TakeLab

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

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 772

Model za otkrivanje i razgraničavanje značenja višeznačnih riječi hrvatskoga jezika

Marko Bekavac

Mentor: Doc. dr. sc. Jan Šnajder

Zagreb, lipanj 2014.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ODBOR ZA DIPLOMSKI RAD PROFILA

Zagreb, 10. ožujka 2014.

DIPLOMSKI ZADATAK br. 772

Pristupnik: **Marko Bekavac**
Studij: Računarstvo
Profil: Računarska znanost

Zadatak: **Model za otkrivanje i razgraničavanje značenja višeznačnih riječi hrvatskoga jezika**

Opis zadatka:

Leksička višeznačnost jezika predstavlja ozbiljnu prepreku u strojnoj obradi teksta. Razvijen je niz postupaka za razrješavanje višeznačnosti riječi temeljenih na statističkoj obradi korpusa i strojnome učenju. Postupci se većinom oslanjaju na unaprijed definiran skup značenja riječi (tzv. rječnik značenja), dok postupci temeljeni na nadziranom strojnom učenju dodatno iziskuju i veliku količinu ručno označenih podataka, koji za mnoge jezike nisu raspoloživi. Dodatan problem predstavlja zrnatost značenja: uobičajeno korišteni rječnici značenja kao što je WordNet za mnoge su primjene suviše detaljni, što nepotrebno usložnjuje model te smanjuje točnost postupaka razrješavanja višeznačnosti.

U okviru diplomskoga rada potrebno je proučiti postupke za razrješavanje višeznačnosti, posebice postupke za nenadzirano otkrivanje i razgraničavanje značenja. Razraditi model nenadziranog otkrivanja značenja riječi hrvatskoga jezika temeljen na statističkoj analizi supojavljivanja riječi u korpusu. Razraditi postupak vrednovanja takvog modela koji ne ovisi o unaprijed definiranom rječniku značenja. Izgraditi odgovarajući ispitni skup višeznačnih riječi s ručno označenim značenjima. Razviti programsku implementaciju modela te provesti iscrpno eksperimentalno vrednovanje na zadatcima otkrivanja i razgraničavanja značenja, uključivo detaljnu analizu pogrešaka. Radu priložiti izvorni i izvršni kod razvijenog sustava, označene skupove podataka i potrebnu dokumentaciju te citirati korištenu literaturu.

Zadatak uručen pristupniku: 14. ožujka 2014.
Rok za predaju rada: 30. lipnja 2014.

Mentor:

Doc. dr.sc. Jan Šnajder

Djelovođa:

Doc. dr.sc. Tomislav Hrkać

Predsjednik odbora za diplomski rad profila:

Prof. dr.sc. Siniša Srbljić

Velika zahvala dobrovoljcima označavačima na uloženom trudu i velikoj pomoći u realizaciji ovog rada. Abecednim redoslijedom to su Domagoj Vočanec, Elena Orozović, Eugen Rožić, Jurica Šprem, Luka Bekavac, Natan Šujansky, Sven Majerić, Svjetlana Bekavac, Teo Bekavac, Valentina Vočanec.

SADRŽAJ

1. Uvod

Jedna od karakteristika prirodnog jezika je njegova nejednoznačnost. Često se može naići na riječi koje bi, upotrijebljene u drugom kontekstu, imale potpuno drugačije značenje. Višeznačnost (polisemija) je do te mjere ugrađena u jezik da se često koristi i u svrhu humora ili literarnog izražaja. Čovjeku koji razumije jezik je u većini slučajeva jednostavan zadatak odrediti o kojem se točno značenju radi. Značenje ovisi o kontekstu u kojem se riječ nalazi, a čovjeku je analiza konteksta intuitivna; ukoliko poznaje kontekst riječi, može s gotovo potpunom sigurnošću odrediti njeno značenje.
S druge strane, računalu je takav zadatak težak. S obzirom da je potrebna visoka razina razumijevanja semantike konteksta, problemu višeznačnosti potrebno je pristupiti iz perspektive obrade prirodnog jezika. Različiti pristupi problemu, dakle, moraju uključivati izvore semantičkog znanja koje računalo može koristiti, ili obradu veće količine teksta ne bi li se našla statistička pravilnost, odnosno, ne bi li nesemantičke značajke konteksta na neki način otkrivale semantiku i ukazivale na ispravno značenje.
Primjerice, promatrajući riječ „red“, moguće je razabrati nekoliko njenih značenja. „Red čokolade“ (engl. *bar*) i „čekati u redu“ (engl. *queue*) ne označavaju istu vrstu reda, kao ni „franjevački red“ (engl. *order*), „Taylorov red“ (engl. *series*) ili „zakon i red“ (engl. *order*). Promatrana riječ prevodi se u (često) različite riječi, koje međusobno nemaju isto značenje niti se mogu koristiti u istom kontekstu. Ljudima koji čitaju navedene primjere potpuno je jasno što „red“ znači u svakom kontekstu, uz uvjet da dovoljno poznaju domenu u kojoj se pojedino značenje nalazi (primjerice, matematiku u slučaju Taylorovog reda). Ukoliko promatramo riječ „linija“, spominjanje riječi „aerodrom“, „glasnoća“, „ravnalo“ i „prijeći“ može asocirati na drugačije značenje (redom, to su „zrakoplovna linija“, „glazbena linija“, „crta“, „granica“). Iako spomenute riječi ne sudjeluju u frazama koje opisuju pojedino značenje, često su dovoljno indikativne da bismo ipak mogli shvatiti o kojem je značenju riječ.

Za neke je postupke (primjerice, strojno prevođenje) nužno znati pravo značenje riječi. Međutim, postupci za razrješavanje višeznačnosti zahtijevaju vrlo kvalitetne resurse (skupove značenja) i ručno označene podatke (primjere za svako značenje), koji

za neke jezike ne postoje. Zbog toga je zanimljiv razvoj sustava koji automatski dolazi do tih podataka. U ovom je radu naglasak na otkrivanju značenja riječi hrvatskog jezika temeljenog na jednojezičnom korpusu (velikom skupu tekstova na jednom jeziku, u ovom slučaju hrvatskom). Na temelju supojavljivanja riječi u korpusu izgrađen je težinski neusmjereni graf, čiji su vrhovi zatim grupirani tako da svaka grupa opisuje kontekste karakteristične za pojedino značenje. Korišten je niz algoritama grupiranja te su uspoređeni njihovi rezultati. Kao višeznačne riječi odabrane su one koje se neovisno o morfološkim promjenama ne razlikuju, ali prema nekom kriteriju imaju barem dva različita značenja. Prema tome, morfološki homonimi, homografi i homofoni nisu uzimani u obzir; pretpostavljeno je da su riječima ispravno određene leme (lingvistički ispravni osnovni oblici riječi) i vrste riječi. Promatrane su, dakle, riječi koje imaju istu lemu i istu vrstu, ali više značenja. Cilj rada je analizirati različite algoritme temeljene na grafu supojavljivanja.

U nastavku je struktura rada. Poglavlje 2 donosi kratak osvrt na problem višeznačnosti u obradi prirodnog jezika te pregled relevantih radova iz područja, s posebnim naglaskom na pristup otkrivanju značenja pomoću grafa supojavljivanja. Poglavlje 3 opisuje korišteni korpus i način izgradnje grafa supojavljivanja te potom i težinskoga grafa koji modelira semantičke poveznice između riječi. U poglavlju 4 prikazani su algoritmi korišteni za grupiranje vrhova grafa, odnosno otkrivanje značenja promatranih riječi. Također, opisuje metodu razrješavanja višeznačnosti temeljenu na otkrivenim značenjima. Poglavlje 5 donosi opis metodologije evaluacije otkrivanja značenja riječi i razrješavanja višeznačnosti te rezultate evaluacije. U poglavlju 6 donesen je zaključak na temelju obavljenoga rada i rezultata, kao i ideje za budući rad i poboljšanja.

