Laboratorij za tehnologije znanja (KTLab)
Zavod za elektroniku, mikrolektroniku, računalne i inteligentne sustave
Fakultet elektrotehnike i računarstva
Sveučilište u Zagrebu

Interni dokument

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1813

Sustav za pretraživanje zbirke često postavljenih pitanja na hrvatskom jeziku

Lovro Žmak

Zagreb, rujan 2009.

Zahvaljujem se mentorici Prof.dr.sc. Bojani Dalbelo-Bašić na strpljenju i pruženoj podršci,
Mr.sc. Janu Šnajder za mnoge savjete,
te ostatku KtLab tima za pruženu pomoć

zadatak

Sadržaj

Popis tablica

Popis slika

1. Uvod

U današnjem su svijetu rasprostranjenost i potreba za informacijama i dalje u porastu. Posredstvom medija mnoštvo je informacija lako dostupno širokom broju ljudi. Iako rasprostranjenost informacija udovoljava - premda ne u potpunosti - velikom dijelu ljudskih potreba za njima, ista ta rasprostranjenost može pridonijeti i smanjenom zadovoljstvu u pogledu dostupnosti informacija. Naime u velikom broju raznih dostupnih informacija često je teško naći onu željenu.

Zbog problema pronalaska relevantnih informacija razvijeni su razni pristupi za njihovu pretragu. Danas često korišten pristup jest pretraga Interneta raznim web-tražilicama. Ovim pristupom smanjuje se opseg informacija koje je potrebno pretražiti kako bi se dobilo relevantne informacije. Iako ovaj pristup zadovoljava većinu ljudskih potreba za informacijama, ne zadovoljava specifičnije prohtjeve koji se tiču užih domena.

Za uže domene razne tvrtke osiguravaju željene informacije svojim korisnicima unutar svoje domene. Dostupnost informacija unutar svoje domene tvrtke osiguravaju preko korisničke podrške: raznim pozivnim centrima, u poslovnicama, putem e-maila, na svojim web stranicama i sl. Uz rastuću potrebu za informacijama i rastući broj ljudi koji koristi naprednije usluge, korisnička podrška naglo raste i počinje stvarati zamjetljive troškove.

Zbog porasta troškova sve veći broj tvrtki seli korisničku podršku iz pozivnih centara i poslovnica na Internet, komunikaciju putem e-maila i web stranica. Jedan od popularnih načina uobličavanja informacija za prezentaciju korisnicima jest baza često postavljanih pitanja. Baza često postavljanih pitanja sastoji se od najčešćih pitanja koja korisnici postavljaju i uobličena je u cjeline oblika pitanje-odgovor.

Često postavljana pitanja mogu rasteretiti korisničku službu najčešćih pitanja te time i smanjiti trošak održavanja korisničke podrške. Iako ovaj pristup djeluje prosperitetno, problem nastaje kada se broj često postavljanih pitanja poveća. Dok je broj pitanja malen, ljudi ih s lakoćom sva pročitaju i eventualno saznavaju

odgovor na svoje pitanje, ali kada je baza pitanja velika, vrlo će se malo korisnika potruditi naći odgovor u bazi. Zato se razvijaju razne metode za automatsko pretraživanje baze često postavljanih pitanja.

U području pretraživanja informacija razvijeni su razni sustavi za pretraživanje baze često postavljanih pitanja, mnogi od kojih daju obećavajuće rezultate. U novije vrijeme sustavi se sve više orijentiraju na upite postavljene prirodnim jezikom, a ne ključnim riječima kao što je to sada slučaj kod većine web-tražilica. Razvijeni sustav koji bi dohvaćao relevantna često postavljana pitanja za određeni upit uvelike bi smanjio trošak održavanja korisničke službe.

Cilj ovog rada jest proučiti razvijene sustave za pretraživanje te primijeniti sakupljena znanja pri izradi sustava za pretraživanje često postavljanih pitanja na hrvatskom jeziku.

2. Sustavi za odgovaranje na pitanja

Odgovaranje na pitanje (engl. *question answering, QA*) jedno je od područja pretraživanja informacija koje se bavi odgovaranjem na pitanja postavljena prirodnim jezikom. Za razliku od pretraživanja dokumenata (engl. *document retrieval*) kod kojeg se za određeni upit dohvaćaju relevantni dokumenti za taj upit, cilj odgovaranja na pitanja jest dohvatiti konkretan odgovor na postavljeni upit, a ne dokument ili dokumente u kojima se taj odgovor nalazi. Također, naglasak se stavlja na upite koji su postavljeni prirodnim jezikom, a ne na one koji se sastoje od ključnih riječi.

QA sustavi postoje od 1960-tih, kada su u SAD-u razvijena dva takva sustava: BASEBALL i LUNAR. Sustav BASEBALL odgovarao je na pitanja o SAD-ovoj baseball ligi, a LUNAR o geološkim analizama stijena sa mjeseca. Domena baze znanja tih prvobitnih sustava je veoma ograničena, a unutar svoje domene LUNAR sustav je točno odgovarao na 90% pitanja postavljenih od strane neizučenih korisnika. Godišnja konferencija o pretraživanju teksta (engl. *Text Retrieval Conference, TREC*) uključila je 1999. godine među svoje radionice i radionicu o odgovaranju na pitanja(Voorhees, 2000). Za sad TREC u radionici QA ocjenjuje kvalitetu sustava koji mogu odgovarati na pitanja čiji su odgovori činjenice, liste ili definicije. Godine 2004 najbolji sustav na TREC-u točno je odgovorio na 77% pitanja čiji je odgovor jedna činjenica. Posljednjih godina QA-sustavi se, osim na nekoliko osnovnih vrsta pitanja, sve više orijentiraju na druge vrste pitanja(npr. vremenski razmak, uzrok nečega, usporedbu dvaju entiteta u nekoj domeni, itd.), kao i na pitanja koja nisu povezana s tekstom nego s video i audio zapisima. TREC i dalje za tri navedena tipa pitanja pruža bazu pitanja i točnih odgovora, koja se svake godine mijenja.

QA-sustavi pretražuju odgovor na upit u određenoj bazi znanja, koja može biti više ili manje specijalizirana za neku domenu (medicina, tehničke znanosti,

podaci o stijenama na mjesecu) ili može biti domenski neovisna, znači sadržavati nedomensko, opće znanje. Težina izrade QA-sustava direktno ovisi o domeni i raspoloživim resursima unutar nje (ontologije, znanje specifično za domenu), pa su sustavi sa specifičnijom domenom i boljim resursima u načelu jednostavniji za izradu i kvalitetniji u davanju točnih odgovora. Također, baza znanja sustava može biti ručno izgrađena, zbirka tekstovnih dokumenata ili čak cijeli web. Izrada QA-sustava ovisi i o tome kako je napravljena baza znanja s obzirom da je ručno izgrađena baza pouzdanija nego baza sakupljena s weba te znanje u ručno izrađenoj bazi ima čvršću strukturu nego u bazi sakupljenoj s weba.

Kvalitetan QA-sustav mora moći riješiti mnogo problema koji se javljaju pri pretraživanju odgovora na upit. Upiti mogu biti višeznačni, gramatički netočno postavljeni, nepotpuni ili bez odgovora u bazi znanja. Sve ove probleme sustav treba biti sposoban riješiti na odgovarajući način, razriješiti višeznačnosti s obzirom na kontekst pitanja i baze znanja, prepoznati i ispraviti gramatičke pogreške te prepoznati da u bazi nema odgovora za dani upit. Problema pri pretraživanju odgovora na upit ima mnogo više nego što je ovdje navedeno, te razni QA-sustavi tim problemima pristupaju na razne načine.

Jedan od problema s kojim se susreće svaki QA-sustav jest prepoznavanje tipa pitanja. Pitanja mogu biti vremenska, načinska, činjenična, intervalna, itd. Prepoznavanje tipa pitanja je veoma bitno jer o tome ovisi i način dohvaćanja odgovora na pitanje.

QA-sustavi orijentirani su na dohvaćanje dijelova odgovora na upit i sintezu tih dijelova u cjeloviti odgovor. QA-sustav iz svoje baze znanja dohvaća sve relevantne činjenice koje mogu odgovoriti na postavljeni upit. Potom se te činjenice filtriraju, tj. odbacuju se činjenice koje ne odgovaraju kontekstu ili tipu upita, kao i one koje nisu dovoljno značajne za postavljeni upit. Iz tako filtriranih činjenica QA-sustav gradi odgovor na postavljeni upit.

Područje QA obuhvaća i područje pretraživanja često postavljanih pitanja. Pošto je QA-sustave jednostavnije napraviti kada je baza znanja sustava strukturirana, i pošto tada, takvi sustavi daju i kvalitetnije rezultate, prirodno je istražiti QA-sustave čija je baza znanja baza često postavljanih pitanja. Često postavljana pitanja imaju relativno čvrstu strukturu (zna se što je pitanje a što odgovor na pitanje) koja olakšava izradu QA-sustava. Također, često postavljana pitanja su često specifična za određenu domenu te je time znanje i domenski ograničeno. Naravno, ograničiti bazu znanja QA-sustava na bazu često postavljanih pitanja ima svojih prednosti i mana. Takvi sustavi će biti precizniji i lakši

za izradu, ali neće sadržavati odgovore na sva pitanja na koja bi mogao odgovoriti QA-sustav s odgovarajućom bazom znanja u nestrukturiranom obliku. Također, nestrukturirani tekstovni resursi su lakše dostupni nego često postavljana pitanja, koja k tome još valja i kvalitetno sakupiti, budući da se gubitkom njihove strukture gubi i jedna od prednosti ovog pristupa. Unatoč manama, baza koja se sastoji od često postavljanih pitanja u određenoj domeni često je dovoljno opširna da odgovori na većinu upita koje bi korisnici mogli postaviti sustavu.

2.1. Terminologija

Prije same razrade područja često postavljanih pitanja potrebno je utvrditi terminologiju koja će se koristiti u ostatku rada. U nastavku je dan pregled nekih često korištenih pojmova, koje je važno raspoznavati i međusobno razlikovati.

upit, korisnički upit – pitanje (upit) koje korisnik postavlja sustavu i na koje korisnik očekuje odgovor od sustava.

FAQ – (engl. *frequently asked question*), često postavljano pitanje.

FAQ – (engl. *frequently asked questions*), također, može se odnositi i na cijelu zbirku često postavljanih pitanja.

par, FAQ-par – par koji se sastoji od jednog pitanja i jednog odgovora na to pitanje.

pitanje, FAQ-pitanje – pitanje iz FAQ-para. Pojam *pitanje* koristit će se u smislu FAQ-pitanja a ne korisničkog upita.

odgovor, FAQ-odgovor – odgovor iz FAQ-para, odgovor na FAQ-pitanje iz FAQ-para

FAQ-zbirka, baza FAQ-parova – niz FAQ-parova. FAQ-zbirku za neko područje najčešće izrađuju stručnjaci za to područje. FAQ-parovi unutar zbirke su u biti pojedinačni dokumenti zbirke.

izvorna zbirka – FAQ-zbirka koja je sakupljena jednom ili više metoda iz neke baze FAQ-parova, te eventualno još i obrađena. Takva zbirka nije podatna za vrednovanje ili treniranje sustava, s obzirom da sadrži samo FAQ-parove, ali ne i vezu između FAQ-parova i korisničkih upita.

ispitna zbirka – zbirka koja sadrži i korisničke upite i FAQ-parove, te veze među njima.

matrica zbirke – rijetka matrica koja opisuje zbirku u vektorskom obliku. Svaki redak matrice predstavlja jedan dokument iz zbirke, a svaki stupac jednu

značajku. U i-tom retku i j-tom stupcu matrice nalazi se vrijednost j-te značajke i i-tog dokumenta.

2.2. Često postavljana pitanja

U raznim sustavima i na web-stranicama postalo je uobičajeno neke korisne i većem broju ljudi zanimljive informacije prikazati u obliku često postavljanih pitanja. Iako je u većini slučajeva veličina baze FAQ-parova za neki sustav ili web-stranicu relativno malena (do pedeset FAQ-parova), nezanemariv broj sustava i stranica sadrži baze od nekoliko stotina do nekoliko tisuća FAQ-parova, dok zajednica Usenet sadrži barem nekoliko stotina tisuća FAQ-parova. Za baze do pedesetak FAQ-parova za korisnika nije problem naći odgovarajući odgovor na upit koji ga zanima pregledavanjem svih FAQ-parova, ali već u slučaju od stotinjak FAQ-parova korisnik ne želi pročitati sva pitanja da bi našao odgovor, nego mu treba biti omogućeno pretraživanje baze FAQ-parova. Uobičajeno pretraživanje dokumenata po ključnim riječima oslanja se na činjenicu da dugački dokumenti sadrže velik broj riječi te je i mogućnost nalaženja ključnih riječi u relevantnom dokumentu relativno velika. FAQ-parovi, uobičajeno, sadrže relativno malen broj riječi te je vjerojatnost nalaženja ključnih riječi manja i potrebno je upotrijebiti druge načine pretraživanja.

Auto-FAQ i FAQ Finder jedni su od prvih sustava razvijenih za inteligentno pretraživanje FAQ-parova. Auto-FAQ sadrži vlastitu bazu FAQ-parova, a metode pretraživanja baze orijentirane su na ključne riječi uz ograničenu jezičnu obradu upita. FAQ Finder (Burke et al., 1995) sadrži bazu FAQ-parova koja je podijeljena u više datoteka od kojih svaka datoteka sadrži jednu kategoriju FAQ-parova, npr. kuhanje, Hrvatska, automobili. FAQ Finder koristi sustav za pretraživanje informacija SMART (Buckley, 1985) kako bi dohvatio relevantne datoteke za postavljeni korisnički upit te, nakon što korisnik odabere jednu od ponuđenih datoteka, FAQ-par koji odgovara na upit traži se unutar te datoteke. Za pretragu FAQ-para najsličnijeg korisničkom upitu FAQ Finder se koristi raznim metodama iz područja pretraživanja informacija koje uključuju i jezičnu obradu upita.

Jedan od pristupa pretraživanju FAQ-parova jest tretirati sakupljene FAQ-parove kao zbirku dokumenata za QA-sustav (Soricut i Brill, 2004). Umjesto da se traži najsličniji FAQ-par koji odgovara na korisnički upit, FAQ-parovi se koriste za slaganje odgovora na upit. Korisnički upit pretvara se u upit koji sustav za pretraživanje prepoznaje. Takav upit sastoji se od blokova (engl. *chunk*)

dobivenih plitkim statističkim parserom (engl. *chunker*) koji je treniran odgovorima u FAQ-parovima, kako bi se na taj način premostila razlika između upita i FAQ-parova. Upit sastavljen od blokova prosljeđuje se sustavu za pretraživanje koji sakuplja relevantne dokumente za upit. Sakupljeni dokumenti prosljeđuju se sustavu za filtriranje koji smanjuje opseg dokumenata na količinu koju je moguće obraditi te segmentira dobivenu zbirku na rečenice. Dobivena zbirka predaje se sustavu za ekstrakciju odgovora, koji iz nje dohvaća relevantne rečenice i iz njih slaže odgovor na postavljeni korisnički upit. Navedena metoda bliska je metodama pretraživanja iz područja QA, a FAQ-parove koristi samo kao bazu znanja za dohvaćanje odgovora.

Pretraživanju FAQ-parova pristupa se i na drukčiji način, u kojem se FAQ-parovi ne koriste za pretraživanje rečenica od kojih se može složiti odgovor na upit, već se cijeli FAQ-par prezentira kao odgovor na postavljeni upit. Da bi takvo pretraživanje bilo moguće, FAQ-parovi moraju ispunjavati dva uvjeta (Burke et al., 1997):

1. FAQ-par mora biti *potpun* – svaki FAQ-par mora u sebi sadržavati pitanje i odgovor na to pitanje

2. FAQ-par mora biti *lokalan* – svi bitni podaci za prepoznavanje relevantnog FAQ-para za neki upit moraju biti sadržani u samom FAQ-paru. To znači da jedan FAQ par sadrži potpuno pitanje (npr. "Koliko je jak novi Citroen C5?" jest potpuno pitanje, a "Koliko je jak taj automobil?", ako je negdje prije u zbirci FAQ-parova spomenut Citroen C5, nije potpuno pitanje) ili potpun odgovor (npr "Citroen C5 ima 138 konjske snage" jest potpun odgovor, a "138 konjskih snaga" nije potpun odgovor).

