Analiziranje sadržaja tekstova upotrebom ontologije i semantičke sličnosti

Dejan Prodanović, Bojan Furlan, Boško Nikolić

***Sadržaj* — U radu je opisan novi pristup za analiziranje tekstova. Pristup je baziran na primeni onotlogije za određenu oblast i semantičke sličnosti. Rezultati analize su primenjeni na studentski forum za oblast arhitekture i organizacije računara na Elektrotehničkom fakultetu Univerziteta u Beogradu. Predavačima su izdvojeni samo oni postovi čiji sadržaj pripada nameni foruma i povezani su sa temama iz kursa.**

***Ključne reči* — semantička sličnost, ontologija, semantičke mreže, analiza foruma.**

I. Uvod

Postoje brojni pristupi za pronalaženje informacija (*Information Extraction*) iz teksta napisanog na prirodnom jeziku. Neki su bazirani na *text mining* tehnikama i statističkom pristupu, dok se drugi zasnivaju na principima računarske lingvistike i obrade prirodnog jezika. Prednosti statističkog pristupa su u većoj preciznosti sa povećanjem korpusa za obradu, jednostavnija realizacija i duža istorija razvoja i upotrebe. Sa druge strane, ukoliko je potrebno obraditi kraći tekst (komentar, pitanje ili odgovor), što je slučaj kod foruma ili blogova, ovakav pristup neće dati najbolje rezultate imajući u obziru mali broj reči koje se mogu analizirati. Iz tog razloga se za kraće tekstove može upotrebiti neka od tehnika računarske lingvistike. Međutim, alati za automatsko pronalaženje informacija mogu biti nedovoljno precizni i izostaviti informacije vredne za dalju analizu i izvršavanje. Tradicionalni pristupi za pronalaženje informacija, čak i prošireni sa korišćenjem koncepata umesto reči/izraza ne daju uvek prava poklapanja, ako nema preklapanja konkretnih koncepata koji reprezentuju semantiku. Kako se opisi entiteta tretiraju kao nezavisne celine, relacije koje nisu eksplicitno navedene u opisu entiteta obično se ignorišu. Pojam semantičke sličnosti je proširen razmatranjem bitnih svojstava i relacija između koncepata pomoću ontologije.

Koncept dodeljivanja metrike skupovima izraza ili dokumenata zasnovane na sličnosti njihovog značenja je jedan od ključnih za razumevanje prirodnih jezika, jer omogućava pravljenje smislenih poređenja i zaključivanja. Zbog toga određivanje semantičke sličnosti ima važnu ulogu u automatskoj kategorizaciji teksta, mašinskom prevođenju, pronalaženju informacija i drugim oblastima veštačke inteligencije. Problem semantičkog poređenja kratkih tekstova ima poseban značaj, jer su kratki tekstovi u širokoj upotrebi na Internetu, u formi natpisa i opisa proizvoda, anotacija slika i web stranica, kratkih novinskih naslova, vesti, komentara, itd. Ovaj problem takođe ima važnu ulogu u oblastima vezanim za obrazovanje i učenje, kao što su studentski forumi i elektronski razgovori, automatsko testiranje i ocenjivanje zadataka, itd.

Slični problemi postoje i na forumima koji se koriste kao podrška učenju i kao dodatni resursi za realizaciju pojedinih kurseva. Jedan od takvih foruma je i forum za oblast arhitekture i organizacije računara studenata Elektrotehničkog fakulteta Univerziteta u Beogradu. Sami studenti su moderatori ovog foruma i koriste ga intezivno kao izvor dodatnih informacija za predmete iz ove oblasti. Za predavače veći deo sadržaja foruma je nekoristan (tipa diskusije studenata šta je predavano na kom času, koje gradivo se pojavljuje na kolokvijumima, mogućnost zamene termina laboratorisjkih vežbi, ...), ali može sadržati i značajne informacije. Način diskusije je neformalan, pa predavač može indirektno doći do informacija koji delovi kursa su najzanimljiviji studentima, koje teme im predstavljaju najveći problem, kada im je potrebna dodatna literatura ili objašnjenje. Zato bi predavačima od velike pomoći bio alat koji ima mogućnost izdvajanja samo onih postova čiji sadržaji zaista pripadaju nameni foruma i koji govore o temama vezanim za sadržaj kursa. U ovom radu je opisan jedan takav sistem, koji za analizu postavljenih tekstova koristi ontologiju i semantičku sličnost. U drugom poglavlju je kroz kratak pregled radova opisan problem i predstavljen predlog rešenja, dok su u trećem i četvrtom poglavlju opisane njegove najbitnije faze. U petom poglavlju je dat zaključak.

