutim, time se bitno povećao i ulazni vektor obeležja reči, što čini model složenijim i memorijski zahtevnijim, a obuku i testiranje sporijim. Takođe je primetno da je granična učestanost slovnih *n*-grama za ubacivanje u vektor obeležja dobro postavljena u inicijalnom eksperimentu, kao i minimalna i maksimalna vrednost *n* za njih – kombinacija bigrama, trigrama i kvadrigrama nije najbolja (ali ne zaostaje mnogo za najboljim rezultatom), međutim predstavlja dobar kompromis između poboljšanja WER i usporavanja obuke i dekodovanja. Dva augmentovana obeležja takođe daju svoj mali doprinos, pošto je WER rezultat lošiji bez njih nego sa njima. Pri reskorovanju, aproksimacija 4-gramima nije optimalna po pitanju WER jer 5-grami ostvaruju nešto bolji rezultat, ali opet po cenu sporije procedure. Po pitanju složenosti neuronske mreže, vidi se da suviše kompleksna mreža može da dovede do pogoršanja WER, što je posledica prevelikog prilagođavanja skupu podataka za obuku. Manje složene arhitekture sa 3 TDNN (i 2 LSTMP) sloja su postigle nešto bolje rezultate od osnovne varijante, što ukazuje na njihovu bolju moć generalizacije.

Tabela 18. Uporedni rezultati raznih referentnih RNNLM testova

Modifikacija	WER [%]	Modifikacija	WER [%]	Modifikacija	WER [%]
-	6,37	$P_{ngram} > 0{,}00025$	6.45	5-gram reskor.	**6,34**
$V \approx 5000$	6,41	$P_{ngram} > 10^{-5}$	**6.34**	2+1 sloj	6,40
$V \approx 15000$	6,39	$n \in \{2,3\}$	6,49	3+2 sloja	**6,35**
$V \approx 20000$	6,37	$n \in \{3,4\}$	6,40	3+2 sloja po 512 neurona	**6,35**
$V \approx 50000$	**6,34**	$n \in \{2,3,4,5\}$	**6,34**	3+2 sloja po 1536 neurona	6,43
$V \approx 75000$	**6,30**	bez F_{length}	6,40	4+3 sloja po 512 neurona	**6,36**
$V \approx 100000$	**6,30**	bez $F_{unigram}$	6,38	4+3 sloja po 1536 neurona	6,41
$P_{ngram} > 10^{-3}$	6.54	3-gram reskor.	6,45	5+4 slojeva	6,38

Podebljani su WER rezultati koji su bolji od osnovne varijante. Nijedan od njih nije dovoljno bolji da bi opravdao rezultujuće povećanje kompleksnosti mreže i obuke, odnosno usporavanje dekodovanja. Arhitekture koje, sa druge strane, dolaze u obzir su one sa manje slojeva ili neurona po sloju, jer uglavnom donose čak i malo poboljšanje rezultata, a pri tome i smanjuju složenost sistema.

POS RNNLM REZULTATI

Konačno, RNN modeli jezika su obučeni i na POS tagovanom korpusu sa morfološkim sufiksima, pripremljenom preko alata PreRnnlmProc. Obučeni modeli jezika su se koristili pri reskorovanju na grafovima dekodovanja koje je proizveo POS trigram LM. I u ovom slučaju rečnik je sadržao oko 250000 različitih reči.

U odnosu na referentnu RNNLM obuku, promenjen je vektor obeležja za reči iz rečnika. Kao što je opisano u poglavlju 5, u skladu sa ostalim obeležjima i Kaldi-RNNLM alatom, u vektore obeležja dodavani su *one-hot* vektori za definisane morfološke kategorije, POS slovni trigrami na morfološkim sufiksima reči, kao i

one-hot vektor najčešćih lema. Broj najčešćih lema je uvek bio u skladu sa brojem najčešćih reči za koje se definiše element u tom podvektoru ($L = V$).

U podsekciji „Rezultati sa četiri osnovne morfološke kategorije“ opisani su rezultati sa korišćenjem samo 4 osnovne morfološke kategorije – vrsta reči, padež, gramatički broj i gramatički rod. Nakon toga, dati su rezultati sa svim mogućim kategorijama koje je alat AnTagger mogao da vrati (što je preko 30 POS kategorija). U ovim eksperimentima, da bi se utvrdio individualni uticaj svakog od načina ubacivanja morfoloških informacija u model jezika, testiran je i RNNLM koji je obučen samo na tagovanom korpusu, bez dodavanja bilo kakvih morfoloških obeležja, a zatim su obučeni i testirani modeli jezika sa samo po jednom vrstom dodatnih morfoloških obeležja, i konačno je izvršen čitav niz testova sa svim vrstama POS obeležja, sa raznim parametrima neuronske mreže, vektora obeležja i reskorovanja, kao i kod referentnih RNNLM eksperimenata.

REZULTATI SA ČETIRI OSNOVNE MORFOLOŠKE KATEGORIJE

Moguće vrednosti osnovnih POS kategorija date su u tabeli 6. Uz konstantu, *one-hot* vektor za 5 specijalnih reči, unigram log-verovatnoću i dužinu reči, određeno je da se $V = 9992$ reči uzima za *one-hot* vektor najčešćih reči, što su reči koje se u korpusu za obuku javljaju makar 263 puta, a svoje obeležje ima i isti broj najčešćih lema (u njihovom slučaju, to su leme koje se u odgovarajućim lematizovanim tekstovima javljaju makar 157 puta). Osim toga, tu je i 7962 različitih slovnih *n*-grama (bigrama, trigrama i kvadrigrama), uz 1205 početnih i 1080 završnih. Konačno, postoji i 26 novih obeležja koja pripadaju POS *one-hot* vektorima za 4 morfološke kategorije (12 vrednosti u podvektoru za vrste reči, 8 u podvektoru za padeže, 2 za gramatičke brojeve, i 4 za rodove), odnosno isto toliko POS slovnih trigrama. Ukupan broj obeležja reči iznosi 30291. Vektor obeležja za reč *sam_gla_jed* (reč *sam*, glagol, u jednini), ažuriran na opisani način u odnosu na tabelu 5, sa svim mogućim novim morfološkim obeležjima, prikazan je u tabeli 19.

Kao i u osnovnom referentnom testu, određena je arhitektura neuronske mreže sa 4 TDNN sloja i sa po jednim LSTMP slojem između svaka dva (ukupno ima 7 slojeva), uz *embedding* dimenzionalnost 1024, što je istovremeno bila i dimenzionalnost izlaza TDNN slojeva i memorijski kapacitet LSTM jedinica, dok su

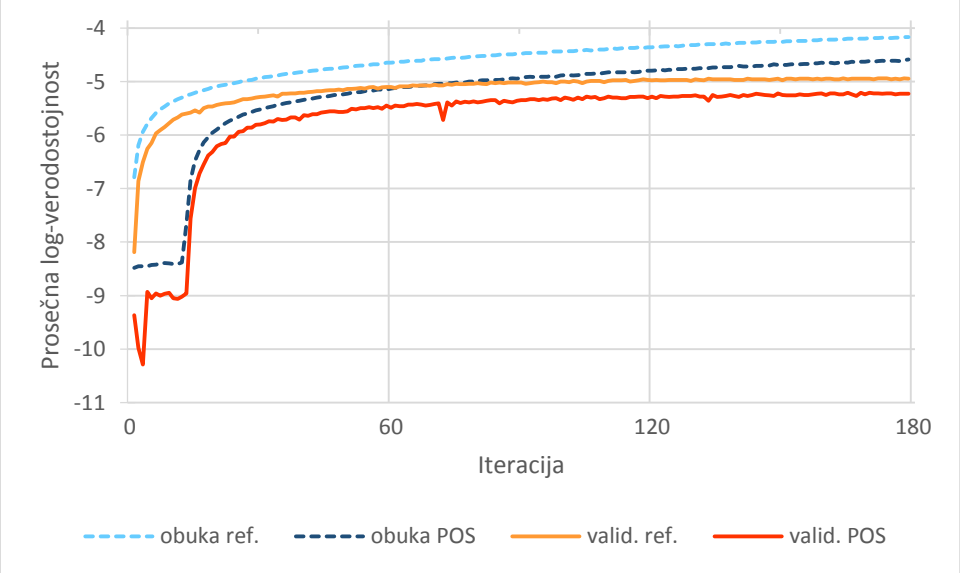
dimenzije rekurentne i ne-rekurentne projekcije iznosile po 256. Pri reskorovanju, vršila se aproksimacija istorija 4-gramima.

Tabela 19. Vektor obeležja za reč *sam_gla_jed* u testovima sa 4 POS kategorije

Indeks obeležja	Vrsta obeležja	Obeležje	Vrednost
0	konstanta	konstanta	0,01
6	unigram	unigram log-verovatnoća	0,00771
7	dužina	dužina reči	0,0051
37	reč	česta reč *sam_gla_jed*	0,21
10000	lema	česta lema *biti*	0,039
19996	POS – vrsta	vrsta: glagol	0,023
20016	POS – broj	broj: jednina	0,014
20078	završni *n*-gram	3-gram *–am$*	0,12
20515	završni *n*-gram	2-gram *–m$*	0,048
20787	završni *n*-gram	4-gram *–sam$*	0,16
21903	početni *n*-gram	2-gram ^*s-*	0,03
21904	početni *n*-gram	3-gram ^*sa-*	0,069
21911	početni *n*-gram	4-gram ^*sam-*	0,14
22640	*n*-gram	2-gram *-am-*	0,065
28154	*n*-gram	2-gram *-sa-*	0,057
28171	*n*-gram	3-gram *-sam-*	0,11
30271	POS trigram	3-gram *gla*	0,024
30275	POS trigram	3-gram *jed*	0,014

Navedena su samo obeležja različita od nule. Sva koja nisu navedena, jednaka su 0. U odnosu na slučaj bez morfoloških obeležja, dimenzionalnost ovog vektora je veća za oko 10000 (uglavnom zbog podvektora najčešćih lema, a tu su i POS one-hot vektori za vrstu reči i gramatički broj, odnosno POS slovni trigrami), ali i dalje je daleko manja od broja reči u rečniku (oko 250000).