2. Problem višeznačnosti riječi

Problem višeznačnosti riječi bitan je jer ljudi koriste višeznačne riječi bez previše razmišljanja o tome te ih isto tako i razumiju. U svakom obliku jezika, bilo pisanom, bilo govorenom, višeznačnost je prisutna. Bilo kakav zadatak u kojem računalo iz zapisa u prirodnom jeziku treba izdvojiti informacije ili iz nekog drugog razloga, uvjetno rečeno, razumjeti značenje zapisa jezika, mora se moći nositi s pojavom višeznačnosti. U nastavku su vrlo općenito opisani postupci pronalaženja točnog značenja (razrješavanje višeznačnosti), postupci pronalaženja skupa značenja (potrebni za razrješavanje višeznačnosti) i kratak pregled radova vezanih uz temu.

2.1. Razrješavanje višeznačnosti

Zadatak otkrivanja značenja riječi svodi se na klasifikaciju, tako da sva pojavljivanja jednog značenja promatrane riječi budu u istoj klasi, a istovremeno niti jedan par pojavljivanja različitih značenja te riječi ne bude u istoj klasi, naziva se razrješavanje višeznačnosti (engl. *word sense disambiguation*).

Povijesno, jedan od prvih problema u analizi teksta gdje se višeznačnost riječi pokazala kao poteškoća je zadatak strojnog prevođenja. Višeznačne riječi u jednom jeziku ne moraju biti višeznačne riječi (s istim skupom značenja) u drugom jeziku, dakle, potrebno je poznavati točno značenje kako bi se riječ mogla prevesti. Moderni sustavi za strojno prevođenje daju bolje rezultate što im je bolji podsustav za razrješavanje višeznačnosti (Agirre i Edmonds, 2007). Prema (Carpuat i Wu, 2007), određivanje kategorija značenja pomoću leksikona lošiji je pristup od onog u kojem se do pravog značenja dolazi na temelju konteksta, najčešće fraze u kojoj se riječ nalazi.
Osim toga, važna primjena je u dohvatu informacija (*information retrieval*). Jasno je da je bitno razumijeti značenje upita kako bi se mogle dohvatiti relevantne informacije. Problem u ovakvoj primjeni je potencijalno kratak upit, što otežava određivanje konteksta u kojem se višeznačna riječ nalazi. No, čak i u slučaju kad je preciznost otkrivanja značenja slaba, ono poboljšava rezultate u domeni dohvata informacija, a veća

preciznost (iznad 90%) znatno utječe na rezultate sustava (Schütze i Pedersen, 1995).
Za ekstrakciju informacija i analizu sadržaja također je bitno poznavati značenje riječi. Zadatak je ovdje ponešto drugačiji te je rjeđe potrebno odrediti značenje općenitih riječi, a češće nekih stručnih pojmova ili pojmova vezanih uz specifičnu domenu. Također, ovdje se zadatak razrješavanja višeznačnosti često miješa sa zadacima poput otkrivanja imenovanih entiteta.
Prema (Navigli, 2009), potreba za otkrivanjem značenja postoji u raznim sustavima za obradu teksta, leksikografiji i, u novije vrijeme, semantičkom webu.
Svaki od navedenih zadataka ima različitu potrebu za razrješavanjem višeznačnosti. Pretraživanje informacija u određenoj domeni znanja zahtijeva finiju granulaciju značenja određenog pojma od one potrebne u sustavu za strojno prevođenje općenite primjene. Također, u zadatku strojnog prevođenja nekad je potrebno prevoditi cijele fraze, koje mogu biti višeznačne, dok se u analizi kraćih i strukturiranijih upita u dohvatu informacija takav problem rijetko susreće. Moguće je doći do zaključka da analiza i razrješavanje višeznačnosti nemaju veliku svrhu sami za sebe, ali igraju bitnu ulogu kao dijelovi drugih sustava te da je zbog toga pri rješavanju problema višeznačnosti važno razmišljati o njegovoj budućoj primjeni.

2.2. Otkrivanje značenja riječi

Kako bi bilo moguće automatski odrediti značenje riječi, potrebno je imati skup značenja svake promatrane riječi. Naime, nema smisla otkrivati značenje unutar strojnog prevođenja ako ne postoji funkcija koja preslikava otkriveno značenje u ispravan prijevod. U dohvatu podataka, nema koristi od razrješavanja višeznačnosti ukoliko nisu poznata značenja, tako da svako značenje odgovara nekoj drugoj mogućoj temi.
Iako se možda čini da je najjednostavniji pristup rješavanju tog problema izgradnja leksikona značenja, takvog da skup značenja određuju ljudi te za svako značenje stručnjaci daju definiciju značenja, uz nekoliko primjera upotrebe (odnosno, primjera konteksta) ili riječi koje su semantički povezane s pojedinim značenjem (primjerice, sinonimi određenog značenja riječi). Glavni nedostatak takvog pristupa je povezan s načinom na koji leksikon nastaje; njegova izrada je vremenski zahtjevna te iziskuje ogromnu količinu rada i znanja o domeni. Nadalje, leksikon je ograničen samo na ograničen skup riječi, tako da specifični izrazi, riječi koje se u jeziku rjeđe pojavljuju, ali i nove riječi najčešće nisu zastupljeni u leksikonu. Leksikoni brzo zastarijevaju jer, osim što nedostaju novije riječi, nedostaju i moguća nova značenja postojećih riječi, tako da je potreban neprekidan trud i ulaganje u održavanje leksikona. Osim toga, leksikon ra-

zvijan za potrebe jednog sustava može biti neprilagođen potrebama drugog sustava. Rješenje može biti u automatiziranom pronalaženju skupova značenja pojedine riječi, kao i nekog oblika definiranja svakog od pronađenih značenja. Takve su metode u obradi prirodnog jezika inherentno slabije točnosti od ljudskog označavanja ili kategorizacije, ali mogu biti fleksibilnije i prilagodljivije zadatku te zahtijevati manje rada i ljudskog vremena. Postavlja se pitanje koliko kvalitetne i točne mogu biti takve, automatske metode te jesu li usporedive s ljudskom odlukom. Također, o kojim se točno metodama radi te koje su prednosti i mane pojedine metode.

Postupak otkrivanja skupa značenja naziva se otkrivanje značenja riječi (engl. *word sense induction*, *sense inventory induction*). Prema (Schütze, 1998), razgraničavanje značenja riječi (engl. *word sense discrimination*) grupiranje je pojavljivanja višeznačnih riječi takvo da su zajedno grupirana pojavljivanja istog značenja. Granica između tih pojmova nije uvijek potpuno određena, ali se u ovom radu svodi na razliku između stvaranja grupa značenja riječi i klasifikaciju pojedinih primjera u te grupe.

2.3. Postupci otkrivanja značenja riječi

U postupku otkrivanja značenja riječi (ali i razrješavanja višeznačnosti) moguće je koristiti izvore znanja, kao što su već postojeći leksikoni značenja (koji se zatim nadopunjavaju, kojima se mijenja granulacija značenja ili se koriste na neki drugi način), ostale leksičke baze podataka poput WordNeta,[1] koje sadrže i konceptualno–semantičke podatke te daju provjerenu informaciju o semantičkoj povezanosti između dvije riječi. Mogući problem vezan uz izvore znanja je pitanje postoje li uopće, a ako postoje, koliko su opširni i kvalitetni. Dok za engleski jezik postoji pregršt kvalitetnih izvora znanja, situacija je znatno slabija za hrvatski jezik.

