Modelovanje Znanja i Klasifikacija Naučnih Radova Pomoću Topic Modeling Algoritma

Knowledge Modeling and Classification of Scientific Papers Based on Topic Modeling

Vladisav Jelisavčić[1], Bojan Furlan[2], Jelica Protić[2], Veljko Milutinović[2]

[1]Matematički institut SANU

[2]Elektrotehnički fakultet u Beogradu

Sadržaj - U ovom radu biće predstavljeno nekoliko algoritama za topic modeling, primenjenih u svrhu modelovanja znanja iz baze naučnih radova. Takođe, data je evaluacija jednog *topic modeling* algoritma primenjena na dati domen. Dobijene teme su najpre evaluirane pomoću ključnih reči dostavljnih uz svaki rad i upoređen je uticaj različitih metoda pretprocesiranja na kvalitet dobijenih tema. Na kraju pomoću dobijenih tema izvršena je klasifikacija radova po naučnim oblastima i diskutovani su pravci za dalje istraživanje.

Abstract – *Several topic models are presented with application in modeling knowledge from scientific papers. Additionally, an evaluation of a topic model applied to the given domain has been provided. Inferred topics were first evaluated using corresponding keywords supplied with each paper and the influence of different methods of textual preprocessing on topic quality was compared. Finally, using inferred topics a supervised classification of papers according to the scientific area was made and future research directions are suggested.*

1. UVOD

Modelovanje znanja iz tekstualnog korpusa je proces čiji je cilj dobijanje sažete reprezentacije, odnosno, suštine (eng. gist) sadržane tekstom koju je zatim moguće predstaviti u odgovarajućem mašinski struktuiranom obliku. Tekst koji je potrebno modelovati napisan je ljudski razumljivim jezikom, tzv. prirodnim jezikom. Modelovanje znanja predstavlja upravo proces ekstrakcije i prevođenja suštine iz teksta napisanog prirodnim jezikom u jasno definisan format.

Kod prirodnih jezika, pronalaženje suštine nije trivijalan proces. Naime, usled same prirode ljudske komunikacije, moguće je isti smisao izraziti na više načina, različitim izborom reči. Zato kažemo da je suština koju pokušavamo da ekstrakujemo iz teksta prikrivena izborom reči

Segmentacijom teksta u abstraktne teme (eng. topic modeling) moguće je pojednostavniti analizu velikih količina nestruktuiranog teksta.

Naučni radovi predstavljaju nestruktuiran tekst iz jedne ili više srodnih naučnih oblasti. Automatsko zaključivanje uske naučne oblasti na osnovu samog teksta rada, usled konstantnog razvoja nauke, nije nimalo jednostavan problem. U ovom radu biće predstavljeno nekoliko algoritama za topic modeling, primenjenih u svrhu rešavanja pomenutog problema.

Polazna pretpostavka ovog rada je da metapodaci, koji su najčešće prisutni uz svaki naučni rad, u izvesnoj meri predstavljaju suštinu samog rada. Svaki rad najčešće sadrži listu ključnih reči koju dostavlja sam autor. Smisao ove liste je da omogući lakše pretraživanje i autor je sam generiše, birajući reči koje najviše opisuju oblast kojom se rad bavi. Samim tim, lista ključnih reči predstavlja na izvestan način sažetak celokupnog rada i može biti pogodna za evaluaciju dobijenog modela. Takođe, radovi su najčešće svrstani u jednu ili više oblasti kojima se rad bavi, što je takođe dodatna informacija o sadržaju odnosno suštini koju želimo da izdvojimo iz rada.

U nastavku rad je struktuiran na sledeći način: u odeljku 2. su prezentovane osnove algoritama za modelovanje tema i predstavljeni ključni koncepti. U odeljku 3. je predstavljen softverski alat koji je korišćen u radu, u odeljku 4. opisan je način kreiranja korpusa za evaluaciju. U odeljku 5. opisan je postupak evaluacije i disukutovani su dobijeni rezultati . Na kraju je dat zaključak i pravci za dalji rad, kao i spisak literature.

2. PREGLED POSTOJEĆIH METODA

U ovom radu je korišćen Topical N-Grams [6] algoritam za modelovanje i pronalaženje tema u tekstualnim dokumentima. TNG spada u grupu takozvanih „topic modeling“ algoritama koji se zasnivaju na bajesovskom zaključivanju. Da bi smo predstavili ovaj algoritam, prezentovaćemo prvo nekoliko algoritma koji su bili osnov za nastanak TNG, i objasniti ključne koncepte neophodne za njegovo razumevanje.