Sama obuka je ponovo rađena u 30 epoha, odnosno 180 iteracija, i trajala je, kao i u referentnom slučaju, oko 19 sati. Kretanje prosečne vrednosti log-verodostojnosti reči kroz iteracije, i na skupu za obuku i na validacionom skupu rečenica, uporedno za referentni i POS slučaj, dato je na slici 15. Primećuje se nešto bolja (manje negativna) prosečna log-verodostojnost za referentni slučaj, za oba skupa podataka. Pretpostavlja se da je to posledica iste činjenice koja je izneta u pojašnjenju zašto je perpleksivnost POS trigram modela lošija od perpleksivnosti referentnog trigram modela – većeg ukupnog fonda reči, uz nepromenjenu veličinu rečnika (buduća istraživanja mogu ovu pretpostavku detaljnije testirati). Ovde imamo istu situaciju kao kod trigrama i što se tiče perpleksivnosti – perpleksivnost ovog RNNLM na istom test skupu rečenica iznosio je oko 345,2, što je značajno lošije od referentnog RNNLM (236,5), ali to ni ovde, kao što nije ni kod trigrama, ne mora da najavljuje lošije performanse modela jezika na testu prepoznavanja govora.



Slika 15. Kretanje log-verodostojnosti tokom RNNLM obuka

Vrednosti log-verodostojnosti su konzistentno bolje tokom cele obuke kod referentnog RNNLM u odnosu na LM sa morfološkim obeležjima, i na skupu za obuku, a i na validacionom skupu rečenica.

Da bi bilo moguće reskorovanje, inicijalno je pušteno dekodovanje test skupa trigram modelom jezika obučenom na tagovanom korpusu sa samo 4 glavne kategorije. WER rezultat sa tim modelom bio je identičan POS trigram testu – 8,3%. Zatim je pristupljeno testovima prepoznavanja govora sa RNNLM. POS model sa 4 osnovne morfološke kategorije ostvario je ukupan WER rezultat od 6,13%, odnosno

CER rezultat od 1,9%. Dakle, ostvareno je poboljšanje od oko 4% relativno u odnosu na referentni RNNLM. Najveće poboljšanje je, kao i kod trigrama, ostvareno na audio knjigama – 3,56% WER, što je relativno poboljšanje od 7%, a to je praćeno mobilnom bazom – 0,79% WER, što je unapređenje za 5%, i na kraju radio emisijama, na kojima je dobijeno relativno poboljšanje od svega 2,5% (9,4% WER). Pretpostavlja se da veća poboljšanja nisu bila moguća zbog grešaka u tekstualnom korpusu, konvencijama transkribovanja audio baza i ograničenja AnTagger alata. Mnogi od ovih nedostataka su primećeni, međutim, njihovo potpuno uklanjanje bi zahtevalo detaljan prolazak kroz ceo korpus od strane nekolicine stručnih ljudi, kao i ažuriranje akcenatsko-morfološkog rečnika, što je izuzetno naporan i spor posao. Tome u prilog govori i najmanje poboljšanje na radio emisijama, koje imaju najveći broj pomenutih problema. Buduća istraživanja će svakako raditi i na ovim problemima. U tabeli 20 su sumirani svi navedeni rezultati. Još treba napomenuti trajanje reskorovanja sa ovakvim skupom obeležja – oko 70 minuta, što je dosta duže nego u referentnom slučaju, ali, s obzirom na to koliko materijala sadrži test baza podataka, i dalje ne predstavlja značajno usporavanje odziva ASR sistema u praksi.

Tabela 20. Rezultati testova sa RNN modelima jezika

LM	WER ukupno [%]	WER audio knjige [%]	WER radio emisije [%]	WER mobilna baza [%]	PPL
RNNLM ref.	6,37	3,84	9,63	0,83	**236,5**
RNNLM POS 4 kat.	**6,13**	**3,56**	**9,40**	**0,79**	345,2

Kao i kod trigrama, POS model daje lošiju perpleksivnost, ali ipak postoji i vidno WER poboljšanje.

Broj svih tipova grešaka je opao. Najveća razlika je u zamenama, kojih je bilo nešto ispod 10000 (a oko 10500 u referentnom testu), dok je umanjenje u broju umetanja i brisanja bilo za ispod 100 instanci (1870 umetanja i 3800 brisanja, umesto 1950 i 3830). U listama najčešćih grešaka, pogotovo kod zamena, vidi se smanjen broj pojedinih morfoloških grešaka. Više reči o tome biće u podsekciji „Rezultati sa svim morfološkim kategorijama“.

Obavljeni su i eksperimenti u kojima je ispitan uticaj dodavanja svake vrste morfoloških obeležja ponaosob. Bez bilo kakvih dodatnih obeležja (samo sa POS sufiksima u korpusu za obuku i rečniku), WER rezultat RNN modela bio je 6,33%, što je neznatno bolje od referentnog; samo sa L = 9992 najčešćih lema u vektoru obeležja reči, WER se popravio na 6,27%; samo sa dodatkom POS slovnih trigrama, WER je iznosio 6,19%; samo POS *one-hot* vektori su snizili WER na 6,16%, što je vrlo blizu rezultatu sa svim morfološkim obeležjima (6,13%). Zaključak je da dodatni *one-hot* vektori morfoloških kategorija imaju ubedljivo najveći doprinos, ali da i leme i POS slovni trigrami mogu da doprinesu blagom poboljšanju stope greške, i sami za sebe, a i u kombinaciji sa ostalim morfološkim obeležjima.

REZULTATI SA SVIM MORFOLOŠKIM KATEGORIJAMA

U odnosu na model opisan u podsekciji „Rezultati sa četiri osnovne morfološke kategorije“, ovde je korišćen pun spektar morfoloških kategorija. Za razliku od vektora obeležja iz tabele 19, umesto 26, postoji preko 150 obeležja koja pripadaju *one-hot* vektorima različitih POS kategorija, i isto toliko POS slovnih trigrama. Broj najčešćih reči i lema koje se uzimaju u obzir za obeležja iznosi 9987 (reči sa makar 263 instanci u korpusu, odnosno leme sa barem 157 instanci). Ukupan broj svih različitih obeležja bio je 30513.

Prvi eksperiment je rađen bez bilo kakvih dodatnih morfoloških obeležja, odnosno, samo sa morfološkim sufiksima u tekstu za obuku RNNLM, odnosno u rečniku. Arhitektura neuronske mreže je identična kao u slučaju RNN za četiri osnovne kategorije. Ukupna stopa greške na nivou reči iznosila je 6,33%, što je vrlo blizu referentnom RNNLM eksperimentu (i identično istom eksperimentu samo sa 4 osnovne kategorije). Zaključak je da morfološki sufiksi, sami za sebe, ne mogu da doprinesu poboljšanju RNN modela jezika, odnosno performansi ASR sistema u kom se taj LM nalazi.

U narednom eksperimentu dodate su samo najčešće leme kao obeležja. Svi ostali parametri su ostali isti. Rezultat je bio blago poboljšanje WER na 6,26%. Dakle, leme daju izvestan doprinos, tako što svim pojavnim oblicima najčešćih lema dodeljuju sopstveno obeležje, dok bez njih takvo obeležje imaju samo pojedini česti pojavni oblici.

Sledeći eksperiment je ispitao uticaj POS slovnih trigrama, bez pomoći lema i POS *one-hot* vektora. To je, uz pomenute morfološke *one-hot* vektore, jedan od korišćena dva načina direktnog ubacivanja morfoloških informacija u sam RNNLM. Rezultujuća stopa greške iznosila je 6,23%, što je malo bolje nego što je dobijeno dodavanjem lema. Sa kombinacijom lema i POS slovnih trigrama, WER je dostigao 6,17%.

Naredni test se ticao morfoloških *one-hot* vektora. Bez pomoći lema i POS slovnih trigrama, dodavanje ovih preko 150 novih obeležja popravilo je WER na 6,15%, što definitivno znači da ova morfološka obeležja imaju najveći uticaj na poboljšanje performansi ASR sistema od svih ponuđenih. Kombinovanjem sa lemama, ova obeležja su uspela da spuste WER na 6,13%.

U eksperimentu u kom su se sva pomenuta morfološka obeležja koristila zajedno, u identičnim uslovima kao za prethodne testove, dobijen je WER od 6,08%, što je poboljšanje od oko 4,5% u odnosu na referentni RNNLM, i dodatnih 1% u odnosu na slučaj sa 4 osnovne POS kategorije. Zbog bliskosti ovog rezultata i rezultata za samo osnovnim kategorijama, zaključak je da dodavanje brojnih dodatnih kategorija trenutno nije previše opravdano. Međutim, buduća istraživanja bi prvo trebalo da ispitaju da li neke od kategorija imaju manji značaj, ili možda čak i više odmažu nego pomažu neuronskoj mreži. Očekuje se da bi odabir optimalne grupe morfoloških kategorija, uz svakako potrebne izmene i dorade u akcenatsko-morfološkom rečniku i alatu AnTagger, mogao dalje da spusti stopu greške. Ako posmatramo po delovima baze podataka za testiranje, u odnosu na varijantu sa 4 kategorije, opet su najviše poboljšane performanse na audio knjigama (za 2%, dobijen je WER od 3,49%), dok je poboljšanje na radio emisijama minimalno (WER je sada bio 9,37% umesto 9,4%), a na mobilnoj bazi nije ni bilo pomaka (WER je ostao na 0,79%). Ukupna stopa greške na nivou karaktera bila je 1,88%, što je opet minimalan korak unapred (u odnosu na 1,9%). Perpleksivnost izračunata na test rečenicama je neznatno bolja u odnosu na RNNLM sa osnovnim morfološkim kategorijama – 344,5, ali i dalje ni blizu referentnom RNNLM. Ovi rezultati su rezimirani u tabeli 21. Kao i kod testa sa 4 kategorije, obuka je trajala 19 sati, a reskorovanje oko 70 minuta.