Nadalje, moguće je koristiti druge resurse, kao što su paralelni višejezični korpusi (Diab i Resnik, 2002), posebno korisni za primjenu u strojnom prevođenju. Jedan od nedostataka takvog pristupa u općenitom je slučaju mogućnost istovremene višeznačnosti riječi u više jezika, gdje se skup značenja promatrane riječi kroz više jezika djelomično preklapa.

Korištenje jednojezičnog korpusa tekstova korak je prema izgradnji vlastite baze podataka o semantičkim odnosima riječi, najčešće bez ljudske intervencije (nenadzirano). Naime, prema (Harris, 1954), riječi koje su na sličan način distribuirane unutar velike kolekcije tekstova, odnosno, pojavljuju se zajedno u sličnim kontekstima, semantički

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

su povezane. Tako su međusobno vrlo povezani sinonimi i antonimi, ali i riječi u ostalim semantičkim odnosima (hipernimi, meronimi i slično). Uobičajena je izgradnja strukture u kojoj su pohranjeni podaci o supojavljivanju, na temelju koje se različitim postupcima grupiranja može doći do podatka o skupovima konteksta, na temelju čega je moguće definirati skup značenja. Nedostaci takvih metoda su relativno velika količina šuma i moguće zanemarivanje značenja koja se u korpusu rjeđe pojavljuju, ali su i dalje značajna.

2.4. Osvrt na povijest problematike višeznačnosti

Problem višeznačnosti riječi već je dugo aktualna tema u strojnoj obradi prirodnog jezika. Već je 1949. godine Warren Weaver (izdano u (Weaver, 1955)) zaključio da riječ sama po sebi ne indicira svoje pravo značenje, već je bitno promatrati dovoljno veliko okruženje riječi kako bi bilo moguće nedvosmisleno odrediti značenje. Godinu poslije, Kaplan (1950) dolazi do zaključka kako je okruženje od po dvije riječi sa svake strane promatrane riječi jednako informativno kao i cijela rečenica. Godine 1957. nastaje ideja da se kao izvor značenja pojedine riječi koristi rječnik te da se na temelju skupa riječi u opisu značenja može odrediti značenje riječi (uspređujući taj skup i okruženje promatrane riječi u tekstu). U (Madhu i Lytle, 1965) rješenje problema višeznačnosti autori vide u statističkoj obradi teksta, što je velik korak prema današnjem viđenju problema i njegovom rješavanju. Oni su računali učestalost pojedinog značenja u pojedinoj domeni tekstova te prema informacijama o domeni određivali značenje. Sljedećih je desetak godina oslabio entuzijazam oko rješavanja problema višeznačnosti; mnogi su autori tvrdili da je takav zadatak izuzetno težak, pa čak i nemoguć.
Kasnije, zajedno s ponovnim procvatom područja umjetne inteligencije, dolazi i do ponovnog interesa za problem višeznačnosti. Metode se fokusiraju na razumijevanje teksta te koriste ručno definirane izvore znanja o značenjima, što se pokazalo neadekvatnim za veću količinu riječi. Zbog toga pravi uspon metoda kreće osamdesetih godina dvadesetog stoljeća, kad opsežni leksikografski resursi postaju dostupnima, što omogućuje automatsku akviziciju informacija. Izuzetno je utjecajan i bitan članak (Lesk, 1986), u kojem je predstavljen algoritam razrješavanja višeznačnosti koji se i danas koristi; u originalnom obliku, izmijenjen ili samo kao metoda za usporedbu rezultata. Algoritam nalaže da, ukoliko promatramo dvije riječi u rečenici, ispravna značenja tih riječi su ona značenja čije se definicije najviše preklapaju (odnosno, imaju najveći broj zajedničkih riječi). Time je definirana čvrsta veza između leksikografije i razrješavanja višeznačnosti. Ta je veza dodatno ojačana pojavom WordNeta, koji se i danas učestalo

koristi kao izvor leksičkih informacija.
Nastavljajući sa statističkim metodama, u devedesetim godinama dvadesetog stoljeća dolazi do široke primjene strojnog učenja na ovaj problem. Jedan od najranijih članaka koji spajaju strojno učenje na korpusu i razrješavanje višeznačnosti je (Brown et al., 1991).
Bitan korak za razvoj metoda je natjecanje Senseval,[2] koji je dodatno potaknuo razvoj resursa za izradu i evaluaciju sustava za otkrivanje značenja i razrješavanje višeznačnosti. Senseval i njegov kasniji razvoj u SemEval detaljnije je opisan u poglavlju 5.
U novije vrijeme zadatak otkrivanja značenja postaje sve popularniji, kao i srodne metode u razrješavanju višeznačnosti, temeljene na nenadziranom strojnom učenju.

2.5. Srodni radovi

Budući da je u radu korišten model temeljen na grafu supojavljivanja, u nastavku su predstavljeni radovi također temeljeni na grafu supojavljivanja. Ostale metode modeliranja semantičke veze i vezani radovi ukratko su opisani u 3.4. Ostale metode otkrivanja značenja riječi predstavljene su u 4.2.8.

Jedan od prvih radova u kojima je predstavljena ideja korištenja grafa supojavljivanja kao temelja za određivanje značenja riječi je (Widdows i Dorow, 2002) (nakon čega su uslijedili i (Dorow i Widdows, 2003) i (Dorow et al., 2004)). Autori grade graf supojavljivanja imenica te grupiraju njegove vrhove na temelju sličnosti između skupova susjednih vrhova. Kasnije je algoritam rafiniran, tako da bolje pronalazi nedjeljive kontekste, odnosno koristi supojavljivanja drugog stupnja. Algoritam je detaljnije opisan u potpoglavlju 4.2.2.

Rad (Véronis, 2004) predstavlja algoritam HyperLex (detaljnije opisan u 4.2.4), gdje je ideja prvo odabrati reprezentativne vrhove grafa, a zatim ostale vrhove spojiti s njima, tako da grupe budu što prirodnije i kompaktnije. (Agirre et al., 2006) donosi nadogradnju implementacije HyperLexa, ali i inačicu algoritma PageRank, prilagođenu grafovima supojavljivanja. Preinaka na PageRanku inspirirala je preinaku napravljenu na algoritmu HITS, predstavljenu u ovom radu.

Ovaj se rad velikim dijelom oslanja na (Di Marco i Navigli, 2013), koji donosi iznimno kvalitetan usporedni pregled rada algoritama B–MST, SquaT++, HyperLex i algoritma „pokvarenog telefona“ (opisanih u poglavlju 4) u primjeni na otkrivanje značenja riječi.

[2] `http://www.senseval.org/`

Teorijska podloga postupcima vezanima uz problem višeznačnosti, najviše razrješavanje višeznačnosti opisana je u (Agirre i Edmonds, 2007).

Za razliku od većine prethodnih radova gdje su korištene samo imenice, sustav opisan u ovome radu koristi imenice, glagole i pridjeve kao vrhove grafa, zbog čega je moguće otkrivanje značenja svake od te tri vrste riječi. U ovom je radu opisano sedam različitih algoritama grupiranja vrhova grafa (od kojih algoritam HITS vjerojatno nikad prije nije bio korišten u ovakvom kontekstu) te, prema trenutnom saznanju, ovaj rad prvi donosi usporedbu rada svih tih algoritama na istom skupu podataka, kao i prikaz performansi pojedinog algoritma na svakoj od vrsta riječi te kroz tri pojasa frekvencija riječi u grafu. Nadalje, rad nudi opis postupka optimizacije parametara algoritama. Prema trenutnom saznanju, do ovog rada nije opisana metoda evaluacije otkrivanja značenja bez njegove primjene unutar okvira razrješavanja višeznačnosti ili bez upotreba metoda za preslikavanje grupa riječi u prethodno poznata značenja iz rječnika. Također je predstavljena metoda izrade zlatnog standarda na temelju ručno grupiranih skupova riječi. Na kraju, predstavljena je jednostavna metoda razgraničavanja temeljenja na prethodno otkrivenim značenjima.