2.1 LSA

LSA je statistički metod za analizu skrivene povezanosti podataka i koristi se u pretraživanju informacija, mašinskom učenju iz teksta, obradi prirodnih jezika i srodnim oblastima. Osnovna ideja ovog algoritma je nalaženje skrivene strukture tema (eng. *topic*) ili koncepata u tekstualnom korpusu, koja u sebi sadrži značenje za koje kažemo da je sakriveno "šumom" prirodnog jezika. Kako autor nekog teksta ima na raspolaganju veliki izbor reči, jedan isti koncept se može izraziti na više različitih načina. Usled toga, možemo reći da je koncept u izvesnoj meri "sakriven" slučajnim izborom reči kojim je opisan. Nasuprot formalnim jezicima, iz prirodnih jezika, usled njihove raznovrsnosti i slobodne strukture, nije jednostavno automatski izdvojiti semantički smisao. U pokušaju semantičke ekstrakcije, LSA kao i mnogi drugi algoritmi, aproksimira suštinu (eng. gist) teksta grupisanjem reči u koncepte na osnovu frekvencije pojavljivanja. Na ovaj način, semantičke celine teksta se mogu posmatrati kao skupovi međusobno povezanih reči.

LSA je nastao u pokušaju da se reši problem traženja relevantnih dokumenata pomoću ključnih reči. Traženje relevantnih dokumenata nije nimalo jednostavno, jer jednostavno upoređivanje reči najčešće ne daje zadovoljavajuće rezultate. Osnovna zamisao je da reči nose značenje, tj. koncepte, i da je ono što je zaista potrebno, upravo poređenje koncepata. LSA se suočava sa ovim problemom tako što preslikava i reči i dokumenta u takozvani "konceptni" ili "semantički" prostor. Poređenje se dalje vrši u redukovanom, semantičkom prostoru i na taj način se u velikoj meri rešava problem šuma nastalog usled prirode jezika.

Dokumenti se najpre predstavljaju kao vektori u N-dimenzionalnom prostoru pojmova, gde je N broj različitih pojmova, tj. različitih reči u korpusu. Ovako definisan korpus, tj. vektorski prostor, može se zapisati u obliku matrice. Semantički prostor može se dobiti redukcijom dimenzija ove matrice. LSA algoritam se zasniva na dekompoziciji singularnih vrednosti (DSV), dobro poznatoj algebarskoj metodi koja faktorizuje ovu matricu na proizvod tri različite matrice, i zatim redukuje broja njihovih dimenzija. Upravo ovom redukcijom vrši se klasterizacija dokumenata koji sadrže iste reči i dolazi do semantičkog prostora. Formalan opis LSA algoritma predstavljen je u [1].

Ovaj algoritam, kao i ostali algoritmi koji se pominju u ovom tekstu, se bazira na takozvanom bag-of-words modelu. U ovom modelu tekst je predstavljen kao neuređeni skup reči, pritom zanemarujući njihov redosled. Zbog ovog pojednostavljenja dati model je neosetljiv na sintaksu i morfologiju posmatranog teksta.

2.2. PLSI

Glavni nedostatak LSA algoritma je zasnovanost na algebarskoj dekompoziji i nedostatak teorijske osnove za modelovanje tema. Naime, konceptni prostor koji se dobija pomoću DSV nema statističkih osnova. Kao odgovor na pomenuti problem nastala je nova grupa algoritama u vidu generativnih probabilističkih algoritama za modelovanje tema (eng. *topic models*).

Generativni modeli opisuju proceduru za generisanje dokumenata kroz niz probabilističkih koraka. Ukoliko bi znali raspodelu reči po dokumentima u nekom korpusu, mogli bi da modelujemo korpus generisanjem reči po nekoj raspodeli, za svaki dokument posebno. U ovakvom scenariju, ne modeluje se redosled reči, već se dokumenti smatraju kao kolekcija reči, što se između ostalog uklapa u bag-of-words model definisan ranije. Na ovaj način, kroz nedeterminističko generisanje reči modeluje se kompleksniji proces pisanja teksta. Nakon definisanja generativnog procesa, moguće je koristeći bajesovsku logiku doći do inverznog procesa, tj. definisati odgovarajući algoritam zaključivanja na osnovu kojeg je moguće dobiti raspodele skrivenih promenljivih.

