TakeLab

Laboratorij za analizu teksta i inženjerstvo znanja
Text Analysis and Knowledge Engineering Lab
Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva
Unska 3, 10000 Zagreb, Hrvatska

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 3792

Izgradnja i odabir značajki za klasifikaciju dokumenata na hrvatskome jeziku

Sandra Trkulja

Zagreb, lipanj 2014.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA ZAVRŠNI RAD MODULA

Zagreb, 13. ožujka 2014.

ZAVRŠNI ZADATAK br. 3792

Pristupnik: **Sandra Trkulja**
Studij: Računarstvo
Modul: Računarska znanost

Zadatak: **Izgradnja i odabir značajki za klasifikaciju dokumenata na hrvatskome jeziku**

Opis zadatka:

Sadržajna klasifikacija teksta jedan je od osnovnih zadatka dubinske analize teksta. Uobičajeno se u tu svrhu koriste modeli strojnog učenja temeljeni na vektorskoj reprezentaciji dokumenta kao vreće riječi. Točnost klasifikacije uvelike ovisi o načinu izgradnje i odabiru značajki, kao i o karakteristikama samih dokumenata te klasifikacijske sheme.

U okviru završnoga rada potrebno je proučiti postupke za klasifikaciju dokumenata temeljene na strojnome učenju te postupke za izgradnju i odabir značajki. Razraditi radni okvir koji će omogućiti ispitivanje niza postupaka za izgradnju i odabir značajki, uključivo postupke za izgradnju značajki temeljenih na n-gramima i distribucijskim značajkama te postupke za odabir značajki temeljene na statističkim mjerama i heurističkoj optimizaciji. Razviti programsku implementaciju radnoga okvira te provesti iscrpno eksperimentalno vrednovanje skupova značajki na ručno označenim zbirkama dokumenata na hrvatskome jeziku. Radu priložiti izvorni i izvršni kod razvijenog sustava, skupove podataka i programsku dokumentaciju te citirati korištenu literaturu.

Zadatak uručen pristupniku: 14. ožujka 2014.
Rok za predaju rada: 13. lipnja 2014.

Mentor:

Doc. dr.sc. Jan Šnajder

Djelovođa:

Doc. dr.sc. Tomislav Hrkać

Predsjednik odbora za završni rad modula:

Prof. dr.sc. Siniša Srbljić

SADRŽAJ

POPIS SLIKA

POPIS TABLICA

1. Uvod

Ovaj završni rad bavi se načinima reprezentacije teksta u svrhu ispravne klasifikacije dokumenata na hrvatskome jeziku. Za klasifikaciju dokumenata obradom prirodnog jezika najčešće se koriste metode nadziranog strojnog učenja koje između ostalog zahtijevaju da se svaki dokument predstavi računalu kao niz značajki. Najpopularniji modeli za izgradnju značajki za prikaz dokumenta koriste skup ili vreću riječi. Ovaj rad razmatra alternativne načine izgradnje značajki za klasifikaciju dokumenata u svrhu dobivanja boljih rezultata.

U svojoj suštini, skup i vreća riječi računalu jednostavno prikazuju sve riječi koje su se pojavile u dokumentu, a ako se radi o vreći riječi, onda se predaje i broj pojavljivanja te riječi unutar dokumenta. Međutim, takva reprezentacija dokumenata ne uzima u obzir važnosti pojedinih riječi niti važnost pojedinih izraza koji ponekad mogu biti razmjerno bitni u postupku klasifikacije dokumenata. Generalno gledajući, riječ koja se pojavljuje u samom naslovu nekog dokumenta najčešće ima mnogo veću važnost od neke druge riječi koja se pojavljuje na kraju dokumenta. Distribucijske značajke korištene u ovom radu nastoje riješiti taj problem. Nadalje, znamo da ponekad koristimo višerječne fraze za opis neke pojave, pa prilikom klasifikacije ima smisla promatrati ih kao jednu cjelinu. To često rješava i problem višeznačnosti riječi koje tvore pojedinu frazu. Za to se u ovom radu koriste n-grami.

Navedene značajke testirane su u odnosu na vreću riječi korištenjem *stroja potpornih vektora*. Dobiveni rezultati upućuju na to da distribucijske značajke ne doprinose kvaliteti klasifikacije, dok korištenje n-grama poboljšava rad klasifikatora.

U poglavlju 2 se opisuju korištene značajke — vreća riječi pojašnjena je u odjeljku 2.1, distribucijske značajke u 2.2, a n-grami u odjeljku 2.3. Nakon toga slijedi opis korištenog klasifikatora koji se nalazi u poglavlju 3. Rezultati su prikazani u poglavlju 4, a na kraju rada se nalazi zaključak (poglavlje 5).

2. Značajke

Kada govorimo o nadziranom strojnom učenju podrazumijeva se da imamo na raspolaganju dovoljno velik korpus klasificiran od strane stručnjaka. U našem slučaju korpus se sastoji od članaka iz novina „Vjesnik“ koji su klasificirani u 13 kategorija od strane autora tih članaka. Autori članaka smatraju se stručnjacima. Svaki članak predstavlja jedan uzorak, a svi uzorci zajedno čine korpus nad kojim se obavlja strojno učenje.

Osim označenog korpusa, potrebno je imati i klasifikator. Klasifikator se može promatrati kao model koji na ulazu ima jedan dokument, a na izlazu javlja odluku o razredu u koji smješta taj uzorak.

Značajke u kontekstu strojnog učenja predstavljaju atribute kojima se opisuju uzorci. Svaka značajka predstavlja jedan atribut uzorka koji može poprimati različite vrijednosti u ovisnosti o prisutnosti atributa u uzorku. Kod klasifikacije teksta jedna značajka najčešće predstavlja jednu riječ, a vrijednost značajke iznosi 0 ili 1, ovisno o tome pojavljuje li se ta riječ u dokumentu ili ne, no može biti i broj pojavljivanja te riječi u dokumentu ako se radi o vreći riječi. Takav skup značajki podrazumijeva da je izgrađen rječnik. Rječnik se sastoji od svih riječi koje se pojavljuju u korpusu s dvije iznimke. U obzir se uzimaju samo riječi koje se pojavljuju barem dva puta unutar korpusa jer iz jednog pojavljivanja riječi klasifikator sigurno ne može ništa naučiti. Riječi koje su jako rijetke općenito ne doprinose poboljšanju rada klasifikatora, pa je granica od 2 pojavljivanja riječi odabrana proizvoljno. Ta granica se najčešće kreće od 2 pojavljivanja riječi u korpusu, pa sve do 50-ak pojavljivanja, ovisno o primjeni. U rječnik ne ulaze niti zaustavne riječi zbog toga što one ne daju značenje za klasifikaciju dokumenata. Primjeri zaustavnih riječi su: $bilo$, $bismo$, $koji$, je. Sve ostale riječi ulaze u rječnik. Važno je napomenuti da u rječnik ulaze riječi lematizirane na svoj morfološki jedinstven oblik (za imenice i pridjeve je to nominativ jednine, a za glagole infinitiv). Primjerice, ako se u korpusu nalaze riječi „$tipkovnice$“ i „$tipkovnicom$“, u rječnik će biti zapisana samo jedna riječ i to nominativ jednine tih riječi što je „$tipkovnica$“. Korpus je u cijelosti lematiziran, tako da su sve riječi morfološki normalizirane prilikom stvaranja rječnika, ali i tijekom korištenja korpusa za učenje i testiranje klasifikatora.