3. Korpus i izgradnja grafa supojavljivanja

U ovome je radu prikazan postupak otkrivanja skupa značenja temeljenog na grafu supojavljivanja. Izgradnja grafa započinje obradom korpusa. To uključuje indeksiranje, filtriranje i pretraživanje korpusa, kao i razne pokuse kojima je potvrđena opravdanost i nužnost tih koraka. Nakon toga slijedi korak učitavanja supojavljivanja u strukturu grafa, takvu da može pohraniti veliku količinu informacija, a istovremeno pružiti brzi pristup bilo kojem podatku vezanom uz graf. Zadnji je korak računanje težina na bridovima između riječi, koje modeliraju snagu njihove međusobne semantičke povezanosti.

3.1. Korpus fHrWaC i rječnik značenja

Korpus weba. Korišteni korpus fHrWaC[1] filtrirana je verzija korpusa hrWaC.[2] Filtriranjem su uklonjeni netekstni sadržaji, problemi u zapisu znakova i strane riječi (Šnajder et al., 2013). Originalni korpus sastoji se od velikog broja (oko 3.5 milijuna) dokumenata prikupljenih s web stranica domene `.hr` te sadrži preko 1.2 milijarde jezičnih elemenata (Ljubešić i Erjavec, 2011). Korpus je tematski i stilski raznolik; web korpusi nisu ograničeni na samo jedan stil (primjerice, članke), niti su filtrirani prema tom kriteriju. Također, autori su eksperimentalno pokazali raznolikost tema, među kojima su politika, zakon, financije, sport, oglasi, automobili, tehnologija, religija, umjetnost i zdravlje. Tekstovi su očišćeni od nepoželjnih i nevažnih sadržaja te morfološki označeni i lematizirani (koristeći alate opisane u 3.5). Konačno, svaki tekstni element (riječ ili interpunkcijski znak) predstavljen je uređenom trojkom, gdje je prvi član nepromijenjena verzija riječi, kakva se nalazi u tekstu, drugi je član lema riječi, a treći pripadajuća morfološka oznaka.

[1] `http://takelab.fer.hr/data/fhrwac/`
[2] `http://nlp.ffzg.hr/resources/corpora/hrwac/`

Rječnička baza. Drugi korišteni resurs bila je rječnička baza,[3] izgrađena na temelju nekoliko novijih rječničkih i leksikografskih izdanja (više informacija na priloženoj poveznici). Uz svaku riječ iz skupa (oko 116 000 riječi) stoji objašnjenje, a uz većinu i primjeri uporabe, sintagmatski izrazi, frazeološki izrazi, sinonimi i antonimi. Iako takav izvor znanja nije dovoljan za samostalno korištenje kao izvor semantičkog znanja, može dati okvirne podatke i smjernice za rad, kao i usporedbu za kvalitetu rezultata. Podaci nisu izravno korišteni u radu sustava, ali su utjecali na njegovo oblikovanje, najviše u koraku evaluacije elemenata sustava.

Iako je uz svaku riječ iz rječničke baze naveden i skup njenih značenja te opis svakog od značenja, pokusi su pokazali da taj skup nije primjeren korištenju u razrješavanju višeznačnosti.

Prvi je razlog slaba definiranost pojedinog značenja. S obzirom na to da semantička struktura nije eksplicitno dana (i dijelom je nepotpuna), glavni izvor informacija su definicije značenja. Nažalost, koristeći algoritme temeljene na preklapanju rječničke definicije i konteksta promatrane riječi (Lesk, 1986), nije moguće dobiti dovoljno kvalitetne rezultate. Čak se niti korištenjem informacija o semantičkim vezama (primjerice, koristeći graf supojavljivanja) ne mogu dobiti dovoljno kvalitetni rezultati, iako je poboljšanje značajno. Jednostavno rečeno, priložene leksikografske definicije značenja ne daju dovoljno dobru aproksimaciju konteksta da bi se na temelju njih moglo automatski pronaći pravo značenje. Pokusi su pokazali da se takav pristup rezultatima ne može mjeriti s modernim pristupima temeljenima na nadziranom ili polunadziranom strojnom učenju.

Drugi je razlog neodgovarajuća, najčešće prevelika granuliranost značenja. Značenja su većinom podijeljena prema vrlo suptilnim i, dijelom, subjektivnim kriterijima. Ponegdje je razlika između dva značenja vrlo mala i apstraktna, pa niti ljudi ne mogu sa sigurnošću utvrditi točnu razliku između njih, a pogotovo je težak zadatak označiti višeznačnu riječ u određenom kontekstu s nekim od tih značenja. Također, dio značenja ne postoji u suvremenom ili laičkom jeziku; neka značenja se niti jednom ne pojavljuju u promatranom korpusu. Radi se o arhaizmima, stručnim leksikografskim izrazima ili prenesenim značenjima koja se iznimno rijetko koriste izvan književnih djela (pogotovo poezije). Osim toga, usprkos činjenici da je baza često ažurirana i nadopunjavana, nedostaju neka često korištena značenja. Nije nužno da se radi o novim značenjima, već i o onima koja su autorima promakla ili za koja su odlučili da nisu dovoljno važna. Primjerice, „mazati se *faktorom*“ relativno je učestala sintagma jasnog značenja, ali to

[3] `http://hjp.novi-liber.hr/index.php?show=baza`

značenje riječi „faktor“ nije navedeno.
U poglavlju 5 prikazana je usporedba skupova značenja određenog na temelju rječničke baze, određenog sustavom za otkrivanje značenja te skupova značenja koje su načinili označavači.

3.2. Izgradnja grafa supojavljivanja

3.2.1. Predobrada korpusa

Prije same izgradnje grafa, potrebno je predobraditi korpusa. Ona ima dvostruku ulogu: ukloniti informacije koje potencijalno štetno djeluju na konačni rezultat i ukloniti nepotrebne informacije, koje ne pridonose rješenju problema, a negativno utječu na vrijeme i složenost izvođenja programskog rješenja. Prolaskom kroz cijeli korpus nastaje popis svih jedinstvenih riječi, odnosno riječi jedinstvene leme i vrste riječi. Uz svaku riječ zabilježen je i broj njenih pojavljivanja u korpusu. Budući da je postupak lematizacije i označavanja proveden automatski (zbog iznimno velikog opsega korpusa), postoji određen broj grešaka. Greške se najviše očituju u postojanju neispravnih lema, obično vrlo malog broja pojavljivanja. S obzirom na činjenicu da postoje riječi koje se vrlo rijetko pojavljuju u korpusu, ali su ipak ispravno lematizirane i označene te da postoje vrlo česte riječi koje su u nekim slučajevima neispravno lematizirane ili označene, koje ukupno imaju broj pojavljivanja veći od nekih ispravnih riječi, filtriranje riječi prema broju pojavljivanja u korpusu dovodi do gubitaka. Zbog toga je cijeli korpus lematiziran i sustavom za morfološku normalizaciju MOLEX (Šnajder et al., 2008).[4] Sustavi za lematizaciju daju različite rezultate, pa je pretpostavka da će, ukoliko oba sustava daju istu lemu i oznaku vrste riječi, ta riječ s većom sigurnošću biti ispravno lematizirana. S druge strane, nije nužno da drugi lematizator za svaku riječ daje ispravan rezultat. Iz tih je razloga odlučeno filtriranje prema dva kriterija: dovoljno je da se f_1 puta riječ (odnosno, par lema–vrsta riječi) pojavi u korpusima lematiziranima pomoću oba lematizatora te f_2 puta u korpusu lematiziranom korištenjem samo prvog lematizatora. Broj f_2 veći je od f_1, a eksperimentalno su određene vrijednosti za f_1 = 100 i f_2 = 500. Sve riječi koje ne zadovoljavaju barem jedan od dva kriterija više se ne uzimaju u obzir. Eksperimenti su pokazali da u odbačenim riječima nema previše onih ispravnih, odnosno, da su gubici nastali korištenjem ovih parametara zanemarivi. Naime, prema Zipfovom zakonu (Zipf, 1949), broj pojavljivanja neke riječi u korpusu obrnuto je proporcionalna njenom rednom broju (ukoliko su riječi sortirane padajući