II. Opis problema i predlog rešenja

U otvorenoj literaturi se može pronaći više radova koji se bave analiziranjem sadržaja tekstova.

U radu [1], opisan je metod za određivanje semantičke sličnosti dva kratka teksta (rečenice ili pasusa) koristeći zajedno mere sličnosti reči zasnovane na rečniku i tekstualnom korpusu, gde se za svaku reč u tekstu identifikuje najbolje poklapanje sa drugom reči u suprotnom tekstu. Potom se taj rezultat pridodaje ukupnoj meri semantičke sličnosti.

U radu [2], predloženo je poboljšanje ovog pristupa koje uključuje algoritam leksičke sličnosti (poređenje na nivou

D. Prodanović, Elektrotehnički fakultet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija; (telefon: 381-11-3218320, e-mail: pd095011p@student.etf.bg.ac.rs)

B. Furlan. Elektrotehnički fakultet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija; (telefon: 381-11-3218320, e-mail: bfurlan@etf.rs)

B. Nikolić, Elektrotehnički fakultet u Beogradu, Bulevar kralja Aleksandra 73, 11120 Beograd, Srbija; (telefon: 381-11-3218320, e-mail: nbosko@etf.rs)

niza karaktera) zajedno sa merom semantičke sličnosti reči zasnovane na tekstualnom korpusu.

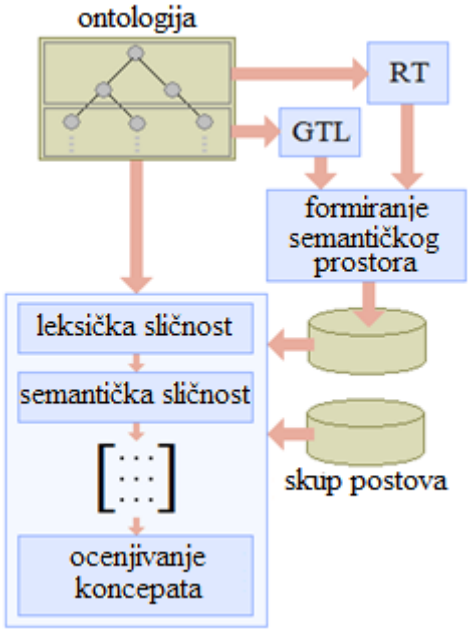
U radu [3], autori su predstavili metod za određivanje sličnosti rečenica koji koristi plitko parsiranje teksta. Iz rečenica se izdvajaju imenice, glagoli, kao i predlozi. Konačna sličnost se računa kao kombinacija sličnosti ove tri vrste izraza.

U radu [4], kombinovane su semantičke i sintaksičke informacije. Sintaksička analiza se vrši kroz proces dubokog parsiranja kako bi se u svakoj rečenici izdvojile fraze. Na kraju se izračunava sličnost između svih pojmova koji imaju istu sintaksičku ulogu pomoću leksičke baze.

Nažalost, za veliki broj jezika, među koje spada i srpski jezik, alati za duboko i plitko parsiranje nisu dostupni što navedene pristupe [3], [4] čini praktično neprimenljivim. Imajući ovo u vidu utvrđeno je da ni jedno od postojećih rešenja se ne može direktno primeniti na problem određivanja STSS (Short Text Semantic Similarity) tako da se može podjednako upotrebiti za engleski, srpski ili neki drugi jezik sa vrlo ograničenim elektronskim lingvističkim resursima. U ovom radu će biti opisan jedan novi pristup rešavanja opisanog problema.

Sve komponente sistema (korisnički interfejs, algoritmi i komunikacija između aplikacije i repozitorijuma za čuvanje semantičkog prostora) su realizovane u programskom jeziku Java. Ontologija je realizovana upotrebom alata *Protégé* [5], dok je za njeno čitanje iz datoteke korišćen alat *Jena* [6]. Za parsiranje web strana korišćen je programski alat *jsoup* [7], koji omogućava da se obrađenim HTML dokumenatima može pristupiti preko standardnih XML interfejsa. Ova komponenta je realizovana kao poseban servis koji periodično vrši obradu samo novog sadržaja.