U ovom radu se koriste tri načina izgradnje značajki. Model vreće riječi koji se najčešće koristi u svrhu klasifikacije dokumenata u ovom radu ima ulogu orijentira. Druga dva načina izgradnje značajki — distribucijske značajke i n-grami — implementirani su i uspoređeni s rezultatima dobivenim korištenjem vreće riječi.

2.1. Vreća riječi

Vreća riječi kao značajka može biti izvedena na više načina. Osnovni cilj vreće riječi jest dati informaciju o broju pojavljivanja svake riječi iz rječnika unutar jednog dokumenta. Međutim, neke riječi su općenito češće, a neke rjeđe od drugih. Riječi koje su česte u gotovo svim dokumentima, a nisu filtrirane izbacivanjem zaustavnih riječi ne nose mnogo informacija o kategoriji kojoj pojedini dokument pripada, a ovakvim pristupom će one redovno poprimati veće vrijednosti i činit će se bitnima, dok nas više zanimaju one koje se pojavljuju u manjem broju dokumenata. Zato se najčešće koristi tf-idf verzija vreće riječi. Tf (engl. term frequency) označava broj pojavljivanja riječi t u nekom dokumentu d, dok idf označava količinu informacije koju pojava te riječi donosi, a računa se kao recipročna vrijednost broja dokumenata koji sadrže riječ t podijeljena s brojem svih dokumenata N, te skalirana logaritmiranjem. Funkcija $brojac(t, d)$ broji pojavljivanja riječi t u dokumentu d, a D predstavlja skup svih dokumenata u korpusu. Formalni zapis tf-idf je u nastavku zadan jednadžbom 2.3.

$$tf(t, d) = brojac(t, d) \tag{2.1}$$

$$idf(t, D) = ln \frac{N}{\mid \{d \in D : t \in d\} \mid} \tag{2.2}$$

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D) \tag{2.3}$$

Prikažimo to na primjeru. Neka se svaki dokument sastoji od jedne rečenice, a rečenice su „Gdje ima ljubavi, niti jedno zlo ne šteti“ i „Gdje nema ljubavi, niti jedno dobro ne koristi“. Izračun tf-idf značajki prikazan je na tablicama 2.1, 2.2 i 2.3.

Promotrimo vremensku i prostornu složenost izračuna značajki vreće riječi. Ako je najdulji dokument duljine l, a rječnik ima r riječi za reprezentaciju jednog dokumenta potrebno je u memoriju učitati cijeli dokument veličine l za što nam treba blok memorije duljine l i blok duljine r da se za svaku riječ iz rječnika upiše broj pojavljivanja te riječi u dokumentu. Za to je potreban jedan prolaz kroz dokument. Dakle ukupna vremenska složenost jest $O(l)$, a prostorna složenost $O(l + r)$. Obje složenosti su linearne, pa je većina korisnika zadovoljna brzinom izlučivanja značajke. Drugačijim izborom strukture podataka za pamćenje broja pojavljivanja riječi možemo smanjiti

Tablica 2.1: Tablica količine informacije idf

t	$idf(t, D)$
gdje	0.00
ima	0.69
ljubavi	0.00
niti	0.00
jedno	0.00
zlo	0.69
ne	0.00
šteti	0.69
nema	0.69
dobro	0.69
koristi	0.69

Tablica 2.2: tf prvog dokumenta

t	$tf(t, d)$	$tfidf(t, d, D)$
gdje	1	0.69
ima	1	0.00
ljubavi	1	0.69
niti	1	0.69
jedno	1	0.69
zlo	1	0.00
ne	1	0.69
šteti	1	0.00

Tablica 2.3: tf drugog dokumenta

t	$tf(t, d)$	$tfidf(t, d, D)$
gdje	1	0.69
nema	1	0.00
ljubavi	1	0.69
niti	1	0.69
jedno	1	0.69
dobro	1	0.00
ne	1	0.69
koristi	1	0.00

pribrojnik r u prostornoj složenosti tako da koristimo primjerice asocijativni spremnik koji će pamtiti samo riječi koje se pojavljuju unutar dokumenta. Međutim, takav izbor može uvećati vremensku složenost izračuna značajke zbog nekonstantne složenosti pristupa elementu asocijativnog spremnika koju diktira implementacijski programski jezik. Pristup elementu može biti logaritamske složenosti $O(\log m)$ ako je spremnik interno implementiran kao stablo ili konstantne složenosti $O(1)$ ako se ključevi spremnika pohranjuju koristeći *raspršeno adresiranje*. Za implementaciju ovog rada korišten je *Python* programski jezik koji koristi raspršeno adresiranje za pohranu ključeva, pa je mogući problem rasta vremenske složenosti izbjegnut, no niti korištenje spremnika s logaritamskom složenošću nije velik problem uzevši u obzir da je rječnik najčešće mnogo veći od samog dokumenta, pa se elementima pristupa relativno malen broj puta.

2.2. Distribucijske značajke

Model vreće riječi je generalno dobar, no postoje situacije kada on može zakazati. Primjerice, može se dogoditi da se neka riječ spominje ukupno četiri puta unutar dokumenta, od toga jedanput u naslovu i još tri puta kroz cijeli dokument, dok se neka druga riječ spominje pet puta, ali samo u zadnjem dijelu dokumenta i odnosi se na povezivanje tog članka s nekom drugom temom. Intuitivno vidimo da prva riječ nosi više težine u tom dokumentu nego druga riječ, no vreća riječi dat će veću težinu drugoj riječi samo zato što se u tekstu pojavljuje više puta. Distribucijske značajke kakve ih opisuju Xue i Zhou (2009) rješavaju taj problem. Na temelju njih su izrađene i distribucijske značajke u ovom radu. One se temelje na trima jednostavnim pretpostavkama za koje intuitivno znamo da vrijede:

1. Što se češće riječ pojavljuje, to je riječ bitnija,
2. Što je riječ više raspršena po dokumentu, to je riječ bitnija,
3. Što je riječ ranije prvi put spomenuta, to je riječ bitnija.

Iz navedenih pretpostavki vidimo da bi značajke koje su izgrađene na temelju ovih pretpostavki u prethodnom primjeru dale puno veću važnost prvoj riječi, nego drugoj riječi. Bitno je napomenuti da su ove pretpostavke najizraženije kod slobodnog stila pisanja, kao što su novinski članci od kojih se sastoji i naš korpus. Činjenica koja pogoduje distribucijskim značajkama jest da se one mogu lako izračunati, pa ne troše mnogo dodatnih memorijskih i računalnih resursa, a klasifikatoru donose dodatne informacije o samom dokumentu.

