Zbornik radova Fakulteta tehničkih nauka, Novi Sad

UDK: 004.9
DOI: https://doi.org/10.24867/06BE08Trpovski

AUTOMATSKA DETEKCIJA STAVKI MENIJA UNUTAR TEKSTOVA RECENZIJA RESTORANA

AUTOMATIC DETECTION OF MENU ITEMS IN RESTAURANT REVIEWS

Igor Trpovski, *Fakultet tehničkih nauka, Novi Sad*

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *Cilj ovog istraživanja jeste prezentovanje jednog pristupa za detekciju stavki menija unutar tekstova recenzija restorana. Nekoliko modela mašinskog i dubokog učenja istrenirano je da detektuje pominjanja hrane unutar recenzija restorana. Nakon toga, nekoliko algoritama poklapanja stringova primenjeno je kako bi se pominjanja hrane uparila sa odgovarajućim stavkama menija. Podaci su prikupljeni sa sajta Donesi.com i ručno anotirani. Svi upotrebljeni modeli i algoritmi su evaluirani.*

Ključne reči: *analiza teksta, obrada prirodnog jezika, prepoznavanje imenovanih entiteta*

Abstract –*The purpose of this research is to present one approach for automatically detecting menu items in restaurant reviews. Several machine and deep learning models were trained in order to detect food mentions. Afterwards, several string matching algorithms were applied in order to match food mentions with corresponding menu items. Data was acquired from the website Donesi.com and manually annotated. All used models and algorithms were evaluated.*

Keywords: *text mining, natural language processing, named entity recognition*

1. UVOD

Zahvaljujući ekspanziji i povećanoj dostupnosti interneta u poslednjoj deceniji, mnogi ga koriste kako bi pronašli informacije koje bi im pomogle pri donošenju svakodnevnih odluka kao što su izbor restorana i hrane za jelo. Postoji veliki broj sajtova koji prikazuju informacije o određenim restoranima, ali pronaći informacije o kvalitetu i popularnosti pojedinačnih stavki menija iz restorana može biti veoma vremenski zahtevan i umarajući posao, jer uglavnom podrazumeva čitanje potencijalno velikog broja recenzija.

Zbog različitih načina pisanja naziva jela unutar recenzija, može biti teško zaključiti na koju se stavku menija pomenuto jelo odnosi. Takođe, ukoliko nekoga interesuje kvalitet neke konkretne stavke menija, umesto da mu automatski budu prikazane samo recenzije u kojima se ta stavka pominje, on opet mora čitati sve recenzije.

NAPOMENA:
Ovaj rad proistekao je iz master rada čiji mentor je bio dr Aleksandar Kovačević, vanredni profesor

Tema ovog rada je automatska detekcija pominjanja hrane unutar recenzija restorana i za svako od njih određivanje na koju stavku menija se ono odnosi. Najveći izazov predstavlja činjenica da su recenzije podaci sa mnogo šuma. Drugim rečima one sadrže veliki broj pravopisnih grešaka, specijalnih znakova i često su pisane upotrebom kolokvijalnih i regionalnih izraza. Tretiranjem pominjanja hrane kao imenovanih entiteta, njihova detekcija unutar tekstova recenzija svodi se na klasičan problem prepoznavanja imenovanih entiteta (eng. Named Entity Recognition - NER). NER je proces koji ima za cilj da locira i klasifikuje pominjanja imenovanih entiteta iz nestrukturiranih tekstova u prethodno definisane kategorije kao što su imena, lokacije, organizacije, itd. Uzima neanotiran i proizvodi anotiran deo teksta sa označenim imenovanim entitetima. Ovaj problem proučavan je i ranije, ali koliko je poznato autoru ovog rada, nikada za recenzije pisane na srpskom jeziku.

