# Klasifikacija tekstova temeljena na skupovima riječi

**Kozina, Ante**

SVEUČILIŠTE U RIJECI

**TEHNIČKI FAKULTET**

Prijediplomski sveučilišni studij računarstva

Završni rad

# KLASIFIKACIJA TEKSTOVA TEMELJENA NA SKUPOVIMA RIJEČI

Rijeka, rujan 2023.

Ante Kozina

0069088428

SVEUČILIŠTE U RIJECI

**TEHNIČKI FAKULTET**

Prijediplomski sveučilišni studij računarstva

Završni rad

# KLASIFIKACIJA TEKSTOVA TEMELJENA NA SKUPOVIMA RIJEČI

Mentor: Prof. dr. sc. Ivo Ipšić

Rijeka, rujan 2023.

Ante Kozina

0069088428

**SVEUČILIŠTE U RIJECI**
**TEHNIČKI FAKULTET**
POVJERENSTVO ZA ZAVRŠNE ISPITE

Rijeka, 13. ožujka 2023.

Zavod: **Zavod za računarstvo**
Predmet: **Programiranje II**
Grana: **2.09.04 umjetna inteligencija**

# ZADATAK ZA ZAVRŠNI RAD

Pristupnik: **Ante Kozina (0069088428)**
Studij: Sveučilišni prijediplomski studij računarstva

Zadatak: **Klasifikacija tekstova temeljena na skupovima riječi**

Opis zadatka:

Opišite postupke gradnje jezičnog modela temeljenog na skupovima riječi (Bag-of-Words modeli). Na temelju tekstova hrvatske Wikipedije realizirajte postupak automatske klasifikacije rečenica u pojedine kategorije tekstova.

Rad mora biti napisan prema Uputama za pisanje diplomskih / završnih radova koje su objavljene na mrežnim stranicama studija.

Zadatak uručen pristupniku: 20. ožujka 2023.

Mentor:

Prof. dr. sc. Ivo Ipšić

Predsjednik povjerenstva za završni ispit:

Prof. dr. sc. Miroslav Joler

## Izjava o samostalnoj izradbi rada

Sukladno članku 7. Pravilnika o završnom radu, završnom ispitu i završetku sveučilišnih prijediplomskih studija Tehničkog fakulteta Sveučilišta u Rijeci izjavljujem da sam ovaj rad samostalno izradio te da on ne sadrži dijelove koji su kopirani iz vanjskih izvora, a da nisu ispravno citirani.

Rijeka, rujan 2023.

Ante Kasina

(potpis)

# Zahvala

Zahvalio bih svojem mentoru prof. dr. sc. Ivi Ipšiću na predloženoj temi i na velikoj pomoći kod realizacije završnog rada.

Također, zahvalio bih svojoj obitelji i prijateljima na njihovom strpljenju i na raznim smišljenim tekstovima koji su korišteni za provjeru ispravnosti klasifikacije.

# Sadržaj

# 1. UVOD

„Bag of words model“ prevedeno „jezični model temeljen na zbirkama riječi“ se koristi u obradi prirodnog jezika i kod sustava za povrat informacija [1]. Često je rješenje problema klasifikacije dokumenta – pridodavanja nekakve kategorije tekstu na temelju njegovog sadržaja [2]. Klasifikacija dokumenata se koristi, među ostalima, u filtriranju neželjene e-pošte, kod jezične identifikacije te za brže i lakše procesiranje upitnika korisničke podrške. Naprimjer, ako su u primljenoj e-mail poruci česti izrazi poput „laka zarada“, „brzo mršavljenje“ ili „prestanak pušenja“ vrlo je vjerojatno da je sadržaj te e-mail poruke prijevara. Na sličan način, ako se u upitniku korisnika pojavljuju riječi „prijava“, „lozinka“, „zaboravljena“ i sl. sistem može automatski poslati odgovor koji sadrži upute za ponovno postavljanje lozinke.

U radu će prvo biti objašnjene osnove o jezičnom modelu te će biti opisan koncept klasifikatora dokumenta – programa koji danom dokumentu odredi kategoriju (temu). Bit će prikazan proces izdvajanja teksta članaka internet izdanja dnevnih novina te njegova obrada u jednostavne rečenice. Nakon, nabrojat će se metode s kojima se odabiru i povezuju riječi različitih oblika ali istih značenja. Bit će nabrojane i različite metode za izračun vjerojatnosti kategorija članaka. Kako bi mogli testirati performanse kombinacija tih metoda, par članaka će biti klasificirano i rezultati će biti prikazani u tablicama. Na kraju će biti donesen zaključak na temelju prikazanih rezultata.

# 2. JEZIČNI MODEL TEMELJEN NA ZBIRKAMA RIJEČI

Jezični model temeljen na zbirkama riječi je način izvlačenja riječi iz nekog sadržaja s ciljem sažimanja i klasificiranja tog sadržaja [3]. Model ne očuvaje redoslijed riječi što bi se moglo zaključiti iz njegovog engleskog naziva: „Bag of words model“ – riječi su „smještene u vreću“, tj. gubi se njihov poredak. Umjesto pojedinačnih riječi može se brojati i učestalost skupa susjednih riječi, takozvanih n-torki. N-torke su za svaki pojedinačni dokument spremljene u strukturu podataka zvanu rječnik, gdje je ključ n-torka, a vrijednost je broj ponavljanja te n-torke. Kod jezičnog modela temeljenog na zbirkama riječi taj rječnik se zove vektor (slika 2.1.).

```
tekst = "Marko je išao igrati nogomet. Išao je i David."

vektor = {'marko': 1, 'je': 2, 'išao': 2, 'igrati': 1,
'nogomet': 1, 'i': 1, 'david': 1}
```

*Slika 2.1. Vektor nastao iz dokumenta, gdje su riječi bodovane po njihovom ponavljanju*

# 3. KLASIFIKATOR DOKUMENTA

Kako bi se metode klasificiranja mogle testirati, napravljen je program kojemu je cilj odrediti kojoj od definiranih kategorija uneseni tekst najviše odgovara. To radi tako da uspoređuje riječi poznatih tekstova kojima su teme već određene s riječima teksta kojega unese korisnik.

Program razlikuje šest kategorija/tema teksta: filmovi/serije, glazba, gospodarstvo, politika, sport i tehnologija. Kategorije su izabrane tako da se uklone moguća preklapanja tema, s ciljem da se preciznost može maksimizirati. Svaka kategorija ima svoju tekstualnu datoteku koja sadrži 20 poveznica članaka internet izdanja dnevnih novina – portala, koje su ručno unesene s obzirom na to kako su ih kategorizirala web-sjedišta.

Iz članaka jedne od šest kategorija, koji su preuzeti putem spremljenih poveznica, stvoreni su vektori koji su međusobno povezani, tvoreći vektor te kategorije. Drugo ime za vektore svih kategorija je vektori klasifikatora. Testirani dokument je zajednički naziv za tekst koji se želi klasificirati, bio on ručno unesen ili izdvojen iz web-stranice korisnički unesene poveznice.

Za stvaranje vektora klasifikatora korišteni su članci portala 24sata [4] i Novi list [5]. Za korisnikov unos kod klasifikacije članka predviđeno je korištenje hrvatskih članaka Wikipedije [6]. U početku je bilo planirano da program može primiti članke s više web-sjedišta, no bilo je teško ukloniti irelevantne dijelove web-stranica jer svako web-sjedište ima svoj dizajn HTML koda. Zbog tog razloga i jer je kod njih izdvajanje teksta članka bilo nešto jednostavnije uzeti su 24sata, Novi list i Wikipedija.

## 3.1. Izdvajanje sadržaja članka

Jer irelevantni dijelovi web-stranica sadrže riječi/rečenice koje nisu povezane s kontekstom samog članka, bilo ih je potrebno ukloniti kako ne bi poremetili klasifikaciju. To posebno dolazi do izražaja kod metoda izračuna vjerojatnosti koje su usredotočene na n-torke čija je učestalost pojavljivanja manja. Primjeri takvih dijelova su izbornici na vrhu s nazivima kategorija te nepovezani predloženi članci i reklame duž web-stranice. U lijevoj listi riječi na slici 3.1., vidi se zašto je izdvajač teksta članka potreban. Riječi poput „novi“, „telefon“, „osmrtnice“ i „pretplata“ nemaju veze sa samim člankom nego se odnose na irelevantne elemente web-stranice.

| | |
|---|---|
| modrić: 11 | modrić: 6 |
| početna: 9 | ancelotti: 6 |
| rijeka: 8 | igra: 5 |
| ancelotti: 8 | madrid: 4 |
| novi: 6 | real: 4 |
| luka: 6 | luka: 3 |
| real: 6 | prvenstva: 3 |
| madrid: 6 | sezone: 3 |
| igra: 5 | zadovoljan: 2 |
| telefon: 5 | igrač: 2 |
| tv: 4 | madrida: 2 |
| prvenstva: 4 | može: 2 |
| minuta: 4 | doprinijeti: 2 |
| sezone: 4 | izjavio: 2 |
| četvrtak: 3 | četvrtak: 2 |
| osmrtnice: 3 | trener: 2 |
| pretplata: 3 | početna: 2 |
| regija: 3 | kola: 2 |
| list: 3 | utakmica: 2 |
| zadovoljan: 3 | neuobičajena: 2 |
| dva: 3 | situacija: 2 |
| kola: 3 | njega: 2 |
| utakmica: 3 | athletic: 2 |