2.2.1. Podjela unutar dokumenata

U gore navedenim pretpostavkama može se vidjeti da se koriste fraze poput „više raspršena riječ“ i „ranije spomenuta riječ“. One impliciraju da je dokument nekako podijeljen na dijelove tako da se može reći da je u jednom dokumentu riječ više raspršena nego u drugom dokumentu. Trivijalna podjela dokumenta je podjela na rečenice, no lako se može uvidjeti da to neće uvijek dobro funkcionirati jer dokumenti generalno mogu biti različitih duljina i broja rečenica, pa redni broj rečenice u dokumentu ne znači mnogo. Potrebno je pronaći neki drugi način raspodjele dokumenata.

Callan (1994) predlaže tri načina raspodjele pojedinog dokumenta, a Kim i Kim (2004) razmatraju dobre i loše strane svake od njih.

Sve se značajke mogu dobiti iz jednostavnog broja pojavljivanja riječi u pojedinim dijelovima dokumenta, tako da nikada ne trebamo imati sve u memoriji, već samo jedan dokument. Govorna podjela dijeli dokument na dijelove kao što su odlomci i rečenice. Negativne strane podjela su već navedene, dok se u ovom slučaju podjela na odlomke čini boljom opcijom zbog toga što se korpus sastoji od novinskih članaka koji često imaju barem približno sličan broj odlomaka. Drugi način podjele dokumenta jest semantička podjela koja dijeli dokument po temama. Iako je takva podjela točnija od prethodne, nju je mnogo teže odrediti pomoću računala. Nadalje, ona uvelike ovisi o algoritmu razdvajanja koji se koristi, pa stoga nije univerzalna metoda. Osim toga iziskuje i dodatne računalne resurse koje u ovom radu ne želimo dodavati, jer bi se tako više udaljili od brzine računanja vreće riječi. Treća metoda podjele dokumenta jest podjela na prozore riječi. Prozor riječi jest niz riječi koje se u dokumentu pojavljuju za redom. Prednost ove metode je što je svaki dio jednake duljine, no teško je odabrati univerzalnu veličinu prozora za cijeli korpus.

Semantička podjela se ne koristi u ovom radu zbog dugotrajnog postupka računanja granica cjelina, već se koriste govorna podjela i podjela na prozore riječi. U govornoj podjeli jednu cjelinu predstavlja jedan odlomak. Korpus je unaprijed razdijeljen na odlomke, pa je postupak pronalaženja granica cjelina trivijalan. Prozor riječi kod podjele dokumenta na prozore riječi može biti izveden na dva načina: kao preklapajući i nepreklapajući prozor. U ovom se radu koristi nepreklapajući prozor, što znači da svaka riječ dokumenta pripada samo jednom prozoru. Isprobane su različite veličine prozora.

U idućem podjeljku su navedene četiri funkcije koje zajedno tvore distribucijske značajke. Svaka od tih funkcija se evaluira za svaku riječ iz rječnika, pa je kraju postupka vektor značajki veličine $4 \times |R|$, gdje je R rječnik, tj. skup svih riječi u korpusu. Taj vektor točno je četiri puta veći od veličine vektora značajki koji se dobije korištenjem vreće riječi zbog toga što taj model evaluira samo funkciju $tfidf(t, d, D)$ za svaku riječ iz rječnika. Za izračunavanje svih funkcija distribucijskih značajki potrebna je samo jedna stvar — broj pojavljivanja svih riječi u svakoj cjelini dokumenta. Možemo pretpostaviti da za svaku riječ u rječniku imamo listu brojeva u kojoj vrijednost na nekom indeksu predstavlja broj pojavljivanja te riječi u cjelini dokumenta s istim indeksom. Dakle, veličina svake liste jednaka je broju cjelina na koje je dokument odijeljen. Izgled jedne takve liste za riječ t koja se u cjelini s indeksom i dokumenta d s n cjelina pojavljuje c_i puta jest $polje(t, d) = [c_0, c_1, ..., c_{n-1}]$.

2.2.2. Korištene značajke

Prvo pojavljivanje riječi

Kao što smo već pretpostavili, distribucijske značajke uzimaju u obzir prvo pojavljivanje riječi u dokumentu jer je riječ to važnija što se ranije pojavljuje. Izračun te značajke dobiva se evaluacijom funkcije $PrvaPojava(t, d)$ koja je zadana jednadžbom 2.4.

$$PrvaPojava(t, d) = \min_{i \in \{0..n-1\}} \begin{cases} i & \text{ako je } c_i > 0 \\ n & \text{ako je } c_i \leq 0 \end{cases} \tag{2.4}$$

Raširenost riječi u dijelovima dokumenta

Raspršenost riječi unutar dokumenta jedna je od mjera navedenih u pretpostavkama. Ona označava koliko se kompaktno riječ pojavljuje unutar dokumenta. Što su pojavljivanja riječi kompaktnije smještena, to je raspršenost riječi manja. Jednu mjeru raspršenosti riječi izražavamo kao broj cjelina dokumenta u kojima se pojavljuje riječ t. Funkcija $BrojCjelina(t, d)$ zadana je jednadžbom 2.5. U ovom radu se osim te koriste još dvije funkcije za računanje raspršenosti riječi u dokumentu, a navedene su u idućim pododjeljcima.

$$BrojCjelina(t, d) = \sum_{i=0}^{n-1} \begin{cases} 1 & \text{ako je } c_i > 0 \\ 0 & \text{ako je } c_i \leq 0 \end{cases} \tag{2.5}$$

Udaljenost prvog i zadnjeg pojavljivanja riječi

Udaljenost prvog i zadnjeg pojavljivanja riječi kao druga mjera raspršenosti riječi u dokumentu mjeri se kao razlika indeksa zadnje i prve cjeline u kojima se riječ pojavljuje. Što se riječ kompaktnije pojavljuje, to je udaljenost manja. Sama funkcija te mjere zadana je jednadžbom 2.7.

$$ZadnjaPojava(t, d) = \max_{i \in \{0..n-1\}} \begin{cases} i & \text{ako je } c_i > 0 \\ -1 & \text{ako je } c_i \leq 0 \end{cases} \tag{2.6}$$

$$UdaljenostPZ(t, d) = ZadnjaPojava(t, d) - PrvaPojava(t, d) \tag{2.7}$$

Varijanca pozicija pojavljivanja riječi

Ova značajka predstavlja treću i posljednju metodu izračuna raspršenosti riječi u dokumentu. Ona koristi statističku raspršenost riječi, te se računa kao varijanca svih

pozicija na kojima se riječ pojavljuje. To znači da se prvo izračuna srednja pozicija svih pojavljivanja riječi u dokumentu, a onda se izračuna srednja vrijednost odstupanja svih pojavljivanja riječi od srednje vrijednosti koja predstavlja varijancu pozicija pojavljivanja riječi i dana je jednadžbom 2.10.

