Zbornik radova Fakulteta tehničkih nauka, Novi Sad

UDK: 681.32:621.391
DOI: https://doi.org/10.24867/21BE20Stanojev

AUTOMATSKA REKONSTRUKCIJA DIJAKRITIČKIH ZNAKOVA U TEKSTOVIMA NA SRPSKOM JEZIKU PRIMENOM MAŠINSKOG UČENJA

AUTOMATIC RECONSTRUCTION OF DIACRITICAL MARKS IN TEXTS IN THE SERBIAN LANGUAGE USING MACHINE LEARNING

Vuk Stanojev, *Fakultet tehničkih nauka, Novi Sad*

Oblast – OBRADA SIGNALA

Kratak sadržaj – R*ad sadrži opis problema rekonstrukcije dijakritičkih znakova u tekstovima na srpskom jeziku. Problem je predstavljen kao klasifikacioni i predstavljene su tri metode mašinskog učenja pomoću kojih su dobijeni rezultati: neuronske mreže sa propagacijom unapred, mreže sa dugom kratkotrajnom memorijom i konvolucione neuronske mreže. Metode su poređene po merama klasifikatora i za najbolji od klasifikatora dat je prikaz rezultata, odnosno primeri kako je redijakritizacija izvršena.*

Ključne reči: *Rekonstrukcija dijakritičkih znakova, Mašinsko učenje, Klasifikacija*

Abstract – *This paper contains an overview of problems that follow reconstruction of diacritical marks in text in Serbian language. The problem is shown as a classification problem and three methods of machine learning are proposed with whom the results are obtained: Feed Forward Neural Networks, Short-Term Memory Neural Networks and Convolutional Neural Networks. Methods are compared based on classification metrics and for the best method, results of diacritic restoration is shown.*

Keywords: *Reconstruction of diacritical marks, Machine learning, Classification*

1. UVOD

Učesnici u elektronskoj pisanoj komunikaciji na srpskom jeziku latiničnim pismom često zamenjuju slova sa dijakritičkim znakom odgovarajućim ekvivalentom bez dijakritičkog znaka. Oni prilikom čitanja ovih tekstova nemaju problem da razumeju šta je napisano, ali prilikom računarske analize teksta, računaru su reči kod kojih su izostavljeni dijakritički znaci nepoznate. Ukoliko postoje dve reči koje imaju ortografski zapis koji se rezlikuje samo u pogledu prisustva ili odsustva dijakritičkih znakova, to za računar može predstavljati izvor dvosmislenosti [1]. Zbog toga je pre bilo koje obrade teksta na računaru neophodno izvršiti rekonstrukciju dijakritičkih znakova.

Iako danas postoje tastature sa slovima koja poseduju dijakritičke znake, ljudi ih i dalje veoma često izostavljaju. Neki razlozi zbog kojih smatramo da se to dešava su sledeći:

- Istorijski gledano, prve tastature koje su se koristile su bile prilagođene engleskom jeziku. Samim tim, nisu imale mogućnost upotrebe slova sa dijakritičkim znacima. Zbog toga su korisnici elektronskih uređaja kao što su računari i mobilni telefoni navikli da ne koriste deo tastature na kojem se danas nalaze slova sa dijakritičkim znacima.

- Nekada je kucanje reči sa dijakritičkim znacima znatno sporije nego da se ona izostave. Na primer, ukoliko je mobilni telefon podešen za rad na srpskom jeziku, tastatura na srpskom jeziku može a i ne mora u svom rasporedu da poseduje slova sa dijakritičkim znacima. Ukoliko ih ne poseduje, potrebno je zadržati prst na odgovarajućem ekvivalentu slova bez dijakritičkog znaka kako bi se pojavila opcija za kucanje slova sa dijakritičkim znakom. Ovakav dizajn tastature usporava kucanje.

- Ukoliko se šalje standardna SMS poruka u njoj je dozvoljeno 160 standardnih karaktera engleskog alfabeta, bez specijalnih karaktera i grafičkih elemenata. Ukoliko se u SMS poruci koriste latinična slova sa dijakritičkim elementom, broj raspoloživih karaktera u okviru jedne SMS poruke smanjuje se na 70 karaktera. Ljudi izbegavaju slova sa dijakritičkim elementima kako bi smanjili trošak telefonskih usluga. Mobilna komunikacija je danas jedan od najzastupljenijih vidova komunikacije. Količina teksta koja se generiše na mobilnim telefonima izuzetno je velika i može doprineti formiranju novih baza tekstova nad kojima je neophodno izvršiti dijakritizaciju.