Tabela 21. Rezultati testova sa morfološkim RNN modelima jezika

LM	WER ukupno [%]	WER audio knjige [%]	WER radio emisije [%]	WER mobilna baza [%]	PPL
RNNLM POS 4 kat.	6,13	3,56	9,40	**0,79**	345,2
RNNLM POS sve kat.	**6,08**	**3,49**	**9,37**	**0,79**	**344,5**

Proširivanjem spiska POS kategorija ostvareni su dalji mali pomaci u pogledu WER i perpleksivnosti.

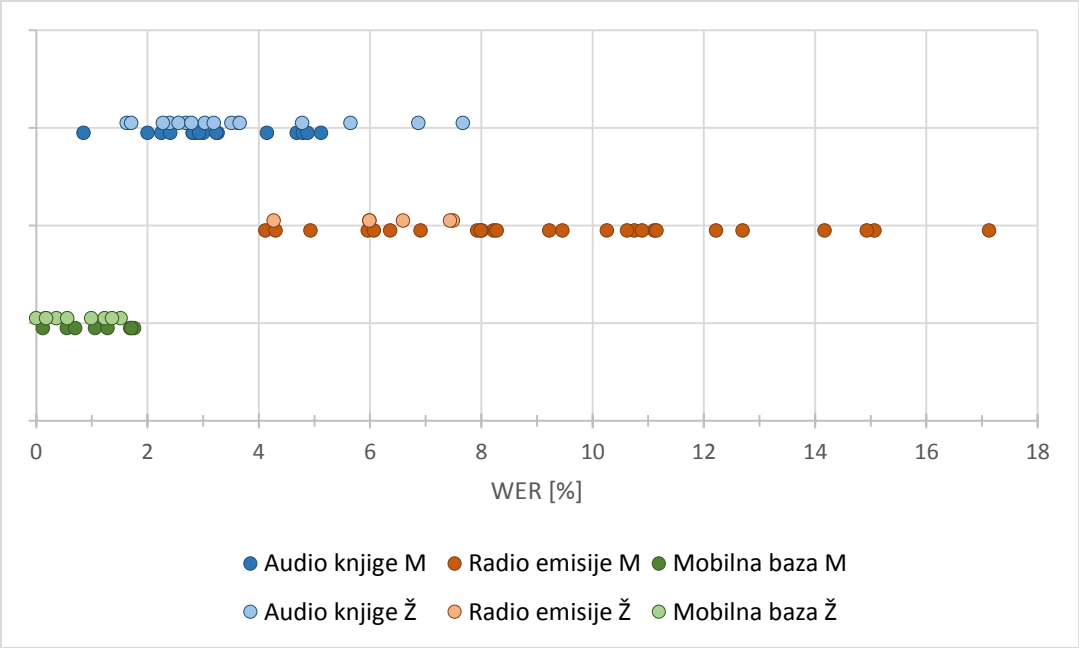
Tabela 22. Primeri nekih čestih grešaka sa brojem pojavljivanja u RNN testovima

Zamene: ref / POS	Umetanja: ref / POS	Brisanja: ref / POS
je → i: 55/64	i: 326/320	je: 551/530
načи → znači: 45/43	je: 234/220	i: 541/523
reko → rekao: 38/37	u: 124/99	u: 226/208
koje → koji: 28/27	da: 100/106	da: 133/125
i → je: 22/24	a: 53/70	a: 97/95
koja → koje: 21/17	na: 36/37	on: 71/69
koji → koju: 19/19	su: 17/12	to: 69/69
sa → sam: 15/16	koji: 11/5	sam: 21/19
bila → bilo: 13/14	kaže: 10/10	koji: 11/8

Smanjenje broja pojedinih grešaka je manje nego očekivano, ali može se ipak primetiti da reči, za koje se može specificirati više, pre svega osnovnih morfoloških kategorija, imaju manje instanci grešaka sa POS modelom.

Karakteristične greške u POS RNNLM testu sa svim morfološkim kategorijama u odnosu na referentni POS RNNLM test date su u tabeli 22. Ukupan broj zamena je najviše opao (za oko 5%), dok se broj brisanja i umetanja razlikuje nešto manje (za između 3,5% i 4%). Može se primetiti da se makar pojedine greške

koje se mogu vezati za morfologiju javljaju ređe, dok se za ostale tipove grešaka može reći da su se uglavnom takođe malo proredile, dok su se poneke ipak i pročestile. U umetanjima i brisanjima i dalje dominiraju veoma kratke, nepromenljive reči (uz *i* i *je*, od kojih su mnoge instance sigurno združene greške sa pojedinim zamenama). Očekivano je da bi svi ovi rezultati mogli da se poboljšaju ispravkama u korpusu za obuku (ali i test rečenicama, pogotovo u delu za radio emisije), odnosno u odgovarajućim procedurama u alatu AnTagger, kao i dopunama srpskog akcenatsko-morfološkog rečnika. Uporedni prikaz rezultata prepoznavanja za pojedine test rečenice, sa referentnim RNNLM i POS RNNLM, dati su u dodatku 5.



Slika 16. POS RNNLM rezultati po govornicima i delovima test baza

Muški govornici su prikazani tamnijim tačkama, a ženski govornici svetlijim. Prosečni WER rezultati su: kod audio knjiga, 3,39% za muškarce i 3,78% za žene; kod radio emisija, 9,96% za muškarce i 6,67% za žene; kod mobilne baze, 1,09% za muškarce i 0,56% za žene.

Na slici 16 su dati WER rezultati za svakog od test govornika pojedinačno, u sva tri dela test baze podataka. Može se primetiti da kod radio emisija postoje velike varijacije – stopa greške se kreće između 4% i čak 17% za različite govornike. Manje varijacije postoje kod audio knjiga (WER je između 1% i 8%, sa najvećom koncentracijom rezultata između 2% i 4%), dok su svi rezultati na mobilnoj bazi bolji od 2% WER. To je i očekivano, uzimajući u obzir prirodu svake od baza podataka, kao i slučajan izbor test govornika. Broj muških i ženskih test govornika je bio u

skladu sa njihovim ukupnim brojem u pojedinim bazama, pa je stoga i, na primer, kod radio emisija, broj ženskih govornika prilično mali u odnosu na broj muških. Ako posmatramo zasebno muške i ženske govornike, test rezultati su nešto bolji za muške kod audio knjiga, dok ženski imaju značajno bolji WER za druge dve test baze. I to je verovatno posledica slučajnog izbora govornika za test, pri čemu su recimo za radio emisije i mobilnu bazu izabrani muški govornici koji su problematičniji (lošiji snimci, više pozadinskog šuma, mucanja i atipičnih izgovora reči, i slično), a uz to imaju i u proseku nešto duže rečenice, dok je kod audio knjiga situacija obrnuta.

Tabela 23. Uporedni rezultati raznih POS RNNLM testova

Modifikacija	WER [%]	Modifikacija	WER [%]	Modifikacija	WER [%]
-	**6,08**	$V = L \approx 50k$	6,13	3+2 sloja po 512 neurona	6,12
$V \approx 5k,$ $L = 0$	6,19	$V \approx 75k,$ $L = 0$	6.12	3+2 sloja po 1536 neurona	6,14
$V = L \approx 5k$	6,18	$V = L \approx 75k$	6.14	4+3 sloja po 512 neurona	6,12
$V \approx 10k,$ $L = 0$	6,12	$V \approx 100k,$ $L = 0$	6,13	4+3 sloja po 1536 neurona	6,09
$V \approx 20k,$ $L = 0$	6,15	$V = L$ $\approx 100k$	6,11	5+4 slojeva	6,19
$V = L$ $\approx 20k$	6,10	2+1 sloj	6,19	-	-
$V \approx 50k,$ $L = 0$	6,13	3+2 sloja	6,10	-	-

Nijedan od rezultata nije bolji od osnovne varijante (podebljano). Veoma male razlike bi mogle opravdati korišćenje dosta jednostavnijih mrežnih arhitektura.

Nakon osnovnih testova, urađen je niz eksperimenata sa različitim podešavanjima RNN i korišćenih obeležja. Rezultati su sumirani u tabeli 23. Može se primetiti da, za razliku od referentnih RNNLM, nema nikakvih poboljšanja pri povećanju broja najčešćih reči i lema koje dobijaju sopstvena obeležja. Takođe, složenije mreže ne daju nikakav dalji napredak, a u praksi se može preporučiti upotreba RNN sa čak i manje slojeva i neurona po sloju u praksi, pošto je ovim

eksperimentima pokazano da takve mreže ne zaostaju gotovo uopšte u odnosu na referentnu strukturu. Uz to, imaju prednost po pitanju trajanja obuke i reskorovanja. Na primer, arhitektura sa tri TDNN sloja je spustila trajanje reskorovanja na oko 50 minuta (a obuka je trajala neznatno manje od referentnih 19 sati), dok se mreža sa 4 TDNN sloja, ali sa po 512 neurona po sloju, obučavala oko 14,5 sati, a reskorovanje je završeno za svega oko 25 minuta. Jednostavnije mreže, pogotovo varijante sa manje neurona po sloju (pošto je to istovremeno i *embedding* dimenzionalnost), imaju i znatno manje memorijske zahteve pri praktičnoj upotrebi.

EKSPERIMENTI SA POVEĆANIM REČNIKOM

U eksperimentima sa povećanim rečnikom ispitan je uticaj povećanja broja reči u rečniku na stopu greške prepoznavanja reči i perpleksivnost. Oni će takođe dati uvid u skalabilnost predloženog sistema.