$$brojac(t,d) = \sum_{i=0}^{n-1} c_i \tag{2.8}$$

$$centroid(t,d) = \frac{\sum_{i=0}^{n-1} c_i \times i}{brojac(t,d)} \tag{2.9}$$

$$VarijancaPozicija(t,d) = \frac{\sum_{i=0}^{n-1} c_i \times |i - centroid(t,d)|}{brojac(t,d)} \tag{2.10}$$

2.2.3. Primjer izračuna distribucijskih značajki

Možemo za primjer uzeti riječ $buba$ čija je lista pojavljivanja u cjelinama nekog dokumenta jednaka $polje(buba, d) = [0, 8, 3, 5, 0, 5, 7]$. Izračun distribucijskih značajki dan je jednadžbama 2.11 - 2.17.

$$PrvaPojava(buba,d) = min\{7,1,2,3,7,5,6\} = 1 \tag{2.11}$$

$$BrojCjelina(buba,d) = 0+1+1+1+0+1+1 = 5 \tag{2.12}$$

$$ZadnjaPojava(buba,d) = max\{-1,1,2,3,-1,5,6\} = 6 \tag{2.13}$$

$$UdaljenostPZ(buba,d) = 6-1 = 5 \tag{2.14}$$

$$brojac(buba,d) = 0+8+3+5+0+5+7 = 28 \tag{2.15}$$

$$centroid(buba,d) = (8\times 1+3\times 2+5\times 3+5\times 5+7\times 6)/28 = 3.429 \tag{2.16}$$

$$\begin{aligned} VarijancaPozicija(buba,d) = {} & (8\times 2.429+3\times 1.429+5\times 0.429+ \\ & +5\times 1.571+7\times 2.571)/28 = 1.847 \end{aligned} \tag{2.17}$$

2.2.4. Analiza vremenske i prostorne složenosti izračuna distribucijskih značajki

Prisjetimo se složenosti izračuna značajki vreće riječi. One su jednake $O(l)$ za vremensku složenost i $O(l+r)$ za prostornu složenost, gdje l predstavlja najveću veličinu dokumenta, a r broj riječi u rječniku.

Promotrimo sada distribucijske značajke. Sve funkcije distribucijskih značajki mogu se dobiti samo ako su poznate frekvencije pojavljivanja riječi u svim cjelinama

dokumenta. Pretpostavimo da dokument ima najviše n cjelina, te da se korpus sastoji od s dokumenata. Tada se u memoriji za svaku cjelinu n pamti broj pojavljivanja svih riječi za koji nam je potrebno $r \times 1$ memorije. Iz toga proizlazi da je ukupna prostorna složenost $O(n \times r)$. Vremenska složenost tog izračuna je jednaka $O(l \times s)$. Ono što još preostaje napraviti jest evaluirati jednadžbe 2.4 - 2.10 za svaku riječ iz rječnika u svim dokumentima za koje nam je potrebno još $O(r \times s \times n)$ vremena. Za to je potrebno još $O(1)$ dodatnog prostora.

Iz navedenog slijedi da je ukupna vremenska složenost distribucijskih značajki $O(l \times s + r \times s \times n)$, a prostorna složenost $O(n \times r)$. Povećanje je u obje složenosti relativno veliko jer niti jedna složenost više nije linearna, ali treba uzeti u obzir da su te složenosti istinite samo u najgorem slučaju. Navedeni račun je potrebno provesti samo ako se ta riječ pojavljuje barem jedanput u dokumentu što je slučaj tek za mali broj riječi (broj riječi $<< r$), pa je ukupni dodatni trošak računalnih i memorijskih resursa u usporedbi s vrećom riječi malen.

2.3. N-grami

U kontekstu obrade prirodnog jezika pojam *n-gram* ima dva značenja. Prvo značenje definira n-gram kao skup od n riječi koje se pojavljuju za redom u nekom tekstu, a u drugom značenju n-gram predstavlja skup od n slova koja se pojavljuju za redom. U ovom radu koristi se prvo značenje te riječi. Dodatno, *unigram* je sinonim za 1-gram, *bigram* za 2-gram i *trigram* za 3-gram. Pokažimo to na primjeru. Rečenica „*Ova rečenica služi kao primjer*“ sadrži *pet* unigrama — *ova*, *rečenica*, *služi*, *kao* i *primjer* — te *četiri* bigrama — *ova rečenica*, *rečenica služi*, *služi kao* i *kao primjer*.

2.3.1. Argumenti za korištenje n-grama i protiv njega

N-grami su korisni za predstavljanje skupova riječi koje se često pojavljuju zajedno. Primjerice riječi *operacijski sustav* zajedno tvore jednu frazu koja se često koristi, pa ima smisla gledati te riječi kao jednu cjelinu. To je upravo ono što n-grami omogućuju. Osim korištenja fraza kao cjeline, n-grami u velikoj mjeri rješavaju problem višeznačnosti i stupnja doslovnosti riječi unutar fraze. Uzmimo za primjer frazu *plava kosa*. Riječ *plava* u svom doslovnom značenju služi kao atribut kojim se iskazuje da je neka stvar plave boje, no znamo da se kod fraze *plava kosa* zapravo radi o žutoj boji, a ne plavoj. Dakle, riječ *plava* je ovdje u prenesenom značenju i nema veze sa svojim doslovnim značenjem. Nadalje, riječ *kosa* se može odnositi na skup vlasi na

glavi, a može označavati i poljoprivredni alat za košnju trave. Jednom kada znamo da se ta riječ nalazi u frazi *plava kosa* možemo bez dileme zaključiti da se radi o prvom značenju te riječi.

Unatoč velikom broju korisnih strana korištenja n-grama, uz njih dolaze i neke neželjene posljedice. Demonstrirajmo ih na primjeru navedenom u članku Caropreso et al. (2001). Neke od mogućih pojava bigrama *informacija pretražiti* su:

1. pretraživanje informacija
2. pretražuje informacije
3. informativno pretraživanje
4. Pretraži informacije!
5. Informiraj pretraživača!

Promotrimo prvo izraze $1. - 4.$. Brzim pregledom po njima možemo ustanoviti da svi imaju isto značenje. Međutim, detaljnijim pogledom može se uvidjeti da su izrazi $1. - 3.$ fraze, a izraz 4. je cijela rečenica. Isto tako, uviđamo da riječi u različitim izrazima mijenjaju vrstu riječi. Primjerice, *pretraživanje* se u prvom izrazu pojavljuje kao glagolska imenica, a u drugom izrazu kao glagol, dok se riječ *informacija* u prvom izrazu pojavljuje kao imenica, a u trećem izrazu kao pridjev. Razlog za definiranje n-grama na način da sve te pojave sakupi u jednu cjelinu zasnovan je na hipotezi da se uporabom različitog poretka riječi te različitih vrsta riječi može predstaviti isti koncept. Međutim, takva generalizacija uzrokuje dva nova problema:

- *pretjerana generalizacija*: vidljiva je iz činjenice da izrazi $1. - 4.$ i izraz 5. koji su objedinjeni istim bigramom ne predstavljaju isti koncept
- *nedovoljna generalizacija*: izraz *pretraživanje zanimljivih informacija* uklapa se među izraze $1. - 4.$, no taj izraz neće biti predstavljen istim bigramom kao i izrazi $1. - 4.$

2.3.2. Funkcije za evaluaciju korisnosti n-grama

U prethodnom odjeljku navedeni su neki od razloga zašto se koriste n-grami. Međutim, postavlja se pitanje kako izlučiti samo one n-grame koji su nam bitni. Primjerice *visok moda* je mnogo zanimljiviji bigram od *visok dječak* jer označava frazu koja predstavlja elitu modne industrije, dok je riječ *visok* u bigramu *visok dječak* samo atribut koji opisuje dječaka.