1.1. Rekonstrukcija dijakritičkih znakova

Dijakritički znaci u srpskom jeziku koriste se za razlikovanje glasova i za razlikovanje slova kada homograf postoji u formi označenog i neoznačenog slova. U srpskom jeziku na latiničnom pismu postoji pet slova sa dijakritičkim znakom: Š, Č, Ć, Ž i Đ. Prilikom pisanja, ona se zamenjuju svojim odgovarajućim slovima bez dijakritičkog znaka, respektivno: S, C, C, Z i Dj. Svako slovo se, dakle, zamenjuje jednim slovom, osim u slučaju slova Đ, koje se zamenjuje sa dva slova. Iako i kombinovano slovo Dž poseduje dijakritički znak, te se u pisanju bez dijakritičkih znakova zamenjuje sa Dz, dijakritizaciju digrafa Dz nije potrebno posmatrati posebno, već se ovaj slučaj može podvesti pod dijakrizitaciju slova Ž.

Problem rekonstrukcije dijakritičkih znakova prisutan je u raznim jezicima i veoma je jezički zavisan. Samim tim, za razne jazike su dati metodi koji rešavaju ovaj problem.

NAPOMENA:
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Milan Sečujski, red. prof.

Jedna od najjednostavnijih metoda bi bila da se formira rečnik sa učestalostima pojavljivanja reči i da se svaka reč za koju postoji dvoumica oko toga da li u njoj postoje dijakritički znaci ili ne, zameni sa verovatnijom [2]. Za sprski, slovenački i hrvatski jezik predložen je model za rekonstrukciju dijakritičkih znakova koji posmatra najverovatniju reprezentaciju slova u datoj reči i kombinuje je sa verovatnoćom da se data reč nađe u datom kontekstu [1]. Najbolji rezultati za rekonstrukciju dijakritičkih elemenata u srpskom, slovenačkom i hrvatskom jeziku su dobijeni korišćenjem bidirekcione rekurzivne neuronske mreže sa tačnošću od 99.7% [2]. Za rekonstrukciju dijakritičkih elemenata na arapskom jeziku predloženi su modeli bazirani na neuronskim mrežama sa propagacijom unapred [3]. Za rumunski jezik korištene su bidirekcione mreže sa dugom kratkotrajnom memorijom [4].

U nastavku rada biće prikazana baza podataka na kojoj će se obučavati klasifikatori formirani u okviru ovog istraživanja, poredиće se performanse klasifikatora i biće prikazani rezultati rekonstrukcije dijakritičkih znakova.

2. BAZA PODATAKA

Baza podataka ukupno ima oko 2.4 miliona reči na srpskom jeziku, napisanih latiničnim pismom. Reči su uzimane iz tekstova koji su pisani različitim stilovima: novinarskom, naučnopopularnom, naučnom i literarnom stilu. Prilikom predobrade baze, svi nizovi znakova koji ne predstavljaju slova zamenjeni su po jednim razmakom. Takođe, sva velika slova su pretvorena u mala.

Rekonstrukcija dijakritičkih znakova je vršena na nivou slova. Posmatraju se isključivo slova koja mogu posedovati dijakritički znak Š, Č, Ć, Ž i Đ , kao i njihovi parovi bez dijakritičkog znaka S, C, C, Z i DJ. Za slova S, Z i DJ dovoljno je odrediti da li bi na mestu pojavljivanja datih slova trebalo da stoji slovo sa dijakritičkim znakom ili ne, dok za C treba odrediti i koji tačno znak treba upotrebiti. Za slova za koja se vrši redijakritizacija, nad delom baze određen je broj pojavljivanja datih slova (tabela 2.1). Iz tabele se vidi da postoje izuzetno velike razlike u pogledu zastupljenosti pojedinih slova u tekstovima na srpskom jeziku. Iako je u standardnim problemima klasifikacije uobičajeno da se u ovakvim slučajevima radi ujednačavanje zastupljenosti pojedinih klasa, odnosno slova, ovde to nije rađeno jer je namera bila da se očuvaju stvarne statistike u tekstu.