FAQ-parove koji ispunjavaju navedene upite moguće je pretraživati i metodama koje pliće analiziraju korisnički upit nego što je to potrebno kod QA-sustava te je njihovo kvalitetno pretraživanje moguće i kada nisu dostupni potrebni resursi za dubinsku analizu upita.

Da bi FAQ-parovi ispunjavali gore navedene uvjete potrebno ih je nakon sakupljanja ručno obraditi. Naknadna obrada može se izostaviti ako se ustanovi da je broj FAQ-parova koji ne ispunjavaju navedene uvjete beznačajno malen i da neće ometati kvalitetan rad sustava (Burke et al., 1997).

Pretraživanje pomoću ključnih riječi

Uobičajen pristup pretraživanju dokumenata pomoću ključnih riječi moguće je, uz određene modifikacije, primijeniti i na pretraživanje FAQ-parova. Modifikacije su potrebne jer su FAQ-parovi relativno kratki te uobičajeno pretraživanje ključnim riječima nije kvalitetno (Sneiders, 1999).

Metoda prioritetnog uspoređivanja ključnih riječi (Sneiders, 1999) radi na sljedeći način. Ključne riječi su podijeljene u četiri kategorije:

1. Obavezne ključne riječi – sadrže smisao upita te se ne smiju ignorirati;
2. Neobavezne ključne riječi – pobliže objašnjavaju smisao upita, ali se mogu ignorirati a da se smisao ne promijeni;
3. Nevažne ključne riječi – veznici, prijedlozi i sl., česte riječi koje nisu povezane sa smislom upita. Značenje tih riječi je slično zaustavnim riječima iz područja pretraživanja informacija, ali se ne mogu potpuno ignorirati jer mogu biti bitne u određenim situacijama prepoznavanja relevantnih FAQ-parova;
4. Zabranjene ključne riječi – pojava ovih ključnih riječi u upitu i FAQ-paru ukazuje da upit i par nisu kompatibilni (npr. u izrazima "Kako kupiti auto" i "Zašto kupiti auto" riječi *kako* i *zašto* su međusobno zabranjene ključne riječi).

Pri pretraživanju FAQ-zbirke iz korisničkog upita izdvajaju se obavezne, neobavezne, nevažne i zabranjene ključne riječi. Potom se svi FAQ-parovi koji ne sadrže obavezne ključne riječi proglašavaju irelevantnima za korisnički upit. Svi parovi koji sadrže zabranjene ključne riječi u odnosu na upit također se proglašavaju irelevantnima. Neobavezne ključne riječi u tako dobivenim parovima uspoređuju se s neobaveznim ključnim riječima iz upita te se izdvajaju parovi u kojima se one podudaraju. Preostale riječi izdvojenih parova uspoređuju se s preostalim riječima u upitu i na osnovu podudarnosti, s određenim faktorom podudarnosti, FAQ-parovi se proglašavaju relevantnima ili irelevantnima. Dobiveni relevantni FAQ-parovi prezentiraju se kao odgovor na postavljeni upit.

Da bi navedena metoda mogla raditi potrebna je FAQ-zbirka u kojoj su za svaki FAQ-par označene sve vrste ključnih riječi kao i neki njihovi sinonimi. Sinonimi su potrebni da bi se za postavljeni upit mogli dohvatiti i dokumenti koji nemaju identične riječi nego i njihove sinonime. Ovaj pristup pretraživanju FAQ-parova zahtjeva mnogo rada oko pripreme same FAQ-zbirke te je prirodno istražiti

drukčije metode koje koriste ključne riječi ili metode kod kojih nije potrebno označavati FAQ-parove.

Jedna od metoda koja automatski određuje ključne riječi i pretražuje FAQ-zbirku jest dinamičko pretraživanje FAQ-parova pomoću teorije približnih skupova (Chiu et al., 2007). Metoda se koristi alatom koji za određeni dokument automatski određuje ključne riječi. Za svaki FAQ-par u zbirci odrede se ključne riječi te se FAQ-parovi hijerarhijski grupiraju koristeći preklapanje ključnih riječi u pojedinim parovima kao mjeru njihove blizine. Potom se za korisnički upit pomoću teorije približnih skupova odredi kojoj grupi taj upit pripada. FAQ-parovi iz te grupe prezentiraju se kao odgovori na postavljeni upit. Ova metoda nema potrebe za označenom zbirkom FAQ-parova te je pokazano da ima bolje performanse nego metoda prioritetnog uspoređivanja ključnih riječi.

Pretraživanje često postavljanih pitanja metodama pretraživanja dokumenata

Razne su metode za pretraživanje dokumenata već etablirane u znanstvenoj zajednici te se zadnjih desetak godina istražuje mogućnost njihove primjene na pretraživanje često postavljanih pitanja. Neke od tih metoda se ne oslanjaju na pretraživanje po ključnim riječima već koriste razne modele usporedbe i prikaza upita i dokumenata koje se pretražuje.

Dva najčešće korištena modela pri pretrazi FAQ-zbirke jesu jezični model (Liu i Croft, 2004; Huo i Feng, 2004) i model vektorskog prostora (Kim i Seo, 2006; Kim et al., 2007; Burke et al., 1995; Song et al., 2007; Wu et al., 2006).

Jezični model računa vjerojatnost $P(Q|M_d)$ da je upit Q generiran jezičnim modelom M_d, gdje je Q upit a M_d jezični model dokumenta d. Najčešći način izračunavanja vjerojatnosti $P(Q|M_d)$ jest pretpostaviti da se upit Q može interpretirati kao skup nezavisnih pojmova q_i, tj. $Q = \{q_1, q_2, ..., q_n\}$ gdje je n broj pojmova iz upita Q. Tada se vjerojatnost $P(Q|M_d)$ računa kao:

$$P(Q|M_d) = \prod_{i=1}^{n} P(q_i|M_d),$$

gdje je q_i pojam upita Q, a $P(q_i|M_d)$ je zadan jezičnim modelom i može se računati kao:

$$P(q_i|M_d) = \lambda P_{ML}(q_i|d) + (1 - \lambda)P_{ML}(q_i|Col),$$

gdje se vjerojatnosti $P_{ML}(q_i|d)$ i $P_{ML}(q_i|_{Col})$ izračunavaju kao najvjerodostojnije procjene vjerojatnosti (engl. *maximum likelihood estimates, MLEs*) pojma q u dokumentu d i pojma q u zbirci $_{Col}$, a λ je faktor izglađivanja.

Vektorski model gradi vektore v_q za upit q i v_d za dokument d, te potom uspoređuje blizinu tih vektora. Najčešći način uspoređivanja upita i dokumenta je računanje kosinusa kuta između njihovih vektora

$$cos(q,d) = \frac{v_q \cdot v_d}{|v_q||v_d|}$$

Pri određivanju vektora upita i dokumenta pretpostavi se da su oboje nizovi nezavisnih pojmova te se određenom metodom, npr. TfIdf, određuju težine pojedinih elemenata vektora, npr. $v_q = (TfIdf(w_1, q), TfIdf(w_2, q), ..., TfIdf(w_n, q))$, gdje su $w_1, w_2, ..., w_n$ pojmovi upita (dokumenta) q, a n broj pojmova upita (dokumenta) q.

Oba navedena pristupa pokazala su se kvalitetnima u pretraživanju dokumenata, ali u pretraživanju FAQ-zbirke kvaliteta opada jer FAQ-pitanja sadrže mali broj riječi te postoji *leksički jaz* (Berger et al., 2000) između pojmova u upitu i pojmova u FAQ-pitanjima. Leksički jaz nastaje jer korisnici pri postavljanju upita ne koriste iste riječi koje se koriste u FAQ-zbirci, iako je moguće da su riječi iz upita i riječi u FAQ-pitanju, s kojim se upit uspoređuje, semantički slične (npr. upit "Koja je cijena A" i FAQ-pitanje "Koliko košta A" očito su smisaono ista pitanja, iako se leksički razlikuju). Premošćivanje leksičkog jaza između pojmova upita i pojmova FAQ-pitanja jedan je od glavnih problema pri pretraživanju FAQ-zbirke (Berger et al., 2000).

Berger et al. (2000) predlažu tri različite metode premošćivanja leksičkog jaza i za svaku pokazuju koliko pridonosi kvaliteti pretraživanja. Prva predložena metoda jest proširenje upita kod koje se na osnovu uzorka za treniranje traži veze između pojmova u upitima i pojmova u FAQ-parovima. Navodi se da ta metoda može prepoznati veze između riječi koje nisu direktno sinonimi, npr. zašto → zato, url → http, Hrvatska → Zagreb. Povezanost pojma iz upita i pojma iz FAQ-para računa se kao informacijska dobit između ta dva pojma. Kada korisnik postavi upit, upit se proširuje s nekoliko pojmova iz FAQ-parova koji za pojmove u upitu imaju najveću informacijsku dobit. Druga predložena metoda je statistički strojni prijevod FAQ-parova u korisničke upite. Ova metoda za upit q i dani odgovor a računa vjerojatnost da se a prevodi u q, $P(q|a)$. FAQ-parove rangiramo po vjerojatnosti da je upit nastao iz svakog od njih te dobijemo rangirane FAQ

parove po relevantnosti u odnosu na postavljeni upit. Treća predložena metoda je model latentne varijable kao što su LSA i PLSI. Umjesto vektorskog modela u kojem se elementi vektora računaju funkcijom TfIdf, model u kojem se elementi vektora računaju preko LSA ili PLSI može dati bolje rezultate i povezati neke semantičke informacije (Kim et al., 2007).

Pri premošćivanju leksičkog jaza koriste se i izglađivanje upita (Huo i Feng, 2004) te rječnik sinonima (npr. WordNet[1]) (Wu et al., 2006; Song et al., 2007; Burke et al., 1997). Za izglađivanje upita, prikupljeni upiti za treniranje se grupiraju te se prilikom pretraživanja FAQ-zbirke upit uspoređuje s pojedinačnim FAQ-pitanjima, ali i s grupom kojoj FAQ-pitanje pripada. Time se mogu premostiti leksičke razlike između upita i FAQ-pitanja. Za korištenje ječnika sinonima uvodi se nova mjera sličnosti upita i FAQ-pitanja, koja mjeri koliko su oni semantički slični. Navedena se mjera koristi u kombinaciji s nekom od metoda pretraživanja dokumenata i pokazano je da kombinirana metoda daje kvalitetnije rezultate (Wu et al., 2006; Burke et al., 1997).

Također, Kim i Seo (2006) koriste kombinaciju izglađivanja upita i korištenja rječnika sinonima. Pri izgradnji grupa za izglađivanje upiti koji se grupiraju uspoređuju se korištenjem rječnika sinonima, a prilikom pretraživanja vektorski model se izglađuje na ovako grupirane upite.

Daljnja povećanja kvalitete pretraživanja uglavnom se tiču uvođenja novih mjera koje u kombinaciji s navedenim pokrivaju područja pretraživanja koje osnovne mjere ne pokrivaju. Burke et al. (1997) uvode prekrivanje (engl. *coverage*) , koje mjeri koliko je pojmova u upitu pokriveno nekim pojmom u FAQ-pitanju s kojim se upit uspoređuje. Time se mjeri koliko su dobro bitni koncepti u upitu pokriveni FAQ-pitanjem.

Također, određuju se i tip upita i FAQ-pitanja (Tomuro i Lytinen, 2001; Wu et al., 2006) te se na osnovi određenih tipova za upit i FAQ-pitanje mjeri njihova sličnost. Lytinen i Tomuro (2002) postavljaju taksonomiju od 12 konceptualnih tipova pitanja. Tip pitanja se ne utvrđuje na osnovi upitnih riječi koje sadrži (tko, što, koliko, itd.), što bi bio intuitivan pristup, već se pitanja algoritmom k najbližih susjeda klasificiraju u jednu od 12 konceptualnih kategorija (vremenska pitanja, da/ne pitanja, načinska pitanja, pitanja o entitetu, itd.). Tip pitanja svakog FAQ-para određuje se ručno. U kasnijem radu (Tomuro i Lytinen, 2004) izgrađeno je stablo odluke koje klasificira i tip upita i tip pitanja FAQ-para.

[1] http://wordnet.princeton.edu/

Većina navedenih metoda korisnički upit uspoređuje s FAQ-pitanjem. Pošto FAQ-odgovori sadrže veću količinu teksta, moguće je da bi kombinirano uspoređivanje upita s FAQ-pitanjem i s FAQ-odgovorom dalo bolje rezultate nego uspoređivanje samo s FAQ-pitanjem (Tomuro i Lytinen, 2004).

Ispitne zbirke

Ispitne zbirke za sustave pretraživanja često postavljanih pitanja uglavnom su skupljene s weba. Veličina ispitne zbirke veoma varira, od 2,3 milijuna FAQ-parova (Soricut i Brill, 2004) pa do 406 FAQ-parova (Kim i Seo, 2006). Većina ispitnih zbirki ima između 500 i 2000 FAQ-parova, a veličina zbirke koju su koristili Soricut i Brill (2004) bila je potrebna zbog treniranja plitkog statističkog parsera. Također, sakupljaju se i korisnički upiti, najčešće preko weba, pomoću kojih se vrednuju sustavi.

U većini radova na području pretraživanja FAQ-parova označavanje relevantnih FAQ-parova za sakupljene korisničke upite vrše ručno sami autori. Jedino Wu et al. (2006) grade zbirku uz ocjene relevantnosti drugih ljudskih sudaca, dok Lytinen i Tomuro (2002) zbirku grade refraziranjem odabranih upita od strane ljudi koji nisu autori rada.

2.3. Pretraživanje često postavljanih pitanja na hrvatskom jeziku

Za hrvatski jezik, koliko je poznato, ne postoji niti jedan sustav za pretraživanje često postavljanih pitanja. Za pretraživanje FAQ-parova predlaže se metoda koja kombinira četiri različita pristupa: vektorski model TfIdf, korištenje rječnika sinonima, prekrivanje te određivanje tipa pitanja. Navedene metode su odabrane jer je za hrvatski jezik iz postojećih resursa moguće izgraditi resurse potrebne za spomenute metode, dok neki resursi već postoje. Za navedene metode, za razliku od opisanih pristupa, ispitati će se utjecaj pretraživanja FAQ-parova usporedbom upita i s FAQ-odgovorom, a ne samo s FAQ-pitanjem.

Također, ne postoji ispitna zbirka na hrvatskom jeziku kojom se može vrednovati sustav za pretraživanje FAQ-parova te je takvu zbirku potrebno izgraditi.

3. Metoda usporedbe upita i dokumenta

Prilikom dohvaćanja relevantnih dokumenata za određeni upit potrebno je dokumente iz zbirke rangirati po opadajućoj sličnosti s upitom. Tada se najviše rangirani dokument, ili nekoliko dokumenata rangiranih pri vrhu, dohvaćaju kao relevantni za postavljeni upit. Za potrebe rangiranja dokumenata razvijena je mjera sličnosti upita q i pojedinog dokumenta d kao linearna kombinacija četiri osnovne mjere: kosinus, semantička sličnost, prekrivanje i tip pitanja. Mjera je definirana kao:

$$\begin{aligned} Sim(q,d) = \; & A_q \cdot cos(q,d_q) + B_q \cdot semSim(q,d_q) + C_q \cdot cov(q,d_q) + D_q \cdot qSim(q,d_q) \\ & + A_a \cdot cos(q,d_a) + B_a \cdot semSim(q,d_a) + C_a \cdot cov(q,d_a), \end{aligned} \tag{3.1}$$

gdje je q korisnički upit, d je FAQ-par, d_q je FAQ-pitanje, d_a je FAQ-odgovor, a A_q, B_q, C_q, D_q, A_a, B_a, C_a su parametri koji određuju važnost određenog aspekta usporedbe.

Navedene su mjere odabrane tako da se svakom pokrije jedan aspekt usporedbe dokumenta koje druga ne pokriva, npr. kosinus mjera će dobro ocijeniti koliko identičnih riječi ima u upitu i dokumentu, ali neće uopće ocijeniti koliko ima semantički sličnih riječi, dok će semantička sličnost to dobro ocijeniti. Parametrima uz pojedinu mjeru u linearnoj kombinaciji namješta se kolika je važnost određene mjere (odnosno aspekta koji ta mjera pokriva) pri rangiranju dokumenata. Također, pošto se pretražuje FAQ-parove, upit se usporedi posebno s FAQ-pitanjem, a posebno s FAQ-odgovorom.