Postavlja se pitanje kako odvojiti bitne od nebitnih n-grama. Caropreso, Matwin i Sebastiani u članku Caropreso et al. (2001) predlažu četiri funkcije čija vrijednost odgovara važnosti tog n-grama. Te funkcije su zadane formulama 2.18–2.21 .Prilikom interpretacije formula vjerojatnost $P(t_k, c_i)$ se tumači kao vjerojatnost da nasumično odabran dokument d pripada kategoriji c_i te da se n-gram t_k pojavljuje u tom dokumentu. Analogno, vjerojatnost $P(\bar{t_k}|c_i)$ predstavlja vjerojatnost da se u nasumično odabranom dokumentu d koji pripada kategoriji c_i ne nalazi n-gram t_k.

Bitni su n-grami tako filtrirani u četiri skupine — svaka skupina predstavlja najbolje n-grame dobivene evaluacijom jedne od četiri dane funkcije. Postupak filtriranja n-grama sastoji se od prikupljanja svih k-grama gdje je $k \in \{1, ..., n\}$, te evaluacijom funkcija 2.18–2.21 za svaki k-gram. Nakon što je funkcija evaluirana za svaki k-gram, oni se rangiraju počevši od najbolje do najslabije ocijenjenog k-grama. Ono što želimo napraviti jest uzeti samo podskup najboljih n-grama tako da izbjegnemo problem iz prethodnog odjeljka. To se može ostvariti tako da se izlista najboljih p posto svih k-grama, te se u rječnik dodaju svi k-grami gdje je $k \in \{2, ..., n\}$. Znajući da je svaka od četiri funkcije za evaluaciju značajki neovisna od svih ostalih funkcija te uzevši u obzir činjenicu da se rječnici za različite postotke p mogu razlikovati zaključujemo da je potrebno izgraditi $4 \times |p|$ različitih rječnika, gdje je $|p|$ kardinalnost skupa izabranih postotaka. To vrijedi ako se promatraju isključivo bigrami. Ako se pak žele promatrati i trigrami, broj rječnika se povećava na $4 \times |p| \times 2$ jer osim svih rječnika za korištenje bigrama moramo imati isto toliko rječnika za korištenje trigrama. U ovom radu se eksperimentira s vrijednostima parametra p jednakima 10%, 20% i 30%, a promatrani su bigrami i trigrami, pa je ukupni broj korištenih rječnika jednak 24 ($4 \times 3 \times 2$).

Frekvencija pojavljivanja u dokumentu

Frekvencija pojavljivanja n-grama u dokumentu predstavlja vjerojatnost da se n-gram t_k nađe u nekom dokumentu iz kategorije c_i i računa se prema formuli 2.18.

$$FP(t_k, c_i) = P(t_k|c_i) \tag{2.18}$$

Informacijska dobit

Informacijska dobit daje numeričku vrijednost količini informacije koju donosi prisutnost n-grama u određenoj kategoriji i dana je formulom 2.19.

$$ID(t_k, c_i) = P(t_k, c_i) \cdot \log \frac{P(t_k, c_i)}{P(t_k) \cdot P(c_i)} + P(\bar{t}_k, c_i) \cdot \log \frac{P(\bar{t}_k, c_i)}{P(\bar{t}_k) \cdot P(c_i)} \tag{2.19}$$

Hi-kvadrat

Hi-kvadrat mjera za evaluaciju važnosti n-grama osim standardnih parametara koristi još i dodatak g koji je jednak kardinalnosti skupa za učenje. Hi-kvadrat mjera dana je formulom 2.20.

$$\chi^2(t_k, c_i) = \frac{g \cdot [P(t_k, c_i) \cdot P(\bar{t}_k, \bar{c}_i) - P(t_k, \bar{c}_i) \cdot P(\bar{t}_k, c_i)]^2}{P(t_k) \cdot P(\bar{t}_k) \cdot P(c_i) \cdot P(\bar{c}_i)} \tag{2.20}$$

Vjerojatnosni omjer

Posljednja predložena funkcija za evaluaciju značaja n-grama naziva se vjerojatnosni omjer i definirana je formulom 2.21.

$$VO(t_k, c_i) = \frac{P(t_k|c_i) \cdot (1 - P(t_k|\bar{c}_i))}{(1 - P(t_k|c_i)) \cdot P(t_k|\bar{c}_i)} \tag{2.21}$$

3. Klasifikator

Kao što je već spomenuto u poglavlju 2, klasifikator se općenito može promatrati kao model koji na svoj ulaz dobiva vrijednosti vektora značajki za pojedini uzorak, a na svom izlazu daje informaciju o razredu u koji smješta taj uzorak. U našem slučaju su uzorci dokumenti, a razredi su kategorije kojima dokument može pripadati. Prisjetimo se, svaki dokument može pripadati samo jednoj od 13 kategorija.

Da bi klasifikator mogao donositi takve odluke na temelju vektora značajki predanih uzoraka potrebno je prvo istrenirati klasifikator setom za treniranje. Skup za učenje sadrži vektore značajki svih uzoraka i njihove pripadne razrede. Na njima klasifikator *uči* kako prepoznati pojedine razlike na temelju danih značajki. Tek nakon što je klasifikator naučen moguće je procijeniti točnost klasifikatora tako da mu se preda skup uzoraka za testiranje koji se sastoji samo od vektora značajki pojedinih uzoraka. Jednom kada klasifikator da svoju odluku o razredima svih uzoraka iz skupa za testiranje one se uspoređuju sa stvarnim razredima kojima pojedini uzorci pripadaju, pa se tako može evaluirati točnost klasifikatora. U ovom radu korišten je Stroj Potpornih Vektora (*engl. Support Vector Machine*).

3.1. Klasifikator *SVM*

Stroj potpornih vektora (SVM) jest model nadziranog strojnog učenja koji se najčešće koristi kod raspoznavanja uzoraka i klasifikacije. U ovom radu koristi se *linearan* klasifikator SVM. Linearanost klasifikatora označava da su funkcije granice između razreda izgrađene kao linearna kombinacija vektora značajki. Drugim riječima, decizijska funkcija koja određuje granicu između razreda jest hiperravnina dimenzionalnosti jednake dimenziji vektora značajki. Specijalno, u slučaju dvodimenzionalnog prostora značajki granica između razreda će nužno biti pravac, a ne primjerice kružnica ili parabola.

