UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Matjaž Vončina

Predstavitev spletnih novic iz več virov

DIPLOMSKO DELO
NA VISOKOŠOLSKEM STROKOVNEM ŠTUDIJU

Mentor: prof. dr. Marko Robnik Šikonja

Ljubljana, 2012

Univerza
v Ljubljani

Fakulteta za računalništvo
in informatiko

Tržaška 25
1000 Ljubljana, Slovenija
telefon: 01 476 84 11
faks: 01 426 46 47
www.fri.uni-lj.si
e-mail: dekanat@fri.uni-lj.si

Št. naloge: 00551/2011
Datum: 03.11.2011

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **MATJAŽ VONČINA**

Naslov: **PREDSTAVITEV SPLETNIH NOVIC IZ VEČ VIROV**
A PRESENTATION OF WEB NEWS FROM MULTIPLE SOURCES

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija

Tematika naloge:

Spletne novice postajajo pomemben vir obveščanja vse večjega dela populacije. Ker se novice iz različnih virov podvajajo, viri pa se razlikujejo po zanesljivosti, poglobljenosti in hitrosti poročanja, so uporabniki postavljeni pred dilemo, katere vire naj spremljajo.
Izdelajte prototip spletnega programa, ki na enem mestu omogoča spremljanje novic iz več virov, pri tem pa novice razvršča v kategorije in jih združuje z upoštevanjem podobnosti in aktualnosti. Analizirajte že obstoječe podobne rešitve in besedila analizirajte z metodami in orodji za obdelavo naravnega jezika. Simulirajte realno dogajanje in rešitev praktično ovrednotite.

Mentor:
prof. dr. Marko Robnik Šikonja

Dekan:
prof. dr. Nikolaj Zimic

Fakulteta za računalništvo in informatiko • UNIVERZA V LJUBLJANI • 2

IZJAVA O AVTORSTVU

diplomskega dela

Spodaj podpisani Matjaž Vončina,

z vpisno številko 63060527,

sem avtor diplomskega dela z naslovom:

Predstavitev spletnih novic iz več virov

S svojim podpisom zagotavljam,da:

- sem diplomsko delo izdelal samostojno pod mentorstvom prof. dr. Marka Robnik Šikonje
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki »Dela FRI«.

V Ljubljani, dne 15. 2. 2012 Podpis avtorja:

ZAHVALA

Zahvalil bi se svojemu mentorju, prof. dr. Marku Robnik Šikonji, za izpopolnitev ideje, strokovno pomoč, nasvete in ves vloženi čas.

Zahvalil bi se tudi Marini za spodbudo pri študiju in pisanju diplomske naloge.

KAZALO

POVZETEK

Izdelali smo spletno mesto, kjer obiskovalci na enostaven in pregleden način spremljajo aktualne novice o dogajanju po Sloveniji iz več medijev. Združili smo podobne novice v eno skupino, s čimer smo obiskovalcem skrajšali čas iskanja pomembnih novic in jim prihranili obiskovanje več spletišč. Izdelali smo bazo novic in oblikovali proces njihove obdelave. Razvili smo zajemanje novic iz več virov, normalizacijo novic s pomočjo lematizacije, uteževanje besed v novici in predstavitev novice s pomočjo vektorskega prostora. Ta model je bil podlaga za izračun podobnosti novic, kar nam je omogočilo razvoj algoritma za oblikovanje skupin podobnih novic. Izdelali smo prototip spletne strani, kjer smo na podlagi aktualnosti prikazali skupine novic.

KLJUČNE BESEDE:
podobnost besedil, kategorizacija, lematizacija, kosinusna podobnost besedil, novice

ABSTRACT

We built a website, where visitors can find and read current news from Slovenia from multiple sources. We presented news articles in groups of similar news to shorten the time to find important news and to spare visitors browsing of several websites. To achieve this we built a database of news and news processor. We developed a system to read and parse news from multiple sources, news normalization with lemmatization, weighting of words in the news and presenting the news using a vector space model. We used our model to calculate similarity between news, which enabled us to clusters similar news. We built a prototype website to display relevant news clusters.

KEYWORDS:
text similarity, categorization, lemmatization, cosine coefficient, news

1. UVOD

Z razvojem spleta so postale različne vsebine dostopne širši populaciji. S tem smo iskalci informacij spremenili svoje navade in se vse pogosteje obračamo na vsebine objavljene na spletu. Takšen preskok vodi v generiranje še več spletnih informacij in še večje število uporabnikov spleta. Danes je pomemben faktor tudi aktualnost informacij na spletu.

Spremembam so se prilagodili tudi tradicionalni mediji (televizija, radio, časopisi, revije …) in danes imajo prav vsi svoje spletne strani, na katerih objavljajo sveže novice. Pridružili so se jim tudi mediji, ki pišejo novice izključno za objavo na spletu. Spletni mediji imajo veliko prednost pred ostalimi mediji, ker lahko novico hitro objavijo in ta postane bralcem takoj dostopna in jim je na voljo za nedoločen čas. Na televiziji so informativne oddaje vezane na točno določen čas dneva in zahtevajo veliko dela in stroškov za snemanje, gledalci pa morajo biti fizično prisotni pred televizijskim aparatom v pravem trenutku. Tiskani mediji so vezani na dnevno, tedensko ali mesečno izdajo, s čimer niso časovno konkurenčni. Zahtevajo tudi veliko grafičnega oblikovanja in visoke stroške tiskanja. Težnja vseh medijev je po čimprejšnji in ekskluzivni objavi novic. Ta dva faktorja sta ključna pri razlikovanju med mediji, saj privabita največ uporabnikov. Uporabniki ne spremljamo medijev, ki nimajo kvalitetnih in pravilnih informacij. Enako velja za novice na televiziji, radiu, časopisih, in revijah. V nadaljevanju se osredotočamo izključno na novičarske spletne strani.

Tudi uporabniki spleta smo postavljeni pred dilemo, kateri medij naj spremljamo, da bomo najbolje obveščeni in to v čim krajšem času. Zaradi ekskluzivnosti nekaterih novic lahko s spremljanjem zgolj enega medija zgrešimo pomembne informacije, spremljanje večjega števila medijev pa je lahko časovno zelo potratno. Časovna zahtevnost nastane v veliki meri zaradi podvajanja novic pri različnih medijih. Splošne novice, ki niso ekskluzivne za posamezen medij, objavljajo vsi mediji. Pri pisanju novic se mediji v Sloveniji v veliki meri naslanjajo na poročanje Slovenske tiskovne agencije. V nekaterih primerih te novice enostavno kopirajo, v drugih primerih pa jih prepišejo, s čimer se spremeni slog pisanja, uporabljajo se sopomenke, dodajo se pojasnila in podobno, a sama vsebina se za bralca bistveno ne spremeni.

Za bralca novic na spletu bi bil idealen medij tak, ki bi na enem mestu vseboval vse splošne novice in ekskluzivne novice vseh ostalih medijev in to v čim krajšem časovnem obdobju po dogodku oziroma od nastanka informacije. Diplomska naloga poskuša doseči ta cilj.

Namen naloge je spletnim uporabnikom na enostaven in pregleden način predstaviti dogajanje po Sloveniji. Za cilj smo si postavili izdelavo algoritma, ki združuje novice z isto tematiko, ter

postavitev spletnega portala, kjer se s kronološkim pregledom prikazujejo skupine novic z isto tematiko iz več slovenskih medijev. Rezultat diplomske naloge je spletna stran, kjer obiskovalec dobi hiter pregled nad dogajanjem po Sloveniji brez prebiranja več slovenskih medijev.

Nalogo smo razdelili na več poglavij. V drugem poglavju smo predstavili nekaj obstoječih sorodnih sistemov ter njihove glavne značilnosti. V tretjem poglavju smo zapisali teoretično osnovo za reševanje našega problema. Spoznali smo dva pristopa k normaliziciji besedila, besedilo smo transformirali iz niza znakov v model, ki je primeren za učni algoritem, na osnovi katerega smo izračunavali podobnost besedil. Predstavili smo tudi algoritem za kategorizacijo besedil – združevanje podobnih besedil v eno skupino. V četrtem poglavju smo na teoretični podlagi opisali kategorizacijo novic iz več virov. Začeli smo z zajemom novice iz virov RSS, normalizacijo in zapisom v našo bazo. Vsako novico smo predstavili v vektorskem prostoru ter jo primerjali z ostalimi novicami. Z izračunanimi podobnostmi med novicami smo določili skupine novic z isto tematiko ter jih predstavili na spletni strani. Dotaknili smo se tudi nekaj problemov, na katere smo naleteli pri izdelavi praktičnega dela diplomske naloge. Na koncu smo zapisali še nekaj sklepnih ugotovitev, kaj smo z našim delom dosegli ter nakazali možnosti izboljšav in nadgradenj naše predstavitve novic iz več virov.

2. SORODNI SISTEMI

Na spletu je mogoče najti več podobnih sistemov. Med bolj znane sodita storitvi Google News na svetovni ravni in Najdi.si novice na slovenskem področju.