3.1. Mjera sličnosti kosinus

Kosinusna mjera sličnosti upita i dokumenta računa kosinus kuta između vektorskih prikaza upita i dokumenta:

$$cos(q, d) = \frac{v_q \cdot v_d}{|v_q||v_d|}, \tag{3.2}$$

gdje je v_q vektorska reprezentacija upita q, a v_d je vektorska reprezentacija dokumenta d. U slučaju kada su vektorske reprezentacije upita i dokumenta identične, $cos(q, d)$ jednak je 1, a u slučaju da se niti jedan element vektora upita i vektora dokumenta ne podudara, $cos(q, d)$ jednak je 0.

3.1.1. TfIdf shema

Za vektorski prikaz upita i dokumenata koristi se statistička metoda TfIdf. Vrijednost TfIdf sastoji se od dvije komponente: frekvencija pojma (engl. *term frequency*) Tf i inverzne frekvencije pojma u dokumentima (engl. *inverse document frequency*) Idf.

Frekvencija pojma dana je s:

$$Tf(w, d) = \frac{n_{wd}}{\sum_{k \in d} n_{kd}}, \tag{3.3}$$

gdje je n_{wd} broj pojavljivanja pojma w u dokumentu d, a $\sum_{k \in d} n_{kd}$ zbroj broja pojavljivanja svih pojmova u dokumentu d.

Inverzna frekvencija pojma dana je s:

$$Idf(w) = log(\frac{N}{n_{wz}}), \tag{3.4}$$

gdje je N broj dokumenata u zbirci, a n_{wz} broj dokumenata u zbirci u kojima se pojavljuje pojam w.

Vrijednost TfIdf računa se kao umnožak:

$$TfIdf(w, d) = Tf(w, d) \times Idf(w), \tag{3.5}$$

a vektorski prikaz dokumenta kao:

$$v_d = (\, TfIdf(w_1, d),\, TdIdf(w_2, d), ...,\, TfIdf(w_n, d)), \tag{3.6}$$

gdje su w_1, w_2, ..., w_n pojmovi koji se pojavljuju u zbirci, a v_d je vektorski prikaz dokumenta d (na isti način prikazuje se i upit q).

3.2. Semantička sličnost

Semantička sličnosti dviju riječi određuje se na osnovi semantičke udaljenosti tih riječi, $dist(w_1, w_2)$, koja se dohvaća iz semantičkog grafa. Semantički graf sadrži niz riječi i udaljenosti među njima, koje se računaju kao najmanji broj skokova (jedan skok je pretvorba riječi u njezin sinonim) po sinonimima između dvije riječi.

Ako su dvije riječi identične ($dist(w_1, w_2) = 0$) onda je semantička sličnost semWsim(w1,w2) tih riječi 1, a ako između dviju riječi ne postoje skokovi koji bi ih povezali ($dist(w_1, w_2) = \infty$), onda je semantička sličnost tih riječi 0:

$$semWsim(w_1, w_2) = \frac{1}{dist(w_1, w_2) + 1}. \tag{3.7}$$

Semantička sličnost upita i dokumenta određuje se na temelju preklapanja semantički sličnih riječi iz upita i dokumenta na sljedeći način (Song et al., 2007):

$$semSim(q, d) = \frac{1}{2}\left(\frac{\sum_{w_i \in q} maxsemWsim(w_i, d)}{|q|} + \frac{\sum_{a_i \in d} maxsemWsim(a_i, q)}{|d|}\right), \tag{3.8}$$

gdje je q upit, $|q|$ broj riječi u upitu q, d dokument, $|d|$ broj riječi u dokumentu d, w_i riječi iz upita q, a_i riječi iz dokumenta d, a $maxsemWsim(v_i, d)$ definiran kao:

$$maxsemWsim(v_i, d) = max(semWsim(v_i, a_1), semWsim(v_i, a_2), ..., semWsim(v_i, a_{|d|})),$$

gdje su $a_1, a_2, ..., a_{|d|}$ riječi dokumenta, odnosno upita, d.

Sličnost se izračuna dvostrano, tj. prvo se računa preklapanje riječi iz upita sa riječima iz dokumenta, pa obrnuto.

3.3. Prekrivanje

Prekrivanje (engl. *coverage*) je mjera koja računa za koji se postotak riječi iz upita pojavljuje semantički slična riječ u dokumentu koji rangiramo. Ovom mjerom saznajemo koliko dobro dokument pokriva važne pojmove iz upita. Prekrivanje je definirano na sljedeći način (Tomuro i Lytinen, 2004):

$$cov(q,d) = \frac{\sum_{w_i \in q} sgn(\sum_{a_j \in d} semWsim(w_i, a_j))}{|q|}, \tag{3.9}$$

gdje su $a_1, ..., a_{|d|}$ riječi iz dokumenta d, w_i riječi iz upita q, $|q|$ broj riječi u upitu q, $semWsim(w,a)$ dana je formulom (3.7), a $sgn(d)$ definiran kao:

$$sgn(d) = \begin{cases} 0 & , d \leq 0, \\ 1 & , d > 0. \end{cases}$$

3.4. Tip pitanja

Mjera sličnosti tipa pitanja iskazuje kolika je sličnost između tipa FAQ-pitanja i tipa pitanja postavljenog u upitu. Tipovi pitanja osmišljeni su tako da opisuju koja se vrsta odgovora očekuje za određeni tip pitanja. Npr., ako je pitanje "Tko može kupiti A", očekuje se odgovor koji sadrži neki entitet (u ovom slučaju neku osobu). Vrsta pitanja za FAQ-pitanje i za upit određuje se na temelju toga sadrži li pitanje, odnosno upit, neku ključnu riječ za tu vrstu pitanja. Svaki upit, odnosno FAQ-pitanje, može biti svrstan u više tipova pitanja ovisno o ključnim riječima koje sadrži. Ključne riječi koje su upitne zamjenice zadane su u muškom rodu, pošto se ključne riječi u ostala dva roda prije određivanja tipa pitanja pretvaraju u muški rod. Pregled tipova pitanja dan je tablicom 3.1.

Sličnost tipova FAQ-pitanja i pitanja u upitu određuje se na sljedeći način:

$$qSim(q, d_q) = \frac{\sum_{i=0}^{N} \sum_{j=0}^{M} qSim1\,(qType(q,i), qType(d_q,j))}{N \cdot M}, \tag{3.10}$$

gdje je q korisnički upit, d_q je FAQ-pitanje, N je broj tipova za pitanje u upitu, M je broj tipova za FAQ-pitanje, $qType(q,i)$ je i-ti tip pitanja u upitu, $qType(d_q,j)$ je j-ti tip za FAQ-pitanje, a $qSim1$ je dan tablicom 3.2. Ako se $qType(a,k)$ ne može utvrditi onda je $qSim1\,(qType(a,k),...) = qSim1\,(...,qType(a,k)) = 0$.

Tipovi pitanja i tablica sličnosti tipova pitanja određeni su heuristički te je onda tablica sličnosti mijenjana kroz par iteracija vrednovanja kako bi dala što kvalitetnije rezultate pretraživanja.

Tablica 3.1: Tipovi pitanja

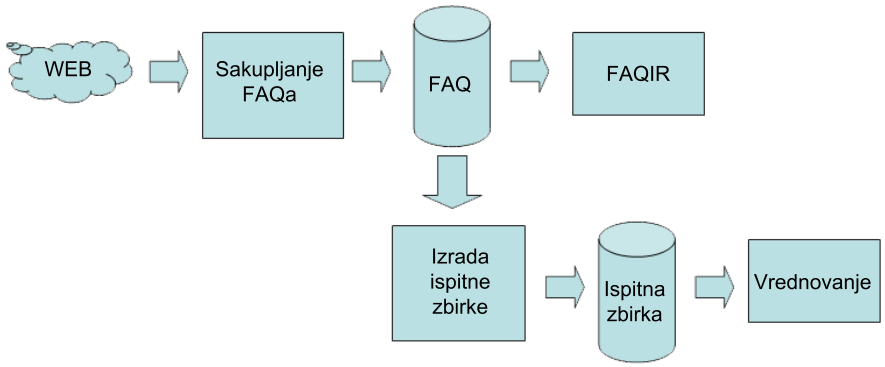
Skraćenica	Tip pitanja	Ključne riječi	Opis
DEF	Definicijsko	Što, Kakav	Odgovor na pitanje je definicija nekog pojma
ENT	Entitet	Tko, Kome, Kojem	Odgovor na pitanje je neki entitet
RAZ	Razlog	Zašto	Odgovor objašnjava razlog ostatka pitanja
NAC	Način	Kako	Odgovor objašnjava kako se nešto radi ili događa
VRM	Vrijeme	Kad, Kada	Odgovor na pitanje je trenutak u vremenu ili neko vremensko razdoblje
KOL	Količina	Koliko, Koliki	Odgovor na pitanje je množina nekog pojma
LOK	Lokacija	Gdje, Kuda	Odgovor na pitanje je lokacija
LIST	Lista	Koji, Kojim	Odgovor na pitanje je popis pojmova
LI	li-pitanje	Da li, Je li, <glagol> li	Pitanje je postavljeno na način "Da li ...", "Je li ...", "Mogu li ...", "Smijem li ...", itd.

Tablica 3.2: Tablica sličnosti tipova pitanja

	DEF	ENT	RAZ	NAC	VRM	KOL	LOK	LIST	LI
DEF	1,0	0,4	0,2	0,2	0,1	0,1	0,1	0,1	0,1
ENT	0,4	1,0	0,2	0,2	0,1	0,1	0,1	0,1	0,1
RAZ	0,2	0,2	1,0	0,8	0,1	0,1	0,1	0,2	0,1
NAC	0,2	0,2	0,8	1,0	0,1	0,1	0,1	0,1	0,2
VRM	0,1	0,1	0,1	0,1	1,0	0,1	0,1	0,1	0,1
KOL	0,1	0,1	0,1	0,1	0,1	1,0	0,1	0,3	0,1
LOK	0,1	0,1	0,1	0,1	0,1	0,1	1,0	0,1	0,1
LIST	0,1	0,1	0,2	0,1	0,1	0,3	0,1	1,0	0,1
LI	0,1	0,1	0,1	0,2	0,1	0,1	0,1	0,1	1,0

4. Implementacija

U okviru rada izgrađeno je pet sustava koji međusobno komuniciraju. Tri sustava - sustav za izgradnju indeksa i matrice zbirke, sustav za sakupljanje i obradu FAQ-zbirke te sustav za izgradnju ispitne zbirke - pomoćni su sustavi i njihov je zadatak izgraditi datoteke koje preostala dva sustava mogu koristiti. Sustav za vrednovanje i sustav za pretraživanje često postavljanih pitanja koriste datoteke izgrađene pomoćnim sustavima. Sustav za pretraživanje često postavljanih pitanja je ciljni sustav, dakle onaj kojemu će krajnji korisnici pristupati. Općeniti tok podataka kroz sustav prikazan je na slici 4.1.



Slika 4.1: Tok podataka u sustavu

Razvoj osmišljenih sustava diktiran je s nekoliko općenitih uvjeta koji su morali biti ispunjeni:

1. ciljni sustav i sustav za izgradnju ispitne zbirke morali su biti javno dostupni korisnicima,
2. svi sustavi morali su biti sposobni međusobno komunicirati,
3. razne strukture nekih sustava trebale su biti čovjeku lako prepoznatljive i pogodne za obrađivanje,

4. sustavi su morali biti sposobni sve svoje metode vršiti nad hrvatskim jezikom.

Pri implementaciji, mogućnosti ispunjavanja navedenih uvjeta su različite, ali nakon odmjeravanja dobrih i loših strana raznih mogućnosti doneseni su sljedeći zaključci prema kojima je implementacija sustava najjednostavnija i najbolje ispunjava navedene uvjete:

1. ciljni sustav i sustav za izgradnju ispitne zbirke biti će implementirani kao web-aplikacija – sustavi su javno dostupni korisnicima sustava;

2. za zapis raznih struktura u sustavima koristi se XML – XML omogućuje zapis struktura na jasno određen način, koji je neovisan o okruženju u kojem se implementira sustav. XML je tekstovni format i unutra njega potiče se da strukture budu logički definirane na apstraktnoj razini i da pokušaju opisati ono što sadrže što sličnije realnom svijetu. Zbog navedenog, XML-format čitljiv je čovjeku. Također, pošto je XML neovisan o okruženju, svi sustavi mogu koristiti strukture opisane XML-om te lako međusobno komunicirati;

3. za razne metode pretraživanja informacija (IR) u sustavima koristi se biblioteka TMT – sustavi koriste razne metode pretraživanja informacija te je potrebno te metode ili razviti ili koristiti neku biblioteku u kojoj su već implementirane. Biblioteka TMT implementira većinu potrebnih metoda zbog čega je ona korištena. Također, metode iz biblioteke TMT dobro rade s hrvatskim jezikom te biblioteka TMT za ulazne podatke prihvaća XML-format za zapis struktura. Pošto je biblioteka TMT korištena u više sustava, ti sustavi lako međusobno komuniciraju;

4. okruženje i programski jezik implementacije moraju imati dobru podršku za XML i nizove znakova – zapis struktura je u XML-u, a pošto se obrađuje mnogo tekstovnihs podataka, može se očekivati da će biti potrebno obavljati mnogo operacija na nizovima znakova.

Prema navedenim zaključcima odabrano je okruženje za implementaciju, kao i programski jezik. Microsoft .NET Framework 2.0 okruženje sadrži biblioteke za manipulaciju XML-dokumentom uz korištenje XPath[1] konstrukata koji olakšavaju dohvaćanje podataka iz XML-a. Također, XML-biblioteka unutar .NET Framework okruženja sadrži mnogo metoda za manipulaciju podacima u samom XML-dokumentu. Okruženje ima dobru podršku za manipulaciju nizovima znakova.

[1]XML Path Language, http://www.w3.org/TR/xpath

.NET Framework okruženje nudi izbor od četiri moguća programska jezika. S obzirom na uvjet da sustav mora biti realiziran kao web-aplikacija, izbor mogućih programskih jezika suzuje se na tri: Visual C#, Visual Basic i Visual J# (za Visual C++ nije podržana izrada Web aplikacija[2]). Biblioteka TMT napisana je u jeziku C++, ali sadrži i adapter za korištenje u programskom jeziku C#.

Zbog navedenih razloga odabrano je okruženje Microsoft .NET Framework 2.0 i programski jezik C#.

Opisivanje XML dokumenata vrši se na sljedeći način:

- sve što je između `<` i `>` je čvor u dokumentu, npr. `<document>`
- sve što je između `[` i `]` ponavlja se 0 ili više puta, npr. `[<category></category>]`
- ostatak teksta je sadržaj čvora u kojem se taj tekst nalazi i opisuje što bi se u tom čvoru trebalo nalaziti
- svi čvorovi koji se ne pojavljuju između `[` i `]` moraju se pojaviti točno toliko puta koliko su navedeni u opisu strukture.

4.1. Sustav za izgradnju indeksa i matrice zbirke

Sustav je zadužen za izgradnju indeksa odnosno matrice zbirke. Tri druga sustava koriste ovaj sustav očekujući različite podatke. Sustav za izgradnju ispitne zbirke očekuje i indekse i matricu zbirke dok sustavi za vrednovanje i pretraživanje često postavljanih pitanja očekuju samo matricu zbirke. Ovisno za koji sustav je potrebna izgradnja, izgrađuje se matrica zbirke te ako je potrebno i indeksi zbirke.

Ulazni podatak je zbirka FAQ-parova, a izlazni podaci su matrica zbirke te ako je potrebno i indeksi zbirke.