[4] `http://takelab.fer.hr/en/molex`

prema broju pojavljivanja). Dakle, manji broj najčešćih riječi čini većinu korpusa, a one riječi koje su dovoljno nisko rangirane gotovo ni ne utječu na korpus, usprkos tome što je broj različitih nisko rangiranih riječi velik.
Koristeći rječničku bazu i uspoređujući njen skup riječi s riječima preostalima nakon filtriranja s raznim parametrima te kasnijom ručnom provjerom rezultata, opravdan je izbor vrijednosti f_1 i f_2. Čak ni bez filtriranja nije očekivano pronaći sve riječi iz rječničke baze (jer dio baze čine arhaizmi i stručni izrazi), ali je povećavajući iznos parametara moguće pronaći točku u kojoj naglo počinje padati broj riječi u presjeku skupova.

U zadnjem koraku filtriranja uklonjene su sve riječi koje nisu imenice, glagoli ili pridjevi. S obzirom da su te tri vrste riječi glavni nosioci značenja i semantičkog sadržaja, ostale riječi mogu se zanemariti. U srodnim radovima autori najčešće koriste samo imenice, ali je jedan od ciljeva ovog rada prikazati uspješnost algoritama ukoliko se koriste i glagoli i pridjevi.

Zaustavne riječi nisu uklanjane prije izgradnje grafa. Nakon što je većina takvih riječi eliminirana filtriranjem po vrsti riječi, one preostale možda mogu nositi neko značenje ili čak ukazivati na značenje neke višeznačne riječi. Također, u poglavlju 3.3 opisan je postupak prema kojem vrlo učestale riječi gube na važnosti osim ako nisu vrlo jako asocirane s nekom drugom riječi.

3.2.2. Brojanje jedinstvenih parova riječi

Graf je definiran kao uređeni par $G = (V, E)$, gdje je V neprazan konačni skup vrhova grafa, a E skup bridova grafa. Vrhovi $v_i, v_j \in V$ susjedni su u slučaju da postoji $e \in E$ takav da $e = (v_i, v_j)$ (e spaja v_i i v_j), a v_i i v_j su incidentni s e. Stupanj vrha v je broj bridova s kojima je incidentan. Neusmjereni graf je onaj graf u kojem bridovi nemaju orjentaciju, odnosno $(v_i, v_j) \equiv (v_j, v_i)$. U težinskom grafu svakome je bridu dodijeljena težina w. U nastavku rada korištene su ove oznake (gdje je moguće).

Graf supojavljivanja definiran je kao težinski neusmjereni graf, gdje vrhovi predstavljaju jedinstvene riječi (skup riječi se bijektivno preslikava u skup vrhova), bridovi predstavljaju supojavljivanje dvije riječi (ukoliko se riječi nigdje ne pojavljuju zajedno, nisu povezane bridom, inače jesu), a težine na bridovima predstavljaju broj supojavljivanja.

Supojavljivanje je definirano kao pojavljivanje dvije različite riječi unutar rečenice, tako da je udaljenost između njih najviše 10 riječi, odnosno, tako da između njih može biti najviše 9 riječi i niti jedna granica rečenice. Ukoliko je najveća dopuštena udalje-

nost manja, može doći do gubitka podataka i naglašavanja povezanosti između riječi koje nemaju nužno semantičku povezanost (primjerice, parovi ime–prezime često spominjanih osoba, riječi iz često korištenih fraza i slično). Ukoliko je dopuštena udaljenost znatno veća, može doći do povezivanja riječi između kojih u rečenici uopće nema semantičke povezanosti, odnosno šuma. Malo veće udaljenosti ne daju zamjetno bolje rezultate, ali povećavaju broj bridova.

No, kako je od početka cilj pokušati što vjernije modelirati semantičku vezu između riječi, graf supojavljivanja nije dovoljno dobar. Problem se nalazi u broju pojavljivanja pojedinih riječi. Vrlo učestale riječi, poput glagola „biti“ i „htjeti“ nalaze se u gotovo svim rečenicama i imaju visok broj supojavljivanja s bilo kojom drugom riječi, ali to ne znači da je semantička veza tih parova jaka. Također, postoje parovi rijetkih riječi koje se uglavnom pojavljuju zajedno, ali zbog niskih učestalosti veza nije prepoznata kao dovoljno snažna. Srećom, postoje uvriježene mjere kojima se kompenziraju učestalosti. Poglavlje 3.3 nosi pregled testiranih mjera, kao i opis eksperimenata.

3.3. Određivanje novih težina bridova

3.3.1. Mjere leksičke asocijacije

Za određivanje novih težina korištene su neke od mjera prikazanih u (Evert, 2008), ali su zbog jednostavnosti korištene drugačije oznake. Vrijednost f_i predstavlja broj pojavljivanja riječi i u korpusu, a f_{ij} broj supojavljivanja riječi i i j. Korištene mjere su često korištene u statistici, ali prilagođene za primjenu u mjerenju asocijacije između para riječi u korpusu. Prema podijeli korištenoj u referenciranom radu, odabrane mjere dijele se u dvije skupine: jednostavne mjere asocijacije i statističke mjere asocijacije.

Jednostavne mjere asocijacije temelje se na usporedbi očekivanog i opaženog broja supojavljivanja. Opaženi broj supojavljivanja označen je s O te odgovara broju supojavljivanja dvije riječi u korpusu.

$$O = f_{ij} \tag{3.1}$$

Očekivani broj supojavljivanja E je broj supojavljivanja koji ovisi samo o frekvencijama riječi u korpusu i pretpostavlja nasumičnu distribuiranost riječi. Ukoliko je N ukupan broj pojavljivanja svih riječi u korpusu, vjerojatnost da je nasumično odabrana riječ upravo riječ i je $\frac{f_i}{N}$. Očekivani broj zajedničkog pojavljivanja riječi i i j je $\frac{f_i}{N} \cdot f_j$. Prema tome,

$$E = \frac{f_i\, f_j}{N} \tag{3.2}$$

.

Uzajamna informacija (engl. *(pointwise) mutual information*, PMI) jednostavna je mjera koja uspoređuje opaženo i očekivano supojavljivanje.

$$PMI = \log_2 \frac{O}{E} \tag{3.3}$$

Ukoliko je očekivano i opaženo supojavljivanje jednako, PMI iznosi 0. U tom je slučaju supojavljivanje dvije promatrane riječi slučajno. Ukoliko je opaženo supojavljivanje veće od očekivanog, PMI poprima pozitivnu vrijednost, a ukoliko je manje, riječi se „odbijaju“. Broj veći od nule indicira postojanje veze između dviju riječi, a visoke vrijednosti PMI odražavaju jaku povezanost, odnosno, učestalo supojavljivanje.