*Slika 3.1. Najčešće riječi (isključujući veznike) sportskog članka portala „Novi list“ bez (lijevo) i sa izdvajačem sadržaja (desno)*

Ako se radi o HTML kodu 24sata, Novog lista ili Wikipedije traži se * * oznaka s jedinstvenom *class* ili *id* vrijednosti koja je ručno pronađena. U slučaju promjene dizajna ili važnih pojedinosti HTML koda web-stranice, ovaj izdvajač sadržaja neće moći pronaći tekst članka, stoga nije dugoročno rješenje. Unutar parova oznaka, * * i * *, traže se * * oznake koje označavaju paragrafe. Sva tri web-sjedišta u svoje web-stranice stavljaju tekst vezan uz članak unutar tih oznaka. Između * * i * * parova još uvijek nije čisti tekst nego postoje razne oznake za njegovo modificiranje: podebljano, nakoso, podcrtano i sl. U sljedećem potpoglavlju opisano je kako se HTML kod između „paragraf“ oznaka dalje obrađuje.

### 3.2. Obrada preostalog HTML koda

Ako poveznica koja se želi obraditi nije s 24sata, Novog lista ili Wikipedije, kod joj se preuzima te se prvo uklanjanju <*style*> i <*script*> oznake i svi znakovi između njih te pronađeni komentari (komentari u kontekstu koda). Kao i u paragrafu prije, preostaje tekst s raznim modifikatorima.

HTML kodu s modifikatorima uklanjaju se oznake <*a*>, <*span*>, <*b*>, <*i*>, <*sup*>, <*strong*>, <*div*>, ali ne i tekst unutar njih. U HTML jeziku postoje različiti kodovi/*string*-ovi koji zamjenjuju znakove koji bi mogli poremetiti interpretaciju koda. Takvi kodovi su ručno pronađeni prolazeći kroz članke klasifikatora i nađeni su im pripadajući ispravni znakovi. Na slici 3.2. su prikazani neki od tih kodova te znakovi s kojima su zamijenjeni koristeći Python metodu *replace()* koja se poziva na *string* objektima. Prvi parametar je *string* koji se traži, a drugi parametar je *string* s kojim se prvi zamijeni.

```
html_text = html_text.replace("&#8220;", '"')
html_text = html_text.replace("&#8221;", '"')
html_text = html_text.replace("&#91;", " ")
html_text = html_text.replace("&#93;", "")
html_text = html_text.replace(" ", " ")
html_text = html_text.replace("\u00a0", " ")
html_text = html_text.replace("\u200b", "")
html_text = html_text.replace("\u201c", "")
html_text = html_text.replace("\u201d", "")
html_text = html_text.replace("\u201e", "")
```

*Slika 3.2. Zamjena HTML „kodova“ sa ispravnim znakovima*

Rezultat je čitljivi tekst isključivo vezan uz temu članka. Takvom tekstu se sva slova pretvaraju u mala slova te su rečenice podijeljene u svoje redove, koristeći interpunkcijske znakove kao separatore. Svaki red predstavlja „jednostavnu“ rečenicu iz koje se lako stvaraju n-torke. Proces je isti ako je korisnik unio svoj proizvoljni tekst – sva se slova pretvaraju u mala te se rečenice odvajaju.

# 4. METODE ODABIRA I POVEZIVANJA N-TORKI

N-torka je konačan niz *n* objekata, a ovdje ona predstavlja niz *n* susjednih riječi. Čisti tekst iz kraja prošlog poglavlja, nakon što je u potpunosti obrađen, se dijeli na n-torke. Svaka n-torka koja se u potpunosti ne sastoji od znakova abecede je odbačena. Za n-torke pretpostavka je da će za velike *n* biti lošija preciznost klasifikatora jer su manje šanse da će se isti skup većeg broja susjednih riječi ponoviti.

U nastavku su navedene metode s kojima je pokušano ili ukloniti nebitne n-torke ili spojiti n-torke koje nemaju isti oblik, ali imaju isto značenje.

## 4.1. Uklanjanje nebitnih riječi

Pod nebitnim riječima smatraju se one riječi iz kojih se ne bih moglo razaznati o čemu se piše u nekom tekstu, tj. koja je njegova tema. Za uklanjanje takvih riječi napravljena je tekstualna datoteka koja sadrži što više zamjenica, brojeva, priloga, prijedloga, veznika, čestica i usklika (pronađeni na [7]). Tekstualna datoteka je ručno pregledana te su izostavljene riječi koje bi ipak mogle imati kontekstualnu važnost. Jedan primjer toga je riječ „oko“ koja je i prijedlog ali i imenica. Ako je u n-torci pronađena nebitna riječ, n-torka se preskače (slika 4.1.).

```
# Preskoči n-torku ako sadrži riječ koja
# se smatra kao nebitna (veznici, čestice...)
for word in n_gram:
    if word in invalid_words:
        valid = False
        break
```

*Slika 4.1. Izbacivanje n-torke ako sadrži nebitnu riječ („invalid_words“ je lista koja sadrži sve nebitne riječi)*

## 4.2. Automatsko određivanje korijena riječi

Ova metoda pokušava algoritmima povezati riječi koje nemaju isti oblik, jer su u drugom padežu ili glagolskom vremenu, ali imaju isto značenje, tj. predstavljaju isti pojam.

### 4.2.1. Prefiks-sufiks algoritam

Sufiksi imenica, pridjeva i glagola prikupljeni su s web-stranica „Hrvatske školske gramatike“ [7] u jednu tekstualnu datoteku i ako su pronađeni na kraju bilo koje riječi, uklonjeni su. Na isti način uklonjeni su i prefiksi čija je lista pronađena na Wikipediji [8]. Prefiksi i sufiksi su sortirani po duljini kako bi se osiguralo da se ukloni najduži mogući dio riječi. Pronađeni su koristeći Python metode *startswith(prefix)* i *endswith(suffix)* te uklonjeni metodama *removeprefix(prefix)* i *removesuffix(suffix)* koje se pozivaju na *string* objektima.

Kod prefiks-sufiks algoritma skoro je neophodno uklanjanje nebitnih riječi zbog njihove kratke duljine. N-torke stvorene iz veznika „pa“, „te“, „ni“ ili „ali“ postaju potpuno beznačajne kad im se uklone sufiksi „a“, „e“ te „i“ – česti sufiksi deklinacije većine imenica. Iako uklanjanje nebitnih riječi pomaže u preciznosti klasifikacije, ne rješava problem preagresivnog uklanjanja sufiksa.

Sufiks „ama“ se dodaje na korijen imenica ženskog roda u dativu, lokativu i instrumentalu množine (e-sklonidba, slika 4.2.) te ga napravljena tekstualna datoteka sufiksa sadrži. Problem stvaraju riječi koje završavaju na „ama“, ali nisu u navedenim padežima. „Mama“, „jama“, „slama“, „tama“ i „dama“ su riječi u nominativu jednine te uklanjanjem sufiksa „ama“ dobiju se jednoslovne n-torke koje su potpuno neupotrebljive. Ovo je samo jedan od mnogih primjera preagresivnog uklanjanja sufiksa. Način na koji bi se ovaj problem mogao riješiti je da se uklanjanje prefiksa/sufiksa dopusti jedino ako duljina rezultata (broj slova) nije premala.

| | jednina | množina |
|---|---|---|
| N | žena | žene |
| G | žene | žena |
| D | ženi | ženama |
| A | ženu | žene |
| V | ženo | žene |
| L | ženi | ženama |
| I | ženom | ženama |

*Slika 4.2. E-sklonidba imenica [7]*

### 4.2.2. Određivanje korijena riječi korištenjem skupa podataka

Umjesto da se ručno pokušavaju ukloniti prefiksi i sufiksi, lakše bi bilo pronaći postojeći skup podataka koji sadrži sve moguće oblike svih riječi uz njihov osnovni oblik.