4.1.1. Izgradnja indeksa i matrice zbirke

Indekse zbirke gradi se pomoću metoda i klasa iz biblioteke TMT. Instancira se objekt iz klase definirane u TMT-u, koji služi za izgradnju indeksa, u koji se mogu dodati dokumenti za koje je potrebno izgraditi indekse. Svaki dokument se sastoji od naslova, sadržaja dokumenta te imena dokumenta, koje mora biti jedinstveno unutar zbirke dokumenta za koju se gradi indekse. Za svaki FAQ-par

[2]Moguće je izraditi dijelove Web aplikacije u C++ tako da se izradi datoteka DLL te se onda napravi adapter za neki drugi jezik.

iz zbirke dodaje se po jedan dokument u navedeni objekt tako da FAQ-pitanje predstavlja naslov dokumenta, FAQ-odgovor predstavlja sadržaj dokumenta, a identifikacijski broj FAQ-para[3] predstavlja ime dokumenta. Metode iz TMT-a za pretraživanje dokumenata po izgrađenim indeksima omogućuju pretraživanje posebno po naslovu i posebno po sadržaju dokumenta te se navedenim načinom pretvaranja FAQ-parova u dokumente u drugim sustavima može vršiti pretraživanje samo FAQ-pitanja ili samo FAQ-odgovora. Nakon dodavanja svih dokumenata u objekt, pokreće se metoda za izgradnju indeksa te se izgrađeni indeksi spremaju u binarnu datoteku.

Biblioteka TMT također gradi i matricu zbirke. Imena čvorova u XML-dokumentima koji sadrže zbirke su odabrana tako da ih TMT prepoznaje te za izgradnju matrice zbirke nije potrebno predprocesirati zbirku FAQ-parova. Matrica zbirke se izgradi posebno za FAQ-pitanja, posebno za FAQ-odgovore te posebno za cijele FAQ-parove, kako bi drugi sustavi mogli pretraživati FAQ-zbirku samo po pitanjima, odgovorima ili FAQ-parovima. XML-dokument koji sadrži zbirku FAQ-parova prosljeđuje se kao ulazni podatak metodama TMT-a za izgradnju matrice zbirke te TMT automatski izgradi matrice i sprema ih u binarnu datoteku.

4.1.2. Izgradnja semantičkog grafa

Sustav za izgradnju indeksa i matrice zbirke također je zadužen i za izgradnju semantičkog grafa. Iz hrvatsko-hrvatskog rječnika u elektronskom obliku prikupe se informacije o sinonimima svih riječi koje se pojavljuju u zbirci za koju se gradi semantički graf (Šarić, 2006). Sinonimi riječi koriste se za inicijalizaciju algoritma koji gradi semantički graf. Za svaku riječ i sve sinonime te riječi u semantički graf zapiše se, za udaljenost riječi i sinonima, udaljenost jedan. Za jednu se riječ pokrene Dijkstrin algoritam koji pronalazi najkraći put preko sinonima između dane riječi i svih ostalih riječi u grafu. Algoritam se pokrene za svaku riječ u grafu te tako dobivene udaljenosti među riječima predstavljaju semantičku udaljenost riječi. Izgrađeni graf koristi se u jednom od načina pretraživanja FAQ-zbirke u sustavu za vrednovanje i sustavu za pretraživanje često postavljanih pitanja.

[3]Objašnjenje što predstavlja identifikacijski broj za svaku zbirku nalazi se u narednim poglavljima.

4.2. Sustav za sakupljanje i obradu često postavljanih pitanja

Sustav je zadužen za automatsko sakupljanje FAQ-parova s weba i obradu sakupljenih parova. Obrada je potrebna zbog mogućnosti dvostrukih i nepotpunih FAQ-parova. Nepotpuni FAQ-parovi su oni koji sadrže FAQ-pitanje ali ne sadrže odgovor i obratno.

Ulazni podaci su HTML-stranice na kojima se nalaze FAQ-parovi, a izlazni podaci su FAQ-zbirka zapisana u XML-formatu u tekstovnoj datoteci te indeksi i matrica FAQ-zbirke zapisani u binarnim datotekama.

4.2.1. Sakupljanje FAQ-parova

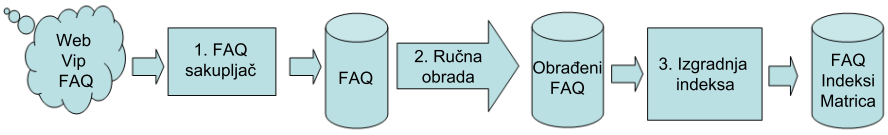
Na slici 4.2 prikazani su koraci pri sakupljanju FAQ-parova. FAQ-parovima u FAQ-bazi, iz koje se sakupljaju FAQ-parovi, pristupa se preko web-stranice na kojima se FAQ-parovi prikazuju korisnicima. Skupe se sve web-stranice koje sadrže FAQ-parove te se obrađuju. Pri obradi web-stranica iterira se kroz HTML-elemente te se za svaki element koji, prema definiranim pravilima za određenu stranicu, sadrži FAQ-par izvuku FAQ-pitanje i odgovor. U svakom pitanju odnosno odgovoru eliminiraju se svi HTML-tagovi koji ne sadrže bitne informacije o strukturi FAQ-pitanja odnosno odgovora, npr.
 tag. Svi HTML-tagovi koji sadrže neki podatak o strukturi, npr. tag, ostavljaju se u pitanju odnosno odgovoru, jer mogu biti značajni za kasniju analizu FAQ-parova.

Tako sakupljeni FAQ-pitanja i odgovori spremaju se kao FAQ-parovi u strukturu koja sadrži sve sakupljene FAQ-parove. Potom se iz strukture eliminiraju svi identični FAQ-parovi, dakle svi FAQ-parovi koji imaju isto i pitanje i odgovor te svi nepotpuni FAQ-parovi. Dobivena struktura se zapisuje u XML-dokument, koji se, ako je potrebno, još i ručno obradi.

Obrađeni XML-dokument prosljeđuje se sustavu za izgradnju indeksa. Dobiveni indeksi i matrica FAQ-zbirke te XML-dokument koji sadrži FAQ-parove zapisuju se u tri zasebne datoteke. Te tri datoteke su izlaz sustava za sakupljanje i obradu često postavljanih pitanja.

4.2.2. Struktura XML-dokumenta koji sadrži FAQ-zbirku

Struktura XML-dokumenta koji sadrži FAQ-zbirku je sljedeća:



Slika 4.2: Shema sustava za automatsko sakupljanje i obradu FAQ-zbirke

```
<documentSet>
    <document>
        <categories>
            [<category>Ime kategorije</category>]
        </categories>
        <content>
            <Question>
                FAQ pitanje
            </Question>
            <Answer>
                FAQ odgovor
            </Answer>
        </content>
    </document>
    [
    <document>
        <categories>
            [<category>Ime kategorije</category>]
        </categories>
        <content>
            <Question>
                FAQ pitanje
            </Question>
            <Answer>
                FAQ odgovor
            </Answer>
        </content>
    </document>
    ]
</documentSet>
```

U daljnjem tekstu navedeni su opisi čvorova, koje atribute moraju sadržavati, koji su im čvorovi djeca, koji im je čvor roditelj te opisi pojedinih atributa. Kategorije FAQ-parova su kategorije koje su definirane u FAQ-bazi iz koje su FAQ-parovi skupljeni.

`<documentSet>` – obavezni atributi: name, description
– čvorovi djeca: `<document>`
– čvor roditelj: -
– opis: korijenski čvor XML-dokumenta
– atribut name: ime zbirke, pomoću ovog atributa raspoznaju se različite zbirke i razne verzije jedne zbirke
– atribut description: pobliže objašnjava odakle je zbirka skupljena i eventualno još informacija ako je potrebno

`<document>` – obavezni atributi: name, id
– čvorovi djeca: `<categories>`, `<content>`
– čvor roditelj: `<documentSet>`
– opis: sadrži jedan FAQ-par i kategorije kojima taj FAQ-par pripada
– atribut name: ime FAQ-para, ako FAQ-par ima ime u bazi u kojoj je sakupljen onda se to stavi u ovaj atribut, ako nema imena u bazi stavi se `FAQ_<id atribut>`
– atribut id: identifikacijski broj FAQ-para, jedinstven unutar XML-dokumenta

`<categories>` – obavezni atributi: -
– čvorovi djeca: `<category>`
– čvor roditelj: `<document>`
– opis: sadrži kategorije kojima FAQ-par pripada

`<category>` – obavezni atributi: category_id
– čvorovi djeca: -
– čvor roditelj: `<categories>`
– opis: sadrži opis jedne od kategorija kojoj FAQ-par pripada
– atribut category_id: identifikacijski broj kategorije

`<content>` – obavezni atributi: -
– čvorovi djeca: `<Question>`, `<Answer>`
– čvor roditelj: `<document>`
– opis: sadrži pitanje i odgovor FAQ-para

`<Question>` – obavezni atributi: -
– čvorovi djeca: -
– čvor roditelj: `<content>`
– opis: sadrži FAQ-pitanje

`<Answer>` – obavezni atributi: -
– čvorovi djeca: -
– čvor roditelj: `<content>`
– opis: sadrži FAQ-odgovor

Primjer XML-dokumenta koji sadrži FAQ-zbirku nalazi se u dodatku A.

4.3. Sustav za izgradnju ispitne zbirke

Sustav je zadužen za izgradnju ispitne zbirke na osnovi ocjena relevantnosti koje daju ljudski suci. Pošto sustav treba biti dostupan većem broju korisnika, realiziran je kao web-aplikacija u obliku web-stranice. Sučelje sustava je uobičajeno za tražilice, dakle pri vrhu web-stranice nalaze se polja za upis upita a nakon pretraživanja odgovori se ispisuju ispod postavljenih upita. Također, na samom vrhu stranice nalaze se postavljeni upiti kako bi korisnici pri ocjenjivanju mogli vidjeti koje su upite postavili.

Ulazni podaci su FAQ-zbirka, indeksi FAQ-zbirke, matrica FAQ-zbirke, koris-

nički upiti i ocjene relevantnosti odgovora na korisničke upite, a izlazni podatak je ispitna zbirka.

4.3.1. Način rada sustava za izgradnju ispitne zbirke

Na slici 4.3 prikazani su koraci pri izradi ispitne zbirke. Pri pokretanju sustav učitava FAQ-zbirku, matricu zbirke te indekse zbirke. Nakon što su navedeni podaci učitani sustav je spreman za rad.

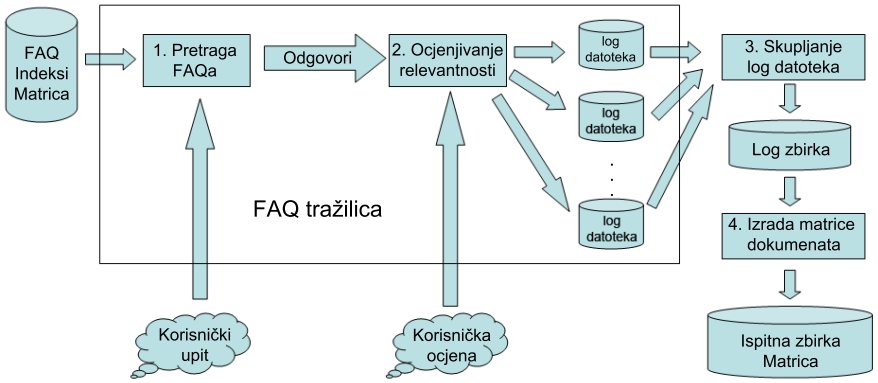
Prilikom pristupanja sustavu korisnici prvo unose korisničko ime i lozinku, koji su potrebni da bi u kasnijim koracima bilo poznato koji je korisnik postavio upite i ocijenio odgovore na upite. Nakon prijave na sustav korisnik postavlja upite te pokreće pretraživanje FAQ-zbirke. Rezultati pretraživanja predočavaju se korisniku te on ocjenjuje redom sve predočene odgovore na upit i svaki relevantan odgovor označuje. Kada završi s ocjenjivanjem odgovora na postavljeni upit, korisnik pokreće spremanje upita i relevantnih odgovora u dnevnik (engl. *log datoteka*) za tog korisnika. Također, sprema se i vrijeme koje je korisniku bilo potrebno da označi relevantne odgovore, kako bi se lakše detektiralo moguće nepouzdano ocjenjivanje. U svakom trenutku, od strane administratora sustava, moguć je pregled korisničkih log datoteka, tj. upita i označenih relevantnih odgovora.

Korisnici, nakon postavljanja svih željenih upita i ocjenjivanja odgovora, obavijeste administratora sustava da su gotovi s ocjenjivanjem. Administrator, nakon što ga svi korisnici obavijeste da su gotovi, pokreće skupljanje svih dnevnika u jedan skupni dnevnik svih korisnika. Podatak o tome koji su korisnici proizveli određene upite i relevantne odgovore na te upite ostaje sačuvan u skupnom dnevniku za kasnije analize. Skupni dnevnik sprema se u XML-dokument u tekstovnoj datoteci.

4.3.2. Struktura XML-dokumenta koji sadrži ispitnu zbirku

Struktura XML-dokumenta koji sadrži ispitnu zbirku je sljedeća:

```
<log>
    [
    <log_entry>
        <user_questions>
            <user_question>
```



Slika 4.3: Shema izrade ispitne FAQ-zbirke

```
            Korisnički upit
        </user_question>
        [
        <user_question>
            Korisnički upit
        </user_question>
        [
    </user_questions>
    <answer_time>
        Vrijeme potrebno za označavanje odgovora
    </answer_time>
    </selected_answers>
     [
     <document>
          .
          .
          .
     </document>
     ]
    </selected_answers>
  </log_entry>
  ]
</log>
```

U daljnjem tekstu navedeni su opisi čvorova, koje atribute moraju sadržavati, koji su im čvorovi djeca, koji im je čvor roditelj te opisi pojedinih atributa.

`<log>` –obavezni atributi: -
–čvorovi djeca: `<log_entry>`
–čvor roditelj: -
–opis: korijenski čvor XML-dokumenta

`<log_entry>` – obavezni atributi: username, empty
– čvorovi djeca: `<user_questions>`, `<answer_time>`, `<selected_answers>`
– čvor roditelj: `<log_entry>`
– opis: sadrži podatke o jednom ocjenjivanju relevantnosti odgovora na postavljene upite
– atribut username: korisničko ime korisnika koji je postavio upite i ocijenio odgovore
– atribut empty: može imati dvije vrijednosti true i false, naznačuje da li je korisnik označio da nema odgovora na postavljene upite

`<user_questions>` – obavezni atributi: -
– čvorovi djeca: `<user_questions>`
– čvor roditelj: `<log_entry>`
– opis: sadrži postavljene upite

`<user_questions>` – obavezni atributi: -
– čvorovi djeca: -
– čvor roditelj: `<user_questions>`
– opis: sadrži jedan postavljeni upit

`<answer_time>` – obavezni atributi: -
– čvorovi djeca: -
– čvor roditelj: `<log_entry>`
– opis: sadrži vrijeme koje je korisniku bilo potrebno da označi relevantne odgovore

`<selected_answers>` – obavezni atributi: -
– čvorovi djeca: `<document>`
– čvor roditelj: `<log_entry>`
– opis: sadrži odgovore na postavljene korisničke upite

`<document>` – opis: identičan je čvoru `<document>` opisanom u 4.2.2. Struktura XML-dokumenta koji sadrži FAQ-zbirku

Primjer XML-dokumenta koji sadrži ispitnu zbirku nalazi se u dodatku A.

4.4. Sustav za vrednovanje

Sustav je zadužen za vrednovanje sustava za pretraživanje često postavljanih pitanja. Vrednovanje sustava temelji se na nekoliko mjera koje se u detalje razrađuju u poglavlju 5.2. . Pomoću tih mjera zaključujemo koliko je kvalitetan izgrađeni sustav za pretraživanje često postavljanih pitanja.

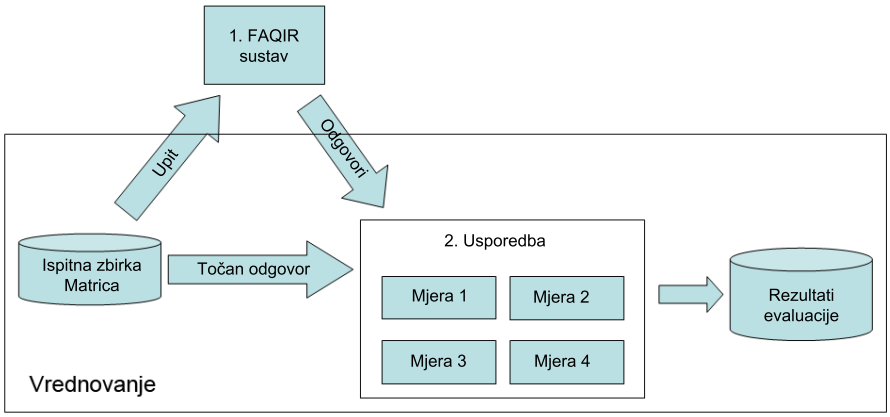
Ulaznih podataka nema jer je izgrađena ispitna zbirka dio podataka u sustavu, a izlazni podatak je XML-dokument s rezultatima vrednovanja.