Na slici 1. prikazan je blok dijagram sa najbitnijim elementima koji se koriste u radu softverskog sistema, gde se mogu uočiti dve faze: *formiranje semantičkog prostora, obrada postova.*



Sl 1. Blok dijagram za fazu kreiranja i čuvanja semantičkog prostora

III. FORMIRANJE SEMANTIČKOG PROSTORA

Ova faza se može predstaviti u sledećim koracima: *formiranje ontologije, pripremanje korpusa za formiranje semantičkog prostora, parsiranje korpusa i čuvanje semantičkog prostora.*

A. Formiranje ontologije

Na početku faze *formiranja ontologije*, definisan je skup tema po kojima se sadržaji postova mogu klasifikovati. Izbor tema je napravljen na osnovu preporuka od stručnih udruženja IEEE i ACM, po kojima je oblast arhitekture i organizacije računara podeljena na osnovna poglavlja i odeljke unutar poglavlja. Svaki odeljak može se poistovetiti sa odgovarajućom temom. Uvedeni su koncepti (termini) koji nedvosmisleno opisuju entitete, njihove međusobne relacije i procese koji se dešavaju unutar računarskog sistema, tako da je moguće svaki koncept smestiti u stablo u kom može biti i/ili roditelj i/ili potomak, oceniti ga pomoću definisanog algoritma i uspostaviti relaciju između posmatranog koncepta i postova u kojima je identifikovan.

B. Pripremanja korpusa za formiranje semantičkog prostora

U ovoj fazi identifikovan je skup reprezentativnih tekstova (RT - Reprezentative Text, blok na slici 1) koji obuhvata tekstove iz knjiga i skripti koji se koriste na predavanjima i vežbama. Kako se ove tekstovi nalaze u različitim tipovima datoteka (pdf, doc), bilo je neophodno njihovo izdvajanje u tekstualne datoteke, koje se koriste kao ulazni parametar za obradu. Uspostavljene su relacije između koncepata na prva dva hijerarhijska nivoa (poglavlja i odeljci) i reprezentativnih tekstova na taj način što za svaki koncept na odgovarajućem hijerarhijskom nivou postoji jedan ili više reprezentativnih tekstova. Za koncept na hijerarhijskoj dubini većoj od dva formirana je lista ključnih reči (tagova), a potom i globalna lista ključnih reči (GTL - Global Tag List, blok na slici 1). Ova lista je rezultat spajanja svih lista ključnih reči i sadrži sve različite reči koje su u njihovom sastavu.

C. Parsiranje korpusa

Parsiranje korpusa (*Corpus parsing*) je neophodno kako bi se uklonile sve suvišne informacije i izdvojio tekst od interesa. Ova faza se izvršava nad korpusom reprezentativnih tekstova i globalnom listom tagova u fazi formiranja semantičkog prostora, odnosno nad korpusom postova u fazi obrade postova i obuhvata sledeće korake:

- *Čišćenje teksta*, odnosno uklanjanje znakova koji spadaju u druga pisma, reči koje sadrže brojeve, uklanjanje datuma, interpunkcije, kao i izjednačavanje malih i velikih slova.
- *Uklanjanje stop reči*, koje je neophodno kako bi se uklonile reči koje imaju zanemarljiv semantički sadržaj, kao što su predlozi, zamenice i veznici, ali se zbog njihove jezičke funkcije često pojavljuju u tekstu. Uklanjanjem ovih reči smanjuje se semantički prostor i povećava tačnost semantičkih algoritama, jer veze između semantički važnih reči postaju više naglašene.
- *Stemovanje (stemming)*, odnosno proces uklanjanja završetka reči predstavlja transformaciju u kojoj može

doći do uklanjanja sufiksa reči, pri čemu se ne gubi osnovni semantički sadržaj. Ovaj postupak se može shvatiti i kao proces normalizacije u kojem se nekoliko oblika reči preslikava u isti oblik pa se na taj način smanjuje broj različitih reči. U radu [8], predložen je opšti sufiksni metod za konstruisanje stemera za jezike sa bogatom fleksijom i oskudnim resursima. Za ovaj pristup eksperimentalno je utvrđena tačnost od 81,83% za srpski jezik. U korpusu analiziranih tekstova delimično je korišćeno latinično pismo sa kodiranjima u ASCII i UTF-8 formatu, odnosno ćirilično pismo sa kodiranjem u UTF-16 formatu. Korišćeni stemer za srpski jezik prihvata kao ulaz reči napisane u *dual1* kodiranju, gde se svaki dijakritik koduje kombinacijom dva nedijakritička slova, pa je bilo potrebno izvršiti konverziju koja prevodi tekst sa latinice i ćirilice u ovo kodiranje, a zatim izvršava stemovanje.