Referentni rečnik (bez morfoloških sufiksa) je generisan na osnovu svih različitih reči iz transkripcija audio baza podataka, reči iz ostatka korpusa i neviđenih test reči, što znači da je u rečnik ukupno ušlo oko 461000 različitih reči. Sa druge strane, rečnik reči sa morfološkim sufiksima je formiran tako što su na skup svih različitih tagovanih reči iz transkripcija audio baza podataka i neviđenih tagovanih reči iz test skupa dodavane najčešće reči iz tagovanog korpusa sve dok se on nije dopunio do iste cifre od oko 461000 reči, radi koliko-toliko fer poređenja (nije potpuno fer jer u slučaju reči sa sufiksima nisu baš sve ušle u rečnik, pošto ih ukupno ima preko 600000).

Parametri trigram i RNNLM obuka bili su identični kao u prethodnim eksperimentima. U tabeli 24 dat je pregled rezultata obučenih trigram modela. Kao što se može videti, perpleksivnost se ponaša na isti način kao i u ranijim testovima. Ostvarene stope greške prepoznavanja reči bile su 8,58% za referentni trigram, odnosno 8,43% za POS trigram model. Ovi rezultati nisu mnogo lošiji od onih sa oko 250000 reči u rečniku, što nagoveštava da je opisani sistem veoma skalabilan. Što se tiče RNNLM, sa istom arhitekturom neuronske mreže kao i do sada (4 TDNN + 3 LSTMP sloja, *embedding* dimenzionalnost 1024, dimenzionalnost projekcionog podsloja 256+256), dobijen je WER od 6,45% za referentni slučaj, odnosno 6,24% za POS slučaj sa svim mogućim dodatnim morfološkim obeležjima. Ako se dobijeni

rezultati uporede sa rezultatima sa 250000 reči, vidi se relativno pogoršanje od samo 1,25% i 2,5%, što znači da je sistem zaista veoma skalabilan, što se od Kaldi-RNNLM procedure i očekivalo da ostvari. Sa morfološkim obeležjima dobijeno je relativno poboljšanje od oko 3,5% u odnosu na slučaj kada se ne koriste. RNNLM rezultati su dati u tabeli 25.

Tabela 24. Poređenje trigram modela jezika sa povećanim rečnikom

LM	broj unigrama	broj bigrama	broj trigrama	WER ukupno [%]	CER ukupno [%]	PPL
trigram ref. 461k	460850	1878710	538334	8,58	**2,50**	**646,3**
trigram POS 461k	460850	2003691	498754	**8,43**	2,54	756,8

Primetna je relativno mala razlika u WER u odnosu na slučaj sa oko 250000 reči (WER je tada iznosio 8,47%, odnosno 8,3%).

Tabela 25. Poređenje RNN modela jezika sa povećanim rečnikom

LM	WER ukupno [%]	CER ukupno [%]	PPL
RNNLM ref. 461k	6,45	1,95	**237,5**
RNNLM POS 461k	**6,24**	**1,92**	368,0

I sa RNNLM je razlika u odnosu na 250k slučaj relativno mala (tamo su stope greške iznosile 6,37%, odnosno 6,08%).

EKSPERIMENTI SA OOV REČIMA

Svi dosadašnji eksperimenti nisu eksplicitno imali OOV reči (reči u test skupu kojih nema u rečniku sistema), jer su sve reči iz test skupa, uključujući i one neviđene u korpusu za obuku, dodavane u rečnik. Naravno, test rečenice nisu učestvovale u samoj obuci, ali takav pristup ipak može da dovede do preoptimističnih rezultata. Zbog toga su ponovljene obuka i testovi i bez dodavanja neviđenih test reči u rečnik sistema.

Procedura formiranja referentnog rečnika se ovde sastojala od ubacivanja svih različitih reči iz transkripcija audio baza za obuku i zatim dopune tog spiska rečima iz tekstualnog korpusa za obuku prema učestanosti, sve dok se nije došlo do broja 250057 (kao u svim ranijim 250k eksperimentima). Isto je rađeno i za POS rečnik, samo na osnovu tagovanih tekstova. Uz ovakve rečnike, broj OOV reči u test skupu je iznosio 4141, odnosno 1,63% svih reči za referentni slučaj, to jest 6716, ili 2,64% za POS slučaj. Veća stopa OOV reči u drugom slučaju rezultat je većeg fonda reči sa sufiksima, odnosno, potencijalnog razdvajanja nekada istih reči na nekoliko različitih, a to istovremeno znači da se odgovarajući parovi rezultata ne mogu uporediti na potpuno fer način, tako da to treba imati u vidu pri analizi rezultata. Dobijene perpleksivnosti obučenih trigram modela jezika na test skupu iznosile su 556,5, odnosno 591,0, što je slično ponašanju u trigram testovima bez OOV reči (631,5, odnosno 718,1). Što se tiče RNNLM, obučeni modeli su dali perpleksivnosti od 205,6 (referentni model), odnosno 319,9 (morfološki model), što je ponovo uporedivo sa ponašanjem u RNNLM eksperimentu bez OOV reči (236,5 i 344,5).

Tabela 26. Poređenje rezultata bez OOV reči i sa OOV rečima

LM	WER ukupno [%]	CER ukupno [%]	PPL
trigram ref. / POS bez OOV	8,47 / 8,30	2,48 / 2,49	631,5 / 718,1
trigram ref. / POS sa OOV	10,51 / 10,91	2,93 / 3,08	556,5 / 591,0
RNNLM ref. / POS bez OOV	6,37 / 6,08	1,93 / 1,88	236,5 / 344,5
RNNLM ref. / POS sa OOV	8,41 / 8,79	2,40 / 2,49	205,6 / 319,9

Rezultati su, kao što je i očekivano, nešto lošiji u odnosu na slučaj bez OOV reči, ali i dalje dovoljno dobri za širok spektar ASR primena.

Rezultati prepoznavanja su, kao što se moglo pretpostaviti, nešto lošiji od ranijih. Pregled rezultata je dat u tabeli 26. Pogoršanje od po par procenata apsolutno i za trigrame i za RNNLM je očekivano s obzirom na to da postoji više hiljada neviđenih reči (to jest 1,6% i 2,6%, respektivno), međutim, rezultat se i dalje smatra

dobrim za razne ASR primene. Zaključak je da obučeni modeli funkcionišu dobro i u realnoj situaciji kada postoje OOV reči, a ako se koriste u sistemu gde je rečnik manje-više unapred poznat, može se očekivati i dalje poboljšanje performansi.

POGLAVLJE VII:

ZAKLJUČAK

U ovoj disertaciji predložen je novi metod ubacivanja morfoloških obeležja u modele srpskog jezika na bazi dubokih neuronskih mreža. Detaljno su objašnjeni način određivanja morfoloških kategorija i njihovih mogućih vrednosti, kao i njihovo korišćenje u Kaldi-RNNLM okruženju prilagođenom za srpski jezik, koje predstavlja jedno od najmodernijih metoda obuke jezičkih modela u svetu. Eksperimentisano je i na trigram modelima, koji proizvode inicijalne grafove dekodovanja tokom primene ovakvog ASR sistema, kao i na različitim varijantama RNN modela, uključujući naravno i referentni, bez dodatnih morfoloških obeležja. Ni opisani referentni model koji svaku reč predstavlja vektorom obeležja čiji elementi su iz skupa realnih brojeva, umesto klasičnom 1-od-*N* reprezentacijom, nije do sada primenjivan na srpski jezik. U dosadašnjim pristupima modelovanju srpskog jezika, morfološke informacije su se koristile ili u metodama aproksimacije verovatnoća *n*-grama kod *n*-gram modela jezika u slučaju da konkretan niz reči ne postoji u modelu, ili u podeli reči na klase u izlaznom sloju neuronskih mreža sa faktorisanim izlaznim slojem. U ovom radu se morfološke informacije direktno inkorporiraju u RNN model jezika, za koji se može reći da je varijanta faktorisanog modela jezika (FLM), koji sadrži obeležja niža od nivoa reči (slovne *n*-grame), kao i nekoliko augmentovanih obeležja (unigram log-verovatnoću reči u korpusu za obuku i dužinu reči), uz 1-od-*N* reprezentaciju za samo određen broj najčešćih reči. Detaljno je ispitan uticaj dodavanja *one-hot* podvektora obeležja za svaku morfološku kategoriju, kao i alternativni način ubacivanja istih morfoloških informacija u jezički model preko slovnih trigrama na sufiksima reči koji specificiraju vrednosti njihovih morfoloških kategorija, a takođe i uticaj ubacivanja dodatne 1-od-*N* reprezentacije za najčešće leme u korpusu za obuku. Hipoteza je bila da bi ove dodatne informacije mogle da doprinesu smanjenju broja pojava nekih uobičajenih grešaka ASR sistema, kao što su greške u padežu, gramatičkom rodu ili gramatičkom broju.

Eksperimenti su pokazali da ubacivanje navedenih morfoloških informacija u model jezika može da reši deo problema u automatskom prepoznavanju govora za visoko inflektivne jezike, u koje spada i srpski. Čak je i *n*-gram pristup imao nešto benefita od same zamene reči u rečniku i korpusu za obuku odgovarajućim tagovanim rečima sa morfološkim sufiksima. Referentni RNNLM je daleko nadmašio oba *n*-gram modela, kao i sve ranije rezultate prepoznavanja govora na velikim rečnicima za srpski jezik, dok je RNNLM sa morfološkim obeležjima još smanjio stopu greške prepoznavanja na datom test skupu. Među najčešćim greškama, osim generalno smanjenja njihovog broja po svim tipovima (i zamena, i umetanja, i brisanja), primetno je bilo i smanjeno prisustvo grešaka u padežima, rodovima, brojevima, i sličnih grešaka morfološke prirode.