U najjednostavnijem slučaju ceo korpus može se modelovati jednom raspodelom. Reči svakog dokumenta u korpusu nezavisno se generišu iz zajedničke multinomialne raspodele:

$$p(\boldsymbol{w}) = \prod_{n=1}^{N} p(w_n) \tag{1}$$

Mana ovakvog pristupa je što se svaki dokument modeluje istom raspodelom, iako različiti dokumenti u korpusu mogu imati jako malo sličnosti, tj. različite raspodele. Koncepti izraženi jednim dokumentom koji govori, na primer, o nekoj oblasti iz biologije značajno se razlikuju od koncepata koji se javljaju u tekstu iz astrofizike. Samim tim, izbor reči u tim dokumentima će biti značajno drugačiji.

Usložnjavajući opisani model, uvodimo novu diskretnu slučajnu promenljivu z. Promenljiva z predstavlja temu (eng. topic). Sada, svaki dokument u korpusu može se generisati tako što se prvo odabere tema *z*, a zatim nezavisno izgeneriše *N* reči iz uslovne multinomialne raspodele $p(w|z)$:

$$p(w) = \sum_{z} p(z) \prod_{n=1}^{N} p(w_n|z) \tag{2}$$

Teme se mogu posmatrati kao raspodele reči u dokumentu. Mana ovakvog modela je što pretpostavlja da svaki dokument odgovara tačno jednoj temi. U praksi se pokazalo da ovakav pristup ne pruža dovoljnu granularnost, jer najčešće mnogi dokumenti govore o sličnoj tematici, kao i što mnogim tekstovima odgovara više od jedne tematike. Naravno, sam pojam "tematika" je, u kontekstu automatske obrade i modeliranja prirodnih jezika, komplikovano definisati, ali kao što je već rečeno, to je upravo sama ideja topic modeling-a; grupisanje reči u celine kojima se modeluje kontekst, gde bi pod kontekstom mogli da smatramo skupove reči koje se često pojavljuju zajedno.

Probabilistic latent semantic indexing (pLSI) je model koji možemo dobiti daljim usložnjavanjem prethodno opisanih modela. U ovom modelu, dokument nije generisan isključivo iz jedne teme, već svakom dokumentu odgovara različita raspodela po temema. Dokument se sada generiše tako što se, za svaku od N reči u dokumentu, prvo odabere tema iz odgovarajuće raspodele tema za taj dokument, pa se zatim iz odabrane teme, koja predstavlja raspodelu po rečima, izgeneriše sama reč.

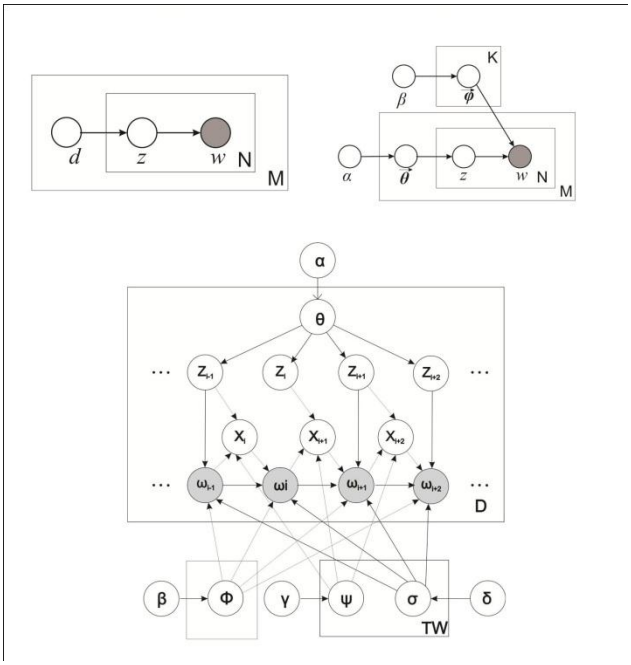
Polazna tačka pLSI modela je da su dokument d i reč w_n uslovno nezavisni ako nam je data skrivena tema z:

$$p(d, w_n) = p(d) \sum_z p(w_n|z)p(z|d) \qquad (3)$$

U pLSI modelu slučajne promenljive *d* i *w* su takozvane obzervabilne ili opažene (eng. observed) promenljive, dok slučajna promenljiva *z* predstavlja skrivenu (eng. latent) promenljivu. Promenljive *d* i *w* su obzervabilne zato što možemo opaziti njihove realizacije u korpusu, realizacije promenljive *w* su konkretne reči, odnosno ako posmatramo samu implementaciju, indeksi u rečniku svih pojmova, dok svaka realizacija slučajne promenljive *d* jednoznačno određuje dokument u okviru korpusa. Realizacije slučajne promenljive *z* ne možemo opaziti u korpusu, njihovu raspodelu moramo dobiti putem bajesovog zaključivanja. Zapravo, raspodela tema po dokumentima $p(z|d)$ i reči po temama $p(w|z)$ je upravo ono što želimo da dobijemo. Ovo se najbolje može uočiti sa slike 1. gde su predstavljene odgovarajuće bajesove mreže u *plate* notaciji, standardnoj notaciji za opisivanje bajesovih mreža. Za zaključivanje parametara traženih raspoleda potrebno je primeniti neki od standardnih algoritama za estimaciju parametara u statističkim modelima kao što je EM (Expectation Maximization) [2].

PLSI model uspešno rešava neke od problema koji su se javili u prethodno opisanom modelu. U pLSI modelu dokumenti mogu sadržati više tema i uslovna raspodela $p(z|d)$ za neki dokument *d* se može protumačiti kao vektor težinskih koeficijenata koji predstavljaju koliko koja tema odgovara datom dokumentu.

Problem koji se javlja kod ovog modela je nemogućnost primene na nekom novom skupu dokumenata izvan korpusa na kojem je istreniran. Slučajna promenljiva *d* koju smo uveli, jednoznačno određuje dokument u okviru trening skupa dokumenata, što povlači sa sobom da se dobijena raspodela $p(z|d)$ ne može primeniti na prethodno neviđen dokument, jer *d* može uzimati samo one vrednosti koje odgovaraju dokumentima trening korpusa. Usled ovoga, pLSI nije praktičan za dinamičku upotrebu, u situacijama kada je potrebno analizirati dokumente izvan trening korpusa.



Slika 1. Grafički prikaz odgovarajućih bajesovih mreža opisanih algoritama u plate notaciji. Na gornjoj slici su opisani PLSI i LDA modeli redom, dok je na donjoj slici opisan TNG model.

Još jedan problem sa kojim se susrećemo je broj parametara koje je potrebno proceniti. Porast broja parametara sa brojem dokumenata je krajnje nepoželjan jer može uzrokovati probleme sa overfitovanjem. Naime, model koji bi dobili za dovoljno veliki broj dokumenata bi bio usko specifičan i, čak i kad ne bi postojao prethodno navedeni problem, ne bi mogao da se primeni na novi skup podataka.

2.3 LDA

Latent Dirichlet Allocation (LDA) model [3] je dalje unapređenje pLSI modela. Ovaj model uspešno rešava probleme pLSI modela, kao što su zavisnosti broja parametara od broja dokumenata, što ujedno predstavlja problem i kompleksnosti samog algoritma i overfitovanja. Takođe, LDA rešava i problem zbog kojeg je pLSI primenjiv samo na statičan, trening skup podataka što omogućava primenu unapred treniranog modela na nekom novom, do tada neviđenom, test skupu. Samim tim, LDA nalazi brojne primene za koje pLSI zbog svoje statičnosti, tj. osobine da se mora primeniti na fiksan korpus, nije adekvatan.

U odnosu na pLSI, kod kog je neophodno estimirati $D * k + k * V$ parametara, LDA odlazi korak dalje i definiše raspodelu tema po dokumentima kao k-parametarsku skrivenu slučajnu promenljivu umesto $D * k$ individualnih parametara koji su eksplicitno vezani za trening skup. Usled kompleksnosti samog modela, nije moguće naći analitičko rešenje za jednačine zaključivanja pa je neophodno koristiti neki od aproksimativnih algoritama kao što je Gibsovo sempliranje [4].