Wječnik [9] je internet rječnik zasnovan na dobrovoljnim dodacima i izmjenama korisnika te je „sestrinski projekt Wikipedije“. Sadrži više od 10 000 hrvatskih riječi [9] – većinom imenica. Uspoređujući to s najmanje 250 000 riječi koje hrvatski jezik zapravo ima [10], činjenicu da velik broj web-stranica tih riječi uopće ne sadrži njihovu deklinaciju i da su tablice deklinacija formatirane nekonzistentno, ovaj skup podataka se ne čini kao dobar kandidat za implementaciju. Pored Wječnik-a pronađen je i GitHub repozitorij [11] koji sadrži tekstualnu datoteku s velikim brojem hrvatskih riječi uključujući njihove deklinacije, no taj skup podataka također pati od problema loše formatiranosti.

croDict [12] je web-sjedište koje služi kao prevoditelj riječi s engleskog i njemačkog na hrvatski. Osim toga sadrži i popise imenica te glagola na tim jezicima, uključujući i njihove deklinacije, tj. konjugacije. U polje za pretraživanje se upiše riječ kojoj se traži osnovni oblik i izabere se je li imenica ili glagol. Jer je potrebno da osnovni oblik riječi dohvati program, nije moguće koristiti se elementima sučelja web-stranice. Srećom, osim polja za pretraživanje i opcije za vrstu riječi, informacije o riječi moguće je definirati unutar poveznice (slika 4.3.). Jer program ne zna odrediti vrstu riječi potrebno je isprobati obje opcije. U slučaju da traženu riječ ne prepoznaje, na vrhu web-stranice se ispiše: „Der gesuchte Begriff konnte nicht gefunden werden“,

što je prevedeno s njemačkog: „Traženi pojam nije pronađen“. Nakon slanja zahtjeva za HTML kodom i njegovog primitka, ako program nije pronašao upozorenje o nepronalasku, pokušava dohvatiti riječ iznad tablica koja predstavlja osnovni oblik unesene riječi – nominativ jednine za imenice te infinitiv za glagole (slika 4.4.).


*Slika 4.3. Poveznica web-stranice s označenim riječima (lijevo – vrsta riječi (imenica/glagol), desno – riječ kojoj tražimo osnovni oblik) koje se izmjenjuju za dobitak osnovnog oblika riječi*

**Deklinacija od *igračka***

**Jednina i množina** od *„igračkama“*

| | *Jednina* | *Množina* |
|---|---|---|
| **Nominativ** | igračka | igračke |
| **Genitiv** | igračke | igračaka / igračka / igrački |
| **Dativ** | igrački / igračci | igračkama |
| **Akuzativ** | igračku | igračke |
| **Lokativ** | igrački / igračci | igračkama |
| **Vokativ** | igračko | igračke |
| **Instrumental** | igračkom | igračkama |

*Slika 4.4. Isječak web-stranice poveznice sa slike 4.3. – osnovni oblik riječi ispisan na vrhu [12]*

Kako bi se olakšalo opterećenje croDict poslužitelju, umanjila šansa zabrane pristupa web-stranicama zbog prevelikog broja zahtjeva te ubrzala klasifikacija, odlučeno je da će se rezultati upita web-stranici spremiti lokalno u *json* datoteku, gdje je ključ riječ s kojom pretražujemo, a vrijednost je osnovni oblik te riječi (slika 4.5.). Sada, ako se ponovno traži osnovni oblik iste riječi, neće se ponovno slati zahtjev nego će se uzeti iz lokalnog rječnika. Za još veće ubrzanje programa i umanjenje broja zahtjeva, umjesto da program spremi samo riječ s kojom je pretraživano i njen osnovni oblik, preuzme i spremi sve oblike riječi te ih poveže.

```
"motiv": "motiv",
"motivi": "motiv",
"motiva": "motiv",
"motivu": "motiv",
"motivima": "motiv",
"motive": "motiv",
"motivom": "motiv",
```

*Slika 4.5. Isječak „json“ datoteke koji prikazuje sve oblike riječi „motiv“ spremljene u rječnik*

Rezultat je metoda koja može spojiti sve oblike velikog broja imenica i glagola, čak i onih s nepravilnim deklinacijama/konjugacijama. Na slici 4.6. se vidi definirani „tekst“ *string* koji sadrži imenice nepravilnih deklinacija te ispis vektora tog teksta koji je stvoren funkcijom *get_vector_from_simple_sentences()* (koja u pozadini koristi skup podataka stvoren iz croDict-a).

```
tekst = "vrabac vrapci ronilac ronioca orahu orasi zadatkom zadaci"
vektor = get_vector_from_simple_sentences([tekst])

print(vektor)
>{'vrabac': 2, 'ronilac': 2, 'orah': 2, 'zadatak': 2}
```

*Slika 4.6. Primjer kako se korištenjem croDict skupa podataka povezuju različiti oblici riječi*

Ako za zadanu riječ croDict nije mogao pronaći valjani osnovni oblik, ta riječ je lokalno spremljena u tekstualnu datoteku, kako bi se spriječilo ponovno slanje zahtjeva stranici.

# 5. METODE IZRAČUNA VJEROJATNOSTI

Nakon što su vektori svih dokumenata definirani, sadržeći n-torke tih dokumenata i njihov broj ponavljanja, putem njih je potrebno izračunati vjerojatnost da je testirani dokument neke kategorije (teme). Načini na koje su vektori klasifikatora i vektor testiranog dokumenta uspoređeni i obrađeni te različite formule izračuna preciznosti su navedene u nastavku.

## 5.1. Binarno bodovanje

Najjednostavniju metodu izračuna vjerojatnosti zapravo niti ne zanima broj ponavljanja n-torke nego samo njeno postojanje, otkuda i ime metode [3]. Stoga, u vektoru dokumenta, broj ponavljanja možemo zamijeniti jedinicom. Za svaku kategoriju napravljen je skup koji sadrži jedinstvene n-torke koje se pojavljuju i u vektoru te kategorije i u vektoru testiranog dokumenta. Za izračun vjerojatnosti da je testirani dokument neke kategorije, veličina skupa kategorije podijeljena je sa zbrojem veličina skupova svih kategorija. Formula (5.1) prikazuje izračun te vjerojatnosti ($P_k$) u obliku postotka.

$$P_k = \frac{|V_k \cap V_{test}|}{\sum_{k\prime \in K} |V_{k\prime} \cap V_{test}|} * 100, \quad \sum_{k' \in K} |V_{k'} \cap V_{test}| > 0 \tag{5.1}$$

gdje je:

| | |
|---|---|
| $k$ | kategorija |
| $K$ | skup svih kategorija |
| $V_k$ | skup jedinstvenih n-torki kategorije $k$ |
| $V_{test}$ | skup jedinstvenih n-torki testiranog dokumenta |

## 5.2. Broj ponavljanja

Kod ove metode nije samo važno da n-torka postoji u dokumentima, nego je važan i njen broj ponavljanja. Kao i prije, za svaku kategoriju napravljen je skup koji sadrži jedinstvene n-torke

koje se pojavljuju i u vektoru te kategorije i u vektoru testiranog dokumenta, no sada je potrebno zadržati njihove količine ponavljanja. Za određenu n-torku skupa uspoređuju se količine ponavljanja unutar vektora kategorije i vektora testiranog dokumenta, te se uzima ona koja je manja. Zbrojem tih manjih vrijednosti svih n-torki skupa kategorije dobivamo „rezultat“ kategorije. Vjerojatnost da testirani dokument pripada nekoj kategoriji računa se dijeljenjem „rezultata“ te kategorije sa zbrojem „rezultata“ svih kategorija. U formuli (5.2) vidi se izračun vjerojatnosti $P_k$ (u obliku postotku) da testirani dokument pripada kategoriji *k*.

$$P_k = \frac{\sum_{t \in (V_k \cap V_{test})} min[v_k(t),\ v_{test}(t)]}{\sum_{k' \in K} \sum_{t \in (V_{k'} \cap V_{test})} min[v_{k'}(t),\ v_{test}(t)]} * 100, \qquad (5.2)$$

$$\sum_{k' \in K} \sum_{t \in (V_{k'} \cap V_{test})} min[v_{k'}(t),\ v_{test}(t)] > 0$$

gdje je:

| | |
|---|---|
| $t$ | n-torka |
| $k$ | kategorija |
| $K$ | skup svih kategorija |
| $V_k$ | skup jedinstvenih n-torki kategorije *k* |
| $V_{test}$ | skup jedinstvenih n-torki testiranog dokumenta |
| $v_k(t)$ | broj ponavljanja n-torke *t* u kategoriji *k* |
| $v_{test}(t)$ | broj ponavljanja n-torke *t* u testiranom dokumentu |

## 5.3. „tf-idf“

„Term frequency-inverse document frequency“ ili skraćeno *tf-idf*, mjera je koja pokazuje koliko je neka n-torka važna u skupu dokumenata, a u kontekstu ovog rada, koliko je važna u kategoriji, tj. skupu dokumenata kategorije [3, 13]. Računa se umnoškom učestalosti n-torke (*tf*) u skupu dokumenata kategorije i inverznom učestalosti dokumenta n-torke (*idf*) u istom tom skupu dokumenata kategorije. Jednadžba (5.3) prikazuje taj umnožak, gdje *t* predstavlja n-torku, a *k* kategoriju.

$$tfidf(k,t) = tf(k,t) * idf(k,t) \qquad (5.3)$$

gdje je:

$tf(k,t)$ učestalost n-torke *t* u kategoriji *k*

$idf(k,t)$ inverzna učestalost dokumenta n-torke *t* u kategoriji *k*

Jer metoda *tf-idf* prebacuje važnost s n-torki koje se ponavljaju puno na n-torke koje se ponavljaju malo, riječi web-stranice koje nisu vezane uz članak, poput reklama, mogu poremetiti vektore klasifikatora i time i preciznost. Zbog istog tog razloga, ova bi metoda trebala moći smanjiti važnost nebitnih riječi koje se često ponavljaju, poput veznika i čestica, umanjujući potrebu za korištenjem već definirane metode za uklanjanje nebitnih riječi.