Kreiran je skup podataka sačinjen od recenzija restorana dobavljenih sa Donesi.com [1], popularnog sajta za naručivanje hrane u regionu Srbije, Crne Gore i Bosne i Hercegovine. Kako bi bilo moguće trenirati modele da prepoznaju koji delovi teksta predstavljaju hranu, a koji ne, bilo je neophodno manuelno anotirati pominjanja hrane unutar recenzija. Modeli koji su korišćeni za ovaj problem su: Conditional Random Fields (CRF) [2], bidirekcioni Long Short-term Memory (LSTM) [3] i bidirekcioni Gated Recurrent Units (GRU) [4]. Za bidirekcione LSTM i bidirekcione GRU modele iskorišćeno je nekoliko različitih načina za kodiranje ulaza: one-hot reprezentacija reči, one-hot reprezentacija reči i karaktera i FastText [5] vektorska reprezentacija reči. Takođe isproban je hibridni pristup korišćenja CRF-a zajedno sa bidirekcionim LSTM i bidirekcionim GRU modelima isto koristeći prethodno spomenute metode za kodiranje ulaza. Kako bi odredili koja stavka menija odgovara određenom entitetu hrane isprobano je nekoliko različitih metoda poklapanja stringova: potpuno poklapanje, podstring poklapanje, fuzzy poklapanje i parcijalno poklapanje. Cilj rada je uporediti sve iskorišćene metode, kao i odrediti onu koja daje najbolje rezultate za zadatak za koji je namenjena.

2. METODOLOGIJA

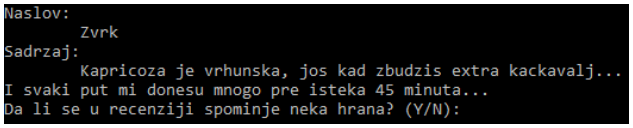
U ovom poglavlju prezentovane su primenjene metodologije za detekciju stavki menija unutar recenzija restorana. Takođe opisani su i alati upotrebljeni u njenoj implementaciji.

2.1. Prikupljanje podataka

Svi potrebni podaci pribavljeni su sa sajta Donesi.com. Koliko je poznato autoru ovog rada, on nema API za prikupljanje podataka, pa je bilo neophodno napisati web crawler. Napisan je u jeziku Python koristeći biblioteku Selenium [6], a parsiranje HTML stranica izvršeno je pomoću biblioteke BeautifoulSoup4 [7]. Odlučeno je da se dobave svi podaci sa područja Srbije. Konkretno, dobavljeno je 169.486 recenzija za 1.658 različitih restorana. Takođe prikupljeni su jelovnici svih restorana.

2.2. Filtriranje recenzija

Filtriranje podataka odrađeno je radi stvaranja skupa podataka koji sadrži samo one recenzije restorana koje u svom naslovu ili tekstu sadrže jedno ili više pominjanja hrane. Da bi se olakšao ovaj zadatak implementiran je mali program čiji je interfejs prikazan na slici 1. Program iterira kroz skup podataka i za trenutnu recenziju prikazuju se naslov i tekst. Osoba koja izvršava ovaj zadatak odlučuje da li se unutar recenzije nalazi pominjanje hrane. Ako odluči potvrdno, recenzija biva sačuvana unutar nove kolekcije podataka. Novokreirani skup podataka sastoji se iz 20.079 filtriranih recenzija. Neophodno je napomenuti da pre čuvanja recenzije unutar nove kolekcije, ukoliko je njen tekst napisan u Ćirilici, on biva konvertovan u Latinicu. Takođe svi latinični specijalni karakteri su zamenjeni. Ova dva prethodna koraka odrađena su kako bi se sve recenzije svele na isti način pisanja i time olakšao posao algoritmima koji će biti korišćeni u narednim koracima.