2.4 TNG

Modeli koje smo do sada prezentovali se baziraju isključivo na bag-of-words pretpostavci; dokumenti se posmatraju kao neuređeni skupovi reči i pri tome se zanemaruje sva informacija koja se može dobiti iz redosleda reči. Ovakav pristup, iako u velikom broju slučajeva daje dobre rezultate, nije primenjiv kada je potrebno detektovati sintagme u tekstu. Na primer, ako neki tekst govori o neuralnim mrežama, reč neuralna će se u tekstu često nalaziti pored reči mreža, češće od bilo koje druge reči. Sintagma, tj. n-gram, u ovom slučaju nosi više informacije od pojedinačnih reči od kojih je sastavljen. Često je upotreba ovakve informacije neophodna i donosi mnogo bolje rezultate od prostog bag-of-words unigram modela. Još izraženiji je primer često korišćen u literaturi, problem koji nastaje u pretraživanju pojma "Bela kuća". Same reči "bela" i "kuća" ne nose dovoljno informacije i mogu se pojavljivati u temi o nekretninama, arhitekturi, uređivanju enterijera itd. Nasuprot tome, izraz "Bela kuća" nosi mnogo više informacije, i drastično sužava broj tema u kojima se može pojaviti. Na taj način možemo dobiti više informacije o temama o kojima zaključujemo iz korpusa i, samim tim, model će bolje odgovarati korpusu i njegovoj semantici koju pokušavamo da ekstrakujemo.

U problemu modelovanja znanja iz naučnih radova, ovakav pristup je neophodan. Stručni termini, kao što je malopre naveden primer neuralna mreža, najčešće su n-grami od dve, pa i nekoliko reči. Bez ovakvog pristupa mnogi stručni termini bi se izgubili prilikom generisanja tema. Osnovna ideja modeliranja znanja pomoću topic modela je utvrđivanje korelacije između termina i stručnih izraza koji se mogu javiti u nekoj naučnoj oblasti. Grupisanjem stručnih termina u teme, i zatim određivanjem raspodele po temama za neki dokument, mogli bismo grubo modelirati kojim naučnim oblastima pripada. Stoga možemo zaključiti da je uočavanje n-grama neizostavno u primeni topic modeling-a u ovom slučaju.

TNG model, za razliku od prethodno opisanih modela, uvodi dodatne slučajne promenljive čija vrednost određuje da li je neka reč izolovana ili je deo veće celine, tj. n-grama. Grafički prikaz modela je prikazan na slici 1. dok su implementacioni detalji dati u [5].

3. KORIŠĆENI ALATI

Za generisanje tema iz korpusa korišćen je MALLET Topic Modeling Toolkit. MALLET je biblioteka za mašinsko učenje realizovana u programskom jeziku Java. Namenjena je pre svega obradi prirodnih jezika, klasifikaciji dokumenata, klasterizaciji, topic modeling-u i drugim primenama mašinskog učenja na tekstualne podatke, mada se može koristiti i u druge svrhe.

MALLET podržava nekoliko topic modela. Od modela opisanih u ovom radu, podržani su Latent Dirichlet Allocation kao i Topical N-grams. Pored navedena dva modela, ova biblioteka podržava i Pachinko Allocation kao i Hierarchical LDA modele, koji pored modelovanja samih tema iz teksta, mogu takođe da modeluju i njihove uzajamne odnose.

Pored alata za topic modeling, ova biblioteka poseduje alate za klasifikaciju dokumenata, kao i za obeležavanje sekvenci (eng. sequence tagging). Podržano je nekoliko algoritama za klasifikaciju kao što su Naive Bayes, Maximum Entropy i Decision Trees.

4. KREIRANJE KORPUSA

Za kreiranje korpusa za evaluaciju korišćena je baza naučnih radova Cogprints. Ova arhiva sadrži preko dve hiljade radova iz oblasti kognitivnih nauka. Svaki rad je svrstan u jednu ili više oblasti među kojima su psihologija, lingvistika, informatika, filozofija, biologija. Pored oblasti, svaki rad sadrži i metapodatke - spisak ključnih reči, zbog čega je upravo korišćena ova baza.

Radovi su preuzeti u izvornom tekstualnom obliku. Zatim je izvršeno pretprocesiranje datih radova. Jedan od ciljeva je upravo bio i ispitivanje kako različite metode pretprocesiranja utiču na rezultate modelovanja. Motiv za ovakav pristup potiče iz načina prezentacije teksta.

U govornom jeziku sve reči ne nose podjednaku količinu informacije u rečenici. Reči kao što su imenice i glagoli nose više informacija od značaja nego funkcionalne reči kao što su predlozi i veznici. Kako u ovom radu koristimo isključivo *bag-of-words* model predstavljanja teksta, u kojem se zanemaruje redosled reči u rečenici, ovakve reči je moguće izbaciti iz ulaznog teksta. Ovo je postignuto koristeći odgovarajući Part-of-Speech tagger. Takođe, ni sve reči iste vrste ne nose istu količinu informacije. Neke imenice su informativnijeg karaktera od drugih jer se pojavljuju ređe u okviru korpusa.