4.4.1. Način rada sustava za vrednovanje

Na slici 4.4 prikazan je općeniti način rada sustava za vrednovanje. Iz ispitne zbirke odabere se prvi upit i FAQ-parovi koji odgovaraju na taj njega te se upit šalje sustavu za pretraživanje često postavljanih pitanja. Povratna informacija je niz rangiranih FAQ-parova za koje je sustav za pretraživanje zaključio da najbolje

odgovaraju na postavljeni upit. Rangirani parovi i točni odgovori iz ispitne zbirke uspoređuju se pomoću više različitih mjera. Rezultat uspoređivanja se zapiše u memoriju te se dohvaća sljedeći upit i FAQ-par i cijeli se proces ponavlja za svaki zapis u ispitnoj zbirci. Sustav za pretraživanje često postavljanih pitanja omogućava sustavu za vrednovanje namještanje parametara pri pretraživanju, kako bi se moglo provesti vrednovanje za razne kombinacije parametara sustava za pretraživanje.

Svi rezultati uspoređivanja zapisani u memoriji koriste se za izračunavanje vrijednosti raznih mjera za vrednovanje sustava. Vrijednosti mjera zapisuju se u XML-datoteku koja sadrži podatke o vrednovanju.



Slika 4.4: Shema sustava za vrednovanje

4.4.2. Struktura XML-dokumenta koji sadrži podatke o vrednovanju

Struktura XML-dokumenta koji sadrži podatke o vrednovanju je sljedeća:

```
<evaluation>
    <measure>
        [ <value /> ]
    </measure>
    [
    <measure>
        [ <value /> ]
    </measure>
    ]
```

```
</evaluation>
```

U daljnjem tekstu navedeni su opisi čvorova, koje atribute moraju sadržavati, koji su im čvorovi djeca, koji im je čvor roditelj te opisi pojedinih atributa.

`<evaluation>`
– obavezni atributi: name
– čvorovi djeca: `<measure>`
– čvor roditelj: -
– opis: korijenski čvor XML-dokumenta
– atribut name: ime vrednovanja, služi za raspoznavanje raznih metoda vrednovanja

`<measure>`
– obavezni atributi: name
– mogući atributi: value
– čvorovi djeca: `<value>`
– čvor roditelj: `<evaluation>`
– opis: sadrži rezultate vrednovanja za određenu mjeru
– atribut name: ime vrednovanja, služi za raspoznavanje raznih metoda vrednovanja
– atribut value: vrijednost izračunate mjere, ako je jedan broj

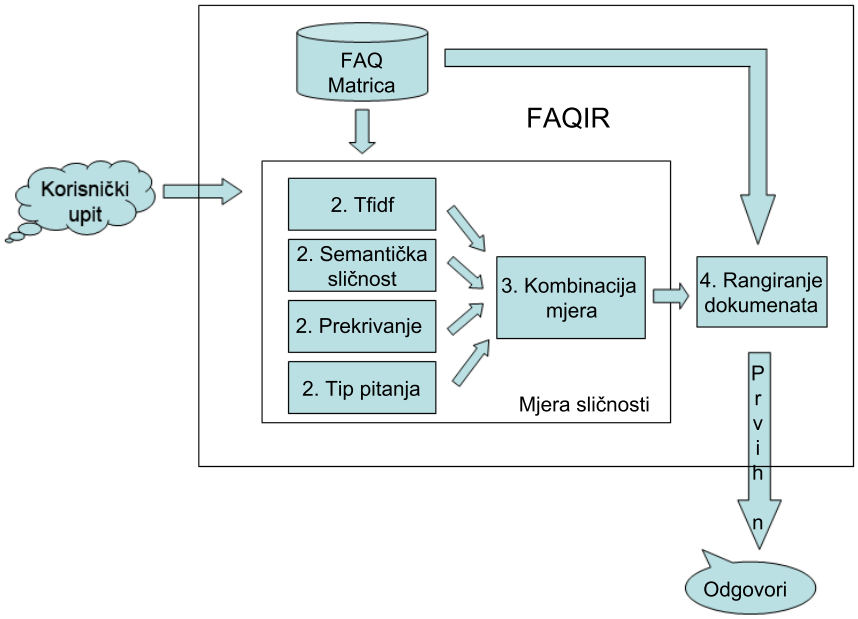
`<value>`
– obavezni atributi: -
– čvorovi djeca: -
– čvor roditelj: `<measure>`
– opis: sadrži podatak o jednoj vrijednosti jedne mjere, ako mjera ima više vrijednosti (npr. za crtanje grafa)

Primjer XML-dokumenta koji sadrži podatke o vrednovanju nalazi se u dodatku A.

4.5. Sustav za pretraživanje često postavljanih pitanja – FAQIR

Sustav za pretraživanje često postavljanih pitanja – FAQIR - je ciljni sustav, tj. svi prethodni sustavi su zapravo koraci u izgradnji FAQIR-a. Sustavom za sakupljanje i obradu često postavljanih pitanja sakupljena je zbirka FAQ-parova koju FAQIR pretražuje. Sustav za izgradnju ispitne zbirke izgradio je zbirku koja se koristi u sustavu za vrednovanje. Pomoću sustava za vrednovanje namješteni su razni parametri FAQIR-a u cilju što veće pouzdanosti i točnosti.

Ulazni podatak u sustav je korisnički upit a izlazni podatak je niz FAQ-parova koji odgovaraju na postavljeni upit.



Slika 4.5: Shema sustava FAQIR

Način rada FAQIR-a prikazan je slikom 4.5. Korisnik pristupi sustavu, upiše upit i pokrene pretraživanje. Postavljeni upit uspoređuje se sa svim FAQ-parovima iz FAQ-zbirke te, kombinacijom četiri mjere sličnosti, FAQIR daje svoju ocjenu sličnosti upita i svakog FAQ-para. Parametri pri kombinaciji mjera sličnosti utvrđeni su empirijski prilikom vrednovanja sustava.

FAQ-parovi u zbirci poredaju se silazno prema izračunatim ocjenama sličnosti te FAQIR korisniku prikazuje određen broj najviše rangiranih odgovora na postavljeni upit. Broj FAQ-parova koji se prikazuju korisniku izračunava se na temelju

empirijski dobivenih parametara prilikom vrednovanja sustava. Na osnovu dobivenih parametara sustav korisniku prikazuje samo relevantne odgovore na njegov upit. Cilj nije, kao u uobičajenim tražilicama, prikazati sve dokumente koji mogu biti relevantni za upit, nego točno one odgovore koji odgovaraju na postavljeni upit. Ako FAQIR u svojoj zbirci nema odgovor na postavljeni upit, prepoznaje da odgovora nema, te o tome obavještava korisnika .

Nakon prikazanih odgovora korisnik može ponovno postaviti upit FAQIR-u te se ponavlja cijeli proces pretraživanja zbirke FAQ-parova.

5. Vrednovanje

Razvijeni sustav za pretraživanje često postavljanih pitanja koristi kombinaciju četiri različite mjere za usporedbu sličnosti korisničkog upita i FAQ-parova, na osnovu koje se određuje koji FAQ-parovi najbolje odgovaraju na korisnički upit. U znanstvenoj zajednici ispitane su sve te mjere (Kim i Seo, 2006; Tomuro i Lytinen, 2004; Burke et al., 1995), kao i kombinacije nekih mjera (Kim i Seo, 2006; Huo i Feng, 2004; Tomuro i Lytinen, 2004; Burke et al., 1997; Song et al., 2007) i pokazana je njihova učinkovitost. Konkretna metoda koju se predlaže nije ispitana te je potrebno provesti vrednovanje sustava. Također, zanimljivo je usporediti mjere sličnosti međusobno, kao i različite kombinacije mjera te vidjeti koje u kombinaciji daju bolje rezultate, a koje se preklapaju, tj. u kombinaciji i svaka posebno daju sumjerljive rezultate.

Da bi se moglo provesti vrednovanje sustava bilo je potrebno izgraditi ispitnu zbirku. Ispitna zbirka sastoji se od trojki: korisnički upit – FAQ-parovi – ocjene relevantnosti FAQ-parova. Za svaki korisnički upit u zbirci ljudski su suci ocijenili koji su FAQ-parovi, kojih može biti više od jednog, relevantni za taj upit, tj. odgovaraju na postavljen korisnički upit. Za sve druge FAQ-parove, koje suci nisu ocijenili relevantnima, pretpostavlja se da ne sadrže odgovor na taj konkretan korisnički upit.

Sljedeća poglavlja daju pregled izrade ispitne zbirke te korištenja iste pri vrednovanju sustava.

5.1. Izrada ispitne zbirke

Izrada ispitne zbirke provedena je u dvije faze: sakupljanje i obrada često postavljanih pitanja te izrada ispitne zbirke iz izvorne zbirke.

5.1.1. Sakupljanje i obrada često postavljanih pitanja

U prvoj je fazi automatski sakupljeno 1334 FAQ parova iz online korisničke VIP-ove FAQ-baze[1]. Svaki FAQ-par sadrži pitanje i jedan odgovor na to pitanje. Potom su uklonjeni svi duplikati FAQ-parova iz FAQ-zbirke te je dobivena zbirka od 1222 jedinstvenih FAQ-parova. Od tih je FAQ-parova 500 parova bilo potrebno obraditi, jer nisu bili lokalni ili potpuni, a jedan je par, kojeg nije bilo moguće preraditi, eliminiran iz zbirke.

Dobivena izvorna zbirka sastoji se od 1221 FAQ-para. FAQ-pitanja koja se sastoje od samo jedne rečenice ima 1082, a višerečeničnih pitanja ima 139. Višerečenična pitanja su npr. oblika: "Korisnik sam usluge A. Mogu li koristiti uslugu B?" ili "Korisnik sam usluge A. Koliko košta korištenje usluge A?". Detaljni podaci o jezičnim karakteristikama izvorne zbirke prikazani su u tablici 5.1.

Tablica 5.1: Jezične karakteristike izvorne zbirke

	Min. riječi	Max. riječi	Prosječno riječi	Upitni oblik	Izjavni oblik
pitanja	2	63	12	1202	19
odgovori	1	252	42	–	–

5.1.2. Izrada ispitne zbirke iz izvorne zbirke

Izrada ispitne zbirke provedena je u dva koraka. U prvom koraku deset studenata FER-a bilo je zaduženo sastaviti barem 12 upita za koje misle da bi bili postavljeni od strane korisnika VIP-ovih usluga. Napomenuto im je da ne čitaju ništa iz izvorne FAQ-zbirke, kako bi postavljeni upiti bili što originalniji. Također, tako postavljeni upiti bolje odražavaju upite koje bi korisnici usluge postavili, pošto korisnici ne žele pretraživati zbirku pitanja, nego samo što brže dobiti odgovor na postavljeni upit. Studenti se nisu smjeli međusobno konzultirati o upitima koje postavljaju za sve vrijeme trajanja izrade ispitne zbirke.

U drugom koraku studenti su trebali osmišljene upite refrazirati (Lytinen i Tomuro, 2002) na prosječno pet, najmanje tri, a najviše deset načina. Jedan od načina refraziranja trebao je biti refraziranje u višerečenični upit. Također,

[1] www.vipnet.hr/cw/show?idc=8724663

poželjno je bilo da se verzije upita razlikuju po obliku, ali i da imaju isti oblik samo s nekim riječima promijenjenim u sinonime te je poželjno bilo napraviti i jedan izjavni oblik upita. Dopuštena je bilo koja kombinacija navedenih načina refraziranja. Bilo je bitno da se refraziranjem upita ne mijenja, ne proširuje niti suzuje smisao originalnog upita, pošto su se originalni i refrazirani upiti ocjenjivali zajednički. U tablici 5.2 navedeni su primjeri mogućih refraziranja jednog upita.

Tablica 5.2: Refraziranje upita

Koja je cijena razgovora u roamingu?	–originalni upit
Koja je cijena pričanja u inozemstvu? Koja je cijena razgovora u inozemstvu?	–zamjena riječi sinonimima
Kolika je cijena roaminga?	–promjena strukture rečenice
Koliki je trošak razgovora u inozemstvu?	–promjena strukture i zamjena sinonimom
Cijena razgovora u roamingu.	–izjavni upit
Trošak pričanja u inozemstvu.	–izjavni upit i zamjena sinonimima
Korisnik sam vaše mobilne mreže. Koja je cijena kada pričam u roamingu?	–višerečenični upit, manja promjena strukture, zamjena sinonimom
Korisnik sam tarife Model 50. Koji je trošak razgovora u inozemstvu?	–pogrešno refraziranje, suzuje smisao originalnog pitanja
Da li mogu koristiti roaming ako sam korisnik tarife Model 200?	–pogrešno refraziranje, mijenja smisao originalnog upita

Tako osmišljeni upiti koristili su se kao upiti za razvijeni sustav za pretraživanje FAQ-zbirke. Nakon postavljanja upita, sustav bi pretražio izvornu zbirku te vratio između 50 i 150 FAQ-parova. Upit i FAQ-par iz izvorne zbirke su se uspoređivali na tri načina: sličnost upita i FAQ-pitanja, sličnost upita i FAQ-odgovora i sličnost upita i cijelog FAQ-para. Pretraživanje se vršilo pomoću neko-

liko metoda (Tf, TfIdf, jezični model s izglađivanjem, pretraživanje fraza, pretraživanje po ključnim riječima, proširenje upita sinonimima) te se svaka metoda koristila uz variranje različitih mogućih parametara za tu metodu (razne metode izračunavanja TfIdf vrijednosti, razni faktori izglađivanja, varijacija koji dio upita je fraza, varijacija koje ključne riječi moraju biti uključene u odgovor a koje ne). Cilj ovakvog pretraživanja nije bio dohvatiti što relevantniji odgovor što bliže vrhu liste odgovora, nego pokušati dohvatiti sve odgovore koji bi na bilo koji način mogli biti relevantni za dane upite. Također, relevantni odgovori nisu rangirani po relevantnosti nego slučajnim redoslijedom, kako bi se izbjeglo pridodavanje veće pažnje više rangiranim odgovorima. Sve navedene metode upotrijebljene su kako bi se čim više povećala mogućnost dohvaćanja svih relevatnih odgovora za dane upite. Studenti su sve odgovore pročitali te svaki binarno ocijenili, odgovara li na postavljene upite ili ne. Ova metoda izrade ispitne zbirke, u kojoj ljudski suci za niz upita i odgovora na te upite daju binarne ocjene relevantnosti pojedinih odgovora, ustaljena je u izradi ispitne zbirke (Wu et al., 2006; Voorhees i Tice, 2000).

Ovako sakupljeni upiti, odgovori na upite i ocjene relevantnosti čine ispitnu zbirku koja se koristi za treniranje i vrednovanje sustava za pretraživanje često postavljanih pitanja. Odgovor na jedan upit može biti više FAQ-parova. Ispitna zbirka sadrži 419 postavljenih upita i 526 FAQ-parova koji odgovaraju na te upite. Pošto jedan FAQ-par može biti odgovor na više korisničkih upita, ispitna zbirka sadrži više identičnih FAQ-parova, svaki kao odgovor na drugi upit. Eliminiranjem duplikata za svaki FAQ-par iz ispitne zbirke saznajemo da ispitna zbirka sadrži 291 jedinstveni FAQ-par, dakle pokriva 23,8% izvorne zbirke od 1221 FAQ-parova. Od tih 291 jedinstvenih FAQ-parova, 259 FAQ-pitanja sastoji se samo od jedne rečenice, a 32 FAQ-pitanja je višerečenično, dok jednorečeničnih upita ima 362 a višerečeničnih 57. U ispitnoj zbirci nalaze se 92 upita bez odgovora, a važni su za vrednovanje jer sustav treba prepoznati da u svojoj FAQ-zbirci ne sadrži odgovore na neke upite (Lytinen i Tomuro, 2002). Detaljni podaci o jezičnim karakteristikama ispitne zbirke nalaze se u tablici 5.3, a podaci o razdiobi korisničkih upita u tablici 5.4.