3.1.1. Klasifikator *liblinear*

U ovom radu je korištena *liblinear* implementacija klasifikatora SVM. Klasifikator liblinear je u današnjoj formi opisan u radu Fan et al. (2008), dok je izvorni algoritam predložen u Cortes i Vapnik (1995). On u podrazumijevanom načinu rada za treniranje koristi samo jedan parametar. On određuje iznos „kazne“koja djeluje na decizijsku funkciju u slučajevima kada klasifikator u postupku računanja optimalne decizijske funkcije krivo klasificira uzorak. Ta je vrijednost nužno pozitivna te može iznositi 2^{-14} – 2^{14}, a najčešće se dobiva empirijskim istraživanjem na podacima.

U slučaju da su uzorci klasificirani u više od dva razreda liblinear prati *jedan-nasuprot-svih* model u kojem se granice određuju jedanput za svaki razred. Svaka granica odjeljuje jedan razred od svih ostalih razreda. Prilikom treniranja klasifikatora na našem korpusu korišten je takav model određivanja linearnih decizijskih funkcija jer je korpus klasificiran u ukupno 13 razreda.

Prilikom rada u podrazumijevanom načinu rada liblinear računa linearne decizijske funkcije bez slobodnog člana b. To znači da je umnožak vektora značajki s vektorom težina decizijske funkcije čisto vektorski produkt. Drugim riječima, sve decizijske funkcije prolaze kroz ishodište koordinatnog sustava. Korištenje člana b u liblinear biblioteci omogućeno je korištenjem opcije *-b*. U ovom radu je korišten član b vrijednosti 1. Uočimo da je dodavanje slobodnog člana vektorskom umnošku ekvivalentno dodavanju homogene komponente u vektor značajki. Dakle, isti efekt bi se postigao kada bi se na kraj svakog vekora značajki dodala vrijednost 1.

4. Rezultati

Rezultati koji su dobiveni treniranjem i testiranjem klasifikatora prikazani su u odjeljcima 4.2.1–4.2.3, a pojedinosti oko mjera kvalitete klasifikatora i korištenog korpusa opisane su u odjeljku 4.1.

4.1. Mjere točnosti klasifikatora i opis korpusa

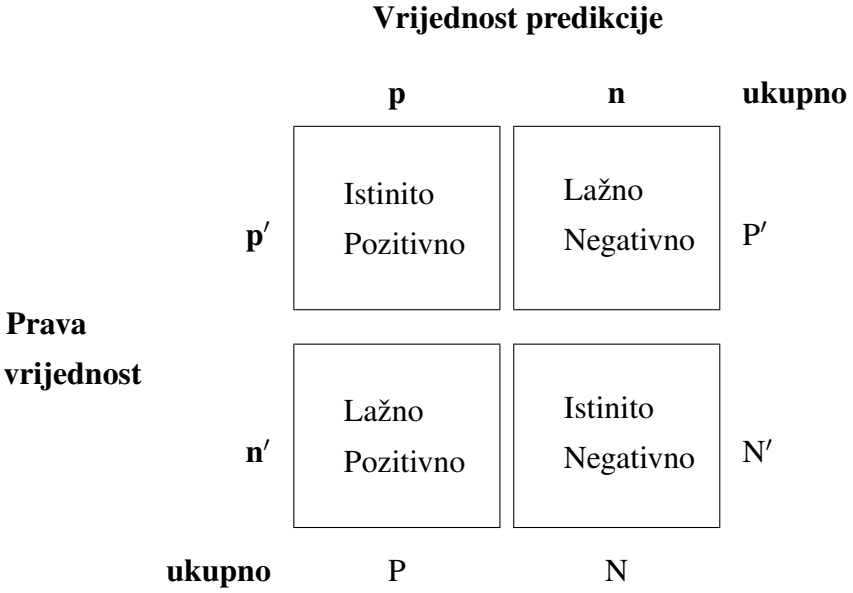
4.1.1. Mjere točnosti klasifikatora

Rezultati su prikazani mjerama *preciznosti*, *odziva* i F_1. One su izabrane zbog toga što donose mnogo više informacija o kakvoći klasifikatora od mjera na koje smo navikli u svakodnevnom životu kao što je postotak točno klasificiranih uzoraka. Uzmimo za primjer klasifikaciju stabala neke šume u dvije kategorije: *bjelogorica* i *crnogorica*. Neka je udio bjelogorice u šumi jednak 99%. Možemo izgraditi klasifikator koji će imati 99%-tnu uspješnost tako da svako stablo svrsta u kategoriju bjelogorica, no očito je da problem nije rješen. Iz takvih razloga se uvode navedene mjere.

Mjere korištene u ovom radu mogu se izračunati korištenjem funkcija iz matrice konfuzije prikazane na slici 4.1. Istinit pozitivni dio matrice se često označuje kraticom TP koja je akronim engleskog naziva *true pozitive*. Analogno, kratica FN *(engl. false negative)* predstavlja lažno negativan, FP je lažno pozitivan, a TN označuje istinito negativan dio matrice.

Za objašnjenje matrice konfuzije možemo se poslužiti već navedenim primjerom klasifikacije stabala. Recimo da želimo odvojiti sva stabla crnogorice. Svaki uzorak kojemu je klasifikator predvidio kategoriju može se svrstati u jedno od četiri polja u matrici konfuzije. Istinita polja pripadaju uzorcima koji su pravilno klasificirani, a lažna pogrešnoj klasifikaciji. Iz razloga što je potrebno odvojiti crnogoricu, pozitivna polja označuju da je klasifikator svrstao stablo u crnogoricu, a negativna u bjelogoricu. Sva stabla crnogorice koje je klasifikator prepoznao kao crnogoricu pripadaju skupini istinito pozitivno. Stabla crnogorice koje je klasifikator greškom klasificirao u bjelogo-

ricu dio su lažno negativne kategorije. Sva stabla bjelogorice koje klasifikator greškom prepoznao kao crnogoricu pripadaju lažno pozitivnoj skupini, a sva stabla bjelogorice koje je klasifikator ispravno uvrstio u bjelogoricu određuju istinito negativno polje.



Slika 4.1: Matrica konfuzije

Sve što još preostaje jest definirati mjere koje se u ovom radu koriste. *Preciznost* u prethodnom primjeru predstavlja omjer ispravno klasificirane crnogorice i svih stabala klasificiranih u crnogoricu te je dana formulom 4.1. *Odziv* — označen s R — je definiran kao udio crnogorice koja je točno klasificirana i dan je formulom 4.2. F_1 mjera jest harmonijska sredina preciznosti i odziva i prikazana je formulom 4.3

$$P = \frac{TP}{TP + FP} \tag{4.1}$$

$$R = \frac{TP}{TP + FN} \tag{4.2}$$

$$R = 2 \cdot \frac{P \cdot R}{P + R} \tag{4.3}$$

4.1.2. Opis korpusa

Korpus na kojim su obavljeni eksperimenti sastoji se od 258869 članaka iz novina „Vjesnik„koji datiraju od 31. svibnja 1999. do 1. studenog 2009. godine. Korpus je raspodjeljen u 13 razreda: *crn*, *gle*, *gos*, *kom*, *kul*, *pis*, *spo*, *sss*, *sta*, *tem*, *unu*, *van* i *zag*. Nazivi kategorija su kratice koje odgovaraju temama članaka. Broj članaka prisutnih u svakoj kategoriji te udio svake kategorije u članku prikazan u tablici 4.1.