Najdi.si novice so največji slovenski agregator novic, saj zajemajo preko 100 slovenskih virov. Zajemanje poteka preko javnih ali za Najdi.si prilagojenih virov RSS. Nekaj izbranih vidnejših slovenskih medijev tudi uredniško izpostavlja po eno aktualno novico, ki se kaže na vstopni strani Najdi.si novic. Najbolj izpostavljen del spletne strani je tako postavljen uredniško, vse ostale vsebine so razvrščene avtomatsko. Med vročimi zgodbami so izpostavljene skupine novic z največ prispevki z različnih medijev. Pomembna utež pri razvrščanju je čas, tako da se na vrhu izpisuje skupina z relativno velikim številom novic in aktualno vsebino.

Najdi.si novice ponujajo branje novic v več rubrikah – Slovenija, Svet, Gospodarstvo, Lepota in zdravje ter druge. Med vročimi zgodbami se pojavljajo zgolj novice iz rubrik Slovenija, Svet, Gospodarstvo in Šport. Pri razvrščanju novic znotraj ene skupine je osnova kronološko zaporedje objavljenih novic. Razvit je tudi sistem, ki onemogoča manipuliranje medijev za izboljšanje pozicije znotraj skupine na nepošten način in preprečuje prikaz vsebinsko slabših vsebin na prvih pozicijah [9].

Posamezne kategorije novic so obogatene z informacijami o zadnje objavljenih in najbolj branih novicah. Ob straneh se pojavljajo vremenska napoved, prometne informacije, anketa, menjalni tečaji, tečajnica vrednostnih papirjev ter podobne informacije. Omogočeno je tudi izločanje novic, ki so za večji obseg vsebine na strani medija plačljive. Tako obiskovalec ne bo kliknil na novico in na koncu razočaran ugotovil, da je vsebina plačljiva. Izločanje je mogoče tudi po viru in po času objave.

Najdi.si novice so postavljene s pomočjo odprtokodnega projekta Apache Lucene [9]. To je knjižnica za naprednejše iskanje po besedilih na različnih platformah napisana v programskem jeziku Java.

Google News je agregator novic z vsega sveta, ki trenutno zajema preko 50 tisoč različnih virov. Poleg globalne različice obstaja še preko 70 regionalnih različic za različne države in jezike sveta, vendar med njimi ni Slovenije [7]. Kdor želi svojo spletno stran vključiti, se mora prijaviti in dobiti potrditev njihove ekipe. Ena od večjih razlik od Najdi.si novic je, da ne zajema virov RSS. Iskalni pajek preišče vsebine spletnih strani, jih indeksira in razvrsti v skupine. Ne zanašajo se izključno na vire RSS, temveč poskrbijo za to, da sami poiščejo prave

vsebine. Za lažje iskanje vsebin priporočajo, da mediji pripravijo prilagojen zemljevid strani, kjer se na točno določen strukturiran način opiše posamezne novice.

Google ponuja nekaj zanimivih značilnosti. Pri skupini novic, ki govori o eni temi, so prikazani tudi video prispevki na spletni strani YouTube.com na kanalih medijskih hiš. S tem so obogatili posredovanje novic. Druga značilnost je izpostavljenost nekaj zanimivih ključnih besed, o katerih mediji v zadnjem času veliko poročajo. Na tem seznamu se večinoma pojavljajo imena oseb in podjetij ter aktualni dogodki. Na vstopni strani so privzeto prikazane novice iz različnih kategorij. Ta prikaz lahko obiskovalec prilagodi lastnemu zanimanju. S pomočjo drsnikov izbere količino novic iz posamezne kategorije, ki jih želi spremljati, ali vpiše ključno besedo. Čeprav novice niso lokalizirane za Slovenijo, obstaja možnost spremljanja novic v bližini obiskovalca. Te novice so v angleškem jeziku in niso razdeljene po kategorijah. Opazno je pomanjkanje virov, saj so vključene le novice Slovenske tiskovne agencije in tujih tiskovnih agencij, zato so informacije zelo okrnjene. Na Google News so, podobno kot na Najdi.si novicah, uredniško izbrani prispevki (t. i. »Editor's Pick«), vendar niso tako izpostavljeni.

Pri razvrščanju prispevkov znotraj skupine uporabljajo več pristopov. Zavedajo se, da so zadnje objavljene novice bolj poglobljene in ponujajo več informacij, a so pri tem vseeno pazljivi, saj nekateri mediji niso tako ažurni in poskušajo le slediti drugim z objavljanjem zgolj osnovnih informacij. Upoštevajo tudi razlikovanje med mediji po strokovnosti. Če zgodba govori o športnem dogodku, so specializirani športni mediji verjetno boljši in bolj poglobljeni kot splošni mediji. Podoben princip upoštevajo tudi pri lokalnih novicah, kjer lokalni mediji bolje poznajo okolje, novinarji poznajo več oseb in podobno. S tem so prispevki kvalitetnejši v primerjavi s splošnimi mediji ali tiskovnimi agencijami. Če opazijo, da uporabniki preskočijo prvo novico in raje kliknejo na tretjo ali četrto, to upoštevajo pri nadaljnjem pozicioniranju novic znotraj skupine [8].

3. TEORETIČNI DEL

Besedila so lahko med sabo identična, skoraj enaka s šumom, le deloma enaka, podobna po vsebini ali pa se povsem razlikujejo. Iskanje podobnosti med besedili oziroma prekrivanje vsebine v različnih besedilih je naloga, ki zahteva sistematičen pristop [12].

Besedilo lahko analiziramo na različnih predstavitvenih nivojih. Lahko ga analiziramo kot niz bitov ali kot zaporedje besed, stavkov in odstavkov. Analiza je lahko statistična, lingvistična ali kombinacija obojega. Za učinkovito procesiranje besedil lahko uporabimo različne podatkovne strukture ali modele predstavitve besedila. Kot primere predstavitev lahko naštejemo vektorski prostor, predstavitev besedila kot graf, drevo pripon … Za vsak scenarij je potrebno identificirati primeren model predstavitve besedila in ustrezne tehnike analize.

Predstavitev podobnosti besedila je odvisna od našega cilja. Najbolj enostaven in verjetno najpogostejši cilj je odkrivanje identičnih besedil, kjer se podobnost izraža z dolžino najdaljšega skupnega dela besedila. Za ta primer obstajajo učinkoviti algoritmi, vendar odpovejo v primeru skoraj identičnih besedil s šumom. V takih primerih se podobnost lahko izrazi z najdaljšim skupnim delom besedila pri dani stopnja šuma. Druga mera podobnosti je prekrivanje vzorčnega besedila in zbirke besedil, kjer se prav tako lahko upošteva šum. Ta mera pomaga odkrivati besedila združena iz več virov. Tretja mera temelji na leksikalnem pristopu in ugotavlja, ali dve besedili uporabljata enak stil pisanja in enake izraze. Ta metoda ne pomaga pri odkrivanju podobnosti besedil, odkriva pa besedila, ki jih je napisal isti avtor. Za nas pomembna mera podobnosti odkriva besedila, ki govorijo o isti temi. Takšna mera nam pomaga pri kategorizaciji besedil v skupine (angl. clustering). Kot najbolj primeren pristop za reševanje našega problema se je pokazala kategorizacija besedil s pomočjo vektorskega prostora, ki ga bomo v nadaljevanju podrobneje predstavili.

Celoten pristop z vektorskim prostorom lahko razdelimo v tri dele. Najprej normaliziramo besedilo, ga predstavimo z vektorjem in nato izračunamo podobnost dveh besedil [13].

3.1. Normalizacija besedila

Za boljše strojno primerjanje besedila, ga je pred obdelavo potrebno normalizirati. Za to sta pomembna postopka krnjenje in lematizacija. Ta dva postopka nista univerzalna, zato morata biti prilagojena vsakemu posameznemu jeziku. Pri krnjenju se v določenih pogojih posamezni besedi odstrani končnica. Kot rezultat krnjenja besede *egiptovski* dobimo koren *egipt*. Pri krnjenju lahko pride do neželenih napak, ki so odvisne tudi od kvalitete krnilnika. Tako lahko v primeru krnjenja glagola *delati* in samostalnika *delo* dobimo isti rezultat v obliki korena *del*.

Za izboljšanje rezultatov se je v takšnih primerih bolje poslužiti lematizacije. Kot rezultat lematizacije vedno dobimo lemo – osnovno obliko besede, ne glede na končnico.

Primeri lematizacije:

- samostalnik *beseda*: besed, beseda, besedah, besedam, besedama, besedami, besede, besedi, besedo;
- glagol *brati*: berem, bereš, bere, bereva, bereta, beremo, berete, berejo, bral, brala, brali;
- pomožni glagol *biti*: sem, si, je, sva, sta, smo, ste, so, nisem, nisi, ni, nisva, nista, nismo, niste, niso, bil, bila, bili, bom, boš, bo, bova, bosta, bomo, boste, bodo, bojo.

Mnenja glede potrebe po krnjenju in lematizaciji so deljena. Nekateri menijo, da postopka bistveno ne prispevata k uspešnosti metode, medtem ko so drugi mnenja, da sta nujno potrebna. Te polemike veljajo predvsem za angleški jezik, v slovenskem jeziku je pregibnih besed več in je zato takšna polemika odveč [3].