Alat Kaldi-RNNLM ne zahteva dodatne eksterne alate i procedure, tako da je obučene modele jezika lako ubaciti u postojeće sisteme za prepoznavanje govora koji su već bazirani na Kaldi paketu alata. Korišćeni metod efikasnog reskorovanja trigram grafova dekodovanja RNN modelom se pokazao kao veoma dobar i efikasan, tako da ne degradira značajno performanse ASR sistema ni u pogledu njegove brzine dekodovanja, to jest odzivnosti. Uzimajući u obzir sve navedeno, ovakav sistem se može koristiti za prepoznavanje na vrlo velikim rečnicima, na primer za potrebe diktiranja.

Osim dalje optimizacije arhitekture neuronske mreže i primene novih tehnika koje će tek postati dostupne, ASR rezultati bi se pre svega mogli poboljšati pronalaženjem i ispravljanjem štamparskih i drugih grešaka u postojećim tekstualnim korpusima za obuku modela jezika, kao i doradama i ispravkama u alatu AnTagger za morfološko tagovanje tekstualnog korpusa i srpskom akcenatsko-morfološkom rečniku, pošto su primećene greške i nedoslednosti u njihovom ponašanju. Ovaj proces će sigurno biti dugačak i naporan, ali može biti od neprocenjivog značaja za dobijanje još kvalitetnijih modela jezika.

Takođe bi se trebalo pozabaviti izborom validacionog skupa, koji bi mogao da se izabere ili tako da ima jednak broj rečenica svih funkcionalnih stilova, ili samo rečenice iz željenog stila ili domena upotrebe. U dosadašnjim obukama, validacioni skup se uvek sastojao od uzastopnih rečenica iz novinskog dela korpusa.

Za konkretne primene, Kaldi-RNNLM procedura bi dalje mogla biti modifikovana tako da dozvoli dodeljivanje težinskih koeficijenata pojedinim delovima korpusa za obuku, da bi se na taj način u neku ruku omogućila adaptacija modela jezika na neki ciljani funkcionalni stil, ili ciljani način govora u željenoj aplikaciji.

Osim toga, treba se osvrnuti i na tretman vlastitih imenica. U srpskom jeziku imamo na hiljade različitih imena, prezimena, toponima i naziva organizacija, u svim svojim različitim oblicima, tako da nije realno očekivati da se svi oni precizno modeluju pre svega trigram, ali ni RNN modelom. Treba pronaći način da se pojedine klase vlastitih imenica modeluju zajedno (na primer, sva muška imena u nominativu jednine, i slično), a da se onda u toku praktične primene ili testiranja odluči koja je od instanci prepoznate klase u pitanju (na primer, na osnovu akustičkog poklapanja).

U svakom slučaju, mogućnosti primene obučenih morfoloških modela srpskog jezika su vrlo široke, jer bi mogli biti od izuzetnog značaja u raznim ASR aplikacijama koje zahtevaju rad na širokom spektru reči bez strogo definisanih pravila – na primer, u sistemima za diktiranje za različite domene, a uz to i rad u realnom vremenu. Konačno, kvalitetni jezički modeli koji se mogu uspešno primenjivati za razne zadatke generalno predstavljaju koristan metod očuvanja jezika, što je još jedan od doprinosa opisanih istraživanja u ovoj disertaciji.

DODATAK 1:

PRIMER TRIGRAM MODELA JEZIKA

Ovo je primer dela trigram modela jezika u standardnom ARPA-MIT formatu koji je obučen na tagovanom korpusu sa morfološkim sufiksima na rečima za sve moguće morfološke kategorije, sa ukupno 250057 reči u rečniku. Za svaki *n*-gram specificira se log-verovatnoća (levo) i opciono *back-off* cena (desno). <*s*> je uobičajena oznaka za BOS (početak rečenice), a <*/s*> za EOS simbol (kraj rečenice).

```
\data\
ngram 1=250059
ngram 2=2026862
ngram 3=525507

\1-grams:
-1.755192       </s>
-99             <s>                                     -0.6992929
-4.845538       abažur_imn_mur_zaj_nom_jed
-6.569226       abadžija_imn_mur_zaj_nom_jed            -0.0338946
-6.364002       abakus_imn_mur_zaj_nom_jed              -0.0141994
…
-5.452569       nečija_zam_zpr_nom_zer_jed              -0.1677872
…
-6.03567        zvučno_prl_pug                          -0.1107127
-5.287687       zvučnu_prd_poz_aku_zer_jed              -0.4060341
-4.454375       z_slo                                   -0.2604614

\2-grams:
-4.966195       <s> abdicirao_gla_nmf_npz_npv_gpr_mur_jed
-5.380256       <s> Abraham_imn_mur_vla_ime_nom_jed
-5.065751       <s> Abramovič_imn_mur_vla_pzm_nom_jed
…
-2.896101       nekoliko_brj_bnp zakona_imn_mur_aps_gen_mno
…
-3.161911       z_slo značili_gla_nmf_npz_npv_gpr_mur_mno
-1.762942       z_slo z_slo                             -0.1882376

\3-grams:
-1.067647       iz_pre_gen Žablja_prd_poz_nom_zer_jed zbog_pre_gen
…
-0.1581085      dozvoljena_prd_poz_nom_zer_jed žalba_imn_zer_zaj_nom_jed </s>
…
-1.233773       došli_gla_nmf_npz_npv_gpr_mur_mno do_pre_gen toga_zam_zim_gen
…
-1.808292       ali_vez_val mi_zam_lic_nom_mno_1lc još_prl_pui
…
-0.2475194      z_slo č_slo k_slo
\end\
```

DODATAK 2:

PRIMER REČNIKA IZGOVORA

Ovo je primer rečnika izgovora u kojem su reči tagovane morfološkim sufiksima za sve moguće morfološke kategorije. Rečnik je dat u standardnom Kaldi CMU (Carnegie Mellon University) formatu, u kom pojavni oblik reči prati niz fonema koji se nalaze u izgovoru te reči. Jedna reč može imati pridruženo i više od jednog izgovora. Fonemi vokala imaju oznake za akcenat (0-5), pri čemu je '0' oznaka za nenaglašene vokale. 'Yv' je oznaka za „šva" fonem, „SIL" za fonem tišine, a „!SIL" za reč eksplicitne tišine.

```
!SIL SIL
abažur_imn_mur_zaj_nom_jed A0 B A2 ZH U5 R
abadžija_imn_mur_zaj_nom_jed A4 B A0 DZ I0 J A0
abakus_imn_mur_zaj_nom_jed A0 B A3 K U0 S
…
automatsko_prd_poz_aku_srr_jed A0 U0 T O2 M A0 T S K O0
automatsko_prd_poz_nom_srr_jed A0 U0 T O2 M A0 T S K O0
automatsku_prd_poz_aku_zer_jed A0 U0 T O2 M A0 T S K U0
automatu_imn_mur_zaj_dat_jed A0 U0 T O0 M A3 T U0
automat_imn_mur_zaj_aku_jed A0 U0 T O2 M A5 T
automat_imn_mur_zaj_nom_jed A0 U0 T O2 M A5 T
…
nečemu_zam_zim_dat N E4 CH E0 M U0
nečem_zam_zim_dat N E4 CH E0 M
nečija_zam_zpr_aku_srr_mno N E4 CH I0 J A0
nečija_zam_zpr_nom_srr_mno N E4 CH I0 J A0
nečija_zam_zpr_nom_zer_jed N E4 CH I0 J A0
nečijeg_zam_zpr_gen_mur_jed N E4 CH I0 J E0 G
nečijeg_zam_zpr_gen_srr_jed N E4 CH I0 J E0 G
…
zvučno_prd_poz_aku_srr_jed Z V U1 CH N O0
zvučno_prd_poz_nom_srr_jed Z V U1 CH N O0
zvučno_prl_pug Z V U1 CH N O0
zvučnu_prd_poz_aku_zer_jed Z V U1 CH N U0
zv_akr Z E1 V E1
zv_akr Z I1 V I1
zv_akr Z Yv V Yv
z_slo Z Yv
```

DODATAK 3:

PARAMETRI KALDI-RNNLM OBUKE

Ovo su pojedini parametri Kaldi-RNNLM obuke koji se specificiraju u skriptovima za obuku, za osnovni RNN model jezika koji uključuje sva moguća morfološka obeležja.

```
--num-tdnn-layers=4
--embedding-dim=1024
--lstm-rpd=256
--lstm-nrpd=256
--num-word-samples=1024
--unigram-factor=200.0
--unk-word='<unk>'
--use-constant-feature=true
--special-words='<s>,</s>,<brk>,<unk>,!SIL'
--include-unigram-feature=true
--include-length-feature=true
--top-word-features=9987
--use-lemma-features=true # novi parametar (dodavanje najčešćih lema u vektor obeležja)
--use-pos-features=true # novi parametar (ubacivanje one-hot morfoloških obeležja)
--min-ngram-order=2
--max-ngram-order=4
--min-frequency=1.0e-04
--include-pos-ngram-features=true # novi parametar (ubacivanje POS slovnih trigrama među obeležja)
--epochs=30
--num-jobs-initial=1
--num-jobs-final=1
--dev-sents=20000
```

---

konfiguraciona datoteka za neuronsku mrežu:

```
input dim=1024 name=input
relu-renorm-layer name=tdnn1 dim=1024 input=Append(0, IfDefined(-1))
fast-lstmp-layer name=lstm1 cell-dim=1024 recurrent-projection-dim=256 non-recurrent-projection-dim=256
relu-renorm-layer name=tdnn2 dim=1024 input=Append(0, IfDefined(-2))
fast-lstmp-layer name=lstm2 cell-dim=1024 recurrent-projection-dim=256 non-recurrent-projection-dim=256
relu-renorm-layer name=tdnn3 dim=1024 input=Append(0, IfDefined(-2))
fast-lstmp-layer name=lstm3 cell-dim=1024 recurrent-projection-dim=256 non-recurrent-projection-dim=256
relu-renorm-layer name=tdnn4 dim=1024 input=Append(0, IfDefined(-1))
output-layer name=output include-log-softmax=false dim=1024
```

DODATAK 4:

PRIMERI REČENICA IZ POJEDINIH DELOVA KORPUSA

Novinski korpus:

- U tekstu koji je objavila Verska informativna agencija VIA, podseća se i na poslovni moral u Srbiji gde novi bogataši, živeći u raskoši, zaboravljaju na bližnje i na one koji su u prošlosti, sa mnogo manje imetka, osećali odgovornost za svoj rod i veru i bili veliki zadužbinari.