D. *Čuvanje semantičkog prostora*

Čuvanje semantičkog prostora na disku je neophodno u cilju izbegavanja njegovog ponovnog kreiranja pri svakom pokretanju obrade. Ono se najčešće realizuje na dva načina: korišćenjem baze podataka ili datoteke koja sadrži serijalizovane objekte. Za čuvanje semantičkog prostora izabrana je baza podataka, dok se ontologija čuva u xml-serijalizovanoj datoteci.

IV. OBRADA POSTOVA

Na slici 2 je prikazan glavni deo koda za obradu postova. U ovoj fazi utvrđuje se semantička sličnost između pročitanih reči u obrađenim postovima i reči iz globalne liste tagova kako bi se formirala ontološka komatrica. Ova struktura je potrebna u fazi ocenjivanja i utvrđivanja relacija između koncepata i obrađenih postova i za njeno izračunavanje korišćene su dve pomoćne strukture [9]: matrica leksičke i matrica semantičke sličnosti.

```
// pročitaj  korpus obrađenih postova u odgovarajući niz
Text[] preprocessedTextArray = readPreprocessedTextArray();
// pročitaj ontološko stablo
ConceptTree conceptTree = readConceptTree();
// iz ontološkog stabla pročitaj obrađenu globalnu listu tagova
GlobalTagList globalTagList = conceptTree.getGlobalTagList();

// obrada svakog pojedinačnog posta
for (Text text : preprocessedTextArray)
{
   // obrada svake pročitane reči u obrađenom postu
   for (String[] tokenArray=text.getTokenArray())
   {
      for (Tag tag : globalTagList)
      {
         // računanje ontološke komatrice
         Matrix m = calc_ontology_comatrix(tokenArray, globalTagList);
         // ocenjivanje koncepata
         score_the_concepts(m, conceptTree);
      }
   }
}
```

Sl 2. Glavni deo koda za obradu postova

A. *Izračunavanje ontološke komatrice*

Matrica leksičke sličnosti je predstavljena matricom M_1 dimenzija $m \times n$, gde je m broj pročitanih reči iz obrađenog posta, n broj pročitanih reči iz globalne liste sinonima za koncepte, $\alpha_{ij} \in (0,1); i = 1, \dots, m; j = 1, \dots, n$. Svaka vrednost α_{ij} predstavlja sličnost na nivou niza karaktera između reči-kolone i reči-reda. Redovi matrice se koriste za pročitane reči iz obrađenog posta. Kolone matrice se koriste za reči iz globalne liste obrađenih sinonima za koncepte. Leksička sličnost između dve reči se računa pomoću algoritma najduže podsekvence. Vrednost nula označava potpuno različite sadržaje dva niza karaktera, dok jedan ukazuje na identične.

$$M_1 = \begin{bmatrix} \alpha_{11} & \cdots & \alpha_{1n} \\ \vdots & \ddots & \vdots \\ \alpha_{m1} & \cdots & \alpha_{mn} \end{bmatrix} \quad (1)$$

Matrica semantičke sličnosti je predstavljena matricom M_2 dimenzija $m \times n$, gde je m broj pročitanih reči iz obrađenog posta, n broj pročitanih reči iz globalne liste obrađenih sinonima za koncepte, $\beta_{ij} \in [0,1]; i = 1, \dots, m; j = 1, \dots, n$. Svaka vrednost β_{ij} u matrici predstavlja semantičku sličnost između reči-kolone i reči-reda. Redovi matrice se koriste za pročitane reči iz obrađenog posta. Kolone matrice se koriste za reči iz globalne liste obrađenih sinonima za koncepte. Vrednost nula označava potpuno različite semantičke sadržaje dve reči, dok vrednost jedan ukazuje na identične. Semantička sličnost između dve reči određuje se izračunavanjem kosinusne sličnosti njihovih kontekstnih vektora dobijenih iz baze podataka.