Prema prezentiranim podacima vidljivo je da su jezične karakteristike izvorne i ispitne zbirke sumjerljive te da ispitna zbirka dobro predstavlja izvornu zbirku često postavljanih pitanja. Također, broj upita po FAQ-paru i broj FAQ-parova u ispitnoj zbirci je dostatan te se pomoću izgrađene ispitne zbirke može vrednovati sustav.

Tablica 5.3: Jezične karakteristike ispitne zbirke

	Min. riječi	Max. riječi	Prosječno riječi	Upitni oblik	Izjavni oblik
Pitanja	4	63	7	287	4
Upiti	1	25	8	372	47
Odgovori	1	218	30	–	–

Tablica 5.4: Razdioba korisničkih upita

Odgovorenih	Bez odgovora	Jednostavnih	Složenih
327	92	362	57
Upita po FAQ-paru	FAQ-parova po upitu		
1,44	1,26		

5.2. Mjere za vrednovanje

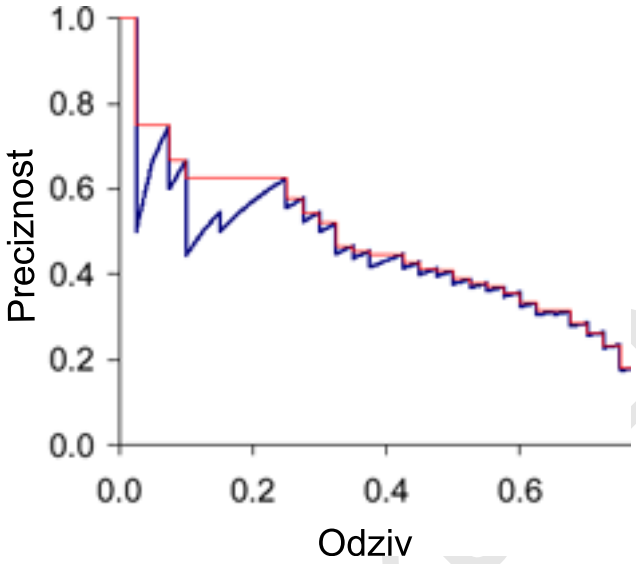
Uobičajene mjere za vrednovanje pretraživanja informacija - preciznost, odziv i F mjera - mjere su koje su predviđene za vrednovanje skupa dokumenata. Za potrebe vrednovanja rangiranih dokumenata navedene se mjere ne mogu upotrijebiti na isti način kao i za skup dokumenata te uvodimo drukčiji način prezentiranja rezultata tih mjera i neke druge mjere.

5.2.1. Preciznost i odziv

Rezultat pretraživanja rangiranih dokumenata smisleno je predstaviti kao prvih k dokumenata s najvišim rangom. Za svaki skup od takvih k dokumenata, tj. za k od nula do ukupnog broja dokumenata, možemo izračunati preciznost i odziv. Pomoću dobivenih vrijednosti preciznosti i odziva crtamo graf preciznost-odziv, koji nam prikazuje kako se te vrijednosti kreću s obzirom na broj dohvaćenih dokumenata.

Graf preciznost-odziv najčešće ima pilasti oblik (slika 5.1. Ako $k+1$ dokument nije relevantan, preciznost se smanjuje u odnosu na preciznost kada se dohvati samo k dokumenata, dok odziv ostaje isti. Ako $k+1$ dokument jest relevantan, tada i preciznost i odziv rastu. Ovo ponašanje je bitno jer se može pretpostaviti da je važnije korisniku prikazati nešto više dokumenata s većom preciznošću umjesto

manje dokumenata uz manju preciznost.



Slika 5.1: Graf preciznost-odziv (vidi Manning et al., 2007, str. 113)

Točke grafa preciznost-odziv izračunavaju se za svaki upit pri vrednovanju, te se potom skupni graf preciznost-odziv računa kao srednja vrijednost preciznosti za pojedine upite.

Interpolirana srednja preciznost u 11 točaka

Graf preciznost-odziv je veoma informativan utoliko što daje dobar pregled situacija u kojima se isplati povećati broj dohvaćenih dokumenata. Ipak, zanimljivo je pokušati sve te informacije skupiti u nekoliko brojeva ili u samo jedan. Uobičajen način da se to napravi jest izračunati interpoliranu srednju preciznost u 11 točaka (Manning et al., 2007). Interpolirane vrijednosti preciznosti vide se na slici 5.1 crtane crvenim vodoravnim crtama. Interpolirana preciznost u nekoj točki grafa je najviša vrijednost preciznosti za odziv veći ili jednak od odziva u toj točci grafa. U 11 točaka grafa za vrijednosti odziva 0.0, 0.1, 0.2, . . . , 1.0 izmjerimo interpoliranu preciznost te dobivamo interpoliranu preciznost u 11 točaka. Potom za te vrijednosti izmjerimo srednju vrijednost preko svih upita i dobijemo interpoliranu srednju preciznost u 11 točaka.

Podaci o interpoliranoj preciznosti u 11 točaka za graf preciznost-odziv sa slike 5.1 nalaze se u tablici 5.5.

Tablica 5.5: Interpolirana preciznost u 11 točaka (vidi Manning et al., 2007, str. 114)

Odziv	Interpolirana preciznost
0,0	1,00
0,1	0,67
0,2	0,63
0,3	0,55
0,4	0,45
0,5	0,41
0,6	0,36
0,7	0,29
0,8	0,13
0,9	0,10
1,0	0,08

5.2.2. Srednja vrijednost prosjeka preciznosti

Srednja vrijednost prosjeka preciznosti (engl. *mean average precision, MAP*) je mjera koja opisuje kvalitetu sustava preko svih vrijednosti odziva.

MAP se računa na sljedeći način. Izračunamo preciznost za jedan korisnički upit za svaki k za koji je k-ti dokument, u rangiranom poretku dokumenata, relevantan za dani upit. Potom računamo prosjek tako izračunatih preciznosti za svaki upit te izračunamo srednju vrijednost dobivenih prosjeka preciznosti.

$$MAP = \frac{1}{|Q|} \sum_{q_j \in Q} \frac{1}{m_j} \sum_{i=1}^{m_j} Preciznost(R_{im_j}),$$

gdje je Q skup upita, q_j jedan upit iz skupa upita Q, m_j broj relevantnih dokumenata za upit q_j, a $Preciznost(R_{im_j})$ preciznost za prvih i rangiranih dokumenata.

5.2.3. R-preciznost

R-preciznost za jedan upit računa preciznost za prvih Rel rangiranih dokumenata, gdje je Rel broj relevantnih dokumenata za dani upit. R-preciznost za sve upite se računa kao srednja vrijednost svih R-preciznosti za pojedine upite.

$$RPrecision = \frac{\sum_{q_i \in Q} RPrecision1(q_i)}{|Q|}, \tag{5.1}$$

gdje je Q skup upita, q_i jedan upit iz skupa Q, $|Q|$ broj upita iz skupa Q, a $RPrecision1(q_i)$ je preciznost dohvaćenih dokumenata kada je broj dohvaćenih dokumenata jednak broju relevantnih dokumenata za upit q_i.

5.2.4. Srednji recipročni rang

Srednji recipročni rang (engl. *mean reciprocal rank, MRR*) je mjera za vrednovanje sustava koji prezentiraju rangirane rezultate. Recipročni rang relevantnog dokumenta za jedan upit je recipročni rang najviše rangiranog relevantnog dokumenta za taj upit. MRR se računa kao srednja vrijednost recipročnih rangova za svaki upit.

$$MRR = \frac{1}{|Q|} \sum_{q_j \in Q} \frac{1}{toprank_{q_j}},$$

gdje je Q skup upita, q_j jedan upit iz skupa Q, a $toprank_{q_j}$ rang najviše rangiranog relevantnog dokumenta za upit q_j.

5.2.5. Odbacivanje

Odbacivanje (engl. *rejection*) je mjera kojom se mjeri koliko uspješno sustav može zaključiti da u svojoj zbirci dokumenata nema dokumenta koji odgovara na postavljeni upit. Odbacivanje se računa kao postotak upita za koje ne postoji odgovor u zbirci, a sustav je točno zaključio da odgovor ne postoji. O odbacivanju ima smisla govoriti samo pri određenoj vrijednosti preciznosti i odziva, pošto sustav koji niti za jedan upit uopće ne prezentira niti jedan odgovor ima odbacivanje od 100%.

$$rejection = \frac{\sum_{q_j \in Q} rejectionS(q_j) \cdot rejectionZ(q_j)}{\sum_{q_j \in Q} rejectionZ(q_j)},$$

gdje je Q skup upita, a q_j jedan upit iz skupa Q,

$$rejectionS(q_j) = \begin{cases} 1 & \text{, ako sustav odluči da za } q_j \text{ nema relevantnih dokumenata,} \\ 0 & \text{, inače.} \end{cases}$$

$$rejectionZ(q_j) = \begin{cases} 1 & \text{, ako za } q_j \text{ nema relevantnih dokumenata u zbirci,} \\ 0 & \text{, inače.} \end{cases}$$

Također, odbacivanje se određuje uz fiksni iznos mjere sličnosti dva dokumenta. Svi dokumenti iznad granica odbacivanja, tj. oni za koje mjera sličnosti daje veći rezultat od granice odbacivanja, prezentiraju se kao relevantni, a svi ispod granice, pa makar su svi dokumenti za određeni upit ispod granice, odbacuju se. Granica odbacivanja utvrđuje se eksperimentalno, namještajući je tako da dobijemo karakteristike sustava koje želimo.

Graf odbacivanja

Graf odziv-odbacivanje (Burke et al., 1997) prikazuje koliki je odziv pri određenom odbacivanju, te omogućuje namještavanje sustav između prezentiranja određenog broja relevantnih dokumenata i prezentiranja praznog skupa za upite za koje ne postoje relevantni dokumenti. Informativni su i grafovi MAP-odbacivanje, MRR-odbacivanje i R-preciznost-odbacivanje jer također omogućuju namještanje sustava između kvalitete prezentiranja točnih odgovora i kvalitete prezentiranja praznog skupa dokumenata kada za upit nema relevantnih odgovora u zbirci.

5.3. Vrednovanje sustava

Vrednovanje sustava vrši se nad dvjema ispitnim zbirkama. Izgrađena ispitna zbirka razdvoji se na dvije zbirke i to tako da jedna sadrži sve upite za koje je označeno da postoji odgovor u izvornoj zbirci (u daljnjem tekstu ispitna zbirka 1), a druga sve upite iz ispitne zbirke (u daljnjem tekstu ispitna zbirka 2). Ispitna zbirka 1 koristi se za vrednovanje sustava na osnovi preciznosti, odziva, srednje vrijednosti prosjeka preciznosti, R-preciznosti i srednjeg recipročnog ranga, a ispitna zbirka 2 koristi se za vrednovanje sustava na osnovi odbacivanja.

Vrednovanje sustava provodi se kroz četiri eksperimenta. U prvom eksperimentu mjeri se odnos četiri osnovne metode mjerenja sličnosti, TfIdf, semantička sličnost, prekrivanje i tip pitanja. Sve osnovne metode u ovom eksperimentu mjere sličnost upita i FAQ-pitanja, a ne upita i cijelog FAQ-para. U drugom eksperimentu za tri osnovne metode, TfIdf, semantička sličnost i prekrivanje, mjeri se koliko na kvalitetu pretrage utječe uspoređivanje upita s FAQ-pitanjem, a koliko kombinacija uspoređivanja upita s FAQ-pitanjem i FAQ-odgovorom. U

trećem eksperimentu utvrđuje se koliko određene kombinacije mjera pridonose kvaliteti pretraživanja. Kao osnovna metoda, s kojom se sve druge uspoređuju, određen je TfIdf. Ispituje se koliko svaka od osnovnih metoda, u kombinaciji s TfIdf-om, kao i kombinacija svih osnovnih mjera, pridonose kvaliteti pretraživanja. U četvrtom eksperimentu mjeri se odbacivanje za sustav, tj. koliko uspješno sustav prepoznaje da u svojoj bazi znanja ne sadrži odgovor na neki upit. Rezultati sva četiri eksperimenta prezentirani su u sljedećem poglavlju, a detaljniji podaci o vrednovanju nalaze se u dodatku B.

5.4. Rezultati vrednovanja

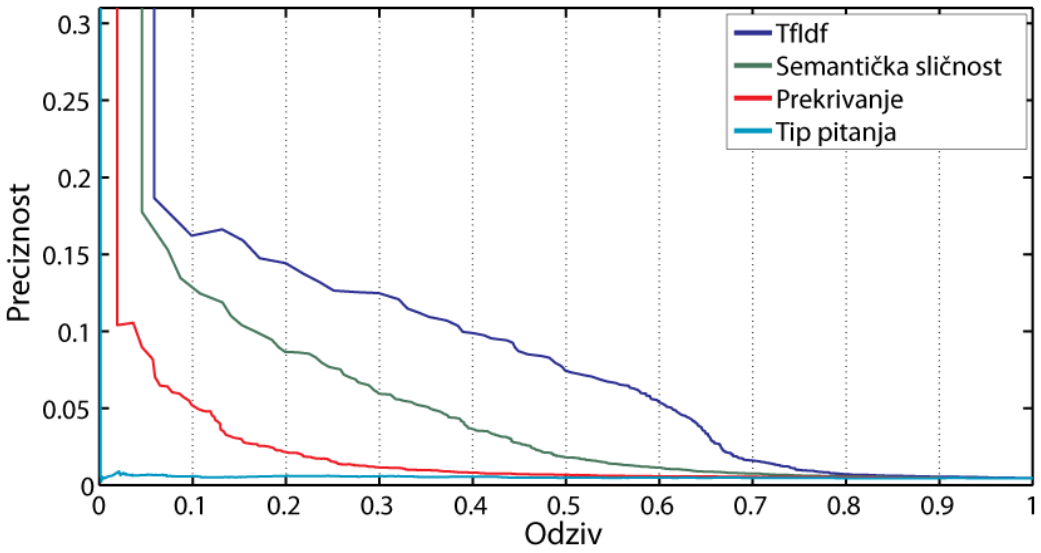
Eksperiment 1: osnovne mjere

U prvom eksperimentu uspoređene su četiri osnovne metode mjerenja sličnosti upita i FAQ-para. Sve osnovne metode određuju sličnost upita i FAQ-para na osnovi sličnosti upita i FAQ-pitanja. Dobiveni rezultati prikazani su u tablici 5.6 i na grafu na slici 5.2.

Tablica 5.6: Osnovne metode

	MAP(%)	R-preciznost(%)	MRR
TfIdf	19,32	15,28	0,3152
Semantička sličnost	13,51	10,55	0,2634
Prekrivanje	6,78	5,94	0,1663
Tip pitanja	1,22	0,52	0,0240

Metoda TfIdf, za MAP i R-preciznost, daje za barem 50% bolje rezultate nego ostale metode. MRR prikazuje prosječnu vrijednost recipročne vrijednosti ranga prvog dohvaćenog relevantnog FAQ-para te 1/MRR možemo interpretirati kao prosječni rang prvog dohvaćenog relevantnog FAQ-para. Vrijednost 1/MRR za TfIdf jednaka je 3,17 što znači da je prvi relevantni FAQ-par u prosjeku dohvaćen među prva četiri prezentirana, a u većini slučajeva i među prva tri prezentirana FAQ-para. Tip pitanja daje slabe rezultate, što je i očekivano, jer se u zbirci nalazi puno FAQ-pitanja s istim tipom te se samo na osnovi vrste pitanja ne može dobro zaključiti koja su FAQ-pitanja relevantna za dani upit. Također, tip pitanja se određuje plitkom analizom teksta, na osnovi upitnih ključnih riječi te se dva pitanja, koja se semantički razlikuju, mogu svrstati u istu kategoriju.



Slika 5.2: Osnovne metode

Dubljom analizom teksta može se bolje utvrditi tip pitanja te takvom analizom teksta tip pitanja daje bolje rezultate (Tomuro i Lytinen, 2001; Lytinen i Tomuro, 2002).