Slika 1. *Interfejs programa za filtriranje recenzija*

2.3. Anotacija pominjanja hrane

Kako bi bilo moguće obučiti modele koji detektuju pominjanja hrane unutar recenzija restorana, bilo je neophodno anotirati delove naslova i teksta recenzije koji ih sadrže. BILOU šema kodiranja korišćena je za predstavljanje pominjanja hrane. Pomoću nje model se može obučiti da prepozna početne, unutrašnje i krajnje tokene entiteta čiji naziv sadrži više tokena, kao i entitete čiji naziv sadrži samo jedan token. O klasa se koristi za predstavljanje tokena koji ne pripadaju nekom entitetu. Na primer, entitet hrane "Piletina sa kikirikijem" označava se sa labelama "B-FOOD", "I-FOOD" i "L-FOOD", odnosno entitet "Kapricoza" sa labelom "U-FOOD". Anotiranje je vršeno pomoću alata MAE [8]. Najpre je neophodno kreirati datoteku koja sadrži klase koje će se koristiti prilikom anotiranja. U ovom slučaju, svaki token mogao je biti anotiran sa jednom od ovih klasa: B-FOOD, I-FOOD, L-FOOD ili U-FOOD. Nakon toga bilo je neophodno kreirati datoteku u XML formatu koja sadrži tekst za anotaciju. Budući da recenzije sadrže naslov i tekst, izvršena je njihova konkatenacija sa karakterom "\n" između. Dakle, za svaku recenziju koristili smo taj konkatenirani tekst za kreiranje odgovarajućih XML datoteka. Posle završene anotacije jedne recenzije, u njenoj odgovarajućoj XML datoteci kreira se lista svih anotiranih tokena sa dodeljenim klasama. Zbog vremenskih ograničenja anotirano je 10 000 recenzija.

2.4. Obučavajući i test skup podataka

Za podelu teksta u rečenice, tokene, POS tagovanje i lematizaciju korišćena je Python biblioteka ReLDI [9-13], jer je razvijena specifično za obradu južnoslovenskih jezika. Takođe je korišćena i Python implementacija stemera za Srpski jezik [14]. Skup podataka od 10 000 recenzija sa anotiranim entitetima hrane i procesiranih od strane ReLDI biblioteke i stemera za srpski jezik nasumično je podeljen na obučavajući skup od 9.000 i test skup od 1.000 recenzija.

2.5. Detekcija pominjanja hrane

Prvobitno je isproban algoritam mašinskog učenja CRF, jer se kroz razna istraživanja pokazao kao najbolji algoritam tog tipa za problem prepoznavanja imenovanih entiteta. Razlog za to je što je po prirodi sekvencioni model, tj. za sekvencu primera predviđa sekvencu labela. Drugim pri predikciji labele za trenutni primer on uzima u obzir predikcije za prethodne. U slučaju linear-chain CRF modela koji je ovde korišćen uzima se u obzir samo labela prethodnog primer. Kada govorimo o NER-u, sekvencu predstavlja rečenica odnosno skup tokena.

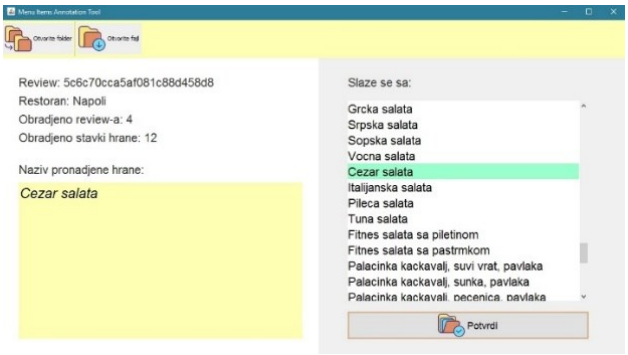
CRF omogućava specificiranje određenog broja dodatnih atributa koji će dodatno opisati svaki token. U ovom radu su za ulaz u CRF model pored tokena korišćeni su sledeći atributi: POS tag, lema, stem, token pisan svim velikim slovima, token pisan svim malim slovima, token pisan velikim početnim i svim ostalim malim slovima, da li je token broj, da li je token znak interpunkcije, da li je token napisan samo velikim slovima, da li je token napisan sa pa prvim početnim, a svim ostalim malim slovima i da li token predstavlja krajnji token rečenice. Takođe je korišćena okolina od 3 tokena oko trenutnog, odnosno 3 prethodna i 3 naredna tokena, kao i POS tagovi svih tih tokena. Svi prethodno navedeni atributi izabrani su proučavanjem sličnih rešenja i generalnih upotreba CRF algoritma za NER, kao i otkrivanjem šablona u korišćenim podacima. Iskorišćena je implementacija CRF algoritma u programskom jeziku C++ zvana CRF++ [15]. Važno je napomenuti da su pri treniranju korišćeni baš svi navedeni atributi, tj. ovaj rad se nije bavio određivanjem optimalnog skupa atributa. Takođe za proces treniranja korišćeni su predefinisani parametri CRF++ modela.