Vse besede v besedilu ne predstavljajo vsebine. To so tako imenovane funkcijske besede: vezniki, predlogi, pomožni glagoli in podobno. Takšne besede je potrebno izločiti iz proučevanja. Tako ostanejo samo besede, ki imajo pomen za določeno temo. Izločanje besed lahko sloni na frekvenci besed, kjer se kot funkcijske štejejo besede z visoko frekvenco. Frekvenco dobimo s štetjem ponovitev določene besede v celotni zbirki besedil, ki jih proučujemo. V praksi je težko uporabiti takšno avtomatsko izločanje funkcijskih besed, saj prihaja do napak pri izločanju besed s pomenom. Namesto tega se uporablja seznam besed za odstranjevanje (angl. stopwords). S tem postane metoda izločanja funkcijskih besed odvisna od jezika in jo je potrebno prilagoditi za vsak jezik posebej, če se že v prejšnjem koraku nismo odločili za krnjenje ali lematizacijo besedila.

Pri normalizaciji besedila ne smemo pozabiti na ločila, saj bi jih brez posebnega postopka lahko vključili v množico besed, ki določajo vsebino besedila. Ločila je potrebno izločiti iz nadaljnje obdelave, prav tako pa se je potrebno izogniti razlikovanju med malimi in velikimi črkami.

3.2. Predstavitev besedil v vektorskem prostoru

Drugi korak pri kategorizaciji besedila je transformacija besedila iz niza znakov v model, ki bi bil primeren za učni algoritem. Večina današnjih sistemov za iskanje informacij ne temelji na semantiki besedila. Pomen besedila predstavljajo zgolj besede, ki besedilo sestavljajo. V teh sistemih je pomen povedi *»Vidim, kar jem.«* in *»Jem, kar vidim.«* popolnoma enak. Vrstni red

besed, ki sestavljajo besedilo, nima nobenega vpliva na njihov pomen. Ker ne upoštevajo semantične in sintaksne informacije, so ti pristopi pogosto poimenovani vreča besed (angl. bag-of-words) [4]. Besedilo predstavimo kot vektor, kjer vsaka različna beseda predstavlja dimenzijo vektorskega prostora [6]:

$$d_i = (w_{i1}, w_{i2}, \dots, w_{iT})$$

kjer je w_{ij} ($i = 1, 2, \dots, N$ in $j=1, 2, \dots, T$) utež besede j v besedilu i. T predstavlja celotno število besed, N pa število dokumentov.

Vsako besedilo predstavimo kot linearno kombinacijo T baznih vektorjev. Uteževanje besed v vektorskem prostoru temelji na statistiki posameznih besed. Končna mera podobnosti besedil je odvisna od definicije uteži. Utež definiramo kot [1]:

- $w_{ij} = tf_{ij}$, kjer je tf_{ij} frekvenca besede j v besedilu i oziroma število pojavitev besede j v besedilu i (angl. Term frequency);
- $w_{ij} = tf_{ij}/tf_{i,max}$, kjer je $tf_{i,max}$ največje možna frekvenca besede v besedilu i; utež je enaka 1, če celotno besedilo sestavlja samo ena beseda, ki se pojavlja od 1- do n-krat;
- $w_{ij} = IDF = log(N/df_j)$, kjer je N število vseh besedil v zbirki in df_j število besedil, ki vsebujejo besedo j (angl. Inverse document frequency);
- $w_{ij} = TF\text{-}IDF$ (angl. Term frequency – Inverse document frequency), ki jo na podlagi različnih teoretičnih razlag lahko izračunamo na več načinov. Predstavljamo tri primere izračuna, uporabljajo pa se tudi različne osnove logaritma:
    - $w_{ij} = tf_{ij} * log(N/df_j)$;
    - $w_{ij} = tf_{ij} * log((N-df_j)/df_j)$, kjer se utež še dodatno zmanjša, če se beseda pojavlja v večjem številu besedil v zbirki;
    - $w_{ij} = tf_{ij} * (log(N/df_j) + 1)$, kjer utež ostane bolj odvisna od frekvence besede j v besedilu i.

Skupno vsem naštetim utežem je, da je utež večja, če se beseda v besedilu pojavlja čim večkrat, kar nam pove koeficient TF. Na drugi strani je koeficient IDF, ki upošteva pogostost besede v vseh besedilih v zbirki. IDF je večji, če se beseda pojavlja redko v besedilih v zbirki, manjši pa takrat, ko se beseda pojavlja pogosto. S tem se primerno uteži beseda v besedilu, saj pogoste besede za proučevano besedilo niso pomembne in obratno.

3.3. Mera podobnosti dveh besedil

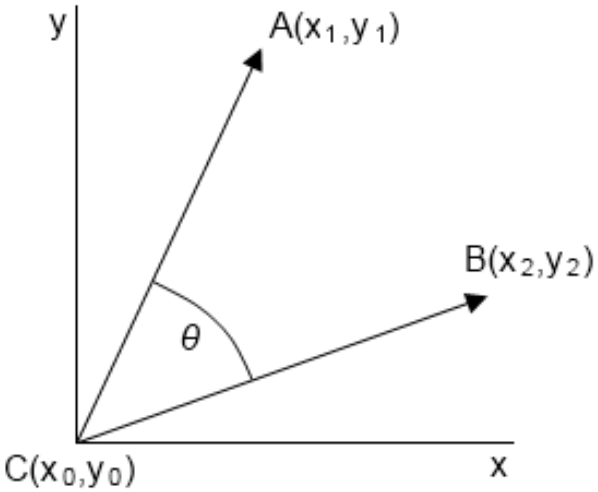
Podobnost besedil v vektorski predstavitvi določimo na podlagi različnih mer, ki temeljijo na produktu vektorjev dveh besedil, kjer prekrivanje besed predstavlja podobnost. Poznamo več

različnih koeficientov. Osnovni je moč presečne množice besed, ki sestavljajo dve primerjani besedili. Vendar ta mera ne upošteva velikosti obeh množic. Kosinusna podobnost, Jaccardov in Diceov koeficient so si podobni v tem, da upoštevajo dolžino obeh besedil in normalizirajo uteži besed ter vrnejo podobnost kot število med 0 in 1 [2, 5].

Pri praktičnem delu naloge smo uporabili kosinusno podobnost, ki jo podrobneje predstavljamo v nadaljevanju.

3.3.1. Kosinusna podobnost besedil

Kosinusni koeficient meri kot med dvema vektorjema. Rezultat je 1, če je kot med vektorjema 0. Rezultat je 0, ko sta vektorja pravokotna, za druge vrednosti kota pa je med 0 in 1. Besedili, ki ju vektorja predstavljata, sta si podobni toliko, kot je kosinus kota med tema dvema vektorjema [1]. Kot med vektorjema smo predstavili na sliki 1.



Slika 1. Kot med vektorjema A in B [1].

Enačba za izračun kosinusnega koeficienta podobnosti besedil je:

$$cos(d_i, d_j) = \frac{d_i \cdot d_j}{\|d_i\|_2 \times \|d_j\|_2} = \frac{\sum_{l=1}^{T} w_{il} \times w_{jl}}{\sqrt{\sum_{l=1}^{T} {w_{il}}^2} \times \sqrt{\sum_{l=1}^{T} {w_{jl}}^2}}$$

kjer d_i predstavlja prvo besedilo, d_j drugo besedilo in T predstavlja celotno število besed.

Ta izračun lahko vidimo kot normalizacijo dolžine vektorjev v postopku izračuna podobnosti besedil.

3.4. Kategorizacija besedil

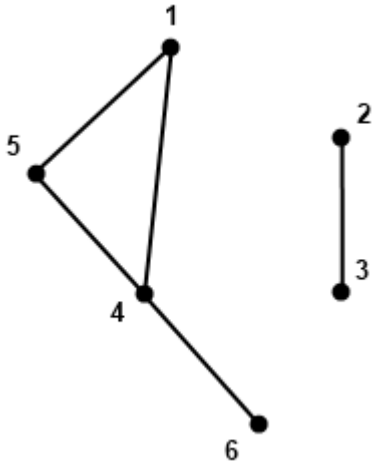
Ko je izračunana podobnost med dvema besediloma, določimo kategorijo, v katero besedilo sodi. Paroma primerjamo vsa besedila v zbirki. To najlažje prikažemo na primeru, kjer so mere podobnosti podane v obliki matrike. Na levi strani tabele 1 je matrika mer podobnosti med šestimi besedili. Na desni strani je predstavljena matrika, kjer so z 1 označeni tisti pari besedil, ki presegajo prag, kjer je mera podobnosti večja od 0,75. Tako velika mera podobnosti označuje tiste pare besedil, ki so si med sabo dovolj podobni [5].

1						
2	0,6					
3	0,6	0,8				
4	0,9	0,7	0,7			
5	0,9	0,6	0,6	0,9		
6	0,5	0,5	0,5	0,9	0,5	
	1	2	3	4	5	6

1						
2	0					
3	0	1				
4	1	0	0			
5	1	0	0	1		
6	0	0	0	1	0	
	1	2	3	4	5	6

Tabela 1. Na levi strani: matrika mer podobnosti med šestimi besedili [5]. Na desni strani: matrika po uporabi praga za določanje dovolj velike podobnosti.

Na sliki 2 so vozlišča posamezna besedila, povezave pa obstajajo med tistimi vozlišči, kjer mera podobnosti presega prag 0,75.



Slika 2. Graf, ki predstavlja matriko mer podobnosti šestih besedil iz tabele št. 1 [5].

Iz grafa je razvidno, da sta nastali dve kategoriji besedil, ki so si podobna. V prvo kategorijo sodijo besedila 1, 4, 5 in 6, v drugo kategorijo pa sodita besedili 2 in 3.