- Olimpijska rukometna reprezentacija Srbije pobedila je danas Sloveniju sa trideset četiri prema dvadeset osam u revijalnoj utakmici u beogradskoj hali Pionir, odigranoj povodom Dana rukometa.

- Šefovi poslaničkih grupa nisu danas postigli dogovor o Ustavnom zakonu zbog toga što neke stranke insistiraju da se zakonom striktno odredi datum izbora.

- Kako Blic saznaje iz izvora bliskih organizatorima ove manifestacije, šarmantna voditeljka jutarnjeg programa je najozbiljniji kandidat da vodi zvaničan deo programa koji će se iz Beograda emitovati u celu Evropu.

- Dva muškarca se vode kao nestala kod obala Švedske, nakon što je juče snažna oluja pogodila ovu zemlju, saopštio je danas Morski spasilački koordinacioni centar.

- Ovo je prvi predmet koji je Međunarodni sud prosledio Srbiji, piše u saopštenju.

Literarni korpus:

- Opšte je uverenje da sportisti nisu preterano obdareni pameću (što, inače, uopšte nije tačno) ali ne mogu da poverujem da je Nevil Strejndž kompletan idiot.

- Jednu ženu grli, a druga mu drži glavu na ramenu.

- Lice mu je bilo izbrazdano dubokim borama od umora i iscrpljenosti.

- A onda sam počeo da i sam primenjujem stari trik svemirskih letača: da povremeno, još tokom putovanja, udišem maramicu natopljenu karakterističnim mirisom planete na koju je trebalo da sletimo.

- Tom ih je kupio od čoveka koji je spasavao piliće iz fabričkih uzgajališta, tako da su do tada uvek živeli u malim kavezima, a sad su postali suviše agorafobični da bi izašli napolje.

- Restoran je bio zaista mali i izgledao bi pretrpano da stolovi nisu bili umešno raspoređeni, tako da su odavali utisak većeg prostora.

Naučni korpus:

- Pokazatelj dobro izbalansirane, pravilne ishrane je zadovoljavajuće stanje ishranjenosti organizma i poželjan telesni sastav.

- Popis iz te godine sprovela je državna vlast posle oslobađanja od Turaka, i to sa namerom da se utvrdi šta je i kome je narod plaćao dok je bio pod Turcima.

- Razlog tome je najčešće nepotpun urod žira, neravnomerno osemenjavanje površine, nemogućnost klijanja semena, zakorovljena površina, nepovoljni uslovi zemljišta, nekvalitetno i šturo seme, oštećen žir od strane insekata ili glodara i niz drugih faktora.

- Donja ivica matrice koristi se za unošenje objektivnih parametara za merenje, odnosno jedinice mera za karakteristike sistema, i to za konkurentski sistem i sopstveni sistem.

- U vezi kvalitativnih i kompetitivnih ograničenja koja treba u oblasti visokog obrazovanja rešavati, pomenuti autor smatra da se glavna pitanja u ovom domenu odnose na definisanje međunarodnih standarda koje bi univerziteti trebalo da ispunjavaju, osim domaćih koje ispunjavaju u okviru svojih država.

- Opšti zaključak je da kultura modifikuje intelektualni razvoj pojedinca kroz podsticanje onih vrednosti koje su značajne za problematiku tipičnu za određenu kulturu.

Administrativni korpus:

- Poslodavac je obavezan da predstavnicima sindikata koji su izabrani u više organe sindikata omogući odsustvovanje sa rada za učestvovanje u radu tih organa, uz priložen poziv.

- Ako je predmet dela iz stava prvog ovog člana vatreno oružje, municija, eksplozivne materije, ili sredstvo na bazi te materije, rasprskavajuće ili gasno oružje čija izrada, prodaja, nabavka, razmena ili držanje nije dozvoljeno građanima, učinilac će se kazniti zatvorom od šest meseci do pet godina i novčanom kaznom.

- Vlada ne može predložiti raspuštanje Narodne skupštine, ako je podnet predlog da joj se izglasa nepoverenje ili ako je postavila pitanje svoga poverenja.

- Saslušanju maloletnika u pripremnom postupku moraju prisustvovati javni tužilac, branilac za maloletnike i roditelj, usvojilac, odnosno staralac maloletnika.

- Presuda doneta u parnici o osporenom potraživanju deluje prema izvršnom dužniku i svim izvršnim poveriocima.

- Članovi organa društva i sa njima povezana lica u smislu ovog zakona, koji vrše funkciju nadzora u tom društvu i u sa njim povezanim društvima u smislu ovog zakona, ne mogu da budu članovi organa koji vode poslove upravljanja i zastupanja društva.

Naučno-popularni korpus:

- Prvi instrumenti za snimanje zvuka bili su uređaji napravljeni u naučne svrhe, zarad snimanja i kasnijeg proučavanja prirode zvučnih talasa.

- Iz toga proizilazi da ne postoje bitne razlike u zvuku na različitim sedištima u prostoriji, ali i da može postojati velika razlika u načinu kako zvuči neki muzički instrument kada se sluša uživo i kada se sluša signal registrovan mikrofonom.

- Operativni sistem predstavlja jezgro računarskog softvera, modul koji u jedinstvenu celinu povezuje sve komponente računarskog sistema: hardver, softver za upravljanje radom tog hardvera, kao i korisničke aplikacije koje efektivno koriste hardver.

- Nastajanje govora dešava se u čovekovom organu govora koji obuhvata pluća, dušnik, grkljan sa glasnicama, ždrelo sa resicom, usnu šupljinu i nosnu šupljinu.

- Zapazio sam da čak i oni ljudi koji tvrde da je sve predodređeno i da to ni na koji način ne možemo da promenimo dobro pogledaju levo i desno pre no što pređu put.

- Bila je to loša vest za pisce kosmičkih vesterna, ali veoma prijatna za nekolicinu nas koji smo u to vreme verovali u crne rupe: bio je to prvi pozitivan nalaz o postojanju neutronskih zvezda.

Razgovorni korpus:

- Imate li još nešto za dodati?

- Morate nam reći gde je uzorak.

- Neću se žrtvovati za nikoga.

- Humor je subjektivan.

- Vreme je da ti kažem istinu.

- On je suviše slab i za tron i za borbu.

DODATAK 5:

UTICAJ MORFOLOŠKIH OBELEŽJA NA PREPOZNAVANJE

Ovo su primeri prepoznavanja pojedinih test rečenica sa RNN modelima jezika, bez dodavanja morfoloških obeležja („ref“), odnosno sa svim mogućim dodatnim morfološkim obeležjima („POS“), uporedno. Razlike u rezultatu naglašene su crvenom bojom i podebljanim tekstom. Kod rezultata sa morfološkim obeležjima prepoznate reči su prikazane bez morfoloških sufiksa radi jasnoće.

ref: **otkud sad** tri četvrt otkuca pola četiri otkuca četvrt do četiri a još nije bilo ni traga od mog trkača

POS: **otkuca** tri četvrt otkuca pola četiri otkuca četvrt do četiri a još nije bilo ni traga od mog trkača

ref: još ni sam ne znam **Bogdan** šta o tome

POS: još ni sam ne znam **bog zna** šta o tome

ref: a ona ga je volela žarom koji mu se učini **životinjske** i bljutav

POS: a ona ga je volela žarom koji mu se učini **životinjski** i bljutav

ref: ja sam onda imao ideju da se **uključi u** neka napredna istraživanja

POS: ja sam onda imao ideju da se **uključe i** neka napredna istraživanja

ref: sutra i **svoje** deci mogu da pokažem i da se pohvalim

POS: sutra i **svojoj** deci mogu da pokažem i da se pohvalim

ref: pogrešio sam lozinku šta da **uradi**

POS: pogrešio sam lozinku šta da **uradim**

ref: ova pećina sadrži neke od najvećih **kristale** koji su ikada otkriveni na Zemlji

POS: ova pećina sadrži neke od najvećih **kristala** koji su ikada otkriveni na Zemlji

ref: nasmešila mi se uprkos tmurnom izrazu **koju** je videla na mom licu i poželela mi prijatno majsko jutro

POS: nasmešila mi se uprkos tmurnom izrazu **koji** je videla na mom licu i poželela mi prijatno majsko jutro

LITERATURA

Abadi M., Barham P., Chen J., Chen Z., Davis A., Dean J., Devin M., Ghemawat S., Irving G., Isard M., Kudlur M., Levenberg J., Monga R., Moore S., Murray D.G., Steiner B., Tucker P., Vasudevan V., Warden P., Wicke M., Yu Y., Zheng X., "TensorFlow: a system for large-scale machine learning," *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pp. 265-283, Savannah, GA, USA, November 2016

Allauzen C., Riley M., Schalkwyk J., Skut W., Mohri M., "OpenFst: a general and efficient weighted finite-state transducer library," *Proceedings of the 12th International Conference on Implementation and Application of Automata (CIAA)*, pp. 11-23, Prague, Czech Republic, July 2007

Arisoy E., Sainath T.N., Kingsbury B., Ramabhadran B., "Deep neural network language models," *Proceedings of the 12th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 20-28, Montréal, Canada, June 2012