Učestalost n-torke (*tf*) relativna je učestalost ponavljanja n-torke s obzirom na ukupni broj n-torki dokumenta u kojem se nalazi. Frekvenciji se izračuna logaritam baze *10* kako bi se umanjila važnost n-torki koje se ponavljaju previše puta [13]. U formuli (5.4) se vidi izračun učestalosti n-torke *t* u dokumentu ili uniji dokumenata (ako je uvrštena kategorija) *d*.

$$tf(d,t) = \log_{10}\left(\frac{v_d(t)}{\sum_{t' \in V_d} v_d(t')}\right), \quad \sum_{t' \in V_d} v_d(t') > 0,\ v_d(t) > 0 \qquad (5.4)$$

gdje je:

$V_d$ skup jedinstvenih n-torki dokumenta *d*

$v_d(t)$ broj ponavljanja n-torke *t* u dokumentu *d*

Inverzna učestalost dokumenta (*idf*) n-torke odražava udio dokumenata u skupu svih dokumenata kategorije koji sadrže tu n-torku. Iako u kontekstu ovog rada to nije potrebno napraviti zbog malog broja dokumenata kategorije, i ovdje se također preporučuje izračun logaritma izraza [13]. Formula (5.5) za izračun inverzne učestalosti dokumenta je sljedeća:

$$idf(k,t) = \log_{10}\left(\frac{N_k}{N_{k,t}}\right), \quad N_k > 0,\ N_{k,t} > 0 \qquad (5.5)$$

gdje je:

$N_k$ broj dokumenata kategorije *k*

$N_{k,t}$ broj dokumenata kategorije *k* koje sadrže n-torku *t*

Za izračun koliko je neka n-torka testiranog dokumenta bitna u kategoriji, uzima se njena važnost u skupu dokumenata kategorije (*tf-idf*). Za povećanje izražaja važnosti n-torke, njen *tf-idf* iznos pomnožimo s njenom relativnom učestalosti (*tf*) u testiranom dokumentu. Te „važnosti" svih n-torki kategorije se zbroje, tvoreći „rezultat" kategorije. Taj rezultat kategorije se, kao i kod drugih metoda, normalizira – podijeli se sa zbrojem rezultata svih kategorija. Konačno, izračun vjerojatnosti (u postotku) da je testirani dokument $d_{test}$ kategorije $k$ je prikazan sljedećom jednadžbom (5.6):

$$P_k = \frac{\sum_{t \in (V_k \cap V_{test})} tfidf(k,t) * tf(d_{test},t)}{\sum_{k' \in K} \sum_{t \in (V_{k'} \cap V_{test})} tfidf(k',t) * tf(d_{test},t)} * 100, \qquad (5.6)$$

$$\sum_{k' \in K} \sum_{t \in (V_{k'} \cap V_{test})} tfidf(k',t) * tf(d_{test},t) > 0$$

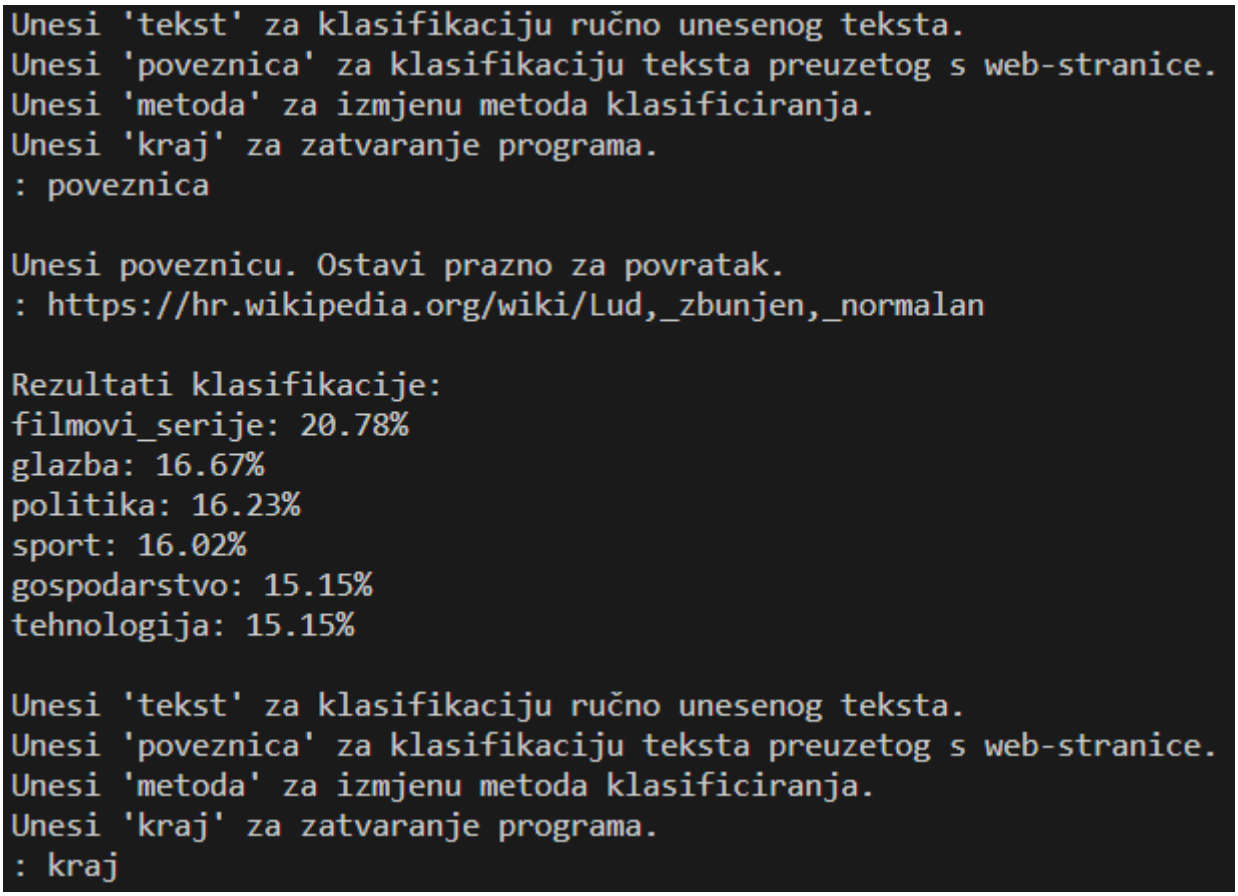
gdje je:

| | |
|---|---|
| $t$ | n-torka |
| $d_{test}$ | testirani dokument |
| $k$ | kategorija |
| $K$ | skup svih kategorija |
| $V_k$ | skup jedinstvenih n-torki kategorije $k$ |
| $V_{test}$ | skup jedinstvenih n-torki testiranog dokumenta |
| $tf(d,t)$ | učestalost n-torke $t$ u dokumentu $d$ |
| $tfidf(k,t)$ | učestalost n-torke - inverzna učestalost dokumenta n-torke $t$ u kategoriji $k$ |

# 6. PROGRAMSKA OPREMA

S GitHub repozitorija (https://github.com/aZina0/bag_of_words_model) klasifikator se može preuzeti i pokrenuti na računalima operacijskog sustava Windows.

Pri otvaranju programa dobiju se upute za upis jedne od četiri riječi (slika 6.1.). Te riječi su „tekst“, „poveznica“, „metoda“ i „kraj“. Unosom „tekst“ korisniku je omogućen unos proizvoljnog teksta kojemu želi odrediti kategoriju. Poželjno je da ovaj tekst bude što duži jer pomaže u preciznosti klasifikacije jer su šanse za preklapanje n-torki veće. Umjesto proizvoljnog teksta, moguće je unijeti poveznicu bilo kakvog internet dokumenta/članka. Ipak, preporučuje se unos poveznica članaka portala 24sata ili Novog lista te hrvatske Wikipedije zbog razloga koji su navedeni ranije (poglavlje 3.1.). Nakon unosa proizvoljnog teksta ili poveznice, pritiskom na tipku „enter“, ispisuju se teme uz njihove vjerojatnosti u obliku postotka, prema kojima su i sortirane. Unosom „metoda“ dobiju se upute za izmjenu metoda izračuna vjerojatnosti. Jer su početne postavke metoda izabrane na temelju zaključka ovog rada, trebale bi dati najbolje rezultate. Unosom riječi „kraj“ program se zatvori.




*Slika 6.1. Prozor programa sa zahtjevom klasifikacije*

Korišten je programski jezik „Python“ zbog lakšeg rukovanja sa *string* objektima, ali i općenito jednostavnije sintakse. Za klasifikaciju teksta s oko 5500 riječi programu je potrebno u prosjeku *~0.27* sekundi. Korištenjem najsloženijih metoda to vrijeme poraste na *~0.32* sekunde.