2.1. Modelovanje podataka

Baza podataka koja je opisana na početku ove glave nije prosleđivana klasifikatoru u originalnom obliku. Način na koji je baza podataka predobrađena je sledeći:

1. Identifikovani su svi indeksi slova iz tabele 2.1, odnosno, pozicije na kojima se tražena slova u bazi nalaze. Kako je naglašeno da dijakritizacija digrafa DZ nije neophodna, za par slova DZ i DŽ nisu identifikovani indeksi
2. U odnosu na pronađeni indeks, uzimano je deset karaktera pre i deset karaktera nakon datog karaktera. Na ovaj način formiran je prozor koji obuhvata 21 karakter. Dati niz od 21 karaktera predstavlja jedan uzorak koji se prosleđuje klasifikatoru. Kako je moguće da se slovo sa dijakritičkim znakom nađe u neposrednoj blizini samog početka ili samog kraj teksta, tekst je dopunjen sa odgovarajućim brojem praznina.
3. Svakom nizu, u zavisnosti od centralnog karaktera, dodeljena je odgovarajuća klasna labela pomoću koje će se vršiti klasifikacija. Labele označavaju klasnu pripadnost tog karaktera, odnosno u našem slučaju, označavaju koje je zapravo slovo reprezentovano pomoću datog niza.
4. Nakon dodeljivanja klasnih labela uzorcima, sa slova Š, Č, Ć i Ž koja su se našla u nizu od 21 karaktera, izbrisani su dijakritički znaci, te su ona pretvorena u S, C, C i Z respektivno. Za slovo Đ uveden je obrnut pristup. Odnosno, svi parovi slova DJ pretvoreni su u slovo Đ (zapravo, mogli su biti pretvoreni u bilo koji specijalni karakter koji će čuvati informaciju da to može biti Đ ili DJ). Ovakvom reprezentacijom, klasifikacija slova Š i S, Ž i Z, Đ i DJ predstavlja binarnu klasifikaciju, a za slova C, Č i Ć imamo problem klasifikacije u tri klase.
5. Svaki niz od 21 karaktera koji predstavlja pojedinačni uzorak transformisan je u matricu veličine 21x28. Svaka vrsta odgovara jednom karakteru, dok kolone čine njegovu „one-hot vector“ reprezentaciju [5]. Alfabet za “one-hot vector” kodovanje je: 'sczđdabefghijklmnopqrtu vwxy '. Kako su slova Š, Č, Ć i Ž prethodno konvertovana u svoje parove bez dijakritičkih elemenata, ona nemaju svoju odgovarajuću „one-hot vector“ reprezentaciju. Iako srpski jezik u svojoj azbuci ne poseduje slova Q, W, X, Y, u tekstovima u bazi javljali su se i strani izrazi i nazivi koji su sadržali ta slova.

TABELA 2.1. BROJ POJAVLJIVANJA SLOVA ZA KOJE SE VRŠI REDIJAKRITIZACIJA

Slova	Broj pojavljivanja slova	Pojavljivanje slova [%]
S	75452	4.23340
Š	13900	0.75086
C	14297	0.79117
Ć	10722	0.62450
Č	12870	0.72110
Đ	4003	0.22296
DJ	12	0.00067
Z	26657	1.53183
Ž	7478	0.41480
DŽ	557	0.03135
DZ	58	0.00326

3. MODELI

Duboko učenje je klasa algoritama mašinskog učenja koji koristi slojeve za progresivno izdvajanje obeležja iz sirovih ulaznih podataka. Slojevi zapravo predstavljaju modele veštačkih neuronskih mreža. Veštačke neuronske mreže predstavljaju složene računarske sisteme nastale povezivanjem veštačkih neurona, inspirisane načinom funkcionisanja neurona u ljudskom mozgu [6].