Bahl L., Brown P., de Souza P., Mercer R., "Maximum mutual information estimation of hidden Markov model parameters for speech recognition," *Proceedings of the 43rd IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 11, pp. 49-52, New York, NY, USA, May 1986

Bakhtin A., Edrenkin I., "Faster RNNLM (HS/NCE) toolkit," *https://github.com/yandex/faster-rnnlm* (datum pristupa: 07.08.2019)

Bengio Y., Simard P., Frasconi P., "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157-166, 1994

Bengio Y., Ducharme R., Vincent P., Jauvin C., "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, pp. 1137-1155, 2003

Bhanuprasad K., Svenson D., "Errgrams – a way to improving ASR for highly inflective Dravidian languages," *Proceedings of the 3rd International Joint Conference on Natural Language Processing (IJCNLP)*, pp. 805-810, Hyderabad, India, January 2008

Bodén M., "A guide to recurrent neural networks and backpropagation," *Tech Rep. T2002:03*, Swedish Institute of Computer Science, 2002

Bogert B.P., Healy M.J.R., Tukey J.W., "The quefrency analysis of time series for echoes: cepstrum, pseudo autocovariance, cross-cepstrum and saphe cracking," *Proceedings of the Symposium on Time Series Analysis*, chapter 15, pp. 209-243, 1963

Bottou L., "Large-scale machine learning with stochastic gradient descent," *Proceedings of the 19th International Conference on Computational Statistics (COMPSTAT)*, pp. 177-187, Paris, France, August 2010

Bourlard H., Morgan N., "*Connectionist speech recognition: a hybrid approach*," Kluwer international series in engineering & computer science, vol. 247, Springer Science+Business Media, New York, NY, USA, 1994

Brown M., de Souza P., Mercer R., della Pietra V., Lai J., "Class-based *n*-gram models of natural language," *Computational Linguistics*, vol. 18, no. 4, pp. 467-479, 1992

Chen S., Beeferman D., Rosenfeld R., "Evaluation metrics for language models," *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pp. 275-280, 1998

Chen S., Goodman J., "An empirical study of smoothing techniques for language modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359-394, 1999

Chen X., Liu X., Qian Y., Gales M.J.F., Woodland P.C., "Recurrent neural network language model training with noise contrastive estimation for speech recognition," *Proceedings of the 40th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5411-5415, Brisbane, Australia, April 2015a

Chen X., Liu X., Qian Y., Gales M.J.F., Woodland P.C., "CUED-RNNLM – an open-source toolkit for efficient training and evaluation of recurrent neural network language models," *Proceedings of the 40th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6000-6004, Brisbane, Australia, April 2015b

Chen X., Liu X., Gales M.J.F., Woodland P.C., "Improving the training and evaluation efficiency of recurrent neural network language models", *Proceedings of the 40th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5401-5405, Brisbane, Australia, April 2015c

Dahl G.E., Sainath T.N., Hinton G.E., "Improving deep neural networks for LVCSR using rectified linear units and dropout," *Proceedings of the 38th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8609-8613, Vancouver, Canada, May 2013

Das A., Jena M.R., Barik K.K., "Mel-frequency cepstral coefficients (MFCC) - a novel method for speaker recognition," *Digital Technologies*, vol. 1, no. 1, pp. 1-3, 2014

Dehak N., Kenny P., Dehak R., Dumouchel P., Ouellet P., "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 4, pp. 788-798, 2011

Delić V., Sečujski M., Jakovljević N., Janev M., Obradović R., Pekar D., "Speech technologies for Serbian and kindred South Slavic languages," *Advances in Speech Recognition*, pp. 141-164, 2010

Deng L., "Integrated optimization of dynamic feature parameters for hidden Markov modeling of speech," *IEEE Signal Processing Letters*, vol. 1, no. 4, pp. 66-69, 1994

Deville J-C., Tille Y., "Unequal probability sampling without replacement through a splitting method," *Biometrika*, vol. 85, no. 1, pp. 89-101, 1998

Duh K., Kirchhoff K., "Automatic learning of language model structure," *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pp. 148-154, Geneva, Switzerland, August 2004

Elman J., "Finding Structure in Time," *Cognitive Science*, vol. 14, pp. 179-211, 1990

Gauvain J., Lee C., "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 2, no. 2, pp. 291-299, 1994

Goodman J.T., "A bit of progress in language modeling: extended version," *Tech. Rep. MSR-TR-2001-72*, Microsoft Research, Redmond, WA, USA, 2001

Hermansky H., "Perceptual linear predictive (PLP) analysis of speech," *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738-1752, 1990

Hochreiter S., Schmidhuber J., "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997

Hopcroft J.E., Ullman J.D., "*Introduction to automata theory, languages and computation*," chapter 9, Addison-Wesley, Boston, MA, USA, 1979

Jakovljević N., Popović B., Janev M., Pekar D., "The impact of the pitch on the estimation of MFCC," *Proceedings of the 19th Telecommunications Forum (TELFOR)*, pp. 651-654, Belgrade, Serbia, November 2011

Jardino M., "Multilingual stochastic *n*-gram class language models," *Proceedings of the 21st International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 161-163, Atlanta, GA, USA, May 1996

Jelinek E., Mercer R., Bahl L., Baker J., "Perplexity – a measure of the difficulty of speech recognition tasks," *Proceedings of the 94th Meeting of the Acoustical Society of America*, vol. 62, no. 1, pp. S63, Miami Beach, FL, USA, December 1977

Kirchhoff K., Yang M., "Improved language modeling for statistical machine translation," *Proceedings of the ACL Workshop on Building and Using Parallel Text: Data-Driven Machine Translation and Beyond*, pp. 125-128, Ann Arbor, MI, USA, June 2005

Kirchhoff K., Vergyri D., Bilmes J., Duh K., Stolcke A., "Morphology-based language modeling for conversational Arabic speech recognition," *Computer Speech & Language*, vol. 20, no. 4, pp. 589-608, 2006

Klakow D., Peters J., "Testing the correlation of word error rate and perplexity," *Speech Communication*, vol. 38, no. 1-2, pp. 19-28, 2002

Kneser R., Ney H., "Improved backing-off for *M*-gram language modeling," *Proceedings of the 20th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 181-184, Detroit, MI, USA, May 1995

Ko T., Peddinti V., Povey D., Seltzer M.L., Khudanpur S., "A study on data augmentation of reverberant speech for robust speech recognition," *Proceedings of the 42nd International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5220-5224, New Orleans, LA, USA, March 2017

Kwon O-W., Park J., "Korean large vocabulary continuous speech recognition with morpheme-based recognition units," *Speech Communication*, vol. 39, no. 3-4, pp. 287-300, 2003

Leggetter C.J., Woodland P.C., "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," *Computer Speech and Language*, vol. 9, pp. 171-186, 1995

Liu X., Chen X., Wang Y., Gales M.J.F., Woodland P.C., "Two efficient lattice rescoring methods using recurrent neural network language models," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 8, pp. 1438-1449, 2016

Müller T., Schütze H., Schmid H., "A comparative investigation of morphological language modeling for the languages of the European Union," *Proceedings of the 12th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 386-395, Montréal, Canada, June 2012

Manning C.D., Schütze H., "*Foundations of statistical natural language processing*," MIT Press, Cambridge, MA, USA, 1999

Martens J., Sutskever I., "Learning recurrent neural networks with Hessian-free optimization," *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pp. 1033-1040, Bellevue, WA, USA, June 2011

Matthews A., Neubig G., Dyer C., "Using morphological knowledge in open-vocabulary neural language models," *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, vol. 1, pp. 1435-1445, New Orleans, LA, USA, June 2018

McCracken D.D., Reilly E.D., "Backus-Naur form," *Encyclopedia of Computer Science*, 4th edition, pp. 129-131, Wiley, Hoboken, NJ, USA, 2003

Mikolov T., Karafiat M., Burget L., Černocký J., Khudanpur S., "Recurrent neural network based language model," *Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1045-1048, Makuhari, Japan, September 2010

Mikolov T., Deoras A., Povey D., Burget L., Černocký J., "Strategies for training large scale neural network language models," *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 196-201, Waikoloa, HI, USA, December 2011a

Mikolov T., Kombrink S., Burget L., Černocký J., Khudanpur S., "Extensions of recurrent neural network based language model," *Proceedings of the 36th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5528-5531, Prague, Czech Republic, May 2011b

Mikolov T., Kombrink S., Deoras A., Burget L., Černocký J., "RNNLM – recurrent neural network language modeling toolkit," *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU) - Demo Session*, Waikoloa, HI, USA, December 2011c

Mikolov T., "Statistical language models based on neural networks," *Ph.D. thesis*, Brno University of Technology, Czech Republic, 2012

Mikolov T., Sutskever I., Deoras A., Le H-S., Kombrink S., Černocký J., "Subword language modeling with neural networks," *Tech. Rep.*, Brno University of Technology: Faculty of Information Technology, Czech Republic, 2012

Mikolov T., Chen K., Corrado G., Dean J., "Efficient estimation of word representations in vector space," *Proceedings of the International Conference on Learning Representations (ICLR)*, pp. 1-12, Scottsdale, AZ, USA, May 2013

Mohri M., "Minimization algorithms for sequential transducers", *Theoretical Computer Science*, vol. 234, no. 1-2, pp. 177–201, 2000

Mohri M., Pereira F., Riley M., "Speech recognition with weighted finite-state transducers," *Computer Speech & Language*, vol. 16, no. 1, pp. 69-88, 2002

Oparin I., Sundermeyer M., Ney H., Gauvain J-L., "Performance analysis of neural networks in combination with *n*-gram language models," *Proceedings of the 37th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5005-5008, Kyoto, Japan, March 2012

Ostrogonac S., Mišković D., Sečujski M., Pekar D., Delić V., "A language model for highly inflective non-agglutinative languages," *Proceedings of the 10th International Symposium on Intelligent Systems and Informatics (SISY)*, pp. 177-181, Subotica, Serbia, September 2012