Nakon svakog upita korisnika i izračuna preciznosti klasifikacije, automatski se izradi tekstualna datoteka „rezultat.txt“ (slika 6.2.) koja sadrži sve kategorije/teme sortirane po vjerojatnosti i njihove n-torke koje se preklapaju s testiranim dokumentom uz pripadajuće bodove, po kojima su i sortirane. Pregledavanjem sadržaja datoteke može se zaključiti zašto program odluči da je vjerojatnost jedne kategorije veća od druge, te će često biti korištena za objašnjavanje dobivenih rezultata.

```
sport: 23.41% -> 155/662
-------------------------
igrati: 7
igrač: 4
real: 4
moći: 4
utakmica: 4
sezona: 4
modrić: 3
godina: 3
prvenstvo: 3
minuta: 3
luka: 2
momčad: 2
trener: 2
kolo: 2
athletic: 2
almerije: 2
nam: 2
sada: 2
```

*Slika 6.2. Isječak s početka tekstualne datoteke koja prikazuje opširniji rezultat klasifikacije uz broj ponavljanja n-torki svih kategorije*

# 7. REZULTATI

Za usporedbu efikasnosti kombinacija metoda kojima je pokušano maksimizirati preciznost, klasificiran je po jedan dovoljno velik hrvatski Wikipedija članak iz svake od ukupno šest kategorija. Klasificiranja se uvijek koriste nekom „vrstom“ n-torki (1-torke, 2-torke...) za podjelu dokumenata klasifikatora i testiranog dokument na dijelove te je to radi kratkoće jednostavno napisano: „klasificiranje korištenjem (npr.) 1-torke“. Rezultati su prikazani u tablicama, gdje redovi predstavljaju unesene članke, a stupci kategorije članaka. Vrijednosti unutar ćelija su vjerojatnosti u postotcima da je članak neke kategorije. Vjerojatnosti za točnu kategoriju su podebljane (tvore dijagonalu). Boja pozadine svake ćelija je zatamnjena ovisno o unesenoj vjerojatnosti kako bi se lakše moglo vidjeti koji su postotci veći. Kasnije taj format će se promijeniti i bit će prikazane samo vjerojatnosti za točnu kategoriju. Za kategoriju filmovi/serije je uzet članak serije „Lud, zbunjen, normalan“, za glazbu članak Olivera Dragojevića, za gospodarstvo članak o inflaciji, za politiku članak predsjednika Zorana Milanovića, za sport članak Luke Modrića i za tehnologiju članak o računalima.

## 7.1. Neobrađene n-torke

Tablice 7.1., 7.2., 7.3. i 7.4. prikazuju rezultate klasifikacija članaka korištenjem 1-torki, 2-torki, 3-torki i 4-torki koje su binarno bodovane. Nikakve metode povezivanja sličnih n-torki nisu korištene.

*Tablica 7.1. Rezultati klasifikacije korištenjem 1-torki i binarnog bodovanja*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **20,63%** | 17,86% | 14,68% | 16,27% | 15,48% | 15,08% |
| Oliver Dragojević | 16,67% | **24,28%** | 14,95% | 14,13% | 14,86% | 15,13% |
| Inflacija | 12,73% | 16,50% | **22,24%** | 16,08% | 13,15% | 19,30% |
| Zoran Milanović | 16,24% | 17,52% | 17,36% | **18,31%** | 14,33% | 16,24% |
| Luka Modrić | 15,58% | 18,25% | 14,88% | 14,83% | **20,88%** | 15,58% |
| Računalo | 14,48% | 18,17% | 17,51% | 15,47% | 12,03% | **22,34%** |

*Tablica 7.2. Rezultati klasifikacije korištenjem 2-torki i binarnog bodovanja*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **21,79%** | 21,79% | 12,82% | 16,67% | 15,38% | 11,54% |
| Oliver Dragojević | 16,25% | **23,14%** | 16,53% | 12,67% | 14,88% | 16,53% |
| Inflacija | 7,74% | 14,29% | **27,98%** | 16,67% | 11,90% | 21,43% |
| Zoran Milanović | 18,89% | 17,22% | 18,33% | **18,89%** | 12,78% | 13,89% |
| Luka Modrić | 14,96% | 18,90% | 14,85% | 13,50% | **23,17%** | 14,62% |
| Računalo | 12,86% | 21,12% | 18,69% | 13,11% | 11,65% | **22,57%** |

*Tablica 7.3. Rezultati klasifikacije korištenjem 3-torki i binarnog bodovanja*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **14,29%** | 28,57% | 28,57% | 14,29% | 0,00% | 14,29% |
| Oliver Dragojević | 9,09% | **36,36%** | 27,27% | 9,09% | 18,18% | 0,00% |
| Inflacija | 7,69% | 23,08% | **23,08%** | 15,38% | 7,69% | 23,08% |
| Zoran Milanović | 21,43% | 14,29% | 7,14% | **21,43%** | 21,43% | 14,29% |
| Luka Modrić | 12,66% | 22,78% | 16,46% | 11,39% | **26,58%** | 10,13% |
| Računalo | 16,67% | 20,83% | 8,33% | 12,50% | 8,33% | **33,33%** |

*Tablica 7.4. Rezultati klasifikacije korištenjem 4-torki i binarnog bodovanja*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **0,00%** | 0,00% | 0,00% | 0,00% | 0,00% | 0,00% |
| Oliver Dragojević | 0,00% | **0,00%** | 0,00% | 0,00% | 0,00% | 0,00% |
| Inflacija | 0,00% | 0,00% | **0,00%** | 0,00% | 0,00% | 0,00% |
| Zoran Milanović | 0,00% | 0,00% | 0,00% | **0,00%** | 100,00% | 0,00% |
| Luka Modrić | 0,00% | 0,00% | 20,00% | 20,00% | **60,00%** | 0,00% |
| Računalo | 0,00% | 0,00% | 0,00% | 0,00% | 0,00% | **0,00%** |

Može se primijetiti da korištenjem najprimitivnijeg bodovanja n-torki, bez uklanjanja nebitnih riječi i bez automatskog povezivanja sličnih riječi, klasifikator za 1-torke točno određuje teme svih članaka, iako nije posve siguran. Kod 2-torki točno određuje tri članka, a za članke „Lud, zbunjen, normalan“ i „Zoran Milanović“ klasifikator najviše vjerojatnosti daje dvjema kategorijama. Kod 3-torki dobivamo uglavnom krive rezultate. Zbog jako maloga preklapanja 4-torki (slika 7.1.), vjerojatnosti u tablici 7.4. su jako nepouzdane. Ipak Wikipedija članak Luke Modrića je točno svrstan kao „sport“. Može se vidjeti da od svih 4-torki koje se preklapaju, samo jedna („na svjetskom prvenstvu u“) zapravo ima veze sa svojom kategorijom (sportom).

```
SPORT: 60.00% -> 3/5
'na svjetskom prvenstvu u'
'nakon što je u'
'je još samo jedan'

GOSPODARSTVO: 20.00% -> 1/5
'nakon što je u'

POLITIKA: 20.00% -> 1/5
'nakon što je u'

FILMOVI_SERIJE: 0.00% -> 0/5

GLAZBA: 0.00% -> 0/5

TEHNOLOGIJA: 0.00% -> 0/5
```

*Slika 7.1. 4-torke koje se nalaze u Wikipedija članku Luke Modrića i u člancima kategorija klasifikatora*

U sljedeće četiri tablice (7.5., 7.6., 7.7. i 7.8.) mogu se vidjeti rezultati klasifikatora korištenjem metode broja ponavljanja. Ovdje također nisu korištene metode uklanjanja nebitnih n-torki niti metode njihovog povezivanja.

*Tablica 7.5. Rezultati klasifikacije korištenjem 1-torki i metode broja ponavljanja*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **20,78%** | 16,67% | 15,15% | 16,23% | 16,02% | 15,15% |
| Oliver Dragojević | 16,67% | **20,45%** | 15,98% | 15,28% | 15,87% | 15,76% |
| Inflacija | 13,65% | 15,98% | **21,41%** | 16,24% | 14,29% | 18,43% |
| Zoran Milanović | 16,12% | 16,91% | 17,03% | **18,24%** | 15,58% | 16,12% |
| Luka Modrić | 15,45% | 17,54% | 16,04% | 15,44% | **19,00%** | 16,52% |
| Računalo | 14,93% | 17,49% | 17,20% | 16,30% | 14,08% | **20,00%** |

*Tablica 7.6. Rezultati klasifikacije korištenjem 2-torki i metode broja ponavljanja*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **23,75%** | 21,25% | 12,50% | 16,25% | 15,00% | 11,25% |
| Oliver Dragojević | 16,35% | **23,56%** | 16,11% | 12,50% | 15,38% | 16,11% |
| Inflacija | 8,56% | 14,44% | **27,81%** | 16,04% | 11,23% | 21,93% |
| Zoran Milanović | 18,53% | 16,81% | 18,10% | **19,40%** | 12,93% | 14,22% |
| Luka Modrić | 15,18% | 18,45% | 15,32% | 13,51% | **22,22%** | 15,32% |
| Računalo | 13,39% | 20,89% | 18,05% | 13,18% | 11,97% | **22,52%** |