Če bi prag, ki določa dovolj visoko mero podobnosti povišali, bi se število povezav zmanjšalo in nastalo bi več kategorij besedil. Če bi bil prag višji od 0,9, ki je v danem primeru najvišja izračunana mera podobnosti, potem si noben par besedil ne bi bil dovolj podoben in vsaki kategoriji bi pripadalo natančno eno besedilo. Imeli bi 6 različnih kategorij. Če bi prag zmanjšali pod 0,7, bi na grafu izrisali še povezavi med vozliščema 2 in 4 ter 3 in 4 in imeli samo 1 kategorijo. Prag določamo empirično na podlagi poizkušanja in preverjanja, ko se na podlagi vzorca dobi najboljša možna kategorizacija.

4. PRAKTIČNI PRIMER

Za postavitev novičarskega portala, ki bi vseboval novice iz več spletnih medijev, smo kot vir podatkov vzeli vire RSS nekaj večjih slovenskih spletnih medijev. Vsako posamezno novico smo primerjali z ostalimi novicami. Tako smo odkrili podobnost med novicami iz različnih medijev, ki opisujejo isto temo oziroma isti dogodek. Na ta način takšno novico izpišemo samo enkrat in bralcu prihranimo čas, ki bi ga moral nameniti brskanju po velikemu številu novic. Hkrati mu nudimo poglobljeno branje o eni temi v več medijih in mu predstavimo sorodne vsebine, ki bi ga utegnile zanimati. V slovenskem prostoru že obstajata dva portala, ki nudita podobno agregacijo več slovenskih novičarskih medijev. To sta Times.si in Najdi.si novice.

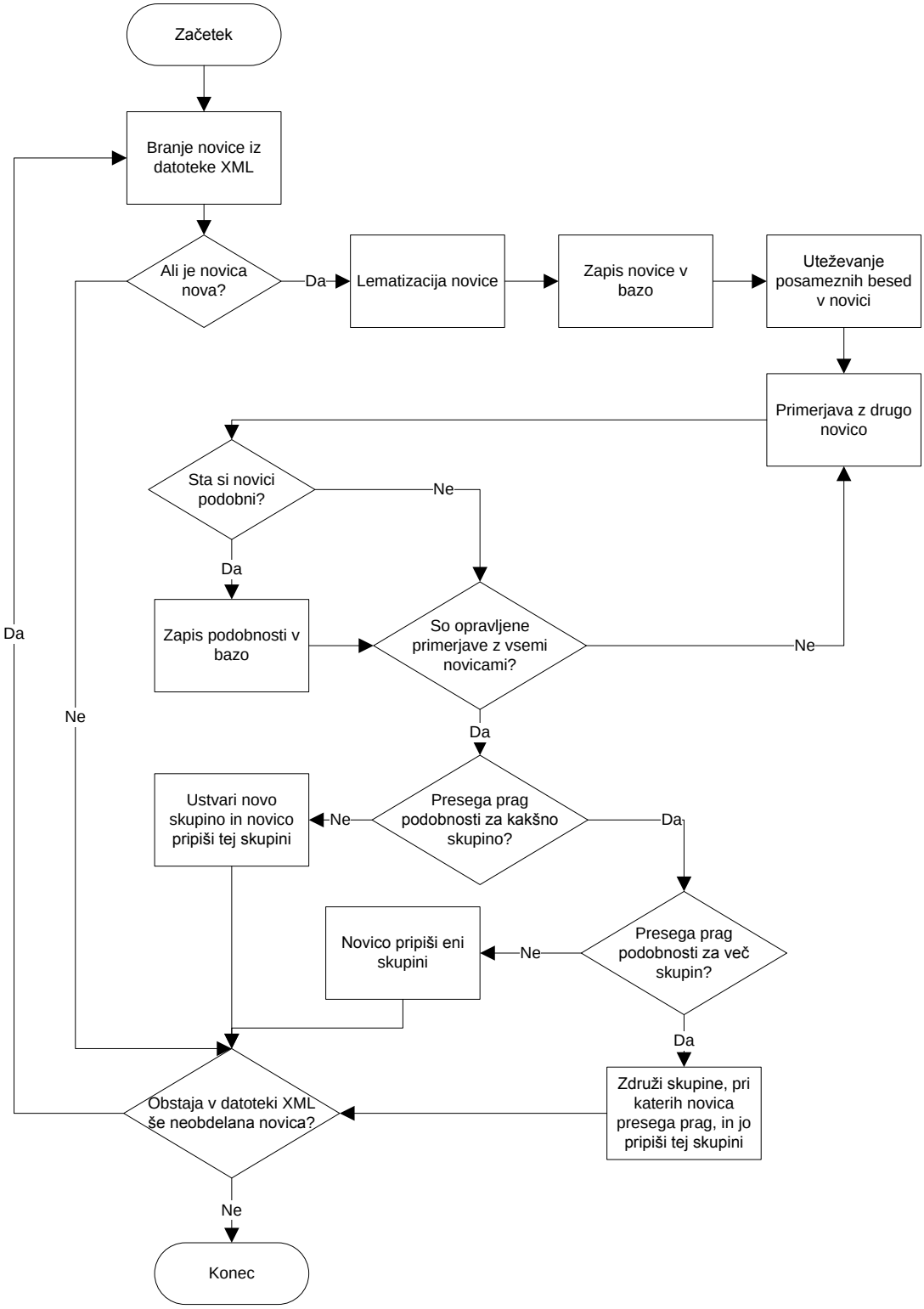
Osnovno kategorizacijo novic izdelajo že mediji sami. Večina spletnih medijev ima novice razporejene v rubrike. Uporabljajo tudi podobna imena rubrik, kot so Slovenija, svet, šport, gospodarstvo … Temu primerno imajo prilagojene vire RSS, kar v večini primerov pomeni, da ena datoteka XML vsebuje zgolj novice iz ene rubrike. Mediji, ki uporabljajo skupno datoteko XML za vse novice, običajno dodajajo dodatno oznako vsaki novici. Pri našem delu smo se osredotočili zgolj na eno rubriko – novice iz Slovenije. Aplikacijo bi razmeroma enostavno prilagodili kategorizaciji novic tudi iz drugih rubrik, spremenili bi samo prikaz na spletni strani, zajemanje virov RSS pa bi potekalo z drugih naslovov.

Za razvoj portala smo uporabili programski jezik PHP, podatke pa smo shranjevali v podatkovni bazi MySQL.

Sam postopek razvoja portala bi lahko razdelili na več sklopov:

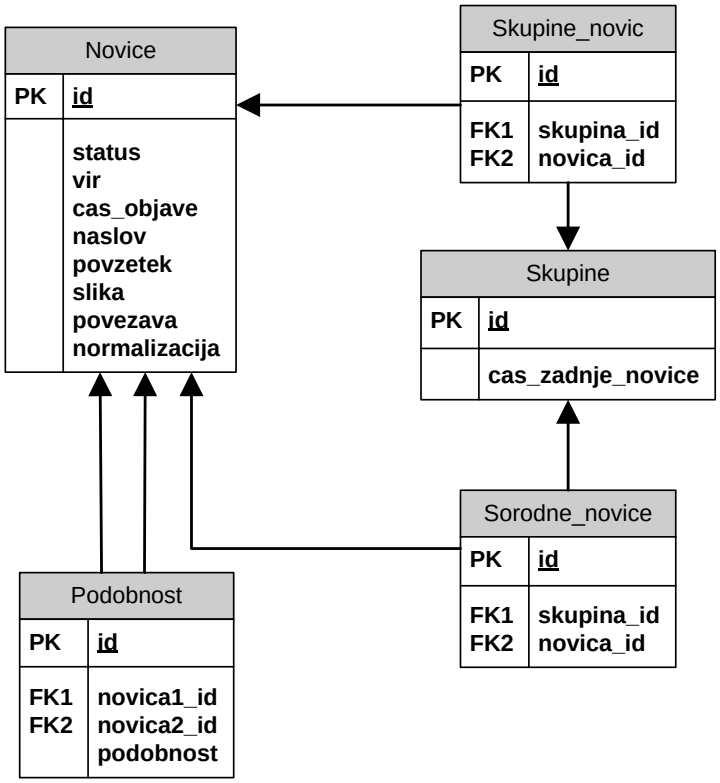
1. zajem virov RSS in shranjevanje novic v bazo MySQL,
2. lematizacija posameznih novic,
3. uteževanje posameznih besed v novici,
4. primerjave novic,
5. izdelava skupin podobnih novic,
6. predstavitev skupine podobnih novic na spletni strani.

Diagram na sliki 3 predstavlja proces obdelave enega vira. V zanki preberemo datoteko XML. Znotraj zanke obdelamo posamezno novico in v novi zanki izračunamo podobnost s preostalimi novicami. Nato določimo, kateri skupini pripada obravnavana novica.



Slika 3. Proces obdelave enega vira.

Osnova za našo aplikacijo je bila dobro definirana baza MySQL, kamor smo shranjevali vse pomembne informacije za obdelavo in prikaz novic na našem spletnem portalu. Konceptualni diagram podatkovne baze predstavlja slika 4.



Slika 4. Konceptualni prikaz podatkovne baze.