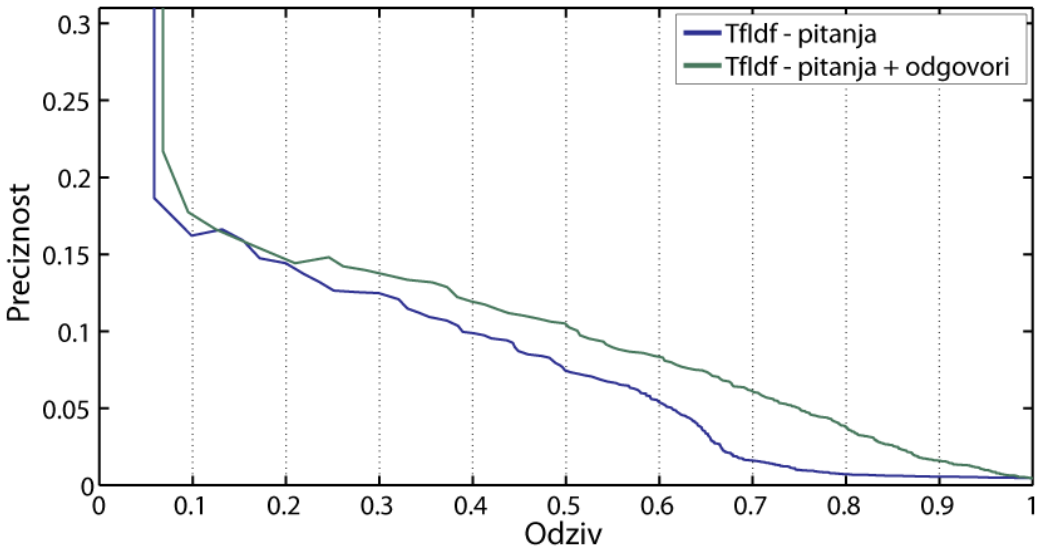
Eksperiment 2: kombinacija pretraživanja po FAQ-pitanju i FAQ-odgovoru

Drugim eksperimentom određuje se korisnost kombinacije uspoređivanja upita s FAQ-pitanjem i upita s FAQ-odgovorom. Za tri metode, TfIdf, semantička sličnost i prekrivanje, rezultati vrednovanja su prezentirani u tablici 5.7 i na grafovima na slikama 5.3, 5.4 i 5.5. Faktori koji određuju koliki je utjecaj FAQ-pitanja, a koliki FAQ odgovora utrđeni su eksperimentalno i navedeni su u tablici B1 u dodatku B.

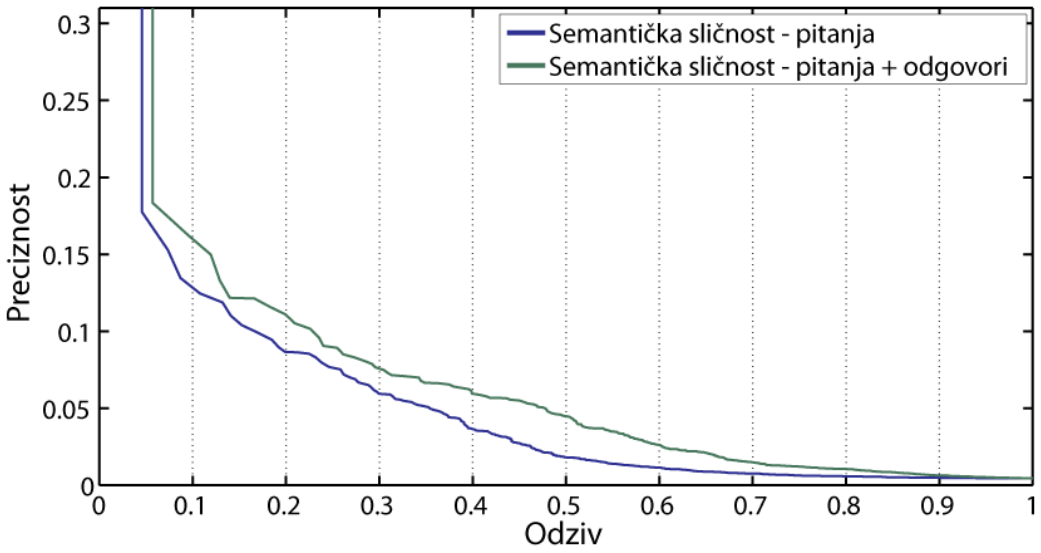
Tablica 5.7: Osnovne metode – pitanja i odgovori

	MAP(%)	R-preciznost(%)	MRR
TfIdf	21,77	15,28	0,3407
Semantička sličnost	16,11	12,28	0,2869
Prekrivanje	9,71	8,22	0,2007

Prema prezentiranim rezultatima, sve tri metode daju bolje rezultate kombinacijom uspoređivanja nego uspoređivanjem upita samo s FAQ-pitanjem. TfIdf

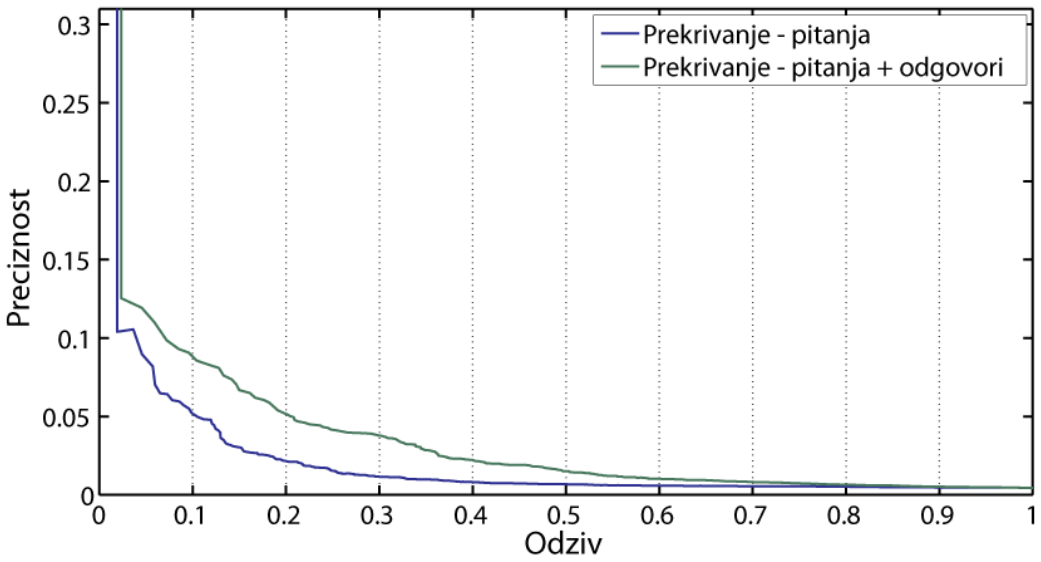


Slika 5.3: TfIdf – pitanja i odgovori



Slika 5.4: Semantička sličnost – pitanja i odgovori

za MAP daje 12,7% bolje rezultate, a za MRR daje 8,1% bolje rezultate. Vrijednost R-preciznosti se nije promijenila što znači da je prvi relevantni dohvaćeni FAQ-par više rangiran, a ostali relevantni FAQ-parovi su niže rangirani nego u slučaju kada upit uspoređujemo samo s FAQ-pitanjem. Semantička sličnost za MAP daje 19,24% bolje rezultate, za R-preciznost daje 16,4% bolje rezultate, a za MRR daje 8,2% bolje rezultate. Prekrivanje za MAP daje 43,3% bolje rezultate,



Slika 5.5: Prekrivanje – pitanja i odgovori

za R-preciznost daje 38,4% bolje rezultate, a za MRR daje 20,7% bolje rezultate. Također, prema prezentiranim grafovima vidljivo je da sve metode daju veću vrijednost preciznosti pri istoj vrijednosti odziva za kombinaciju uspoređivanja. Prema prezentiranim podacima, kombinacija uspoređivanja daje značajno bolje rezultate nego uspoređivanje samo s FAQ-pitanjem.

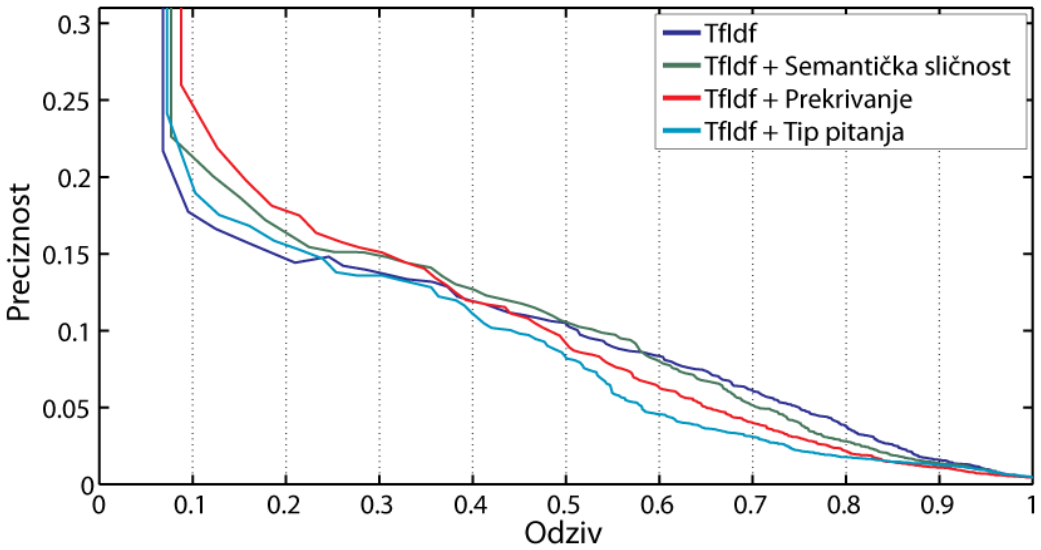
Eksperiment 3: kombinacije mjera

U trećem eksperimentu metoda TfIdf(pitanja+odgovori) se koristi kao osnovna metoda s kojom se uspoređuje značajnost kombiniranja TfIdf-a s drugim metodama. U prvom djelu eksperimenta vrednuje se kombinacija metode TdIdf sa semantičkom sličnošću, prekrivanjem i tipom pitanja te su rezultati vrednovanja prezentirani u tablici 5.8 i na grafu na slici 5.6. U drugom dijelu eksperimenta vrednuje se kombinacija svih metoda te su rezultati prikazani u tablici 5.9 i na grafu na slici 5.7. Pojedini faktori u formuli (3.1) određeni su eksperimentalno i navedeni su u tablici B1 u dodatku B.

Kombinacija TfIdf-a i semantičke sličnosti za MAP daje 6,4% bolje rezultate, za R-preciznost daje 15% bolje rezultate, a za MRR daje 4,8% bolje rezultate. Kombinacija TfIdf-a i prekrivanja za MAP daje 7,1% bolje rezultate, za R-preciznost daje 14% bolje rezultate, a za MRR daje 13% bolje rezultate. Također, prema prezentiranim grafovima vidljivo je da pri nižim vrijednostima odziva obje

Tablica 5.8: Kombinacija metoda

	MAP(%)	R-preciznost(%)	MRR	1 / MRR
TfIdf	21,77	15,28	0,3407	2,9351
TfIdf + semantička sličnost	23,16	17,56	0,3570	2,8011
TfIdf + prekrivanje	23,32	17,43	0,3852	2,5960
TfIdf + tip pitanja	20,90	15,93	0,3530	2,8328



Slika 5.6: Kombinacija metoda

kombinacije metoda daju veću vrijednost preciznosti nego samo metoda TfIdf. Veća preciznost pri nižim odzivima te veća vrijednost MRR-a bitni su za sustav koji vraća mali broj FAQ-parova, jer će se tada prvi relevantni FAQ-par pojaviti više rangiran na listi prezentiranih FAQ-parova. Veća vrijednost R-preciznosti i MAP-a pri kombinaciji metoda upućuje na to da su svi relevantni dokumenti više rangirani nego kod metode TdIdf.

Kombinacija TfIdf-a i tipa pitanja za R-preciznost daje 4,3% bolje rezultate, za MRR daje 3,6% bolje rezultate, dok za MAP daje 4% lošije rezultate. Iz prezentiranog grafa vidljivo je da samo pri niskim vrijednostima odziva kombinacija daje veću vrijednost preciznosti nego sama metoda TfIdf. Neznatno bolji rezultati za R-preciznost i MRR te lošiji rezultat za MAP rezultat su načina određivanja tipa pitanja. Kao što je već navedeno, tip pitanja se određuje plitkom analizom teksta te bi uz dublju analizu teksta i bolje određivanje tipa pitanja i

kombinacija dala bolje rezultate.

Kombinacija TfIdf i semantičke sličnosti odnosno prekrivanja daje, prema prezentiranim podacima, značajno bolje rezultate nego sama metoda TfIdf.

Prema rezultatima prezentiranim u tablici 5.9 i na grafu na slici 5.7, vidljivo je da kombinacija svih metoda daje bolje rezultate nego sama metoda TfIdf, ali u usporedbi s kombinacijom TfIdf-a i prekrivanja daje neznačajno bolje rezultate za MAP (1,3%) i R-preciznost (2,2%) te čak i lošiji rezultat za MRR (4%). Navedeno je posljedica preklapanja aspekta semantičke sličnosti i prekrivanja. Obje metode se temelje na usporedbi sinonima s time da je za prekrivanje potrebno znati koje su riječi međusobno mogući sinonimi, a za semantičku sličnost je potrebno znati i koliko su semantički bliske dvije riječi. Mogući razlog preklapanja aspekata navedenih metoda jest korišteni rječnik i metoda izračuna semantičke udaljenosti dvije riječi u mreži sinonima te bi se preklapanje moglo izbjeći uz korištenje kvalitetnije mreže sinonima.

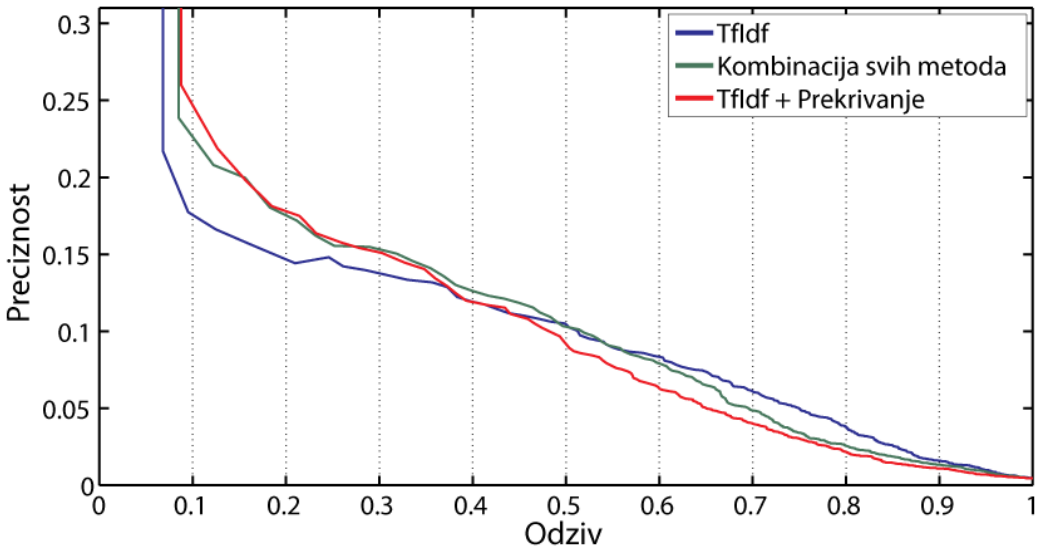
Tablica 5.9: Kombinacija svih metoda

	MAP(%)	R-preciznost(%)	MRR
TfIdf	21,77	15,28	0,3407
TfIdf + prekrivanje	23,32	17,43	0,3852
Kombinacija svih metoda	23,63	17,82	0,3701

Eksperiment 4: odbacivanje

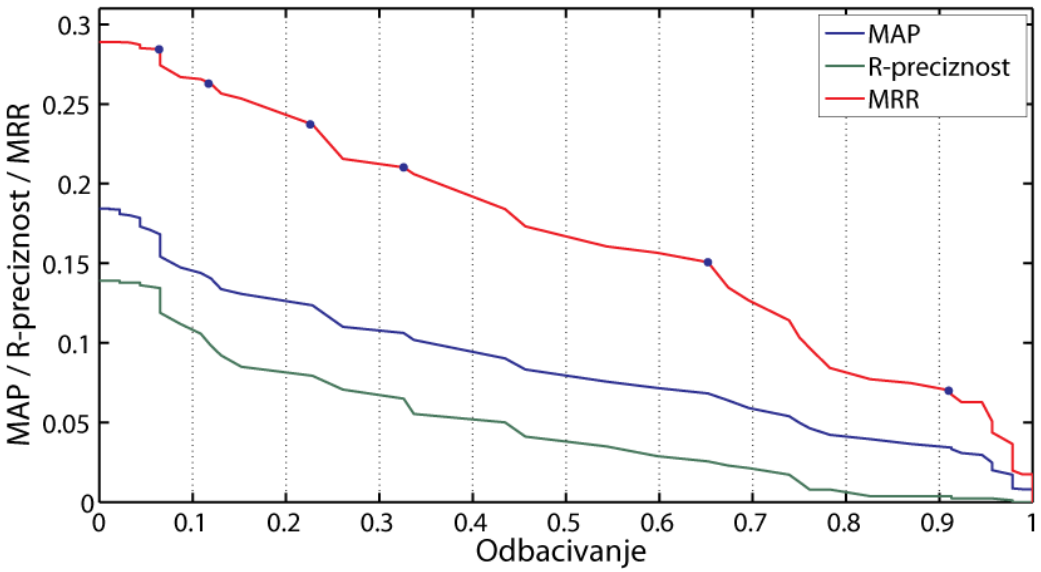
Graf odbacivanja (slika 5.8) prikazuje karakteristiku sustava pri određenim razinama odbacivanja. Značajni dijelovi grafa su intervali u kojima se vrijednosti mjera polako snizuju s porastom odbacivanja i kraći intervali u kojim se vrijednosti mjera naglo snizuju. Mali nagib grafa znači da se može dobiti veliko poboljšanje vrijednosti odbacivanja uz malo smanjenje vrijednosti ostalih mjera. Prilikom namještanja sustava između prepoznavanja da u zbirci nema relevantnih FAQ-parova za dani upit i kvalitete prezentiranja relevantnih dokumenata, odabiru se točke na grafu odbacivanja koje su neposredno prije intervala u kojem se nagib grafa naglo poveća. Odabrane točke predstavljaju optimalne vrijednosti za određivanja vrijednosti mjere sličnosti upita i FAQ-para koja određuje koji se FAQ-parovi pri pretraživanju prezentiraju a koji odbacuju.