Nakon toga isprobani su modeli dubokog učenja LSTM i GRU. Oni predstavljaju varijacije standardnih modela rekurentnih neuronskih mreža i imaju za cilj da otklone probleme često vezane za njih kao što su nestajući gradijent i rukovanje dugoročnim zavisnostima. Korišćeni su birirekcioni LSTM i GRU modeli, jer se tako pri predikciji trenutnog tokena uzimaju u obzir informacije iz prošlih i budućih stanja, tj. informacije o prošlim i budućim tokenima iz sekvence. Kako ovi modeli za ulaz ne mogu uzeti tekstualne podatke oni su morali na neki način biti kodirani. Isprobane su varijante one-hot reprezentacija reči, kao i reči i karaktera.

Takođe isprobane su i FastText vektorske reprezentacije reči. FastText vektori dobijeni su treniranjem neuronske mreže nad skupom podataka od 267 362 rečenica recenzija sa Donesi.com koje sadrže 6 ili više tokena. Iskorišćeni su predefinisani parametri Python FastText implementacije. Takođe iskorišćena je varijanta LSTM i GRU modela sa dodatnim CRF slojem. Razlog za to je što modeli bez tog sloja donose odluke o klasifikaciji svakog tokena zasebno, odnosno ne obezbeđuju predikciju najverovatnije čitave sekvence. To dovodi do narušavanja ograničenja koja BILOU šema postavlja. Npr. dešava se da se "U-FOOD" tag javlja odmah nakon "B-FOOD" taga. CRF sloj, sa druge strane, vrši predikciju klase trenutnog tokena tako što uzima u obzir klasu prethodnog tokena i tako na kraju odredi najverovatniju čitavu sekvencu. Za kreiranje arhitekture svih modela dubokog učenja korišćena je biblioteka Keras [16].

2.6. Anotacija stavki menija

Bilo je neophodno anotirati pominjanja hrane iz recenzija sa odgovarajućom stavkom menija iz restorana na koju se ta recenzija odnosi. Iz tog razloga kreiran je poseban alat za anotaciju. Na slici 2. prikazan je njegov interfejs. Ovaj korak bio je neophodan kako bi se testirali algoritmi koji će automatski odrediti koja stavka menija odgovara kom pominjanju hrane. Kao ulaz u ovaj alat iskorišćen je skup podataka od 1.000 recenzija kod kojih je izvršena anotacija pominjanja hrane. Taj skup se takođe koriste i za testiranje algoritama za detekciju pominjanja hrane. Jedno pominjanje hrane može se odnositi na nijednu, jednu ili više stavki iz menija restorana. Ukoliko se pominjanje hrane očigledno ne odnosi ni na jednu stavku menija ili je nemoguće napraviti izbor između nekoliko stavki menija, doneta je odluka da se u tom slučaju pominjanje hrane anotira sa "None". U suprotnom anotator je morao izabrati jednu stavku menija na koju se pominjanje hrane odnosi. Pri završetku anotiranja poslednjeg pominjanja hrane iz jedne recenzije, generiše se izlazna datoteka u JSON formatu, koja sadrži spisak pominjanja hrane sa anotiranim stavkama menija.



Slika 2. *Interfejs programa za anotaciju stavki menija*

2.7. Uparivanje pominjanja hrane sa odgovarajućim stavkama menija

Kako bi se automatski odredilo na koju stavku menija se pominjanje hrane odnosi isprobano je nekoliko različitih metoda poklapanja stringova: potpuno poklapanje, podstring poklapanje, fuzzy poklapanje i parcijalno poklapanje. Kod svih tekstova su pre primene algoritama za poklapanje velika slova konvertovana u mala.

Potpuno poklapanje je najstrožiji od svih algoritama. Smatra se uspešnim ukoliko se tekst stavke menija potpuno poklapa sa tekstom pominjanja hrane. Ova metoda nije uspela da pronađe previše poklapanja, osim u slučajevima kada je autor recenzije naveo puno ime stavke menija pri njenom pominjanju u recenziji.