Kao takve, imaju veliku moć generalizacije i primenjuju se u raznim problemima nadgledanog i nenadgledanog učenja. Modeli dubokog učenja koji su ispitivani u ovom radu su: neuronske mreže sa propagacijom unapred, mreže sa dugom kratkotrajnom memorijom i

konvolucione neuronske mreže. Ovi modeli su poređeni po merama klasifikatora i broju parametara koji su neophodni da bi se model obučio.

Neuronske mreže sa propagacijom unapred (eng. *Feed Forward Neural Network* – FFNN) su mreže kod kojih podaci putuju samo napred kroz mrežu i predstavljaju najjednostavniji model dubokog učenja. Arhitektura se sastoji od nekog broja slova koji sadrže određeni broj neurona koji su međusobno povezani sa svim neuronima iz prethodnog ili narednog sloja [7].

U zavisnosti od složenosti problema, broj slojeva, kao i broj neurona koji oni sadrže, može se menjati. Korištene su različite arhitekture neuronskih mreža sa propagacijom unapred prilikom klasifikacije. Od mogućih arhitektura koje su ispitivane, najbolji rezultati su dobijeni kada je postojao jedan skriveni sloj koji je sadržao 512 neurona.

Mreže sa dugom kratkotrajnom memorijom (eng. *Long-Short Term Memory* - LSTM) predstavljaju potklasu rekurentnih neuronskih mreža koje su sposobne da nauče dugoročne zavisnosti u podacima. Rekurentne neuronske mreže predstavljaju klasu veštačkih neuronskih mreža kod kojih postoje povratne veze.

Povratne veze omogućavaju da se prethodne informacije očuvaju i prenose iz jednog koraka u sledeći. Primenjuju se u domenima prepoznavanja govora, prevođenja jezika, predviđanja akcija i mnogim drugima. Osnovna gradivna jedinica arhitektura mreži sa dugom kratkotrajnom memorijom jeste LSTM jedinica. Pomoću LSTM jedinice, mreža kombinuje trenutne vrednosti koje joj dolaze sa prethodnim i odlučuje o važnosti uticaja prethodnih vrednosti na trenutne [8, 9].

Prilikom konstrukcije klasifikatora pomoću mreža sa dugom kratkotrajnom memorijom, ispitivane su različite arhitekture u zavisnosti od broja LSTM jedinica koje sadrže. Najbolji rezultati su postignuti kada je korišteno 512 LSTM jedinica.

Konvolucione neuronske mreže (eng. *Convolutional Neural Network* - CNN) se najčešće primenjuju za probleme obrade slike, ali može da se primeni i u drugim oblastima, od kojih je jedna i oblast obrade prirodnog jezika. Konvolucione neuronske mreže tretiraju podatke kao prostorne. Umesto da su neuroni u jednom sloju povezani sa svakim neuronom u prethodnom sloju, oni su povezani samo sa neuronima koji su im bliski i svi neuroni imaju istu težinu.

Na ovaj način arhitektura mreže održava prostorni karakter skupa podataka [10]. Kako se crno-bele slike zapisuju u formi matrica na računaru, a podaci koji se prosleđuju mreži su matrice dimenzije 21x28, konvoluciona neuronska mreža je uspela da nauči prostorne karakteristike podataka i da pokaže dobre performanse.

Prilikom formiranja konvolucione neuronske mreže adaptirana je arhitektura "LeNet-5" sa malim izmenama. "LeNet-5" mreža je radila klasifikaciju slika ručno pisanih cifara koje su veličine 28x28 [11].

Motivacija za adaptaciju date arhitekture jeste dimenzionalnost slika, koja je približna dimenzijama ulaznih podataka u problemu dijakritizacije.