4.1. Zajem virov RSS

Slovenski spletni mediji povzetke novic objavljajo tudi preko virov RSS, ki so strukturirane datoteke XML. Te datoteke smo prebrali in v bazo zapisali strukturirane podatke vsake posamezne novice. Vsaka datoteka XML je vsebovala več novic. Večina datotek XML ima za nas pomembne podatke zapisane v oznakah, nekatere pa dodatne podatke zapisujejo tudi kot atribute oznak. Za vsako novico smo poleg vira potrebovali še naslov, povzetek, sliko, povezavo in čas objave novice. Primer dveh zapisov novice v datoteki XML je na sliki 5, kjer sta predstavljena izseka datoteke XML, ki predstavljata eno novico.

```
<item>
    <enclosure url="http://www.vir1.si/assets/media/p
    <title>1. SNL: Maribor le s tocko iz Nove Gorice,
    <link>http://www.vir1.si/sport/nogomet/1-snl-mari
    <description>V 10. krogu 1. SNL so Mariborcani v
    <pubDate>Wed, 21 Sep 2011 22:35:00 CEST</pubDate>
    <author>Š. Ro., Vir1</author>
    <category>Šport</category>
    <guid isPermaLink="false">174430_2011-09-21 22:35
</item>
```

```
<item>
    <title>Real spet razocaral, Genoa na vrhu</title>
    <link>http://www.vir2.net/sportal/nogomet/tuja_pr
    <description>
    <![CDATA[
    <img src="http://www.vir2.net/sn/img/11/264/63452
    <p>Ljubljana - Sredi tega tedna so nogometaši pro
    </p>]]>
    </description>
    <dc:creator>Vir2</dc:creator>
    <guid isPermaLink="false">F08B995EA01D4D06A2234C3
    <pubDate>Wed, 21 Sep 2011 23:18:00 +0200</pubDate
</item>
```

Slika 5. Primera zapisa novic v datoteki XML.

Z razčlenjevalnikom kode smo iz obeh primerov pridobili vse potrebne podatke za uspešno klasifikacijo in prikaz novic. V obeh primerih oznaka *item* predstavlja posamezno novico. Vir novice nam je bil znan že na začetku, saj smo morali dostopati do točno določenega vira RSS. Vse ostale pomembne podatke smo na primeru iz slike 5 pridobili na način, kot je zapisano v tabeli 2.

PODATEK	VIR 1	VIR 2
Naslov	oznaka *title*	oznaka *title*
Povzetek	oznaka *description*	z razčlenjevanjem kode HTML v oznaki *description*
Slika	atribut *url* oznake *enclosure*	z razčlenjevanjem kode HTML v oznaki *description*; atribut *src* oznake *img* kode HTML
Povezava	oznaka *link*	oznaka *link*
Čas objave	oznaka *pubDate*	oznaka *pubDate*

Tabela 2. Primera pridobivanja podatkov o novici na podlagi slike 5.

Pri zajemanju novic iz virov RSS smo uporabili razčlenjevalnik kode PHP XML Expat. Ta razčlenjevalnik vidi datoteko kot zaporedje dogodkov. Ko se nek dogodek zgodi, pokliče funkcijo, ki ga obdela. Tak tip razčlenjevalnika se osredotoča na vsebino in ne na strukturo datoteke XML. Zaradi tega lahko dostopa do podatkov hitreje od razčlenjevalnikov, ki temeljijo na drevesni strukturi datoteke XML [11].

Zajeli smo 1.000 novic in jih shranili v našo bazo. Te novice so bila naša zbirka, na kateri smo preizkušali razvite metode. Zajemanje smo nadaljevali še naprej, da smo razvili tudi kategorizacijo novic ob dinamičnem povečevanju zbirke.

4.1.1. Težave pri zajemu novic

Zajemanje novic je bilo za vsak vir drugačno, saj je bila struktura datotek XML različna. Vse datoteke smo pregledali in prilagodili razčlenjevalniku datotek XML za vsak vir posebej. Mediji lahko objavijo novo novico vsak trenutek, zato je bilo pomembno, da je aplikacija čim pogosteje zajemala datoteke XML z novicami. Le tako smo lahko zagotovili ažurnost prikaza novic na našem portalu. Na strežniku se je ponavljajoče poganjal proces, ki je obiskal vse naše vire in pregledal datoteke XML, če so obstajale nove novice. Pri tem je bilo potrebno določiti kriterije za novost. Ugotovili smo, da mediji spreminjajo stare novice in morali smo paziti, da nismo večkrat zapisali iste novice. Mediji lahko v naslovu ali povzetku naknadno popravljajo slovnične napake, spremenijo naslov zaradi privlačnosti, zamenjajo povezavo do novice ali slike. Nekateri mediji določene dogodke spremljajo od začetka do konca, zato najprej ustvarijo novico, ki jo osvežujejo tekom poteka dogodka. Ob nekaterih popravkih tudi spremenijo čas objave novice. Verjetnost spremembe novice pada s časovno oddaljenostjo od

prvotne objave. Sprememba novice vpliva tudi na primerjavo novice z ostalimi novicami in na klasifikacijo. Ker je končni izpis skupine novic z enako temo potreboval razvrščanje, smo morali biti previdni, da se zaradi sprememb vrstni red ne bi spreminjal, ali da novica ne bi zamenjala skupine. Za reševanje teh problemov smo preverjali vsak vir posebej, da smo ugotovili specifike medijev in tako omejili vpliv naknadnih sprememb na naš portal.

4.2. Normalizacija novice

Za uspešno primerjavo novic smo besedila primerno obdelali. Ker smo primerjali novice med sabo v obliki vektorjev, smo novice najprej normalizirali. Uporabili smo lematizacijo, saj v slovenskem jeziku daje boljše rezultate kot s krnjenjem.

Ker predstavitev v vektorskem prostoru upošteva vse besede, ki sestavljajo besedilo, smo ugotovili, da je potrebno iz novic odstraniti pogoste besede, ki so v preveliki meri vplivale na rezultate klasifikacije. Vsako slovensko besedilo vsebuje veznike (na primer: *in, ki, ker, ko …*) in predloge (na primer: *v, na, ob, nad …*), ki nimajo pomena za razumevanje. Iz naslova in povzetka novic smo odstranili veznike, predloge in zelo pogost pomožni glagol *biti* v vseh oblikah. Iz preostalih besed smo dobili osnovo za predstavitev v vektorskem prostoru.

Za lematizacijo smo uporabili lematizator JOS ToTaLe text analyser [10], ki so ga razvili na Institutu Jožef Stefan. Lematizatorju smo pošiljali zahteve z besedilom v osnovni obliki. V odgovorih na naše zahteve smo dobili besede, ki so sestavljale novico, skupaj z lematizirano besedo – lemo in oblikoslovno oznako. Oblikoslovne oznake nismo potrebovali, zato smo ta podatek zanemarili.

V nadaljevanju predstavljamo primer lematizacije dveh novic.

- *Novica 1:* Posočje je danes zjutraj stresel potres stopnje 5 po Rihterjevi lestvici. Prebivalci niso poškodovani, nastala je le gmotna škoda.
- *Novica 2:* Okolico Bovca je danes ob 5.00 prizadel potresni sunek stopnje 5 po mednarodni lestvici. Poškodovani so nekateri objekti, o ranjenih prebivalcih ne poročajo.

V tabelah 3 in 4 sta predstavljeni obe novici, kjer je vsaki besedi iz novice pripisana oblikoslovna oblika ter lema besede.

Izvorna beseda	**Oblikoslovna oznaka**	**Lema**
Posočje	Slsei	posočje
je	Gp-ste-n	~~biti~~
danes	Rsn	danes

zjutraj	Rsn	zjutraj
stresel	Ggdd-em	stresti
potres	Somei	potres
stopnje	Sozer	stopnja
5	Kag	5
po	Dm	~~po~~
Rihterjevi	Psnzem	rihterjev
lestvici	Sozem	lestvica
.	.	~~.~~
Prebivalci	Sommi	prebivalec
niso	Gp-stm-d	~~biti~~
poškodovani	Pdnmmi	poškodovan
,	,	~~,~~
nastala	Ggdd-ez	nastati
je	Gp-ste-n	~~biti~~
le	L	~~le~~
gmotna	Ppnzei	gmoten
škoda	Sozei	škoda
.	.	~~.~~

Tabela 3. Lematizacija 1. novice: Posočje je danes zjutraj stresel potres stopnje 5 po Rihterjevi lestvici. Prebivalci niso poškodovani, nastala je le gmotna škoda.

Izvorna beseda	Oblikoslovna oznaka	Lema
Okolico	Sozet	okolica
Bovca	Slmer	bovec
je	Gp-ste-n	~~biti~~
danes	Rsn	danes
ob	Dm	~~ob~~
5.00	Kag	5.00
prizadel	Ggdd-em	prizadeti
potresni	Pdnmetd	potresen
sunek	Sometn	sunek
stopnje	Sozer	stopnja
5	Kag	5
po	Dm	~~po~~
mednarodni	Ppnzem	mednaroden
lestvici	Sozem	lestvica
.	.	.
Poškodovani	Pdnmmi	poškodovan
so	Gp-stm-n	~~biti~~
nekateri	Zn-mmi	nekateri
objekti	Sommi	objekt
,	,	~~,~~
o	Dm	~~o~~
ranjenih	Pdnmmr	ranjen
prebivalcih	Sommm	prebivalec
ne	L	~~ne~~
poročajo	Ggnstm	poročati
.	.	~~.~~

Tabela 4. Lematizacija 2. novice: Okolico Bovca je danes ob 5.00 prizadel potresni sunek stopnje 5 po mednarodni lestvici. Poškodovani so nekateri objekti, o ranjenih prebivalcih ne poročajo.