Pominjanje hrane i stavka menija poklapanju se ukoliko je tekst pominjanja hrane podstring teksta stavke menija ili obrnuto, u zavisnosti od toga koji tekst sadrži više karaktera. Bilo je vrlo uobičajeno da se pominjanje hrane poklopi sa više stavki menija. Isprobana su dva različita načina za rešavanje tog problema. Prvi je da se izabere stavka menija čiji je tekst najkraći, a drugi je da se ne izabere nijedna stavka menija odnosno da se pominjanje hrane označi sa "None".

Mnoga pominjanja hrane sadržala su greške u pisanju. U tim slučajevima prethodne metode nisu mogle biti uspešne. Samim tim isprobane su tri metode fuzzy poklapanja: Damerau-Levenštajnovo rastojanje [17], Jaroova sličnost [18] i Jaro-Winklerova sličnost [19]. Stavka menija koja odgovara pominjanju hrane jeste ona za koju je Damerau-Levenštajnovo rastojanje najmanje, gde je ono takođe manje od zadate vrednosti praga (eng. threshold). Korišćena je Python implementacija ovog rastojanja koja je deo biblioteke strsim [20]. Stavka menija koja odgovara pominjanju hrane pri korišćenju Jaroove sličnosti je ona za koju je ta sličnost najveća, a da je ona pritom veća od zadate vrednosti praga. Isti princip korišćen je i za Jaro-Winklerovu sličnost. Korišćena je Python biblioteka pyjarowinkler [21] koja sadrži implementacije ova dva algoritma.

Pre vršenja parcijalnog poklapanja tekstovi pominjanja hrane i stavki menija su tokenizovani, izvršeno je stemovanje i uklonjene su stop reči, odnosno reči koje ne nose prevelik značaj. Za tokenizaciju i stemovanje korišćene su Python implementacije tokenizera i stemera za srpski jezik. Poklapanje se smatra uspešnim ukoliko se u skupu stemovanih tokena pominjanja hrane nalazi bar polovina stemovanih tokena stavke menija i beleži se tačan broj stemovanih tokena u preseku. Stavka menija sa kojom je najveći broj stemovanih tokena u preseku se uzima kao ona stavka na koju se pominjanje hrane odnosi.

3. EKSPERIMENTALNA EVALUACIJA I REZULTATI

Eksperimentalna evaluacija se vrši radi određivanje performansi korišćenih algoritama za detekciju pominjanja hrane i određivanja stavke menija na koje se svako pominjanje odnosi.

Metrike performansi algoritama za detekciju pominjanja hrane i određivanja stavke menija na koje se svako pominjanje odnosi koje su korišćene u ovom radu su preciznost, odziv i F_1-mera. Važno je napomenuti da je pominjanje hrane koje je detektovao model jedino smatrano tačnim ukoliko se svi njegovi tokeni poklapaju sa tokenima pominjanja označenog kao ispravnog. Drugim rečima nije se razmatralo nikakvo parcijalno poklapanje, odnosno detektovana pominjanja koja se parcijalno poklapaju sa očekivanim smatrana su kao netačna.

Što se tiče detekcije pominjanja hrane svi modeli koji su postigli najbolje metrike su neuralni i koriste FastText vektorske reprezentacije reči. Najbolju preciznost od 0.9376 dostigao je bidirekcioni GRU-CRF treniran u 5 epoha. Najbolji odziv od 0.9335 bidirekcioni LSTM i najbolju F_1-meru od 0.9272 bidirekcioni LSTM-CRF, oba trenirana u 10 epoha. Sa druge strane za uparivanje pominjanja hrane sa odgovarajućim stavkama menija najbolju preciznost od 0.8865 dostiglo je Jaro-Winklerovo poklapanje sa pragom od 0.8, a najbolji odziv od 0.8683 i F_1-meru od 0.8468 dostiglo je parcijalno poklapanje.

4. ZAKLJUČAK

U ovom radu predstavljeno je jedno rešenje za detekciju stavki menija u recenzijama restorana. Kreiran je skup podataka recenzija na srpskom jeziku, tako što su one dobavljene sa sajta Donesi.com. Izvršena je ručna filtracija i anotacija recenzija kako bi se obeležili oni delovi teksta koji sadrže pominjanja hrane. Nekoliko modela mašinskog i dubokog učenja istrenirano je da detektuje pominjanja hrane. Nakon toga, nekoliko algoritama poklapanja stringova primenjeno je kako bi se pominjanja hrane uparila sa odgovarajućim stavkama menija. Svi korišćeni modeli i algoritmi postigli su zadovoljavajuće rezultate.