Ostrogonac S., Pakoci E., Sečujski M., Mišković D., "Morphology-based vs unsupervised word clustering for training language models for Serbian," *Acta Polytechnica Hungarica*, Special Issue on Cognitive Infocommunications, vol. 16, no. 2, pp. 183-197, 2019

Ostrogonac S., "Automatic detection and correction of semantic errors in texts in Serbian," *Proceedings of the 5th International Congress of Applied Linguistics Today – New Tendencies in Theory and Practice – Primenjena Lingvistika*, vol. 17, pp. 265-278, Novi Sad, Serbia, November 2015

Ostrogonac S., "Modeli srpskog jezika i njihova primena u govornim i jezičkim tehnologijama (Models of the Serbian language and their application in speech and language technologies)," *Ph.D. thesis*, University of Novi Sad, Serbia, 2018

Pakoci E., Popović B., Pekar D., "Language model optimization for a deep neural network based speech recognition system for Serbian," *Proceedings of the 19th International Conference on Speech and Computer (SPECOM) – Lecture Notes in Artificial Intelligence*, vol. 10458, pp. 483-492, Hatfield, UK, September 2017

Pakoci E., Popović B., Pekar D., "Improvements in Serbian speech recognition using sequence-trained deep neural networks," *SPIIRAS Proceedings*, vol. 3, no. 58, pp. 53-76, 2018

Pakoci E., Popović B., Pekar D., "Using morphological data in language modeling for Serbian large vocabulary speech recognition," *Computational Intelligence and Neuroscience*, Special Issue on Advanced Signal Processing and Adaptive Learning Methods, vol. 2019, pp. 1-8, 2019

Peddinti V., Povey D., Khudanpur S., "A time delay neural network architecture for efficient modeling of long temporal contexts," *Proceedings of the 16th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 3214-3218, Dresden, Germany, September 2015

Popović B., Pakoci E., Ostrogonac S., Pekar D., "Large vocabulary continuous speech recognition for Serbian using the Kaldi toolkit," *Proceedings of the 10th Conference on Digital Speech and Image Processing (DOGS)*, pp. 31-34, Novi Sad, Serbia, October 2014

Popović B., Ostrogonac S., Pakoci E., Jakovljević N., Delić V., "Deep neural network based continuous speech recognition for Serbian using the Kaldi toolkit," *Proceedings of the 17th International Conference on Speech and Computer (SPECOM) – Lecture Notes in Artificial Intelligence*, vol. 9319, pp. 186-192, Athens, Greece, September 2015a

Popović B., Pakoci E., Jakovljević N., Kočiš G., Pekar D., "Voice assistant application for the Serbian language," *Proceedings of the 23rd Telecommunications Forum (TELFOR)*, pp. 858-861, Belgrade, Serbia, November 2015b

Popović B., Pakoci E., Pekar D., "A comparison of language model training techniques in a continuous speech recognition system for Serbian," *Proceedings of the 20th International Conference on Speech and Computer (SPECOM) – Lecture Notes in Artificial Intelligence*, vol. 11096, pp. 522-531, Leipzig, Germany, September 2018

Povey D., Kuo H-K.J., Soltau H., "Fast speaker adaptive training for speech recognition," *Proceedings of the 9th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1245-1248, Brisbane, Australia, September 2008

Povey D., Ghoshal A., Boulianne G., Burget L., Glembek O., Goel N., Hannemann M., Motlíček P., Qian Y., Schwarz P., Silovský J., Stemmer G., Veselý K., "The Kaldi speech recognition toolkit," *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 1-4, Waikoloa, HI, USA, December 2011

Povey D., Peddinti V., Galvez D., Ghahrmani P., Manohar V., Na X., Wang Y., Khudanpur S., "Purely sequence-trained neural networks for ASR based on lattice-free MMI," *Proceedings of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2751-2755, San Francisco, CA, USA, 2016

Povey D., "Discriminative training for large vocabulary speech recognition," *Ph.D. thesis*, University of Cambridge, UK, 2003

Press O., Wolf L., "Using the output embedding to improve language models," *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, vol. 2, pp. 157-163, Valencia, Spain, April 2017

Qin L., "Learning out-of-vocabulary words in automatic speech recognition," *Ph.D. thesis*, Carnegie Mellon University, Pittsburgh, PA, USA, 2013

Rabiner L.R., Juang B.H., Levinson S.E., Sondhi N.M., "Recognition of isolated digits using hidden Markov models with continuous mixture densities," *AT&T Technical Journal*, vol. 64, no. 6, pp. 1211-1234, 1985

Rabiner L.R., "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, 1989

Rissanen J., "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465-658, 1978

Rosenfeld R., "Two decades of statistical language modeling: where do we go from here?," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1270-1278, 2000

Rozenberg G., Salomaa A., "*Handbook of formal languages: volume I - word, language, grammar*," Springer, Berlin, Germany, 1997

Rumelhart D.E., Hinton G.E., Williams R.J., "Learning internal representations by back-propagating errors," *Nature*, vol. 323, pp. 533-536, 1986

Sak H., Saraçlar M, Güngör T., "Morphology-based and sub-word language modeling for Turkish speech recognition," *Proceedings of the 35th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5402-5405, Dallas, TX, USA, March 2010

Sak H., Senior A., Beaufays F., "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *Computer Science*, vol. 1128, pp. 338-342, 2014

Saon G., Padmanabhan M., Gopinath R., Chen S., "Maximum likelihood discriminant feature spaces," *Proceedings of the 25th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1747-1750, Istanbul, Turkey, June 2000

Sarikaya R., Afify M., Deng Y., Erdogan H., Gao Y., "Joint morphological-lexical language modeling for processing morphologically rich languages with application to dialectal Arabic," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 7, pp. 1330-1339, 2008

Schwenk H., Gauvain J-L., "Training neural network language models on very large corpora," *Proceedings of the Joint Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP)*, pp. 201-208, Vancouver, Canada, October 2005

Sečujski M., "Automatic part-of-speech tagging in Serbian," *Ph.D. thesis*, University of Novi Sad, Serbia, 2009

Shannon C.E., "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 4, pp. 623-656, 1948

Stolcke A., Zheng J., Wang W., Abrash V., "SRILM at sixteen: update and outlook," *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pp. 5-9, Waikoloa, HI, USA, December 2011

Strube H.W., "Linear prediction on a warped frequency scale," *Journal of the Acoustical Society of America*, vol. 68, no. 4, pp. 1071-1076, 1998

Sundermeyer M., Schlüter R., Ney H., "LSTM neural networks for language modeling," *Proceedings of the 13th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 194-197, Portland, OR, USA, September 2012

Sutskever I., Vinyals O., Le Q.V., "Sequence to sequence learning with neural networks," *Advances in Neural Information Processing Systems*, vol. 27, pp. 3104-3112, 2014

Suzić S., Ostrogonac S., Pakoci E., Bojanić M., "Building a speech repository for a Serbian LVCSR system," *Telfor Journal*, vol. 6, no. 2, pp. 109-114, 2014

Verteletskaya E., Šimák B., "Performance evaluation of pitch detection algorithms," *Access Server (Online Journal)*, vol. 2009, pp. 1, 2009

Viikki O., Laurila K., "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communication*, vol. 25, no. 1-3, pp. 133-147, 1998

Waibel A., "Modular construction of time-delay neural networks for speech recognition," *Neural Computation*, vol. 1, no. 1, pp. 39-46, 1989

Walters P., "*An introduction to ergodic theory*," Graduate texts in mathematics, vol. 79, chapter 1, Springer Science+Business Media, New York, NY, USA, 1982

Weizenbaum J., "ELIZA – a computer program for the study of natural language communication between man and machine," *Communications of the Association for Computing Machinery*, vol. 9, no. 1, pp. 36-45, 1966

Welling L., Kanthak S., Ney H., "Improved methods for vocal tract normalization," *Proceedings of the 24th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 761-764, Phoenix, AZ, USA, March 1999

Whittaker E.W.D., Woodland P.C., "Efficient class-based language modeling for very large vocabularies," *Proceedings of the 26th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 545-548, Salt Lake City, UT, USA, May 2001

Woodland P.C., Gales M.J.F., Pye D., "Improving environmental robustness in large vocabulary speech recognition," *Proceedings of the 21st International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 65-68, Atlanta, GA, USA, May 1996

Wu Y., Yamamoto H., Lu X., Dixon P.R., Matsuda S., Hori C., Kashioka H., "Factored recurrent neural network language model in TED lecture transcription," *Proceedings of the 9th International Workshop on Spoken Language Translation (IWSLT)*, pp. 222-228, Hong Kong, China, December 2012

Xu W., Rudnicky A., "Can artificial neural networks learn language models?," *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP)*, pp. 202-205, Beijing, China, October 2000

Xu H., Chen T., Gao D., Wang Y., Li K., Goel N., Carmiel Y., Povey D., Khudanpur S., "A pruned RNNLM lattice rescoring algorithm for automatic speech recognition," *Proceedings of the 43rd International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5929-5933, Calgary, Canada, April 2018a

Xu H., Li K., Wang Y., Wang J., Kang S., Chen X., Povey D., Khudanpur S., "Neural network language modeling with letter-based features and importance sampling," *Proceedings of the 43rd International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6109-6113, Calgary, Canada, April 2018b

Young S.J., Russell N.H., Thornton J.H.S., "Token passing: a simple conceptual model for connected speech recognition systems," *Tech. Rep. F_INFENG/TR38*, University of Cambridge: Department of Engineering, UK, 1989

Young S.J., Odell J.J., Woodland P.C., "Tree-based state tying for high accuracy acoustic modelling," *Proceedings of the ARPA Human Language Technology Workshop (HLT)*, pp. 307-312, Plainsboro, NJ, USA, March 1994