Ko nam je lematizator uspešno vrnil rezultate, smo uporabili še seznam besed za odstranjevanje in na tak način odstranili veznike, predloge in pomožni glagol biti. Te besede smo v tabelah št. 3 in 4 prečrtali in označili z rdečo barvo. To smo storili s spodnjo funkcijo.

```
function remove_stopwords($textarr) {
    $stopwords = array("biti","in","ki","se","ker","pa","ali","ko",
        "medtem","da","še","tako","ta","kot","kako","ter","na","saj",
        "iz","le","pred","kar","od","za","po","ob","do","že","nad",
        "sicer","tudi","kaj","že","zakaj","ne", "pri", "med", "o");
    $text = array_diff($textarr,$stopwords);
    return array_values($text);
}
```

Odstranili smo tudi ločila, ki bi drugače vplivala na vektorsko predstavitev, saj bi bilo vsako ločilo predstavljeno kot ena izmed besed, ki sestavlja novico.

Da smo čas lematiziranja skrajšali, smo vse prejete odgovore shranili v našo bazo v tabelo besed, kjer smo zapisali vsako osnovno besedo in njeno lemo. Pred vsako zahtevo lematizatorju smo preverili, če je beseda že shranjena v naši bazi in je ni potrebno ponovno lematizirati. Pred pošiljanjem zahtev smo združili več novic, da smo zmanjšali čas lematiziranja, ki je bil odvisen od odzivnosti lematizatorja. Čas se je zmanjšal, če smo poslali manj zahtev, ki so vključevale več besed.

4.2.1. Težave pri lematizaciji novic

Velika težava pri lematizaciji novic so pravopisne napake (na primer zatipkane besede), saj jih je težko najti, če se ne ukvarjamo z razumevanjem vsebine novice. Natančnost izračuna podobnosti med novicami in točnost kategorizacije novic se zmanjša. Težava je tudi odvisnost od zunanje aplikacije, ki smo jo uporabili za lematizacijo. V določenih primerih poteče čas za odgovor na zahtevo in je potrebno postopek ponoviti.

4.3. Uteževanje besed

Ko smo vsako posamezno novico normalizirali, smo jo predstavili z vektorjem. 1.000 novic v naši bazi vsebuje 6.182 različnih besed. Za vse različne besede smo prešteli, kolikokrat se pojavi v zbirki. Vsaka različna beseda predstavlja eno dimenzijo v vektorskem prostoru, zato smo dobili 6.182-dimenzionalen prostor. Ker vsaka novica ne vsebuje vseh 6.182 različnih besed, smo za izračun upoštevali le tiste besede, ki sestavljajo posamezno novico. Ostale dimenzije vektorskega prostora smo zanemarili, ker so uteži teh besed enake 0.

4.3.1. Prikaz izračuna uteži za eno besedo

Kot primer prikaza izračuna uteži smo vzeli besedo »mednaroden«, ki se v proučevani novici pojavi enkrat. V zbirki 1.000 novic se skupno pojavi v 29 novicah.

$w_{mednaroden,novica2} = tf_{mednaroden,novica2} * log(N/df_{mednaroden})$
$w_{mednaroden,novica2} = 1 * log(1000/29)$
$w_{mednaroden,novica2} = \underline{1{,}5376}$

4.3.2. Prikaz izračuna uteži za celo novico

V tabeli 5 prikazujemo primer izračuna uteži za besede iz prej navedenih primerov novic.

1: beseda	2: tf_{1j}	3: tf_{2j}	4: df_j	5: $IDF_j = log(N/df_j)$	6: $TF\text{-}IDF_{1j}$	7: $TF\text{-}IDF_{2j}$
5	1	1	2	2,6990	2,6990	2,6990
5.00	0	1	1	3	0	3
bovec	0	1	6	2,2218	0	2,2218
danes	1	1	86	1,0655	1,0655	1,0655
gmoten	1	0	1	3	3	0
lestvica	1	1	2	2,6990	2,6990	2,6990
mednaroden	0	1	29	1,5376	0	1,5376
nastati	1	0	5	2,3010	2,3010	0
nekateri	0	1	41	1,3872	0	1,3872
objekt	0	1	11	1,9586	0	1,9586
okolica	0	1	9	2,0458	0	2,0458
poročati	0	1	6	2,2218	0	2,2218
posočje	1	0	4	2,3979	2,3979	0
poškodovan	1	1	6	2,2218	2,2218	2,2218
potres	1	0	6	2,2218	2,2218	0
potresen	0	1	6	2,2218	0	2,2218
prebivalec	1	1	11	1,9586	1,9586	1,9586
prizadeti	0	1	3	2,5229	0	2,5229
ranjen	0	1	1	3	0	3
rihterjev	1	0	1	3	3	0
stopnja	1	1	3	2,5229	2,5229	2,5229
stresti	1	0	3	2,5229	2,5229	0
sunek	0	1	14	1,8539	0	1,8539
škoda	1	0	11	1,9586	1,9586	0
zjutraj	1	0	13	1,8861	1,8861	0

Tabela 5. Prikaz izračuna uteži za besede v dveh primerih novic.

V prvem stolpcu so zapisane vse različne besede, ki se pojavljajo v novicah iz primera. V drugem stolpcu so predstavljene frekvence besed v novici št. 1, v tretjem stolpcu je enak podatek še za novico št. 2, v četrtem stolpcu pa je število novic iz naše zbirke, ki vsebujejo posamezno besedo. Na podlagi tega podatka smo v petem stolpcu izračunali koeficient IDF, v

zadnjih dveh stolpcih pa smo podali izračun uteži (w_{ij} = $TF\text{-}IDF_{ij}$ = tf_{ij} * $log(N/df_j)$) za posamezno besedo v novici. Vrednost šestega oz. sedmega stolpca smo izračunali kot zmnožek vrednosti drugega oz. tretjega in petega stolpca.

S pomočjo uteži besed v zadnjih dveh stolpcih predstavimo oba primera novic v obliki, s katero lahko izračunamo mero podobnosti novic.

4.3.3. Uteževanje besed novice v zbirki novic, ki se skozi čas veča

Baza 1.000 novic je bila dobra osnova za prenos v prakso. Ko se baza novic skozi čas veča, se spremeni izračun mere podobnosti. Težava nastane pri izračunavanju koeficienta IDF (*IDF = log(N/df)*). Spremenljivka *df* namreč predstavlja pojavnost besede v celotni zbirki novic. Ob zajemu nove novice se ta spremenljivka poveča za vse tiste besede, ki jih nova novica vsebuje. Z zajemom vsake novice se veča tudi spremenljivka *N*.

Če smo besedi v preteklosti pripisovali večji koeficient IDF, ker se je v bazi novic pojavljala redko, se je to lahko spremenilo. To se zgodi v primeru, da se v novih novicah ta beseda pojavlja relativno pogosteje kot v preteklosti. Seveda velja tudi obratno, da se v novih novicah beseda ne pojavlja več tako pogosto kot v preteklosti.

4.4. Podobnost dveh besedil

Za izračun podobnosti dveh novic smo uporabili kosinusno razdaljo.

$$cos(d_i, d_j) = \frac{\sum_{l=1}^{T} w_{il} \times w_{jl}}{\sqrt{\sum_{l=1}^{T} {w_{il}}^2} \times \sqrt{\sum_{l=1}^{T} {w_{jl}}^2}}$$

Na spodnjem primeru smo uporabili podatke iz tabele 5, ki vsebuje izračunane uteži besed.

$$cos(d_i, d_j) = \frac{2{,}6990 \times 2{,}6990 + 0 \times 3 + \cdots + 1{,}8861 \times 0}{\sqrt{2{,}6990^2 + 0^2 + \cdots + 1{,}8861^2} \times \sqrt{2{,}6990^2 + 3^2 + \cdots + 0^2}} = 0{,}3757$$

Postopek smo ponovili za vse pare iz zbirke. Tako smo postopek opravili 499.500-krat.

$$C = \binom{n}{2} = \frac{n \times (n-1)}{2} = \frac{1000 \times (1000-1)}{2} = 499500$$

V našem primeru nismo izračunali uteži vnaprej, saj bi morali rezultate zapisati v tabelo, ampak smo uteži izračunali tik preden smo začeli primerjavo. Vzeli smo prvo in drugo novico ter ju primerjali, nato prvo in tretjo novico in tako naprej do prve in zadnje novice. Nato smo vzeli drugo in tretjo novico, drugo in četrto ter tako naprej do druge in zadnje novice. Kot zadnjo smo izračunali podobnost med predzadnjo in zadnjo novico. Takšnih primerjav smo naredili 499.500.

Kjer je bil rezultat podobnosti med novicama večji od nič, smo ga zapisali v bazo. V tabeli smo za vsak izračun zapisali identifikacijsko številko prve in druge novice ter podobnost. Izračunane podobnosti so bile osnova za določanje, v katero skupino novica sodi.