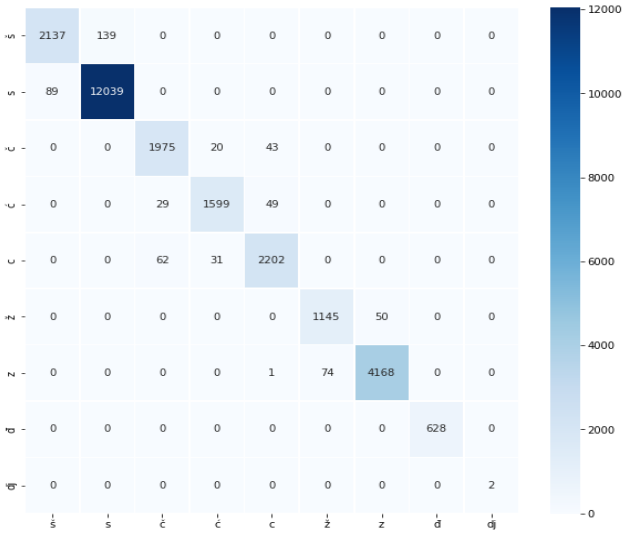
3.1. Performanse modela

Performanse modela koji su navedeni u prethodnoj glavi prikazane su u tabeli 3.1. Iz tabele se vidi da najbolje performanse ima LSTM model. Mada, LSTM model je znatno složeniji od modela formiranih na osnovu FFNN i CNN arhitektura. Iako je LSTM klasifikator znatno složeniji, zbog strukture podataka, klasifikator se veoma brzo obučava i brzo daje rezultate.

Zbog toga, konačni model klasifikatora za koji će biti prikazani rezultati jeste klasifikator baziran na LSTM neuronskoj mreži. Na slici 3.1. je prikazan matrica konfuzije klasifikatora. Iz nje možemo da vidimo da klasifikator uglavnom pogrešno klasifikuje slova koja imaju ortografski zapis koji se samo razlikuje u zavisnosti da li poseduje dijakritički znak ili ne, ali može da se desi i da skroz pogrešno klasifikuje slovo, kao u slučaju gde je jedno slovo Z klasifikovao kao C.

TABELA 3.1. PERFORMANSE RAZLIČITIH MODELA KLASIFIKATORA

Arhitektura	FFNN	LSTM	CNN
Broj parametara za obuku	306.185	1.098.249	274.521
Tačnost klasifikacije	0,966	0,977	0,967
Preciznost klasifikacije	0,850	0,973	0,850
Osetljivost klasifikacije	0,844	0,972	0,850
F1-mera	0.847	0,973	0,850



Slika 3.1. *Matrica konfuzije za LSTM klasifikator*

3.2. Rezultati

Za konačni model klasifikatora, u tabeli 3.2 prikazani su neki primeri slova koja su pogrešno klasifikovana. Iz tabele 3.2 vidimo da mreža ponekad greši kada postoje štamparske greške kao u slučaju „nCi" umesto „noCi". „Koloseum" predstavlja retku reč koja je pogrešno

klasifikovana jer je model nije susreo u skupu za obuku. Za reč „znaCi“ klasifikator je predivdeo da treba da stoji Č umesto C, što je verovatno posledica činjenice da je reč „znači“ češća u srpskom jeziku od reči „znaci“. Postoje i neke veoma neobične greške, kao u primeru reči „saglaSnost“, gde je S prepoznato kao Š.

TABELA 3.2. PRIMERI POGREŠNO KLASIFIKOVANIH SLOVA

Rečenica	Stvarni karakter	Prediktovani karakter
a u casopi**S**u spasic j	**S**	**Š**
rsina pod **S**ecernom re	**Š**	**S**
ovlacenje **Z**andarmerij	**Ž**	**Z**
siran i u **S**amponskoj	**Š**	**S**
poput kolo**S**euma kapit	**S**	**Š**
lepotice n**C**i crveno i	**Ć**	**C**
a je ne li**S**i cinjenic	**Š**	**S**
luposti i **C**arsijske i	**Č**	**C**
ranu sagla**S**nost zaint	**S**	**Š**
ubijenog **S**pijuna bri	**Š**	**S**
da preko **S**ahare goni	**S**	**Š**
da svi zna**C**i ukazuju	**C**	**Č**

4. ZAKLJUČAK

U datom radu predstavljene su različite arhitekture neuronskih mreža koje su vršile rekonstrukciju dijakritičkih znakova u tekstovima na srpskom jeziku. Najbolje performanse pokazala je LSTM mreža i prikazano je nekoliko primera kako ona radi.

Zbog ograničenih resursa koji su bili na raspologanju, obuka mreže i testiranje njenih performansi rađeni su na relativno malom broju reči, odnosno nad delom baze koja je navedena u radu. Za praktične primene ovog modela njegovu obuku treba izvršiti na mnogo većem tekstu, za šta je moguće upotrebiti i tekstove dostupne na internetu. Velika količina podataka može znatno da poboljša performanse dubokih neuronskih mreža.