Skup za učenje čini kronološki prvih 70% korpusa, a preostalih 30% pripada skupu za treniranje. Kako je korpus vremenski raspodijeljen, sve kategorije se pojavljuju u sličnim udjelima i u skupu za učenje i u skupu za treniranje.

Tablica 4.1: Udio kategorija unutar korpusa

Kategorija	Broj članaka	Udio u korpusu (%)
crn	21956	8.5
gle	5285	2.0
gos	20829	8.0
kom	8291	3.2
kul	21961	8.5
pis	11904	4.6
spo	34872	13.5
sss	20943	8.1
sta	5268	2.0
tem	14966	5.8
unu	42459	16.4
van	28450	11.0
zag	21685	8.4

4.2. Eksperimentalni rezultati

Svi eksperimentalni rezultati prikazani su tablicama 4.2–4.5. Prvi stupac tablice sadrži model (u slučaju više varijanti modela naglašeni su i parametri pojedine varijante). U drugom stupcu nalaze se vrijednosti preciznosti, u trećem odziva, a u četvrtom stupcu F_1 mjera.

4.2.1. Vreća riječi

U tablici 4.2 prikazani su rezultati dobiveni korištenjem modela vreće riječi. Ti rezultati su često dovoljno dobri za vektorsku reprezentaciju dokumenta te predstavljaju svojevrsni standard. Alternativni načini vektorske reprezentacije teksta imaju cilj dati bolje rezultate od modela vreće riječi.

Tablica 4.2: Rezultati korištenja modela vreće riječi

Model	Preciznost	Odziv	F_1
Vreća riječi	0.590	0.551	0.570

4.2.2. Distribucijske značajke

Rezultati distribucijskih značajki prikazani su u tablici 4.3. Iz tablice se može vidjeti da razdvajanje dokumenata na odlomke u odnosu na razdvajanje na nepreklapajuće prozore daje bolje rezultate za klasifikaciju dokumenata. Sve tri korištene mjere za izražavanje točnosti klasifikatora su u skladu s takvim zaključkom.To znači da govorna podjela na cjeline kao što su u ovom slučaju odlomci ima veći značaj od prozora stalne veličine. Prilikom razmatranja samih prozora, vidljivo je da točnost klasifikatora raste s povećanjem prozora što upućuje na to da veće cjeline imaju veću ulogu u razmatranju važnosti pozicija riječi u dokumentu.
Ako usporedimo model vreće riječi s modelom distribucijskih značajki možemo uvidjeti da distribucijske značajke same ne donose mnogo novih informacija, te se utrošak dodatnih računalnih resursa ne isplati.

Tablica 4.3: Rezultati korištenja modela distribucijskih značajki

Model	Preciznost	Odziv	F_1
Odlomak	0.590	0.551	0.570
Prozor_{50}	0.458	0.450	0.454
Prozor_{100}	0.455	0.452	0.453
Prozor_{200}	0.454	0.452	0.453

4.2.3. Vreća riječi i n-grami

U tablici 4.4 nalaze se eksperimentalni rezultati rada klasifikatora korištenjem bigrama i vreće riječi, a u tablici 4.5 su podaci o klasifikaciji korištenjem bigrama, trigrama i vreće riječi. N-grami sami po sebi ne nose dovoljno informacija da bi predstavljali cijeli dokument jer je prilikom evaluacije n-grama odbačeno 70, 80 ili 90 posto svih n-grama u svrhu izdvajanja značajnijih od manje značajnih n-grama. Zato su u ovom radu n-grami korišteni zajedno s vrećom riječi. Na kraj svakog vektora značajki dobivenog uporabom modela vreće riječi dodana je frekvencija pojavljivanja tog bigrama u dokumentu te je time dobiven vektor značajki vreće riječi i n-grama koji je predstavljen tablicama 4.4−4.5.

Tablica 4.4: Rezultati korištenja modela filtriranja *bigrama*

Model	Preciznost	Odziv	F_1
$\text{FP}_{10\%}$	0.628	0.569	0.597
$\text{FP}_{20\%}$	0.639	0.571	0.603
$\text{FP}_{30\%}$	0.643	0.572	0.605
$\text{ID}_{10\%}$	0.631	0.568	0.598
$\text{ID}_{20\%}$	0.636	0.565	0.599
$\text{ID}_{30\%}$	0.640	0.564	0.600
${\chi^2}_{10\%}$	0.631	0.566	0.597
${\chi^2}_{20\%}$	0.636	0.563	0.597
${\chi^2}_{30\%}$	0.643	0.563	0.600
$\text{VO}_{10\%}$	0.631	0.566	0.597
$\text{VO}_{20\%}$	0.642	0.568	0.603
$\text{VO}_{30\%}$	0.648	0.567	0.605

Iz tablice 4.4 je baš svi modeli korištenja n-grama s vrećom riječi doprinose poboljšanju rada klasifikatora u usporedbi s korištenjem samo vreće riječi. Generalni trend kod postotka filtriranih n-grama jest poboljšanje klasifikacije s odabirom većeg pos-

totka n-grama. Preciznost klasifikatora raste s povećanjem parametra p u njegovom testiranom rasponu od 10% do 30%. Odziv klasifikatora na povećanje broja prihvaćenih n-grama blago pada što upućuje na početak zasićenosti klasifikatora manje bitnim n-gramima, no F_1 mjera koja pokriva i preciznost i odziv i dalje raste. Čak i s velikim postotkom od 30% prihvaćenih n-grama kvaliteta klasifikatora F_1 mjera pokazuje poboljšanje rezultata. Takvi rezultati ukazuju na to da su bigrami u hrvatskome jeziku korisni za korištenje prilikom reprezentacije dokumenata za klasifikaciju pomoću strojnog učenja.

Usporedimo li četiri korištene funkcije za rangiranje n-grama vidljivo je da funkcija vjerojatnosnog omjera najbolje ocjenjuje važnost pojedinih n-grama. Rezultati korištenja vjerojatnosnog omjera uz prihvaćanje 30% najboljih n-grama uz vreću riječi povećava preciznost klasifikatora za 9.8%, dok je F_1 mjera porasla za 6.1%. Najveće povećanje odziva ima model vjerojatnosnog omjera uz parametar p jednak 30%, a povećanje iznosi 3.8%.