4.4.1. Mere podobnosti novic v zbirki, ki se veča skozi čas

Z večanjem baze zajetih novic se spreminjajo uteži besed. Na podlagi stalno spreminjajočega se koeficienta IDF se spreminjajo uteži besed v tistih novicah, ki smo jih predhodno že obdelali in izračunali podobnosti. To bi zahtevalo ponoven izračun podobnosti ob vsakem dodajanju. Takšen postopek bi bil časovno zelo potraten, ne bi pa prinesel bistvenih sprememb. Zato smo se odločili, da v primeru večanja baze podobnosti za že obdelane novice nismo ponovno izračunavali.

Za vsako novo novico v bazi bi morali izračunati mero podobnosti z vsemi že obstoječimi novicami. Kategorizacija novic je le v tem primeru res dobra, vendar z večanjem baze postane postopek časovno zelo zahteven. Zato smo izračunavanje podobnosti omejili zgolj na novice, ki so novejše od določene meje (na primer 1, 7, 14 ali 30 dni). Potrebno je poskrbeti, da postopek kategorizacije opravimo hitreje, kot mediji objavljajo novice. V nasprotnem primeru bi morali določene novice preskočiti in jih ne obdelati, da bi ostajali aktualni.

4.5. Kategorizacija novic

V eno izmed tabel v bazi smo shranili podobnosti za vse pare novic iz zbirke. Kje je prag podobnosti, ko sta si dve novici še podobni po vsebini, smo določili empirično s poskušanjem, dokler nismo dobili zadovoljivih rezultatov. Če je bil prag postavljen previsoko, je nastalo preveč skupin, nekatere z le eno novico, ki je bila vsebinsko podobna drugim. Če smo prag postavili prenizko, so bile v isto skupino izbrane novice, ki si vsebinsko niso bile podobne.

Nadaljnji postopek je bil avtomatski: vzeli smo prvo novico ter poiskali novice s podobnostjo nad določenim pragom. S tem je nastala skupina novic. Vse novice, ki so bile podobne prvi,

so lahko podobe še drugim. Tako smo za vsako novico, ki je postala del skupine, poiskali podobne novice. Pogoj je bil, da novica še ni bila dodeljena v nobeno skupino.

4.5.1. Postopek kategorizacije novic

Poglejmo postopek na primeru zbirke 6 novic po korakih, ki smo jih predstavili v tabeli št. 6. V matrikah so z »Da« označeni tisti pari novic, ki so si podobni. »Ne« označuje, da si novice medsebojno niso dovolj podobne.

Korak 1:

	A	?	?	A	A	?
1						
2	Ne					
3	Ne	Da				
4	Da	Ne	Ne			
5	Da	Ne	Ne	Da		
6	Ne	Ne	Ne	Ne	Ne	
	1	2	3	4	5	6

Korak 2:

	A	?	?	A	A	?
1						
2	Ne					
3	Ne	Da				
4	Da	Ne	Ne			
5	Da	Ne	Ne	Da		
6	Ne	Ne	Ne	Ne	Ne	
	1	2	3	4	5	6

Korak 3:

	A	?	?	A	A	?
1						
2	Ne					
3	Ne	Da				
4	Da	Ne	Ne			
5	Da	Ne	Ne	Da		
6	Ne	Ne	Ne	Ne	Ne	
	1	2	3	4	5	6

Korak 4:

	A	B	B	A	A	?
1						
2	Ne					
3	Ne	Da				
4	Da	Ne	Ne			
5	Da	Ne	Ne	Da		
6	Ne	Ne	Ne	Ne	Ne	
	1	2	3	4	5	6

Korak 5:

	A	B	B	A	A	?
1						
2	Ne					
3	Ne	Da				
4	Da	Ne	Ne			
5	Da	Ne	Ne	Da		
6	Ne	Ne	Ne	Ne	Ne	
	1	2	3	4	5	6

Korak 6:

	A	B	B	A	A	C
1						
2	Ne					
3	Ne	Da				
4	Da	Ne	Ne			
5	Da	Ne	Ne	Da		
6	Ne	Ne	Ne	Ne	Ne	
	1	2	3	4	5	6

Tabela 6. Prikaz postopka določanja skupin novic na podlagi postopka opisanem v [5].

Prvi korak:

- vzeli smo novico 1 in poiskali podobne novice;

- novica 1 je bila podobna novicama 4 in 5;
- nastala je skupina A z novicami 1, 4, 5;
- nerazvrščene so ostale novice 2, 3, 6.

Drugi korak:

- vzeli smo novico 4, ki je bila prva dodana v skupino A in še nismo naredili preverjanja podobnosti;
- novica 4 je bila podobna novici 5;
- ker je bila novica 5 že del skupine A, je le-ta ostala enaka kot po predhodnem koraku;
- nerazvrščene so ostale novice 2, 3, 6.

Tretji korak:

- vzeli smo novico 5, ki je bila naslednja dodana v skupino A in še nismo naredili preverjanja podobnosti;
- novica 5 ni bila podobna nobeni preostali novici;
- skupina A je ostala enaka kot po predhodnem koraku;
- nerazvrščene so ostale novice 2, 3, 6.

Četrti korak:

- ker smo preverili že vse novice iz skupine A, smo vzeli naslednjo novico, ki še ni bila preverjena – novica 2;
- novica 2 je bila podobna novici 3;
- nastala je skupina B z novicama 2 in 3;
- nerazvrščena je ostala novica 6.

Peti korak:

- vzeli smo novico 3, ki je bila prva dodana v skupino B in še nismo naredili preverjanja podobnosti;
- novica 3 ni bila podobna nobeni preostali novici;
- skupina B je ostala enaka kot po predhodnem koraku;
- nerazvrščena je ostala novica 6.

Šesti korak:

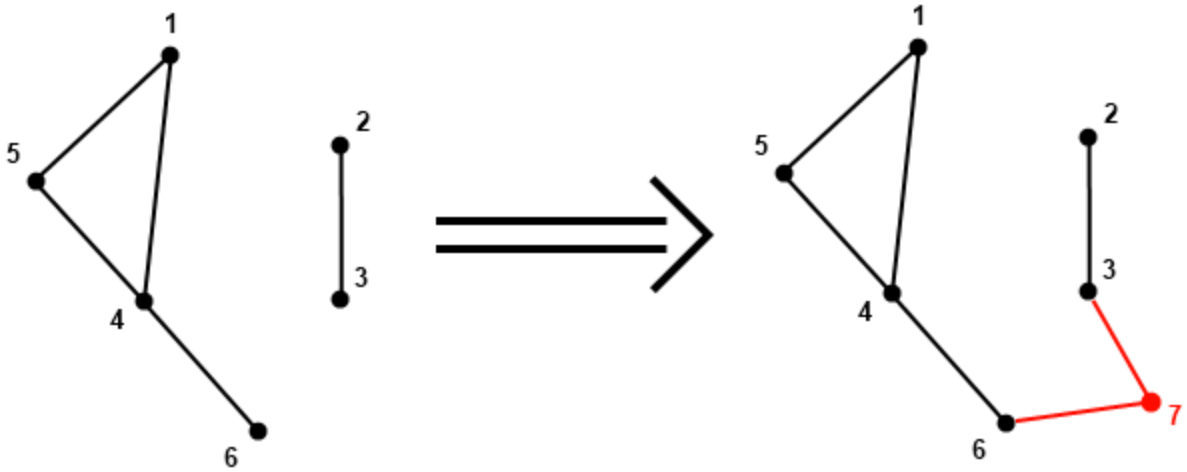
- ker smo preverili že vse novice iz skupine B, smo vzeli naslednjo novico, ki še ni bila preverjena – novica 6, ki je bila hkrati tudi zadnja;
- novica 6 ni podobna nobeni drugi novici;
- nastala je skupina C, ki je imela samo novico 6;

- razvrščene so vse novice.

V bazo smo zapisali pare, ki označujejo identifikacijsko oznako skupine in identifikacijsko številko novice. Če je skupina imela tri novice, smo v tabelo zapisali tri vrstice.

Določanje skupin je porodilo še idejo o drugem pragu. S pragom, ki je postavljen višje, smo v skupino izbrali novice, ki so si podobne po vsebini. Te govorijo o isti temi, razlikujejo se morda le v dodatnih informacijah, ali pa so le preoblikovane in ne ponujajo ničesar novega. Ko smo določili nižji prag, so se med obema pragoma pokazale novice, ki sicer niso podobne novicam v skupini, so si pa sorodne. Tako se je prikaz na spletni strani obogatil. Poleg izpisa podobnih novic iz ene skupine, so se izpisale še sorodne novice. Pri tem se je posamezna novica lahko prikazala kot sorodna pri več skupinah novic.

Ob večanju zbirke novic z zajemanjem novih novic je potrebno za vsako novo dodano novico poiskati, v katero skupino sodi. Če podobnost z eno novico presega prag, ki določa pripadanje skupini, potem sodi v isto skupino novic. Če praga ne presega, se ustvari nova skupino, ki vključuje samo to novo novico. Težava nastane v primeru, ko je prag presežen pri izračunu podobnosti z dvema ali več novicami. V tem primeru se skupine združijo v eno samo, kar predstavljamo na sliki 6.



Slika 6. Združevanje dveh skupin v primeru, ko je nova novica podobna novicam iz dveh skupin.