PMI je pristran rjeđim riječima zbog jake osjetljivosti na svako zabilježeno supojavljivanje u slučaju da je E relativno nizak. Kako bi se riješilo taj problem, korištena je mjera lokalne uzajamne informacije (engl. *local mutual information*, LMI), koja množi izračunati PMI s faktorom O.

$$LMI = O \cdot \log_2 \frac{O}{E} \tag{3.4}$$

Problem s LMI je taj što za riječi koje imaju negativnu asocijaciju ($O < E$) mjera ne mora davati niske rezultate. Ukoliko je E fiksiran, LMI postiže minimum za $O = \frac{E}{e}$ te ponovno raste za niže vrijednosti O.

Jednostavne mjere asocijacije imaju još jednu bitnu manu. Parovi riječi u kojima je jedna riječ vrlo česta, a druga rijetka i često se supojavljuje s prvom, imati će visoku vrijednost asocijacije, iako iz perspektive prve riječi ta asocijacija nije osobito jaka. Primjerice, u sintagmi „diplomski rad“; riječ „rad“ u korpusu se pojavljuje preko milijun puta, dok riječ „diplomski“ ima oko 15 000 pojavljivanja. Iako je očekivano da „diplomski“ bude jako asociran s riječju „rad“, obrnuta situacija nije poželjna. Dva su moguća rješenja ovog problema: korištenje asimetričnih mjera (takvih da $w(i, j) \neq w(j, i)$, gdje je w mjera asocijacije, a i i j su riječi) ili pokušaj kompenzacije drugačijom, simetričnom mjerom. Budući da je većina korištenih algoritama namijenjena za neusmjerene grafove, asimetrične mjere bi stvorile dodatne probleme. Rješenje je, dakle, u simetričnim mjerama koje bolje određuju asocijaciju nesimetričnih parova riječi – statističkim mjerama asocijacije.

Za korištenje statističkih mjera asocijacije nužno je prvo definirati nove varijable, R i C te umjesto po jednog očekivanog i opaženog broja supojavljivanja koristiti njih četiri. Cilj je prikazati četiri mogućnosti pojavljivanja para riječi; supojavljivanje riječi, pojavljivanje samo prve riječi, pojavljivanje samo druge riječi i nepojavljivanje obje

riječi. Redom, broj pojavljivanja svake od četiri mogućnosti označen je s O_{11}, O_{12}, O_{21} i O_{22}. Računanje je obavljeno prema tablicama:

Tablica 3.1: Opažena i očekivana pojavljivanja

(a) Opažena pojavljivanja

	i	$\neg j$	
i	O_{11}	O_{12}	R_1
$\neg i$	O_{21}	O_{22}	R_2
	C_1	C_2	

(b) Očekivana pojavljivanja

	i	$\neg j$
i	$E_{11} = \frac{R_1 C_1}{N}$	$E_{12} = \frac{R_1 C_2}{N}$
$\neg i$	$E_{21} = \frac{R_2 C_1}{N}$	$E_{22} = \frac{R_2 C_2}{N}$

Log–vjerodostojnost (engl. *log–likelihood*, $ln\mathcal{L}$, također i G^2) je, prema literaturi, vrlo dobra aproksimacija računalno puno složenijih mjera asocijacije. Jednako se dobro nosi sa slučajevima kad je $O \ll E$ i $E \ll O$. Oblikom je slična mjerama uzajamne informacije, ali daje bolje rezultate za nesimetrične parove riječi.

$$ln\mathcal{L} = 2 \sum_{xy} O_{xy} \ln \frac{O_{xy}}{E_{xy}}, \qquad xy \in \{11, 12, 21, 22\} \tag{3.5}$$

χ^2-test proširenje je jednostavne mjere *z-score*, koja procjenjuje koliko je dobra hipoteza $O = E$. U neproširenom obliku, visoka apsolutna vrijednost mjere znači da hipoteza loše modelira danu situaciju, odnosno, da supojavljivanje ne odgovara onome očekivanom. *Z-score* daje pozitivnu vrijednost za pozitivnu asocijaciju, a negativnu vrijednost za negativnu asocijaciju. Proširenje na statističku mjeru uzima u obzir sve vrijednosti O_{xy} i E_{xy}.

$$\chi^2 = \sum_{xy} \frac{(O_{xy} - E_{xy})^2}{E_{xy}}, \qquad xy \in \{11, 12, 21, 22\} \tag{3.6}$$

Malo drugačiji pristup ima sljedeća mjera, Diceov koeficijent (engl. *Dice coefficient*, također i *Sørensen coefficient*, *Dice's coefficient*). Umjesto polaženja od hipoteze $O = E$, Diceov koeficijent izravno računa udio riječi i koje se supojavljuju s j i udio riječi j koje se supojavljuju s i. Budući da je potrebno objediniti udijele tako da krajnji rezultat bude jedan broj, koristi se njihova harmonijska sredina kao rigorozna metoda usrednjavanja. Diceov koeficijent poprima vrijednosti blizu 1 ukoliko je asocijacija jaka s obje strane, a u slučaju asimetričnog para daje niže vrijednosti.

$$D = \frac{2O_{11}}{R_1 + C_1} = \frac{2f_{ij}}{f_i + f_j} \tag{3.7}$$

Diceov koeficijent odabrana je mjera za određivanje težina u grafu supojavljivanja kod mnogih autora, primjerice u radu (Di Marco i Navigli, 2013), na kojeg se ovaj rad često oslanja.

Jaccardov indeks (engl. Jaccard index) vrlo je bliska mjera Diceovom koeficijentu. Ona pokazuje omjer supojavljivanja dvije riječi i ukupnog broja pojavljivanja svake od riječi.

$$J = \frac{O_{11}}{O_{11} + O_{12} + O_{21}} = \frac{D}{2 - D} \tag{3.8}$$

Iako Jaccardov indeks ne donosi nove informacije u odnosu na Diceov koeficijent te vrijedi $D(i,j) > D(k,l) \iff J(i,j) > J(k,l)$, različito se ponašaju unutar aritmetičkih operacija, a Jaccardov indeks još više naglašava parove riječi s najvećom asocijacijom.

3.3.2. Odabir mjere asocijacije

Odabir mjere asocijacije koja će najvjernije prikazivati semantičku vezu potrebno je temeljiti na ponovljivim rezultatima pokusa i metodama koje minimiziraju trenutnu subjektivnost označavača. Usprkos tome, nakon odabira mjera, grafovi su pregledani tako što su za svaku riječ iz nasumično odabranog skupa analizirane najpovezanije (povezane bridovima najvećih težina) riječi. Osim potvrde da je metoda odabira mjere uspješna u smislu odabira najbolje mjere, takva analiza je dala uvid u to koliko je najbolja mjera zapravo dobra.

Korištena metoda odabira mjere asocijacije temelji se na javno dostupnom,[5] označenom skupu semantičke povezanosti između parova riječi opisanom u (Janković et al., 2011). Skup sadrži 450 parova riječi i usrednjenu ocjenu sličnosti (promatrajući ocjene svih 12 označavača i ocjene šest označavača s najvišim slaganjem). Kako bi se odredila kvaliteta mjere asocijacije u kontekstu određivanja semantičke sličnosti, korištena je mjera Kendallov τ–koeficijent (engl. *Kendall-τ coefficient*).

$$\tau = \frac{|P_c| - |P_d|}{\frac{1}{2}|P|(|P| - 1)} \tag{3.9}$$

P je skup parova svih parova u označenom skupu, odnosno kombinacija (p_{ij}, p_{kl}), gdje je par riječi $p_{ij} = (i,j)$.