*Tablica 7.7. Rezultati klasifikacije korištenjem 3-torki i metode broja ponavljanja*

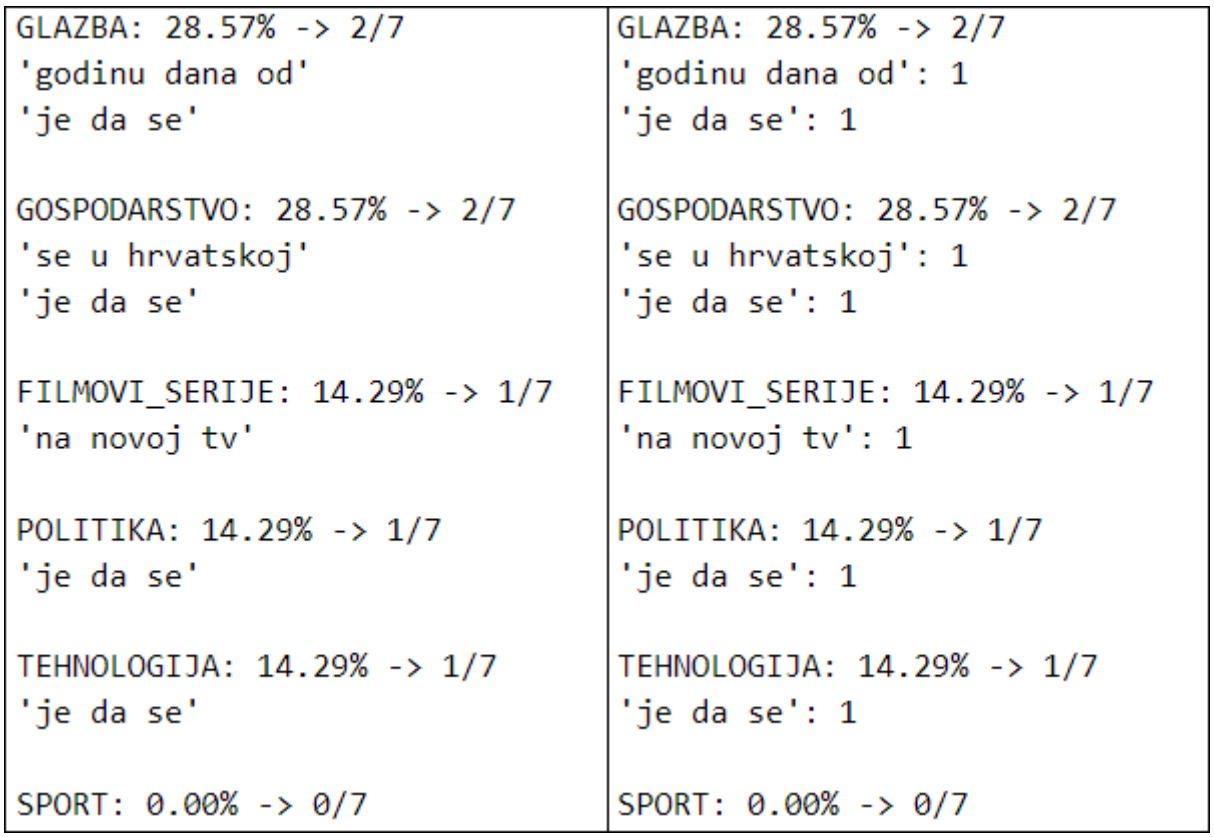
| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **14,29%** | 28,57% | 28,57% | 14,29% | 0,00% | 14,29% |
| Oliver Dragojević | 9,09% | **36,36%** | 27,27% | 9,09% | 18,18% | 0,00% |
| Inflacija | 6,67% | 20,00% | **26,67%** | 13,33% | 6,67% | 26,67% |
| Zoran Milanović | 20,00% | 13,33% | 6,67% | **26,67%** | 20,00% | 13,33% |
| Luka Modrić | 10,64% | 19,15% | 18,09% | 13,83% | **29,79%** | 8,51% |
| Računalo | 17,86% | 17,86% | 7,14% | 17,86% | 10,71% | **28,57%** |

*Tablica 7.8. Rezultati klasifikacije korištenjem 4-torki i metode broja ponavljanja*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **0,00%** | 0,00% | 0,00% | 0,00% | 0,00% | 0,00% |
| Oliver Dragojević | 0,00% | **0,00%** | 0,00% | 0,00% | 0,00% | 0,00% |
| Inflacija | 0,00% | 0,00% | **0,00%** | 0,00% | 0,00% | 0,00% |
| Zoran Milanović | 0,00% | 0,00% | 0,00% | **0,00%** | 100,00% | 0,00% |
| Luka Modrić | 0,00% | 0,00% | 28,57% | 14,29% | **57,14%** | 0,00% |
| Računalo | 0,00% | 0,00% | 0,00% | 0,00% | 0,00% | **0,00%** |

Uspoređivanjem vjerojatnosti tablice 7.1. i tablice 7.5. (1-torke), može se vidjeti da korištenjem metode brojanja ponavljanja n-torki, klasifikator još uvijek točno određuje teme članaka, no u tu najveću vjerojatnost je nesigurniji za *~1,52* postotna poena u prosjeku. Tablica 7.2. i tablica 7.6. prikazuju da je kod 2-torki bodovanje brojem ponavljanja bolje nego binarno bodovanje jer svakom članku točno određuje temu te je u tu najveću vjerojatnost sigurniji za *~0,29* postotna poena u prosjeku. Klasifikacije 3-torki (tablica 7.3. i tablica 7.7.) za članke „Lud, zbunjen, normalan“ i „Oliver Dragojević“ imaju identične vjerojatnosti za svaku kategoriju. Na slici 7.2. se vidi zašto su vjerojatnosti iste – 3-torke se ponavljaju samo jednom pa preslikavaju binarno bodovanje. Tablica 7.8. prikazuje kako rezultati klasificiranja korištenjem 4-torki opet imaju jako nepouzdane rezultate, pa oni neće više biti prikazani.

Ako se ignoriraju 4-torke, zbrajanjem prosjeka promjene vjerojatnosti dobije se da je ukupni rast vjerojatnosti u točnu kategoriju prosječno jednak *~0.03* postotna poena. Uzimajući to u obzir i činjenicu da je većem broju članaka točno određena tema, može se zaključiti da je brojanje ponavljanja n-torki preciznije u kontekstu neobrađenih n-torki.




*Slika 7.2. 3-torke Wikipedija članka „Lud, zbunjen, normalan“ koristeći binarno bodovanje (lijevo) i metodu broja ponavljanja (desno)*

Sljedeće tablice (7.9., 7.10. i 7.11.) prikazuju vjerojatnosti koje su izračunate *tf-idf* metodom.

*Tablica 7.9. Rezultati klasifikacije korištenjem 1-torki i „tf-idf“ metode*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **23,73%** | 19,62% | 12,36% | 17,10% | 13,70% | 13,50% |
| Oliver Dragojević | 16,80% | **25,38%** | 14,94% | 13,98% | 14,25% | 14,66% |
| Inflacija | 11,78% | 17,04% | **22,82%** | 15,98% | 12,36% | 20,02% |
| Zoran Milanović | 16,90% | 17,66% | 17,39% | **17,69%** | 13,36% | 17,00% |
| Luka Modrić | 15,51% | 18,71% | 14,55% | 15,15% | **20,79%** | 15,28% |
| Računalo | 14,59% | 17,94% | 17,56% | 15,47% | 11,20% | **23,25%** |

*Tablica 7.10. Rezultati klasifikacije korištenjem 2-torki i „tf-idf“ metode*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **22,16%** | 23,34% | 11,20% | 16,76% | 15,24% | 11,29% |
| Oliver Dragojević | 16,51% | **24,09%** | 16,11% | 12,31% | 14,48% | 16,49% |
| Inflacija | 6,29% | 13,68% | **30,29%** | 17,38% | 11,72% | 20,64% |
| Zoran Milanović | 19,91% | 17,93% | 18,10% | **18,28%** | 12,21% | 13,57% |
| Luka Modrić | 15,17% | 19,65% | 14,32% | 13,16% | **23,61%** | 14,09% |
| Računalo | 12,81% | 22,66% | 18,60% | 11,95% | 10,56% | **23,42%** |

*Tablica 7.11. Rezultati klasifikacije korištenjem 3-torki i „tf-idf“ metode*

| | filmovi/ serije | glazba | gospodarstvo | politika | sport | tehnologija |
|---|---|---|---|---|---|---|
| Lud, zbunjen, normalan | **12,35%** | 25,22% | 34,09% | 11,05% | 0,00% | 17,29% |
| Oliver Dragojević | 8,92% | **38,32%** | 28,23% | 9,12% | 15,41% | 0,00% |
| Inflacija | 8,77% | 24,45% | **22,21%** | 17,94% | 8,89% | 17,74% |
| Zoran Milanović | 25,03% | 15,31% | 3,86% | **18,22%** | 22,86% | 14,72% |
| Luka Modrić | 12,22% | 24,59% | 16,34% | 11,00% | **25,24%** | 10,61% |
| Računalo | 14,91% | 23,01% | 8,63% | 11,95% | 7,58% | **33,92%** |

U tablicama se vidi da je, uspoređujući s prijašnje dvije metode izračuna vjerojatnosti (ne računajući 4-torke), metoda *tf-idf* u prosjeku nešto bolja – u smislu davanja većih vjerojatnosti točnim kategorijama. No, sveukupno gledajući metoda *tf-idf* je točno odredila kategorije manjem broju članaka. U binarnom bodovanju, članci koji su imali iste najviše vjerojatnosti za više kategorija, njih 4, su u *tf-idf* metodi krivo klasificirani. Uspoređujući tablice 3-torki (tablice 7.3., 7.7. i 7.11.) može se vidjeti da uglavnom prate isti uzorak, ali je uočljivo da metoda *tf-idf* daje lošije rezultate. Vjerojatnost da je članak serije „Lud, zbunjen, normalan“ kategorije „gospodarstvo“ je *34.09%* – zasad najveća vjerojatnost koja je dana krivoj kategoriji (ako se ignoriraju 4-torke).