Iz tog razloga implementiran je *Term-base sampling* algoritam [6] za rangiranje i detekciju stop reči - najmanje informativnih reči u korpusu. Ove reči je poželjno izbaciti tokom pretprocesiranja teksta. Takođe, reči se često mogu pojavljivati u više različitih oblika, tj. mogu imati isti koren reči a drugačije sufikse. Jednostavno leksikografsko poređenje reči, na kojem se zasnivaju topic modeling algoritmi, detektuje izvedene reči i njihove različite oblike kao različite reči. Ovo je moguće u izvesnoj meri korigovati primenom tzv. stemera, posebnih algoritama koji uklanjaju sufikse reči čime se ovaj efekat ublažava. U ovom radu primenjene su opisane tehnike pretprocesiranja i upoređen je njihov međusobni uticaj pri modelovanju znanja.

5. EVALUACIJA

Proces evaluacije izvršen je na sledeći način: Najpre su iz pretprocesiranih radova uočene teme o kojima rad govori pomoću topic modeling algoritama. Zatim, dobijena raspodela po temama evaluirana je pomoću ključnih reči koristeći različite metrike,

Osnovna metrika od koje smo pošli je kosinusna sličnost. Ova metrika ne pruža dovoljnu preciznost prilikom

evaluacije zbog malog broja ključnih reči koje su dostupne uz svaki rad (u proseku ispod 7). Zbog toga smo primenili modifikovane precision i recall metrike. Ove metrike se baziraju na pretpostavci da se u skupu svih elemenata nalaze "razbacani" elementi koje je nepohodno pronaći. U našem slučaju, skup relevantnih elemenata za neki dokument d_i je skup ključnih reči k_i:

$$Precision = \sum_{j=0}^{N} t_{ij} \frac{C(k_i, t_j)}{C(t_j)} \quad (4)$$

$$Recall = \sum_{j=0}^{N} t_{ij} \frac{C(k_i, t_j)}{C(k_i)} \quad (5)$$

U navedenim izrazima, t_j predstavlja temu sa rednim brojem j, k_i predstavlja skup ključnih reči za neki dokument i, dok veličine $C(k_i, t_j)$, $C(t_j)$ i $C(k_i)$ predstavljaju redom broj pojavljivanja ključnih reči iz skupa k_i u temi t_j, ukupan broj pojmova u temi t_j i ukupan broj ključnih reči u skupu k_i. Veličina t_{ij} predstavlja udeo teme t_j u dokumentu d_i, dok N predstavlja ukupan broj tema. Veličine precision i recall daju različit uvid prilikom evaluacije i međusobno se dopunjuju. Iz ovog razloga uvodi se veličina F-score koja je definisana kao njihova harmonijska sredina, čime se jednom veličinom objedinjuje informativni karakter obe veličine.

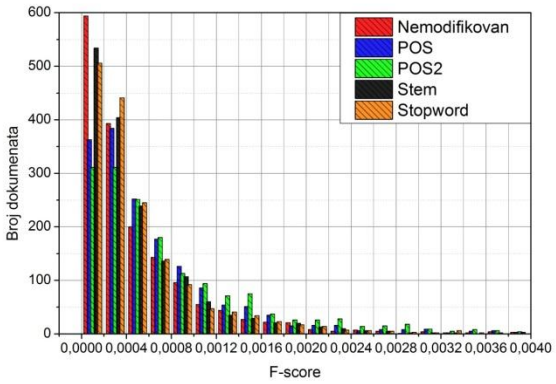
Izvršeno je poređenje po nekoliko kriterijuma koristeći pomenute merike i rezultati su prikazani u Tabeli 1. Korpus od 1639 dokumenata modelovan je sa 50,100,200 reči i 350 tema.

	$\bar{P}$	$\bar{R}$	$\bar{F}$	$\bar{C}$
Nemodifikovan	2,51E-04	0,52494	5,02E-04	0,00568
POS	3,55E-04	0,509	7,09E-04	0,00706
POS2	4,43E-04	0,4506	8,85E-04	0,00732
Stem	2,65E-04	0,51188	5,28E-04	0,00721
Stopword	2,64E-04	0,51408	5,27E-04	0,00706