$(w(i,j) > w(k,l) \wedge a(i,j) > a(k,l)) \Rightarrow (p_{ij}, p_{kl}) \in P_c$

$(w(i,j) < w(k,l) \wedge a(i,j) < a(k,l)) \Rightarrow (p_{ij}, p_{kl}) \in P_c$

$(w(i,j) > w(k,l) \wedge a(i,j) < a(k,l)) \Rightarrow (p_{ij}, p_{kl}) \in P_d$

[5] `http://takelab.fer.hr/data/crosemrel450/`

$(w(i,j) < w(k,l) \wedge a(i,j) > a(k,l)) \Rightarrow (p_{ij}, p_{kl}) \in P_d$

$(w(i,j) = w(k,l) \vee a(i,j) = a(k,l)) \Rightarrow (p_{ij}, p_{kl}) \notin P_c \wedge (p_{ij}, p_{kl}) \notin P_d$

pri čemu je $w(i,j)$ izračunati iznos analizirane mjere asocijacije, a $a(i,j)$ ocjena iz označenog skupa. Najmanji je iznos Kendallovog τ–koeficijenta -1, što označava potpuno krivi poredak, a najveći 1, što označava savršeno preklapanje poredaka dvije liste.

Tablica 3.2: Vrijednosti Kendallovog τ–koeficijenta za različite mjere asocijacije

Mjera asocijacije	6 označavača	12 označavača
PMI	0.378	0.422
LMI	0.419	0.477
$ln\mathcal{L}$	0.424	0.483
χ^2	0.464	0.52
Dice i Jaccard	**0.56**	**0.623**

Iz tablice 3.2 vidljivo je da Diceov koeficijent daleko najbolje oponaša semantičku sličnost između dvije riječi. Budući da Jaccardov indeks daje jednako rangirane parove riječi, isto vrijedi i za njega (rezultati su jednaki). S obzirom na to, Diceov koeficijent korišten je za određivanje težina bridova grafa, a Jaccardov indeks korišten je kao alternativa u pokusima u kojima nije svejedno koji je točan iznos težina dva brida dok god je odnos između njih jednak.

Rezultati u tablici 3.2 zanimljivi su iz još jednog razloga. Općenito, očekivano je da će vrijednosti za manji broj označavača biti veće nego vrijednosti za veći broj označavača. Ovdje to nije slučaj, vjerojatno zbog slabije granulacije bodova po parovima; više parova dijeli mjesto u slučaju šest označavača nego u slučaju 12 označavača.

3.4. Ostale metode modeliranja semantičke veze

Osim grafa supojavljivanja, semantičku sličnost moguće je modelirati drugim metodama, također koristeći informacije o supojavljivanju i distribuciji riječi kroz korpus. U nastavku su opisane metode i njihovi nedostaci, odnosno, prednosti nad grafom supojavljivanja. Ove se metode ne koriste u radu iz razloga pojašnjenih u nastavku, ali i zato što je cilj ovog rada fokusirati se isključivo na metode temeljene na strukturi grafa.

Jednostavna metoda u području distribucijske semantike je vreća riječi (engl. *bag–of–words*) (Harris, 1954). Ukoliko je svaka jedinstvena riječ indeksirana, tekst (doku-

ment, odlomak, rečenica i slično) može se prikazati vektorom koji na poziciji indeksa svake riječi ima njen broj pojavljivanja u tom tekstu. Takav prikaz ne uzima u obzir poredak riječi, njihov međusoban odnos niti gramatička obilježja, ali bilježi broj pojavljivanja pojedine riječi u tekstu. Umjesto broja riječi, element vektora može biti vrijednost dobivena primjenom neke težinske funkcije, primjerice tf–idf, što ujednačava iznose za tekstove različitih duljina i korigira težine ovisno o broju pojavljivanja riječi u drugim dokumentima (kako česte riječi ne bi utjecale previše na rezultate). Budući da je svaki tekst predstavljen vektorom iste duljine, gdje određena pozicija odgovara istoj riječi u svim vektorima, moguće je odrediti sličnost dvaju vektora. Primjerice, kosinusna sličnost (engl. *cosine similarity*) određuje kut između dva vektora u prostoru visoke dimenzionalnosti, što pokazuje na udaljenost dva vektora, odnosno sličnost (ili razliku) tekstova koje oni predstavljaju. U kontekstu otkrivanja značenja, kontekste u kojima se višeznačna riječ pojavljuje moguće je predstaviti vektorima i grupirati nekim algoritmom grupiranja, tako da jedna grupa sadrži slične vektore (kontekste). Za razgraničavanje značenja dovoljno je nekom metodom odrediti kojoj je grupi novi, neviđeni kontekst najbliži.

Problem ove metode je visoka dimenzionalnost vektora, koja može stvarati probleme u vremenskim ili memorijskim zahtjevima izvođenja postupka. Također, ova metoda ne otkriva sakrivenu semantičku strukturu; ukoliko dva različita teksta sadrže riječi koje su sinonimi, oni ne doprinose sličnosti dva teksta, čak štoviše, udaljuju vektore. To je posebno neželjeno kad su tekstovi kratki, kao što je slučaj s predstavljenim kontekstima višeznačnih riječi.

Latentna semantička analiza (engl. *latent semantic analysis* (Deerwester et al., 1990)) rješava problem otkrivanja semantičke strukture koristeći prethodno analizirani korpus. Ideja je izgraditi matricu u kojoj reci predstavljaju riječi iz korpusa, a stupci pojedini dokument. Vrijednosti u matrici govore koliko se puta pojedina riječ pojavljuje u nekom dokumentu ili su izračunate nekom težinskom mjerom (primjerice, kao i u prethodnom slučaju, tf–idf). Dekompozicijom singularnih vrijednosti (engl. *singular value decomposition*) matrica je transformirana u oblik u kojem joj je moguće reducirati rang. Redukcijom ranga smanjeni su memorijska zahtjevnost i šum u podacima. Također, redukcija ranga dovodi do pojave u kojoj neka dimenzija ovisi o više od jedne riječi (kako je bilo u početnoj matrici), već je predstavljena kao linearna kombinacija više riječi, koje su obično semantički slične (semantička sličnost inducirana je na taj način iz korpusa), primjerice, sinonimi. Također, riječi koje su višeznačne mogu utjecati na više različitih dimenzija u novoj matrici; svaka dimenzija predstavlja jedno značenje ili barem kontekst. Negativna strana ove metode je ogromna memorijska slo-

ženost (dekomponirana matrica nije rijetka kao originalna matrica riječ–dokument), kao i slaba interpretabilnost rezultata.

Nasumično indeksiranje (engl. *random indexing* (Sahlgren, 2005)) i latentna Dirichleteova alokacija (engl. *latent Dirichlet allocation* (Blei et al., 2003)) rješavaju navedene probleme te su također metode koje modeliraju teme pomoću korpusa dokumenata i distribucije riječi po tim dokumentima. Nasumično indeksiranje koristi prikaz riječi u prostoru smanjenog broja dimenzija, ali bez gubitka točnosti u računanju udaljenosti među riječima. U tom pogledu ima prednosti nad vrećom riječi i LSA. Latentna Dirichleteova alokacija polazi od pretpostavke da je svaki tekst mješavina latentnih (neopaženih) tema, a svaku temu predstavlja distribucija riječi vezanih uz nju. Koristeći postupak otkrivanja latentnih slučajnih varijabli, gradi se generativni model. Rezultati su obično bolje interpretabilni nego kod LSA. Ipak, ove dvije metode ne donose značajnu razliku u metodama otkrivanja i razgraničavanja značenja od prethodno opisanih.

3.5. Specifikacije programskog ostvarenja

Glavni dio funkcionalnosti implementiran je u programskom jeziku *Java*[6] (verzija 1.7) i testiran na 64-bitnoj verziji operacijskog sustava *Ubuntu 13.10*.[7] Dio funkcionalnosti ovisan je o jezgri operacijskog sustava Linux, tako da je za upotrebu na drugim sustavima nužno načiniti preinake. Korištena je baza podataka *Apache Derby*.[8] Programska biblioteka *Apache POI*[9] korištena je kao API za čitanje dokumenata Microsoft Excela (`.xlsx`) u kojima su oznake potrebne za evaluaciju.