$$M_2 = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1n} \\ \vdots & \ddots & \vdots \\ \beta_{m1} & \cdots & \beta_{mn} \end{bmatrix} \quad (2)$$

U prvoj fazi formiranja ontološke komatrice uklanjaju se svi redovi, odnosno kolone u kojima je vrednost β jednaka nuli i dobija se matrica M_3 dimenzija $m \times n$, gde su m broj redova, odnosno kolona iz matrice preostalih nakon sažimanja matrice M_2. U svakoj ćeliji matrice nalaze se uređeni skupovi $C_{ij} = \left\{\{(c_1, w_1), \dots, (c_k, w_k)\}, TF - IDF_i, S_{ij}\right\}$, gde je $i = 1, \dots, m; j = 1, \dots, n; k \in (0, < broj\ koncepata >)$. U svakom uređenom skupu C_{ij}, c_k je koncept u čijoj listi obrađenih sinonimima se nalazi reč iz kolone j, w_j težinski koeficijent koji predstavlja stepen pripadnosti reči u obrađenom sinonimu i određuje se kao recipročna vrednost broja reči u obrađenom sinonimu (na primer, ako je pročitana reč iz posta *kontroler*, a sinonim *DMA kontroler*, težinski koeficijent iznosi 0.5), $TF - IDF_k$ je proizvod frekvencija pročitane reči u postu i inverzne dokument frekvencije, čija vrednost raste srazmerno broju pojavljivanja reči u dokumentu, ali je kompenzovana frekvencijom pojavljivanja reči u čitavom korpusu, S_{ij} ukupna semantička sličnosti između reči u redu i i koloni j izračunata kao $S_{ij} = 0.45\alpha_{ij} + 0.55\beta_{ij}$, pri čemu su vrednosti koeficijenata utvrđene empirijskim putem.

$$M_3 = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{m1} & \cdots & C_{mn} \end{bmatrix} \quad (3)$$

U procesu sažimanja matrice M_3 uklanjaju se svi redovi, odnosno kolone u kojima su vrednosti C_{ij} prazni skupovi i

dobija se matrica M_4 dimenzija $m \times n$, gde su m i n broj redova i kolona iz matrice M_3 preostalih nakon sažimanja.

B. *Ocenjivanje koncepata*

Ocena koncepta predstavlja meru zastupljenosti određene oblasti u dokumentu gde je koncept pronađen. Posmatrajući međusobne relacije između koncepata, njihove različite hijerarhijske nivoe, kao i to da se koncepti na nižim nivoima mogu predstaviti pomoću liste ključnih reči, izdvojeni su sledeći parametri, koji se koriste u fazi ocenjivanja:

- *Koncentracija* predstavlja meru prisutnosti koncepta i njegovih potomaka u semantičkom podstablu. Težinski koeficijent za koncept određuje se na osnovu broja reči u tagu za taj koncept po sledećem kriterijumu:

$$w_{c_i} = \begin{cases} 0, ncount(c_i) = 0 \\ 0.15, ncount(c_i) > 0 \wedge ntokens(c_i) = 1 \\ 0.2, ncount(c_i) > 0 \wedge ntokens(c_i) = 2 \\ 0.3, ncount(c_i) > 0 \wedge ntokens(c_i) \geq 3 \end{cases} \quad (4)$$

, gde je:

- $ncount(c_i)$ - broj pojavljivanja sinonima u tekstu,
- $ntokens(c_i)$ - broj reči u pronađenom sinonimu.

Koncentracija za određeni koncept se određuje na osnovu njegovog težinskog koeficijenta i koncentracija za sve direktne podkoncepte po sledećem obrascu:

$$score(c_i) = w_{c_i} + \frac{\sum_{j=1}^{child(c_i)} score(c_j)}{child(c_i)} \quad (5)$$

, gde je:

- w_{c_i} - težinski koeficijent za koncept c_i,
- $child(c_i)$ - broj direktnih potomaka koncepta c_i.

- *Relevantnost koncepta u semantičkom podstablu* predstavlja meru zastupljenosti svih njegovih potomaka u dokumentu i određuje se po sledećem obrascu:

$$cr(c_i) = \frac{nsc(c_i)}{nsub(c_i)} \quad (6)$$

, gde je:

- $nsc(c_i)$ - broj podkoncepata za koncept ci računajući i koncept ci, sa koncentracijom većom od definisanog praga,
- $nsub(c_i)$ - ukupan broj podkoncepata za koncept c_i računajući i koncept c_i.