## 7.2. Uklonjene nebitne n-torke

Tablice svih sljedećih potpoglavlja prikazivat će rezultate klasifikacija u kompaktnijem obliku zbog velikog broja mogućih kombinacija metoda. Svaka tablica će sadržavati klasifikaciju svakom metodom izračuna vjerojatnosti i bit će prikazane samo one vjerojatnosti koje su dane ispravnoj kategoriji. Prateći pravila prijašnjih tablica, ti iznosi vjerojatnosti će biti podebljani. Ako je prije iznosa vjerojatnosti prikazan simbol „*“ to znači da ta vjerojatnost nije najveća dana vjerojatnost članku – nekoj drugoj kategoriji vjerojatnost je veća ili jednaka, tj. članak nije ispravno klasificiran. Zbog loših rezultata klasificiranja tekstova korištenjem 3-torki i 4-torki, one neće biti prikazane.

Tablice 7.12. i 7.13. prikazuju rezultate klasifikacija testiranih dokumenata korištenjem 1-torki i 2-torki te metode uklanjanja nebitnih riječi – zamjenica, brojeva, priloga, prijedloga, veznika, čestica i usklika. Cijela n-torka je odbačena ako sadrži ijednu nebitnu riječ.

*Tablica 7.12. Rezultati klasifikacije korištenjem 1-torki (nebitne uklonjene) i svih metoda izračuna vjerojatnosti*

| | binarno bodovanje | broj ponavljanja | tf-idf |
|---|---|---|---|
| Lud, zbunjen, normalan | **24,39%** | **29,05%** | **24,08%** |
| Oliver Dragojević | **28,30%** | **27,22%** | **28,08%** |
| Inflacija | **25,16%** | **28,89%** | **23,99%** |
| Zoran Milanović | **19,14%** | **20,97%** | ***** **17,78%** |
| Luka Modrić | **22,31%** | **24,71%** | **21,18%** |
| Računalo | **24,60%** | **25,28%** | **24,70%** |
| prosjek | **23,98%** | **26,02%** | **23,30%** |

*Tablica 7.13. Rezultati klasifikacije korištenjem 2-torki (nebitne uklonjene) i svih metoda izračuna vjerojatnosti*

| | binarno bodovanje | broj ponavljanja | tf-idf |
|---|---|---|---|
| Lud, zbunjen, normalan | **50,00%** | **50,00%** | **45,17%** |
| Oliver Dragojević | **35,29%** | **35,29%** | **32,92%** |
| Inflacija | **58,33%** | **57,14%** | **59,18%** |
| Zoran Milanović | **35,29%** | **40,91%** | **31,10%** |
| Luka Modrić | **54,55%** | **56,00%** | **53,26%** |
| Računalo | *** 22,22%** | *** 22,22%** | *** 23,07%** |
| prosjek | **42,61%** | **43,59%** | **40,78%** |

Uspoređujući tablicu 7.13. s tablicama neobrađenih n-torki, vidi se da je prosjek vjerojatnosti koje su dane točnim kategorijama porastao skoro dva puta. Razlog tome je vidljiv na slici 7.3. – najčešće neobrađene n-torke sadrže „nebitnu“ riječ. Slična usporedba vrijedi između tablica 7.13. i 7.14. – 1-torka je u prosjeku skoro dva puta lošija. Ignorirajući 3-torke i 4-torke, klasificiranje neobrađenih n-torki određuje krive kategorije istom broju članaka (4). Metoda izračuna vjerojatnosti *tf-idf* kod neobrađenih n-torki daje najbolje rezultate, a ovdje vrijedi obrnuto.

| | |
|---|---|
| SPORT: 22.22% -> 306/1377 | SPORT: 56.00% -> 28/50 |
| 'je u': 15 | 'svjetskom prvenstvu': 3 |
| 'koji je': 12 | 'milijuna eura': 2 |
| 'što je': 11 | 'hrvatske reprezentacije': 2 |
| 'nakon što': 6 | 'europskom prvenstvu': 1 |
| 'je bio': 6 | 'real madrid': 1 |
| 'prvenstvu u': 5 | 'dana kasnije': 1 |
| 'kada je': 5 | 'svjetskog prvenstva': 1 |
| 'bio je': 5 | 'lige prvaka': 1 |
| 'na svjetskom': 4 | 'najboljeg igrača': 1 |
| 'na europskom': 4 | 'najboljih igrača': 1 |
| 'svjetskom prvenstvu': 3 | 'bio najbolji': 1 |
| 'ugovor s': 3 | 'ove sezone': 1 |
| 'je i': 3 | 'luku modrića': 1 |
| 'koji su': 3 | 'izbornik hrvatske': 1 |
| 'je na': 3 | 'nekoliko mjeseci': 1 |
| 'je s': 3 | 'hrvatski reprezentativac': 1 |
| 'na kraju': 3 | 'prijelaznog roka': 1 |
| 'u kvalifikacijama': 3 | 'obrambenog igrača': 1 |
| 'dok je': 3 | 'hrvatski nogometaš': 1 |

*Slika 7.3. Najčešće 2-torke klasifikacije Wikipedija članka Luke Modrića koristeći metodu broja ponavljanja bez (lijevo) i s uklanjanjem nebitnih n-torki (desno)*

### 7.3. N-torke povezane prefiks-sufiks algoritmom i uklonjene nebitne n-torke

Sljedeće tablice (7.14. i 7.15.), na isti način, prikazuju rezultate gdje su članci klasificirani kombinacijom metoda povezivanja n-torki prefiks-sufiks algoritmom i uklanjanja nebitnih n-torki.

*Tablica 7.14. Rezultati klasifikacije korištenjem 1-torki (povezane prefiks-sufiks algoritmom i nebitne uklonjene) i svih metoda izračuna vjerojatnosti*

| | binarno bodovanje | broj ponavljanja | tf-idf |
|---|---|---|---|
| Lud, zbunjen, normalan | **24,49%** | **28,44%** | **26,23%** |
| Oliver Dragojević | **24,45%** | **25,25%** | **23,66%** |
| Inflacija | **24,43%** | **28,25%** | **24,58%** |
| Zoran Milanović | *** 18,22%** | **21,11%** | *** 16,75%** |
| Luka Modrić | **19,21%** | **23,54%** | *** 17,84%** |
| Računalo | **23,83%** | **24,33%** | **23,27%** |
| prosjek | **22,44%** | **25,15%** | **22,06%** |

*Tablica 7.15. Rezultati klasifikacije korištenjem 2-torki (povezane prefiks-sufiks algoritmom i nebitne uklonjene) i svih metoda izračuna vjerojatnosti*

| | binarno bodovanje | broj ponavljanja | tf-idf |
|---|---|---|---|
| Lud, zbunjen, normalan | **55,56%** | **55,56%** | **51,99%** |
| Oliver Dragojević | **30,77%** | **30,77%** | **27,92%** |
| Inflacija | **72,22%** | **70,00%** | **72,87%** |
| Zoran Milanović | *** 30,00%** | **34,62%** | *** 26,15%** |
| Luka Modrić | **54,00%** | **61,02%** | **52,26%** |
| Računalo | *** 30,77%** | *** 30,77%** | *** 31,71%** |
| prosjek | **45,55%** | **47,12%** | **43,82%** |

Uspoređujući tablice 7.12. i 7.13. s tablicama 7.14. i 7.15. vidi se da prefiks-sufiks algoritam nije od velike koristi – čak je u osam situacija članku krivo određena kategorija. Uspoređujući s tablicama neobrađenih n-torki, postotci su u prosjeku veći, ali to je zahvaljujući metodi uklanjanja nebitnih n-torki. U tablici 2-torki (7.15.) vidljivo je da vjerojatnosti (točne) kategorije članka o inflaciji prelaze *70%*, što je neobično visoko i po zaključku o efikasnosti prefiks-sufiks algoritma neutemeljeno. Metoda izračuna vjerojatnosti *tf-idf* je opet najlošija, a metoda broja ponavljanja je opet najbolja.