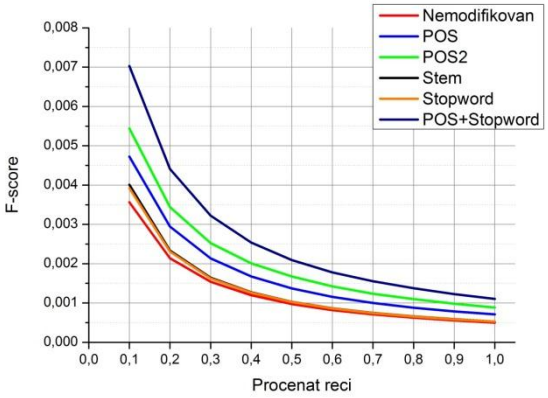
Tabela 1. Zavisnost srednje vrednosti precision ($\bar{P}$), recall ($\bar{R}$) , F-score ($\bar{F}$) i kosinusne sličnosti ($\bar{C}$) od metode pretprocesiranja (redom): nemodifikovan - bez procesiranja, POS – izbačene sve reči izuzev prideva i imenica, POS2 – izbačene sve reči izuzev prideva i nevlastitih imenica, Stem – sve reči su stemovane, Stopword – izbačeno je 5000 najmanje informativnih reči.

Za različito pretprocesirane korpuse dobijeni rezultati su potvrdili predpostavku da modelovanje znanja iz korpusa u kojem su odstranjene funkcionalne reči daje bolje rezultate nego primenom algoritama na nemodifikovanom korpusu. Uklanjanje vlastitih imenica iz korpusa unosi dodatno poboljšanje. Takođe, izvesna poboljšanja su postignuta i stemovanjem. Generisanje stopword liste i njihovo izbacivanje u ovom slučaju je donelo poboljšanje tek u kombinaciji sa prethodno opisanim metodama pretprocesiranja. Histogram evaluiranih korpusa je prikazan na slici 2. Kao što je napomenuto, jedan od ciljeva primene topic modela je dobijanje sažete reprezentacije tekstova što se može postići posmatranjem samo prvih N najznačajnijih pojmova u dobijenim raspodelama reči po temama. Na slici 2. je prikazana srednja vrednost F-score metrike na nivou korpusa u zavisnosti od normalizovanog broja posmatranih reči u okviru tema.

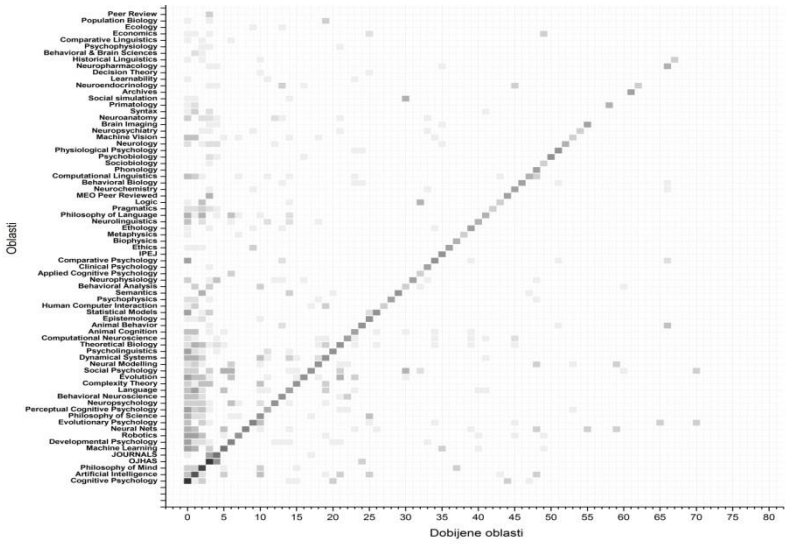
Takođe, izvršeno je poređenje rezultata na osnovu broja tema, jer jedan od ciljeva rada je bila i evaluacija primene topic modeling algoritama na određivanje naučne oblasti kojoj rad pripada. Samom primenom topic modeling algoritama dobijamo raspodelu po skrivenim temama za svaki dokument. Teme, kao što je napomenuto, predstavljaju skupove međusobno statistički povezanih reči i same za sebe ne moraju nužno određivati jednoznačno oblasti kojima pripadaju. U ovom radu korišćen je model supervizovanog učenja oblasti naučnih radova koji se zasniva na određivanju raspodele oblasti po temama za unapred zadat obučavajući skup. Radovi su unapred svrstani u jednu ili više oblasti. Na osnovu ovih metapodataka modelovana je raspodela po oblastima za svaki rad. Zatim je korpus izdeljena na trening i test skup. Na trening skupu obučen je jednostavan Bajesov klasifikator na osnovu generisanih raspodela. Rezultati evaluacije na test skupu prikazani su na Slici 4. Uspešno su prepoznate oblasti u 52.4% slučajeva, što je s obzirom na broj oblasti koje korpus poseduje značajan rezultat.