Rad bi mogao dalje da se proširi izborom optimalnog skupa atributa za treniranje CRF modela. Moglo bi se dodatno poraditi na pronalaženju optimalnih parametara za treniranje FastText neuronske mreže i neuralnih modela korišćenih za NER. Takođe mogao bi se kreirati veći skup podataka sa anotiranim pominjanjima hrane eksperimentisanjem sa polu-automatizovanom anotacijom. Pošto metode koje određuju na koju stavku menija se pominjanje hrane odnosi ne mogu da detektuju sinonime, mogao bi se napraviti rečnik čestih sinonima i on koristiti da se taj problem razreši. Takođe se pored metoda za poklapanje stringova može eksperimentisati i sa korišćenjem metoda mašinskog učenja ili neuronskih mreža.

5. LITERATURA

[1] www.donesi.com

[2] Lafferty, J., McCallum A., and Pereira F., 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

[3] Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. *Neural computation*, *9* no.8, pp.1735-1780.

[4] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*

[5] Bojanowski, P., Grave, E., Joulin, A. and Mikolov, T., 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, *5*, pp.135-146.

[6] Jason Huggins, *et al,* 2004. Selenium, https://www.seleniumhq.org

[7] Leonard Richardson 2014, BeautifulSoup4 https://www.crummy. com/software/BeautifulSoup

[8] Kyeongmin Rim, "MAE2: Portable Annotation Tool for General Natural Language Use". In Proceedings of the 12th Joint ACL-ISO Workshop on Interoperable Semantic Annotation, Portorož, Slovenia, May 28, 2016.

[9] Ljubesic, Nikola, Tomaz Erjavec and Darja Fiser. "Corpus-Based Diacritic Restoration for South Slavic Languages." *LREC* (2016).

[10] Ljubesic, Nikola and Tomaz Erjavec. "Corpus vs. Lexicon Supervision in Morphosyntactic Tagging: the Case of Slovene." *LREC* (2016).

[11] Ljubesic, Nikola, Filip Klubicka, Zeljko Agic and Ivo-Pavao Jazbec. "New Inflectional Lexicons and Training Corpora for Improved Morphosyntactic Annotation of Croatian and Serbian." *LREC* (2016).

[12] Agic, Zeljko and Nikola Ljubesic. "Universal Dependencies for Croatian (that work for Serbian, too)." *BSNLP@RANLP* (2015).

[13] Fišer, D., Ljubešić, N. & Erjavec, T. Lang Resources & Evaluation (2018). https://doi.org/10.1007/s10579-018-9425-z

[14] Milosevic, Nikola "Stemmer for Serbian Language. " *CoRR* abs/ 1209.4471 (2012): n. pag.

[15] Taku Kudo, "CRF++: Yet another CRF toolkit" 2005, https://taku910.github.io/crfpp

[16] Chollet, Francoise et al. "Keras" 2015, https://keras.io

[17] Damerau, F.J., 1964. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, *7*(3), pp.171-176.

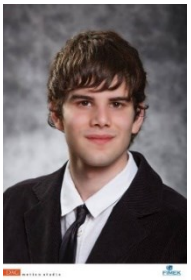
[18] Jaro, M.A., 1989. Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida. *Journal of the American Statistical Association*, *84*(406), pp.414-420.

[19] Winkler, W.E., 1990. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage.

[20] Zhuo Yang Luo. python-string-similarity 2018 https://github.com/luozhouyang/python-string-similarity

[21] Jean-Bernard Ratte. Jaro Winkler Distance 2015, https://github.com/nap/jaro-winkler-distance

Kratka biografija:



Igor Trpovski rođen je u Novom Sadu 1995. god. Master rad na Fakultetu tehničkih nauka iz oblasti Računarstva i automatike – Sistemi za istraživanje i analizu podataka odbranio je 2019.god. kontakt: trpovski@gmail.com