Tablica 4.5: Rezultati korištenja modela filtriranja *bigrama* i *trigrama*

Model	Preciznost	Odziv	F_1
$\text{FP}_{10\%}$	0.604	0.555	0.579
$\text{FP}_{20\%}$	0.617	0.554	0.584
$\text{FP}_{30\%}$	0.626	0.552	0.586
$\text{ID}_{10\%}$	0.603	0.555	0.578
$\text{ID}_{20\%}$	0.613	0.554	0.582
$\text{ID}_{30\%}$	0.624	0.551	0.585
${\chi^2}_{10\%}$	0.607	0.551	0.578
${\chi^2}_{20\%}$	0.616	0.549	0.580
${\chi^2}_{30\%}$	0.625	0.545	0.582
$\text{VO}_{10\%}$	0.603	0.558	0.580
$\text{VO}_{20\%}$	0.614	0.560	0.586
$\text{VO}_{30\%}$	0.623	0.562	0.591

Pogledajmo sada rezultate filtriranja bigrama i trigrama. U usporedbi s vrećom riječi dodatak bigrama i trigrama pozitivno utječe na rezultate klasifikacije. Sve tri mjere su u prosjeku bolje od same vreće riječi. Najboljim modelom u usporedbi s vrećom riječi se pokazao vjerojatnosni omjer uz prihvaćanje bigrama i trigrama koji se nalaze u gornjih 30% svih unigrama, bigrama i trigrama. Povećanje F_1 mjere u odnosu na vreću riječi iznosi 3.7%. Najveće povećanje preciznosti postiže se modelom frekvencije pojavljivanja riječi uz postotak p jednak 30%, a povećanje iznosi 6.1%. Najbolji odziv ima ponovno ima model vjerojatnosnog omjera uz parametar p jednak 30% čije je povećanje jednako 1.9%. Može se primijetiti da sva tri najbolja modela imaju parametar p jednak 30%. Kod bigrama je to također bio slučaj. To je još jedan pokazatelj ispravnosti pretpostavke da ekstrakcija korisnih višerječnih fraza unutar dokumenta doprinosi poboljšanju klasifikacije dokumenata.
Iako su se modeli s trigramima također pokazali bolje rješenje od vreće riječi, poboljšanje je sveukupno manje nego kada u slučaju filtracije samo bigrama. Vreća riječi može se promatrati kao filtracija unigrama, dok je filtracija bigrama i trigrama implementirana rangiranjem istih. Iako se izlučivanje bigrama uz unigrame pokazalo boljim rješenjem od izlučivanja samo unigrama, dodavanje trigrama je smanjilo kvalitetu klasifikatora. Dakle, idealna opcija jest korištenje funkcija vrednovanja bigrama uz značajke vreće riječi.

4.3. Dodatna analiza rezultata

Kao što se vidi iz rezultata u tablicama 4.2 i , model vreće riječi dominira nad distribucijskim značajkama, pa se može reći da distribucijske značajke u hrvatskome jeziku nemaju veliko značenje. Usporedbom rezultata distribucijskih značajki i n-grama može se vidjeti da je kvaliteta klasifikatora s n-gramima i vrećom riječi veća od klasifikatora učenog na distribucijskim značajkama. Među n-gramima se najboljom opcijom pokazalo korištenje bigrama uz vreću riječi, a vjerojatnosni omjer se pokazao najboljim modelom rangiranja n-grama.

5. Zaključak

U ovom radu je isprobano djelovanje distribucijskih značajki i korištenje n-grama u cilju dobivanja boljih rezultata od onih dobivenih korištenjem najčešćeg modela za reprezentaciju — vreće riječi. Dobiveni rezultati upućuju na to da distribucijske značajke koje osim frekvencije pojavljivanja riječi u dokumentu u obzir uzimaju i pozicije riječi unutar dokumenta ne doprinose kvaliteti klasifikacije dokumenata što je suprotno pretpostavkama navedenim u odjeljku 2.2. Korištenje n-grama se pokazalo puno boljim rješenjem. Eksperimentalno je utvrđeno da svi modeli daju najbolje rezultate kada se u obzir uzimaju bigrami, ali ne i trigrami.

Najboljim modelom od svih korištenih u ovom radu se pokazao model vjerojatnosnog omjera opisan formulom 2.21. Vjerojatnosni omjer za rangiranje bigrama uz parametar filtriranja jednak 30% i vreću riječi daje najbolje rezultate za klasifikaciju dokumenata. Dobri rezultati korištenja bigrama općenito upućuju na to da dvorječne fraze imaju relativno velik utjecaj na kvalitetu reprezentacije i klasifikacije dokumenata pisanih na hrvatskome jeziku.

Daljnji eksperimenti koji bi se mogli obaviti kao proširenje ovog rada uključuju eksperimentiranje s parametrom p većim od 30% s obzirom da je utvrđen trend rasta kvalitete klasifikatora s porastom parametra p. Osim toga, mogle bi se kreirati nove funkcije za evaluaciju n-grama koje bi mogle davati bolje rezultate od onih koje se koriste u ovom radu. Kod distribucijskih značajki bi se mogla isprobati semantička podjela dokumenta s obzirom da je podjela na odlomke koja je sličnija njoj davala bolje rezultate od podjele na prozore.

LITERATURA

James P Callan. Passage-level evidence in document retrieval. U *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, stranice 302–310. Springer-Verlag New York, Inc., 1994.

Maria Fernanda Caropreso, Stan Matwin, i Fabrizio Sebastiani. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. *Text databases and document management: Theory and practice*, stranice 78–102, 2001.

Corinna Cortes i Vladimir Vapnik. Support-vector networks. U *Machine Learning*, stranice 273–297, 1995.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, i Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

Jinsuk Kim i Myoung Ho Kim. An evaluation of passage-based text categorization. *Journal of Intelligent Information Systems*, 23(1):47–65, 2004.

Xiao-Bing Xue i Zhi-Hua Zhou. Distributional features for text categorization. *Knowledge and Data Engineering, IEEE Transactions on*, 21(3):428–442, 2009.

Izgradnja i odabir značajki za klasifikaciju dokumenata na hrvatskome jeziku

Sažetak

Ovaj rad istražuje kako prikupljanje distribucijskih značajki i tvorba n-grama riječi utječe na klasifikaciju dokumenata u odnosu na standardni model reprezentacije teksta kao vreće riječi. Korištenje tih modela iziskuje dodatne računalne resurse, ali oni mogu nositi više informacija nego što ih nosi reprezentacija korištenjem vreće riječi. Distribucijske značajke dobivene su ekstrakcijom informacija o pozicijama u dokumentu na kojima se riječi pojavljuju, dok je važnost n-grama ispitana pomoću četiri funkcije za evaluaciju značajki. Rezultati su uspoređeni s modelom vreće riječi.

Ključne riječi: Obrada prirodnog jezika, strojno učenje, vreća riječi, distribucijske značajke, n-gram, stroj potpornih vektora, liblinear.

Feature Construction and Selection for Document Classification in Croatian Language

Abstract

In this work we investigate how does extracting distributional features and using word n-grams for document classification compare to using bag of words — a more traditional model for document representation. Extraction of these features requires additional computational resources, but they can carry more information about the document compared to the bag of words baseline. In order to extract distributional features we use positions of word occurrences in a document. N-grams are rated by evaluation of four feature evaluation functions in order to select only useful n-grams which are then treated as a single feature. Results obtained by using these models are compared with the bag of words model.

Keywords: Natural language processing, machine learning, bag of words, distributional features, n-gram, support vector machine, liblinear.