U radu su posmatrani uzorci koji su uzimali deset karaktera pre i posle karaktera za koji se određuje da li sadrži dijakritički znak ili ne. Ukoliko bismo proširili prozor, memorija koju zauzimaju ulazni podaci bi se znatno povećala. Zbog ograničenih računarskih resursa, ograničili smo se na fiksan prozor od 21 karaktera, iako bi trebalo ispitati i za performanse modela i za veće prozore, obzirom da je LSTM mreža pogodna za rad sa širim kontekstom, odnsno sa dužim vremenskim sekvencama.

Postoje metode koje pokazuju na kojim delovima podataka mreža uči, kao što su Grad-CAM [12], pomoću kojih bi mogla da se odredi optimalna širina prozora. Postoji verovatnoća da je prozor od 10 karaktera pre i 10 karaktera nakon traženog slova već i sam blizu optimuma. Međutim, ako bi se koristio prozor koji obuhvata na primer 100 karaktera pre i 100 karaktera posle karaktera za koji se vrši rekonstrukcija dijakritičkog znaka, mogla bi se tačno utvrditi optimalna veličina prozora za predstavljene tipove neuronskih mreža.

Problem koji je rešavan u ovom radu na neki način predstavlja uklanjanje šuma iz podataka. Pored modela koji su testirani u datom radu za rekonstrukciju dijakritičkih elemenata, trebalo bi testirati i neke druge, kao što je autoenkoder za uklanjanje šuma. Za rad sa sekvencijalnim podacima veoma dobro su se pokazale rekurentne neuronske mreže, tako da će u budućim istraživanjima pažnja biti posvećana i njima.

5. LITERATURA

[1] Nikola Ljubešić, Tomaž Erjavec, Darja Fišer, „Corpus-Based Diacritic Restoration for South Slavic Languages“

[2] Jakub Náplava, Milan Straka, Pavel Straňák, Jan Hajič, “Diacritics Restoration Using Neural Networks”

[3] Ali Fadel, Ibraheem Tuffaha, Bara’ Al-Jawarneh, and Mahmoud Al-Ayyoub, “Neural Arabic Text Diacritization: State of the Art Results and a Novel Approach for Machine Translation”, Proceedings of the 6th Workshop on Asian Translation, pages 215–225 Hong Kong, China, November 4, 2019. Association for Computational Linguistics

[4] Stefan Ruseti, Teodor-Mihai Cotet, Mihai Dascalu: *Romanian Diacritics Restoration Using Recurrent Neural Networks.* September 2020.

[5] https://machinelearningmastery.com/how-to-one-hot-encode-sequence-data-in-python/, 17.08.2022.

[6] Tijana Nosek, Branko Brkljač, Danica Despotović, Milan Sečujski, Tatjana Lončar-Turukalo: *Praktikum iz mašinskog učenja*, Fakultet Tehničkih Nauka, Univerzitet u Novom Sadu

[7] https://en.wikipedia.org/wiki/Feedforward_neural_network, 17.10.2022.

[8] https://towardsdatascience.com/illustrated-guide-to-recurrent-neural-networks-79e5eb8049c9, 17.10.2022

[9] https://colah.github.io/posts/2015-08-Understanding-LSTMs/, 17.10.2022.

[10] https://colah.github.io/posts/2014-07-Conv-Nets-Modular/, 18.10.2022.

[11] Yann LeCun, Leon Bottou, Yoshua Bengio, Patrick Haffner, “Gradient-Based Learning Applied to Document Recognition”, “Proc. Of the IEEE”, Novembar 1998

[12] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra, *Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization*, International Journal of Computer Vision, 2019.

Kratka biografija:

Vuk Stanojev rođen je 25.02.1998. godine u Novom Sadu. Završio je gimnaziju “Jovan Jovanović Zmaj” u Novom Sadu 2017. godine. Nakon toga upisuje Fakultet Tehničkih Nauka, smer Elektronika, energetika i telekomunikacije gde diplomira 2021. godine. Master akademske studije upisuje iste godine na Fakultetu tehničkih nauka, oblast obrada signala.