### 7.4. N-torke povezane skupom podataka i uklonjene nebitne n-torke

Sljedeće tablice (7.16. i 7.17.) prikazuju rezultate klasificiranja korištenjem metoda povezivanja n-torki skupom podataka i uklanjanja nebitnih n-torki.

*Tablica 7.16. Rezultati klasifikacije korištenjem 1-torki (povezane skupom podataka i nebitne uklonjene) i svih metoda izračuna vjerojatnosti*

| | binarno bodovanje | broj ponavljanja | tf-idf |
|---|---|---|---|
| Lud, zbunjen, normalan | **23,02%** | **27,27%** | **23,23%** |
| Oliver Dragojević | **25,15%** | **25,92%** | **24,89%** |
| Inflacija | **22,56%** | **27,50%** | **20,97%** |
| Zoran Milanović | **19,23%** | **21,03%** | **18,36%** |
| Luka Modrić | **19,16%** | **23,64%** | *** 17,83%** |
| Računalo | **23,73%** | **25,42%** | **22,97%** |
| prosjek | **22,14%** | **25,13%** | **21,38%** |

*Tablica 7.17. Rezultati klasifikacije korištenjem 2-torki (povezane skupom podataka i nebitne uklonjene) i svih metoda izračuna vjerojatnosti*

| | binarno bodovanje | broj ponavljanja | tf-idf |
|---|---|---|---|
| Lud, zbunjen, normalan | **55,56%** | **55,56%** | **52,01%** |
| Oliver Dragojević | **36,36%** | **36,36%** | **34,55%** |
| Inflacija | **42,86%** | **43,48%** | **42,53%** |
| Zoran Milanović | **35,00%** | **37,04%** | **33,23%** |
| Luka Modrić | **56,60%** | **60,34%** | **54,57%** |
| Računalo | *** 28,57%** | *** 28,57%** | *** 28,31%** |
| prosjek | **42,49%** | **43,56%** | **40,87%** |

Kombinacija metoda povezivanja n-torki skupom podataka i uklanjanja nebitnih n-torki daje bolje rezultate od kombinacije koja koristi prefiks-sufiks algoritam (tablice 7.14. i 7.15.) po sveukupnom broju točno određenih kategorija. Iznenađujuće je da prefiks-sufiks algoritam ima veće prosječne vjerojatnosti, iako bi teoretski trebao biti lošija metoda određivanja korijena riječi. Razlog tome leži u neobično visokim vjerojatnostima danim članku o inflaciji (tablica 7.15.) koje povećavaju prosjek.

Usporedbom rezultata klasifikacija ove kombinacije metoda (tablice 7.16. i 7.17.) i klasifikacije koja se koristi samo uklanjanjem nebitnih n-torki (tablice 7.12. i 7.13.) vidljivo je da, kao i prefiks-sufiks algoritam, korištenje skupa podataka za određivanje korijena riječi ima mali negativan utjecaj na prosječnu vjerojatnost točne kategorije. Na slici 7.4. vidi se kako je skupom podataka uspješno povezano više sličnih n-torki točne kategorije, no ta kategorija je dobila manju vjerojatnost jer su se u drugim kategorijama n-torke povezale više puta.

Performanse metoda izračuna vjerojatnosti prate isti obrazac, *tf-idf* je najlošija, a metoda broja ponavljanja je najbolja.

| | |
|---|---|
| GOSPODARSTVO: 57.14% -> 8/14 | GOSPODARSTVO: 43.48% -> 10/23 |
| 'druge strane': 2 | 'druge strana': 2 |
| 'najveći problem': 1 | 'najveći problem': 1 |
| 'mjerama monetarne': 1 | 'mjera monetarne': 1 |
| 'trenutno ima': 1 | 'republika hrvatska': 1 |
| 'dvoznamenkastu stopu': 1 | 'trenutno imati': 1 |
| 'proizvođačkih cijena': 1 | 'dvoznamenkastu stopa': 1 |
| 'negativne posljedice': 1 | 'indeks proizvođačkih': 1 |
| | 'proizvođačkih cijena': 1 |
| | 'negativne posljedica': 1 |

*Slika 7.4. Najčešće 2-torke (nebitne uklonjene) klasifikacije Wikipedija članka o inflaciji koristeći metodu broja ponavljanja bez (lijevo) i s korištenjem skupa podataka za povezivanje sličnih n-torki (desno)*

# 8. ZAKLJUČAK

Kombinacija metoda s najvećim prosječnim vjerojatnostima, a da je točno odredila sve kategorije članaka je ona koja koristi 1-torke i uklanjanje nebitnih n-torki te vjerojatnost računa metodom broja ponavljanja. Kombinacija metoda koja zapravo ima najveće prosječne vjerojatnosti je ona koja koristi 2-torke, prefiks-sufiks algoritam za povezivanje n-torki, uklanjanje nebitnih n-torki i istu metodu za izračun vjerojatnosti. Ta kombinacija je jednom članku (o računalima) krivo odredila kategoriju.

Iznenađujuće je što korištenje metoda za automatsko određivanje korijena riječi ima negativni utjecaj na performanse klasifikatora. Također, zanimljivo je koliko različite metode izračuna vjerojatnosti daju slične rezultate. Kompleksnost implementacije *tf-idf* metode nije opravdana jer je imala nešto lošije rezultate od ostalih metoda izračuna vjerojatnosti. Rezultati klasifikacija korištenjem 3-torki i 4-torki su razočaravajući zbog malog broja preklapanja među vektorima klasifikatora i vektora testiranog dokumenta.

Potrebno je reći da su navedeni zaključci doneseni iz konteksta napravljenog programa, koji koristi nesavršene članke koji tvore tekstove klasifikatora i nesavršene članke koji su klasificirani. Pretpostavka je da bi se većina prije navedenih „nedosljednosti“ mogla riješiti povećanjem obujma i broja tekstova klasifikatora.

# LITERATURA


[1] „Bag-of-words model“, s interneta, https://en.wikipedia.org/wiki/Bag-of-words_model, 20. kolovoza 2023.

[2] „Document Classification With Machine Learning: Computer Vision, OCR, NLP, and Other Techniques“, s interneta, https://www.altexsoft.com/blog/document-classification, 19. kolovoza 2023.

[3] „A Gentle Introduction to the Bag-of-Words Model“, s interneta, https://machinelearningmastery.com/gentle-introduction-bag-words-model/, 14. kolovoza 2023.

[4] „24sata“, s interneta, https://www.24sata.hr/, 15. kolovoza 2023.

[5] „Novi list“, s interneta, https://www.novilist.hr/, 20. kolovoza 2023.

[6] „Wikipedija“, s interneta, https://hr.wikipedia.org/wiki/Glavna_stranica, 16. kolovoza 2023.

[7] „Hrvatska školska gramatika“, s interneta, http://gramatika.hr/, 17. kolovoza 2023.

[8] „Hrvatski prefiksi“, s interneta, https://hr.wikipedia.org/wiki/Hrvatski_prefiksi, 27. srpnja 2023.

[9] „Wječnik“, s interneta, https://hr.wiktionary.org/wiki/Glavna_stranica, 15. kolovoza 2023.

[10] „Rječnik hrvatskoga ili srpskoga jezika“, s interneta, http://ihjj.hr/iz-povijesti/rjecnik-hrvatskoga-ili-srpskoga-jezika-tzv-akademijin-rjecnik/40, 1. rujna 2023.

[11] „rjecnik-hrvatskih-jezika“, s interneta, https://github.com/gigaly/rjecnik-hrvatskih-jezika, 16. kolovoza 2023.

[12] „croDict“, s interneta, https://www.crodict.hr/, 17. kolovoza 2023.

[13] Jurafsky D., Martin J. H.: „Speech and Language Processing“, Prentice Hall, Upper Saddle River, NJ, 2000. (https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf)

## SAŽETAK

U ovom radu predstavljen je jezični model temeljen na zbirkama riječi te je napravljen program koji na osnovu tog modela klasificira članke hrvatske Wikipedije. Prikazan je proces obrade HTML koda internet članka kako bi se iz njega mogao izdvojiti samo sadržaj vezan uz taj članak. Opisane su različite metode odabira i povezivanja n-torki. Korištenjem drugog skupa metoda izračunate su vjerojatnosti da je zadani članak određene kategorije i prikazane su u tablicama. Usporedbama tablica zaključeno je koje kombinacije metoda imaju najprecizniju klasifikaciju.



Ključne riječi: jezični model temeljen na zbirkama riječi, klasifikacija dokumenta, učestalost riječi, n-torke



## ABSTRACT

This thesis presents the bag of words model and a created program which uses that model to classify Croatian Wikipedia articles. The process of parsing the HTML code of internet articles is shown, with the goal of extracting only the content related to the article. Several methods of choosing and connecting n-grams are described. Using a different set of methods, probabilities that a given article is of a certain category are calculated and shown in the tables. Using those tables, a conclusion about the best-performing combinations of methods is made.



Keywords: bag of words model, document classification, word frequency, n-grams