- *Relevantnost koncepta u dokumentu* određuje se na osnovu utvrđenih relacija u ontološkoj komatrici između odgovarajućeg koncepta i pročitanih reči u obrađenom postu računanjem kvadratne sredine između relevantnosti koncepta u semantičkom podstablu i $TF - IDF$ za pročitanu reč.

$$dr(c_i) = \sqrt{\frac{cr^2(c_i) + \sum_{i=0}^{n} (TF-IDF)_i^2}{n+1}} \quad (7)$$

, gde je:

- n - broj pročitanih reči u postu za koje postoji relacija sa odgovarajućim konceptom u ontološkoj komatrici.

Ukupna ocena za koncept formirana je na osnovu prethodno opisanih parametara i računa se po sledećem obrascu:

$$totalscore_{c_i} = w_{sc} \cdot score(c_i) + w_{cr} \cdot cr(c_i) + w_{dr} \cdot dr(c_i) \quad (8)$$

, gde su:

- w_{sc}, w_{cr}, w_{dr} - koeficijenti srazmere za koncentraciju, relevantnost koncepta i relevantnost dokumenta sa definisanim konstantama: 0.2, 0.3 i 0.5 respektivno.

V. ZAKLJUČAK

U radu je opisan softverski sistem koji analizira sadržaje postova studentskog foruma i pomaže predavaču sa informacijama o najzastupljenijim temama i konceptima. Sistem je primenjen na oblast arhitekture i organizacije računara korišćenjem tehnika semantičke sličnosti i ontologije baziranoj na ovoj oblasti. Verifikacija realizovanog sistema je izvršena na postovima studentskog foruma Elektrotehničkog fakulteta Univerziteta u Beogradu. Dobijene informacije predavači mogu iskoristiti u svrhu promene i poboljšavanja svojih predavanja. Sistem trenutno predstavlja jednu funkcionalnu celinu, ali može biti deo složenijeg tutorskog softvera, jer je realizovan na modularan i proširiv način.

LITERATURA

[1] R. Mihalcea, C. Corley, and C. Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity," in *National Conference on Artificial Intelligence*, 2006, vol. 21, no. 1, pp. 775–780

[2] A. Islam and D. Inkpen, "Semantic text similarity using corpus-based word similarity and string similarity," *ACM Trans. Knowl. Discov. Data*, vol. 2, no. 2, pp. 1–25, Jul. 2008.

[3] J. Oliva, J. I. Serrano, M. D. del Castillo, and Á. Iglesias, "SyMSS: A syntax-based measure for short-text semantic similarity," *Data Knowl. Eng.*, vol. 70, no. 4, pp. 390–405, Apr. 2011.

[4] L. Li, Y. Zhou, B. Yuan, J. Wang, and X. Hu, "Sentence similarity measurement based on shallow parsing," in *Fuzzy Systems and Knowledge Discovery*, 2009, pp. 487–491.

[5] Protégé, Ontology creation tool, http://protege.stanford.edu

[6] Jena Semantic Web Framework, http://jena.sourceforge.net

[7] jsoup, java HTML parser, http://jsoup.org

[8] V. Kešelj and D. Šipka, "A suffix subsumption-based approach to building stemmers and lemmatizers for highly inflectional languages with sparse resource," *INFOTHECA–Journal Informatics Librariansh.*, vol. IX, no. 1–2, p. 24a–33a, 2008.

[9] B. Furlan, V. Batanović, B. Nikolić, Semantic Similarity of Short Texts in Languages with a Deficient Natural Language Processing Support, DECISION SUPPORT SYSTEMS, Vol. 55, No. 3, pp. 710-719, Jun, 2013

ABSTRACT

This paper describes new approach for text analyzing. The approach is based on applying domain ontology and semantic similarity. Analysis results are applied on the computer architecture and organization student forum on Faculty for Electrical Engineering, University of Belgrade. Teachers were singled out only those posts whose content belongs to the purpose of the forum and are related with the topics from the course.

TEXT CONTENT ANALYSIS USING ONTOLOGY AND SEMANTIC SIMILARITY

D. Prodanović, B. Furlan, B. Nikolić