4.6. Prikaz na spletni strani

Ker je bil eden ciljev naloge olajšanje spremljanja novic, smo ta del natančno definirali. Ker smo se omejili zgolj na zajemanje novic, ki jih spletni mediji objavljajo v rubriki Slovenija, osnovnega menija na naši spletni predstavitvi nismo postavili. Glavno merilo pri prikazu je bila aktualnost ter časovno razporejanje. Bralec mora ob obisku spletne strani najprej dobiti na razpolago zadnje novice. Naj spomnimo, da smo ob zajemanju virov RSS v bazo zapisali

tudi čas objave novice. Na sliki 7 predstavljamo spletno stran portala s petimi skupinami novic.

Slika 7. Vstopna stran portala z najbolj aktualnimi skupinami novic na dan 7. 2. 2012.

V prvem koraku smo v časovnem zaporedju izpisali vse skupine novic. Na vrhu smo izpisali skupino z novico, ki je bila objavljena nazadnje, nato je sledila skupina s predzadnjo novico in tako naprej. Če se je izkazalo, da sta bili zadnja in predzadnja novica del iste skupine, smo predzadnjo preskočili in poiskali še neprikazano predhodno novico.

Takšen postopek se je izkazal za počasnega, zato smo izpis pohitrili tako, da smo v bazi ustvarili tabelo, kamor smo zapisali identifikacijsko oznako skupine ter čas objave zadnje

novice, ki pripada tej skupini. Tako smo ob izpisovanju najprej s poizvedbo SQL pridobili vrstni red skupin.

V drugem koraku je sledil izpis posamezne skupine. Skupina vsebuje novice o eni temi oziroma dogodku. Medij, ki o določeni temi prvi ustvari novico, je najbolj aktualen. Ker svojo novico objavi pred ostalimi, večkrat poroča bolj površinsko in se ne poglobi v podrobnosti. Zato so novice ostalih medijev lahko bolj kvalitetne in bralcem nudijo več, razjasnijo se okoliščine dogodka, pridobi se več virov informacij in podobno. Zato smo ob prikazu skupine novic na vrhu prikazali zadnjo objavljeno novico in po časovnem redu vse do zadnje. Nekatere skupine so vsebovale le eno novico.

Ob prikazu prve novice v skupini smo izpisali naslov, povzetek, čas objave na strani medija, ime medija ter prikazali sliko. Ob izpisu ostalih novic v skupini smo izpisali zgolj naslov in ime medija.

Takšen izpis je smiseln tako v primeru omejene baze 1.000 novic kot tudi v primeru dinamičnega večanja baze novic, ki jih pridobivamo skozi čas. Spremeni se samo to, da se skupine sčasoma večajo in da se spreminja vrstni red izpisa skupin.

Ob izpisu vsake skupine smo izpisali tudi povezavo do podrobnejšega prikaza celotne skupine. Povezavo na spletni strani smo poimenovali »Vse novice o temi«. Na tej strani smo pri vseh novicah izpisali naslov novice, povzetek, čas objave, ime medija ter prikazali sliko. Takšen prikaz na zaslonu zavzame več prostora, zato ni primeren za prikaz vseh skupin naenkrat. Bralec se v takem primeru izgubi v preobilju novic ter težje loči med posameznimi temami. Stran je na desni strani ponudila prostor za prikaz sorodnih novic, ki smo jih pridobili ob izdelavi skupin novic in so bile po meri podobnosti med obema pragoma. Ta sklop smo na spletni strani poimenovali »Morda vas zanima tudi …«. Pri sorodnih novicah smo izpisali zgolj naslov novice in ime medija, sam izpis pa je bralcem omogočil globlje raziskovanje zgodbe. Primer podrobnejšega prikaza ene skupine novic predstavljamo na sliki 8.

Slika 8. Prikaz ene skupine novic.

5. SKLEPNE UGOTOVITVE IN NADALJNJE DELO

Predstavili smo združevanje novic iz več virov po podobnosti vsebine. V eno skupino smo združili vse podobne novice, ki smo jih zajeli iz različni virov RSS. Razvili smo algoritem za izračun podobnosti novic in postavili pravila, ki določajo, kdaj novice sodijo v eno skupino in kako so znotraj skupine razvrščene. Pri delu smo se spoznavali z različnimi metodami obdelave naravnega jezika in specifikami slovenskega jezika.

Pri pisanju diplomske naloge in izdelavi praktičnega primera sem se podrobneje spoznal s programskim jezikom PHP in bazo MySQL. Znanje obdelave besedil in primerjave sorodnosti novic se lahko razširi na področje odkrivanja duplikatov, kategorizacije knjig, določanja ključnih besed, iskalnikov in podobno. Naš spletni portal bi lahko s pridobljenim znanjem obogatili z mnogimi izboljšavami, a predstavlja dobro osnovo za razširitve in nadaljnje delo.

Podobnost smo izračunavali s predstavitvijo v vektorskem prostoru in s kosinusno razdaljo. Dobro bi bilo preizkusiti še druge postopke in druge koeficiente podobnosti. Morda bi z drugačnim pristopom prišli do boljših rezultatov pri razvrščanju novic v skupine in zmanjšali časovno zahtevnost.

Pri našem delu smo se omejili zgolj na novice iz rubrike Slovenija. Sistem bi lahko enostavno prilagodili za obdelavo več rubrik, zajeli bi lahko tudi več virov novic. Pri povečanju obsega novic bi bila potrebna optimizacija algoritma za izračun podobnosti. Primerjali bi lahko več novic in obiskovalcem naše spletne strani izboljšali predloge za branje sorodnih novic. Iz novic bi lahko izluščili ključne besede, na primer imena oseb, in tako pripravili kronološki pregled pisanja medijev o osebah.

Kronološko razvrščanja skupin je možno še izboljšati. Zdaj se na vrhu pojavi tista skupina, ki vsebuje zadnjo objavljeno novico. Pri tem smo domnevali, da so kasneje objavljene novice bolj poglobljene, nismo pa upoštevali, da nekateri mediji svoje prispevke objavljajo počasneje. V tem primeru se lahko zgodi, da aktualnejši mediji že pišejo o novih zgodbah, ko počasnejši objavijo starejšo in se s tem prebijejo na vrh prikaza na naši spletni strani. Obiskovalcem s tem v nekaterih primerih ne ponudimo predstavitve najaktualnejšega dogajanja.

Uporaba vektorske predstavitve in kosinusne podobnosti je pogosta pri iskalnikih. Tudi naš novičarski portal bi lahko obogatili z iskalnikom po novicah. Ob izvedbi iskanja se primerja

vneseno poizvedbo z vsemi novicami. Zadetke se razvrsti tako, da je novica, ki ima največjo podobnost z iskalno poizvedbo pri vrhu, sledijo pa ji novice s čedalje manjšo podobnostjo.

LITERATURA

[1] E. Garcia. (2006, okt.). Cosine Similarity and Term Weight Tutorial. [Splet]. Dostopno na: http://www.miislita.com/information-retrieval-tutorial/cosine-similarity-tutorial.html (Datum dostopa: 10. 11. 2011).

[2] M. Hadjieleftheriou, D. Srivastava, »Weighted Set-Based String Similarity«, v zborniku *IEEE Data Engineering Bulletin, Volume 33*, D. B. Lomet; Los Alamitos, Kalifornija: IEEE Computer Society Press, mar. 2010, str. 25-36.

[3] B. Jerko, »Samodejno indeksiranje povzetkov«, v zborniku *Jezikovne tehnologije: Zbornik B 7. mednarodne multi-konference Informacijska družba IS 2004, 9. do 15. oktober 2004*, T. Erjavec, J. Ž. Gros; Ljubljana: Institut Jožef Stefan, okt. 2004, str. 13-17.

[4] D. Jurafsky, J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*, New Jersey: Pearson Prentice Hall, 2. izdaja, 2009, pogl. 23.1.

[5] C. J. van Rijsbergen, *Information Retrieval*, London: Butterworth-Heinemann, 2. izdaja, 1979, pogl. 3.

[6] I. Sandu Popa, K. Zeitouni, G. Gardarin, »Text Categorization for Multi-label Documents and many Categories«, v zborniku *CBMS '07 Proceedings of the Twentieth IEEE International Symposium on Computer-Based Medical Systems: 20-22 June, Maribor, Slovenia*, P. Kokol, V. Podgorelec, D. Mičetić-Turk, M. Zorman, M. Verlič; Los Alamitos, Kalifornija: IEEE Computer Society Press, jun. 2007, str. 421-426.

[7] About Google News. Dostopno na: http://news.google.com/intl/en_us/about_google_news.html (Datum dostopa: 18. 1. 2012).

[8] How to Best Engage Readers Through Google News. Dostopno na: http://press.org/news-multimedia/videos/how-best-engage-readers-through-google-news (Datum dostopa: 18. 1. 2012).

[9] Interno gradivo podjetja TSmedia, d.o.o.

[10] JOS ToTaLe text analyser. Dostopno na: http://nl2.ijs.si/analyze/ (Datum dostopa: 20. 9. 2011).

[11] PHP XML Expat Parser. Dostopno na: http://www.w3schools.com/php/php_xml_parser_expat.asp (Datum dostopa: 15. 9. 2011).

[12] PHTC in More Details. Dostopno na: http://wiki.uni.lu/mine/PHTC+in+More+Details.html (Datum dostopa: 18. 11. 2011).

[13] Vector Space Model. Dostopno na: http://cogsys.imm.dtu.dk/thor/projects/multimedia/textmining/node5.html (Datum dostopa: 25. 11. 2011).