Za razvijeni sustav optimalne su točke za sljedeće vrijednosti odbacivanja:



Slika 5.7: Kombinacija svih metoda

0,064, 0,117, 0,226, 0,326, 0,652, 0,91 (na grafu na slici 5.8 označeno plavim točkama).



Slika 5.8: Graf odbacivanja

6. Zaključak

U ovom radu predložena je metoda pretraživanja često postavljanih pitanja, izgrađena je ispitna zbirka za vrednovanje te je provedeno vrednovanje predložene metode.

Metoda predlaže da se za pretraživanje ne koristi samo TfIdf nego kombinacija TfIdf-a s mjerom semantičke sličnosti, prekrivanjem i tipom pitanja. Također, predlaže se pretraživanje zbirke često postavljanih pitanja ne samo na temelju FAQ-pitanja, već kombiniranjem pretraživanja na temelju FAQ-pitanja i FAQ-odgovora.

Pokazano je da predložena metoda značajno pridonosi kvaliteti pretraživanja u odnosu na metode koje koriste samo TfIdf i koje pretražuju FAQ-zbirku samo na temelju FAQ-pitanja.

Vrednovanjem kombinacije svih predloženih mjera pokazalo se da se područja semantičke sličnosti i prekrivanja preklapaju te da tip pitanja ne doprinosi bitno kvaliteti pretraživanja u kombinaciji s ostalim mjerama. Razlog preklapanja semantičke sličnosti i prekrivanja je ili u pogrešnom pristupu izgradnji semantičkog grafa ili u neprikladnom rječniku koji se koristio za izgradnju semantičkog grafa. Potrebno je dublje analizirati preklapanje navedenih mjera kako bi se utvrdio razlog preklapanja. Tip pitanja može više pridonijeti kvaliteti pretraživanja, ali je potrebno razviti metodu za dublju analizu teksta kojom bi se određivao tip pitanja. Također, kvaliteta korištene ispitne zbirke nije poznata te je potrebno provesti vrednovanje ispitne zbirke.

U daljnjem razvoju sustava vrednovati će se ispitna zbirka kako bi se utvrdila kvaliteta iste te je li potrebno ponovno izraditi ispitnu zbirku. Za izgradnju semantičkog grafa upotrijebiti će se neki drugi rječnik te utvrditi zašto se semantička sličnost i prekrivanje preklapaju. Također, izgrađeni sustav pruža osnovu za izgradnju naprednijeg sustava – sustava za odgovaranje na pitanja.

Literatura

Rich Ackerman. Vector Model Information Retrieval, 2003. `http://www.hray.com/5264/math.htm`.

Adam Berger, Rich Caruana, David Cohn, Dayne Freitag i Vibhu Mittal. Bridging the Lexical Chasm: Statistical Approaches to Answer-Finding. U *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, str. 192–199, 2000.

Chris Buckley. Implementation of the SMART Information Retrieval System. Technical report, Cornell University, 1985.

Robin D. Burke, Kristian J. Hammond i Edwin Cooper. Knowledge-based information retrieval from semi-structured text. U *AAAI Fall Symposium on AI Applications in Knowledge Navigation and Retrieval*, str. 19–24, 1995.

Robin D. Burke, Kristian J. Hammond, Vladimir Kulyukin, Steven L. Lytinen, Noriko Tomuro i Scott Schoenberg. Question Answering from Frequently Asked Question Files Experiences with the FAQ FINDER System. *AI Magazine*, 18 (2):57–66, 1997.

Deng-Yiv Chiu, Pei-Shin Chen i Ya-Chen Pan. Dynamic FAQ Retrieval with Rough Set Theory. *International Journal of Computer Science and Network Security*, 7(8), 2007.

Hua Huo i Boqin Feng. Retrieval Based on Combining Language Models with Clustering. *Lecture notes in computer science*, str. 847–852, 2004.

Harksoo Kim i Jungyun Seo. High-performance FAQ retrieval using an automatic clustering method of query logs. *Information Processing and Management*, 42 (3):650–661, 2006.

Harksoo Kim, Hyunjung Lee i Jungyun Seo. A reliable FAQ retrieval system using a query log classification technique based on latent semantic analysis. *Information Processing and Management*, 43(2):420–430, 2007.

Xiaoyong Liu i W. Bruce Croft. Cluster-Based Retrieval Using Language Models. U *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, str. 186–193. ACM, 2004. ISBN 1-58113-881-4.

Steven L. Lytinen i Noriko Tomuro. The Use of Question Types to Match Questions in FAQFinder. U *AAAI Spring Symposium on Mining Answers from Texts and Knowledge Bases*, str. 46–53, 2002.

Christopher D. Manning, Prabhakar Raghavan i Hinrich Schutze. *An Introduction to Information Retrieval*. Cambridge, England, 2007.

Philip Resnik. Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to problems of Ambiguity in Natural Language. *Journal of Artificial Intelligence Research*, 11:95–130, 1998.

Eriks Sneiders. Automated FAQ Answering: Continued Experience with Shallow Language Understanding. U *Question Answering Systems. Papers from the 1999 AAAI Fall Symposium*, str. 97–107, 1999.

Wanpeng Song, Min Feng, Naijie Gu i Liu Wenyin. Question Similarity Calculation for FAQ Answering. U *Third Conference on Semantics, Knowledge and Grid*, 2007.

Radu Soricut i Eric Brill. Automatic Question Answering: Beyond the Factoid. U Daniel Marcu Susan Dumais i Salim Roukos, urednici, *HLT-NAACL 2004: Main Proceedings*, str. 57–64, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics.

Anastasios Tombros, Robert Villa i C. J. Van Rijsbergen. The Effectiveness of Query-Specific Hierarchic Clustering in Information Retrieval. *Information Processing and Management*, 38(4):559–582, 2002.

Noriko Tomuro i Steven L. Lytinen. Selecting Features for Paraphrasing Question Sentences. U *Proceedings of the Workshop on Automatic Paraphrasing at Natural Language Processing Pacific Rim Symposium (NLPRS)*, str. 52–62, 2001.

Noriko Tomuro i Steven L. Lytinen. Retrieval Models and Q and A Learning With FAQ Files. U Mark T. Maybury, urednik, *New Directions in Question Answering*, str. 183–194. American Association for Artificial Intelligence, 2004.

Ellen M. Voorhees. The TREC-8 question answering track report. *NIST SPECIAL PUBLICATION SP*, str. 77–82, 2000.

Ellen M. Voorhees i Dawn M. Tice. Building a question answering test collection. U *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, str. 200–207, 2000.

Frane Šarić. Primjena teorije grafova u dubinskoj analizi teksta. Diplomski rad, Fakultet elektrotehnike i računarstva, 2006.

Chung-Hsien Wu, Jui-Feng Yeh i Yu-Sheng Lai. Semantic Segment Extraction and Matching for Internet FAQ Retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 18(7):930–940, 2006.

Indeks

Dodatak A
Primjeri XML-dokumenata

1. XML-dokument koji sadrži FAQ-zbirku

```
<documentSet name="FAQ_Vip"
description="Vip korisnička baza FAQ-a>
    <document name="FAQ-000000" id="0">
        <categories>
            <category category_id="000000">Vipme - Nazovi me
            </category>
        </categories>
        <content>
            <Question>Kakva je to usluga "Nazovi me"?
            </Question>
            <Answer>
                Usluga Nazovi me je nova usluga koju Vipnet
                uvodi za sve svoje Vipme (prepaid) korisnike.
                U SMS poruku upiši broj mobilnog
                telefona osobe koju hitno trebaš te
                pošalji na broj 765.  Nakon toga, ta
                osoba će primiti poruku da te nazove.
            </Answer>
        </content>
    </document>
    <document name="FAQ-000001" id="1">
        <categories>
            <category category_id="000000">Vipme - Nazovi me
            </category>
```

```
        </categories>
        <content>
            <Question>
                U kojem formatu trebam upisati telefonski broj u
                SMS poruci?
            </Question>
            <Answer>
                Isto kao i da pišeš SMS ili zoveš direktno tu osobu.
            </Answer>
        </content>
    </document>
</documentSet>
```

2. XML-dokument koji sadrži ispitnu zbirku

```
<log>
    <log_entry username="ime.prezime" empty="false">
        <user_questions>
            <user_question>Kako aktivirati mobilni internet?
            </user_question>
            <user_question>Kako upaliti mobilni internet?
            </user_question>
            <user_question>Kako se spaja na internet?
            </user_question>
            <user_question>Aktivacija interneta.
            </user_question>
            <user_question>Kako se spaja na net?
            </user_question>
        </user_questions>
        <answer_time>0:15:7.0</answer_time>
        <selected_answers>
            <document name="FAQ-000243" id="243">
                <categories>
                    <category category_id="000017">
                        Vip pretplatnici - Homebox i Homebox Call
                    </category>
```

```
                </categories>
                <content>
                    <Question>
                        Što mi je potrebno da bih spojio računalo i
                        koristio se internetom?
                    </Question>
                    <Answer>
                        Morate spojiti računala s Homebox uređajem
                        LAN kabelom ili putem WLAN-a.  Detaljne
                        upute kako to napraviti možete naći u
                        Uputama za korisnike.
                    </Answer>
                </content>
            </document>
        </selected_answers>
    </log_entry>
</log>
```

3. XML-dokument koji sadrži podatke o vrednovanju

```
<evaluation name="Vrednovanje.  | broj upita:  419 | TfIdfQ: 0,7500
| TfIdfA: 0,2500 | SemSimQ: 0,0000 | SemSimA: 0,0000
| Coverage:  0,0000 | QType:  0,0000 | Cut at:  0,1000 |">
    <measure name="Mean reciprocal rank" value="0,2643" />
    <measure name="Min.  average precision" value="0,0922" />
    <measure name="Rejection" value="0,0000" />
</evaluation>
```

Dodatak B
Detaljni podaci o vrednovanju sustava

U tablici B1 navedeni su detaljni podaci o vrednovanju sustava. Navedeni su parametri za formulu (3.1) te su dane vrijednosti MAP-a, R-preciznosti (RP) i MRR-a za navedene parametre. Navedene vrijednosti su izmjerene za ispitnu zbirku 1 tj. ispitnu zbirku koja sadrži samo upite za koje je označen barem jedan odgovor. Parametri za semantičku sličnost označeni su sa Sem, za prekrivanje s Cov, a za tip pitanja s TP. Odnosi li se neki parametar na uspoređivanje s FAQ-pitanjem ili s FAQ-odgovorom naznačeno je uz ime metode i to Q za FAQ-pitanje i A za FAQ-odgovor.

Prema priloženoj tablici vidi se da je za uspoređivanje upita i FAQ-para metodom TfIdf bitnije FAQ-pitanje, dok je za semantičku sličnost i prekrivanje bitniji FAQ-odgovor. Također, prilikom kombiniranja metoda za usporedbu upita i FAQ-para odnos važnosti FAQ-pitanja i FAQ-odgovora za svaku metodu ostaje isti.

Tablica B1: Detaljni podaci o vrednovanju

TfIdf Q	TfIdf A	Sem Q	Sem A	Cov Q	Cov A	TP	MAP	RP	MRR
1	0	0	0	0	0	0	0,1932	0,1528	0,3152
0	0	1	0	0	0	0	0,1351	0,1055	0,2634
0	0	0	0	1	0	0	0,0678	0,0594	0,1663
0	0	0	0	0	0	1	0,0122	0,0052	0,0240
0,6667	0,3333	0	0	0	0	0	0,2177	0,1528	0,3407
0,3333	0,6667	0	0	0	0	0	0,2023	0,1537	0,3053
0	0	0,3	0,7	0	0	0	0,1611	0,1228	0,2869
0	0	0,7	0,3	0	0	0	0,1476	0,1190	0,2639
0	0	0	0	0,9091	0,0909	0	0,0852	0,0769	0,1912
0	0	0	0	0,0909	0,9091	0	0,0971	0,0822	0,2007
0,2919	0,1460	0,1681	0,3940	0	0	0	0,2316	0,1756	0,3570
0,1469	0,2919	0,1681	0,3940	0	0	0	0,2236	0,1626	0,3480
0,2919	0,1469	0,3940	0,1681	0	0	0	0,2176	0,1628	0,3298
0,1469	0,2919	0,3940	0,1681	0	0	0	0,2128	0,1621	0,3315
0,3334	0,1666	0	0	0,0454	0,4546	0	0,2332	0,1743	0,3852
0,1666	0,3334	0	0	0,0454	0,4546	0	0,2148	0,1593	0,3630
0,3334	0,1666	0	0	0,4546	0,0454	0	0,1980	0,1495	0,3329
0,1666	0,3334	0	0	0,4546	0,0454	0	0,1905	0,1438	0,3294
0,5797	0,2899	0	0	0	0	0,1305	0,2090	0,1593	0,3530
0,2653	0,1327	0,1528	0,3582	0	0,0910	0	0,2363	0,1782	0,3701

Sažetak

Sustav za pretraživanje zbirke često postavljenih pitanja na hrvatskom jeziku

Često postavljana pitanja pružaju strukturiranu bazu znanja za sustav za pretraživanje informacija. Razvijen je sustav koji pretražuje takvu bazu znanja. Uspoređene su četiri različite metode pretraživanja baze kao i kombinacije tih metoda te je ispitano i pretraživanje baze znanja kombinirajući pristup pretraživanja FAQ-pitanja i FAQ-odgovora. Za potrebe vrednovanja razvijenog sustava izgrađena je ispitna zbirka i nad njom je provedeno vrednovanje. Prezentirani podaci pokazuju da navedeni pristup značajno pridonosi kvaliteti pretraživanja često postavljanih pitanja.

Ključne riječi: pretraživanje često postavljanih pitanja, leksički jaz, FAQ-odgovor, izrada ispitne zbirke

Abstract

Retrieval of FAQs written in Croatian languange

FAQs provide a structured knowledgebase for an IR system and such system was built. Four different measures for FAQ retrieval are compared as well as their combinations. Also, FAQ retrieval is done by combining retrieval of FAQ questions and FAQ answers. A test collection was built and used for evaluation. The empirical results show that the proposed methods significantly increase quality of retrieval results.

Keywords: FAQ retrieval, lexical disagreement problem, FAQ answer, building test collection