Slika 2. Uporedni prikaz korpusa za nekoliko različitih metoda pretprocesiranja.



Slika 3. Zavisnost srednje vrednosti evaluiranog korpusa u zavisnosti od dimenzionalnosti semantičkog prostora



Slika 2. Rezultat zaključivanja oblasti na osnovu raspodele po temama koja odgovara test dokumentima. Na X osi su predstavljeni redni brojevi oblasti koje su dobijeni zaključivanjem, Y osa predstavlja stvarno dodeljene oblasti. Na dijagonali su uspešno zaključene oblasti koje čine 52.4 % od ukupnog broja oblasti u test skupu.

6. ZAKLJUČAK I DALJI RAD

Ovaj rad prikazuje evaluaciju algoritama za modelovanje znanja nad različito pretprocesiranim korpusima. Raspodelu po temama dobijenu pomoću topic modeling algoritma evaluirana je pomoću ključnih reči koristeći različite metrike. Najpre pomoću kosinusne distance, a zatim uvodeći modifikovanu precision, recall i F-score metriku. Dobijeni rezultati su u skladu sa pretpostavkom da modelovanje znanja iz korpusa u kojem su odstranjene funkcionalne reči daje bolje rezultate nego primenom algoritama nad nemodifikovanim korpusom. Dodatno uklanjanje vlastitih imenica iz korpusa unosi poboljšanje preciznosti sa izvesnim smanjenjem recall vrednosti. Ova pojava može se objasniti retkim stručnim terminima, koji su greškom klasifikovani kao vlastite imenice. Takođe, izvesna poboljšanja su postignuta i stemovanjem. Generisanje stopword liste reči i njihovo izbacivanje donelo je vidno poboljšanje tek u kombinaciji sa prethodno opisanim metodama.

Takođe, prikazan je i jedan model supervizovanog učenja oblasti naučnih radova koji se zasniva na određivanju raspodele oblasti po temama za unapred zadat obučavajući skup. Na osnovu dostupne klasifikacije po oblastima modelovana je raspodela po oblastima za svaki rad. Primenom topic modeling algoritama dobijena je raspodela po temama za svaki rad. Zatim je baza izdeljena na trening i test skup. Na osnovu trening skupa dobijena je raspodela oblasti po temama. Primenom na test skupu uspešno su prepoznate oblasti u 52.4% slučajeva, što je s obzirom na broj oblasti koje baza poseduje značajan rezultat.

U daljem radu poželjno bi bilo iskoristiti informaciju o pripadnosti naučnoj oblasti i ugraditi je sam model. Novonastali usložnjen model bi mogao doneti dodatna poboljšanja raspodela tema po dokumentima i doprineti boljoj semantičkoj reprezentaciji samih radova. Takodje, jedan od glavnih nedostataka topic modeling algoritama je zahtevanje velikih količina teksta, što kod naučnih radova često nije slučaj jer su javno dostupni isključivo abstrakti. Jedno rešenje ovog problema bi bilo proširivanje skupa svih pojmova iz abstrakta njihovim semantički bliskim pojmovima pomoću rečnika ili ontologije.

LITERATURA

[1] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "*Indexing by latent semantic analysis,*" Journal of the American Society for Information Science, vol. 41, pp. 391–407, 1990.

[2] T. Hofmann, "*Probabilistic latent semantic indexing*," in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ser. SIGIR '99. New York, NY, USA: ACM, 1999, pp. 50–57.

[3] David M. Blei, Andrew Y. Ng, and Michael I. Jordan, "*Latent Dirichlet allocation,"* Journal of Machine Learning Research, 2003.

[4] Thomas L. Gri_ths and Mark Steyvers, „*Finding scientific topics,"* Proceedings of the National Academy of Sciences of the United States of America, 2004.

[5] Xuerui Wang, Andrew Mccallum, Xing Wei, **„***Topical N-Grams: Phrase and Topic Discovery, with an Application to Information Retrieval,"* Seventh IEEE International Conference on Data Mining ICDM 2007 (2007), pp. 697-702.

[6] Rachel Tsz-Wai Lo, Ben He, Iadh Ouni, „*Automatically Building a Stopword List for an information Retrieval System,"* In Proceedings of the Fifth Dutch-Belgian Information Retrieval Workshop, 2005.