Neki od elemenata sustava su preuzeti kao gotova rješenja. Korišteni algoritam *MCL*[10] (van Dongen, 2000) dostupan je unutar operacijskog sustava, odnosno njegovih repozitorija. Lematizator *CST's Lemmatiser*[11] i morfološki označavač *HunPos tagger*[12] koriste modele opisane u (Agić et al., 2013). Ostatak programskog rješenja je samostalno implementiran.

Za optimalan rad sustava procesorska brzina nije presudna; algoritmi koji se izvršavaju nad strukturom cijelog grafa su niske složenosti te na vrijeme izvršavanja najviše

[6] `https://www.java.com/en/`
[7] `http://www.ubuntu.com/`
[8] `https://db.apache.org/derby/`
[9] `https://poi.apache.org/`
[10] `http://micans.org/mcl/`
[11] `http://cst.dk/online/lemmatiser/uk/`
[12] `https://code.google.com/p/hunpos/`

utječe brzina pristupa datotekama na disku (ukoliko se koriste operacije čitanja i pisanja). Nakon što je graf izgrađen, za pojedinu višeznačnu riječ gradi se graf znatno manjih dimenzija, pa upotreba algoritama veće složenosti ne predstavlja problem; oni se izvršavaju naizgled trenutno. S druge strane, memorijski zahtjevi mogući su problem. Naime, u pojedinim je trenucima nužno imati cijelu strukturu grafa supojavljivanja u radnoj memoriji. Problem leži u visokoj prostornoj složenosti, $O(n^2)$, gdje je n broj jedinstvenih riječi, odnosno vrhova grafa. Iako se radi o relativno rijetkoj matrici, u pokusima je Java virtualni stroj (unutar kojeg se program izvodi) zauzimao između 6 i 7 GB radne memorije. Ta činjenica ne čudi, s obzirom da se jednostavnom računicom, za n = 100 000 dolazi iznad 37 GB ukoliko matrica nije rijetka, koriste se decimalni brojevi dvostruke preciznosti, a polovica matrice je zanemarena (jer je matrica simetrična). Uračunavši dodatnu memoriju koja je zauzeta zbog korištenja neprimitivnih tipova podataka, visoki memorijski zahtjevi su opravdani.

4. Otkrivanje značenja riječi

Ideja rada je, nakon što je težinski graf izgrađen, grupirati njegove vrhove kako bi svaka grupa odgovarala nekom značenju višeznačne riječi. U nastavku su prikazani postupci koji prethode grupiranju grafa, algoritmi grupiranja i odabir optimalnih parametara za korištene algoritme.

4.1. Filtriranje vrhova grafa

Zaključak donesen na temelju preliminarnih pokusa je da pristup u kojem se cijeli graf grupira kako bi se otkrili konteksti koji opisuju značenja nije optimalan. Čak i ako se zanemari problem veličine grafa i poteškoća vezanih uz nju (trajanje izvođenja algoritama grupiranja, poteškoće u podešavanju parametara), pristup i dalje ne daje dobre rezultate. Naime, grupiranje cijelog grafa u grupe (kontekste) primjerene za otkrivanje i razgraničavanje značenja svih višeznačnih riječi teško je moguće. Kontekst jednog značenja višeznačne riječi ne mora se potpuno preklapati s kontekstom značenja druge riječi. Primjerice, određeno značenje jedne višeznačne riječi može se pojavljivati u nekoliko konteksta druge višeznačne riječi; u svakom slučaju takav pristup mora dovesti do pada kvalitete rada. Također, općenito govoreći, kontekst jedne riječi ne mora biti kontekst druge. Grupe u cijelom težinskom grafu nisu prirodno odvojene, već su granice među njima vrlo nejasne. Zbog toga je korištena metoda filtriranja slična onoj korištenoj u srodnim radovima, ali uz veći broj parametara.

Moguće je odrediti skup tema s proizvoljnom razinom granulacije (primjerice, metodama opisanima u poglavlju 3.4), ali to nije nužno dovoljno dobro za sve višeznačne riječi. Kod nekih se višeznačnica velika razlika između značenja krije u malim razlikama u kontekstima. Bez posebnog grupiranja za svaku riječ, takva značenja biti će izostavljena ili spojena.

Bez predfiltriranja grafa, tako da ostanu samo riječi vezane uz promatranu višeznačnicu, koje tvore kontekste određene prema točno toj riječi, teško je dobiti dobar (ispravan) skup značenja. Također, filtriranjem se može doći do skupa riječi koji će,

nakon provođenja grupiranja, biti lako interpretabilan.

Predfiltriranje težinskog grafa svodi se na uklanjanje vrhova ili bridova koji ne zadovoljavaju određena svojstva. Počevši od promatrane višeznačne riječi, u skup filtriranih riječi dodaju se one koje imaju dovoljno veliku težinu brida (iznos težine iznad određenog praga w_1. Time se u obzir uzimaju samo one riječi koje su dovoljno bitne u kontekstu promatrane riječi. U skup se zatim dodaju riječi koje s riječima dodanima u prethodnom koraku imaju dovoljno veliku težinu brida (iznad praga w_2, za kojeg obično vrijedi $w_2 > w_1$). Na taj način dodane su riječi koje se ne pojavljuju zajedno s promatranom riječi, ali mogu biti bitne, primjerice sinonimi pojedinog značenja. Drugim riječima, na ovaj se način u skup filtriranih riječi dodaju supojavljivanja drugog reda. Riječi iz skupa filtriranih riječi zatim se spajaju bridovima iz početnoga grafa, tako da težina brida bude iznad praga w_3. Iako algoritmi uglavnom uklanjaju nebitne bridove, nekima je ovaj korak filtriranja bridova od pomoći. U svakom se koraku provjerava broj pojavljivanja riječi; riječi koje su prečeste (frekvencije iznad f_1) i riječi koje su prerijetke (frekvencije manje od f_2). Ovaj korak također nije od presudne važnosti u predfiltriranju, ali u nekim slučajevima može pomoći. Posljednji je korak izbacivanje vrhova nedovoljno visokog stupnja (broja susjednih vrhova manjih od d_1). Ukoliko je minimalan stupanj vrhova jednak 1, uklanjaju se oni vrhovi koji nisu povezani s grafom. Ukoliko je minimalan stupanj jednak 2, uklanjaju se jednostruko povezani lanci vrhova.

Odabir parametara opisan je u poglavlju 4.3. Obično u filtriranju nisu korišteni svi kriteriji, već samo dio njih, ovisno o potrebi i svojstvima korištenih algoritama grupiranja.

Nakon provođenja algoritma grupiranja pokazalo se korisnim ukloniti riječi prevelike ili preniske frekvencije ukoliko taj korak nije proveden u predfiltriranju.

Nakon filtriranja i grupiranja, očekivani rezultat su precizne grupe riječi koje zajedno opisuju određeno značenje. Trebale bi biti dovoljno kratke i interpretabilne da ljudi nemaju problema s njihovim čitanjem i razumijevanjem, a opet dovoljno velike da budu iskoristive u razgraničavanju značenja kod neviđenih konteksta.

4.2. Metode grupiranja vrhova grafa

U statistici i strojnom učenju postoji nekoliko vrsta grupiranja. Particijsko grupiranje (engl. *parititional clustering*, *flat clustering*) grupira elemente u različite particije, među kojima ne postoji eksplicitna struktura. S druge strane, hijerarhijsko grupiranje (engl. *hierarchical clustering*) elemente dijeli u ugniježđene grupe među kojima pos-

toji hijerarhija. Također, ukoliko promatramo čvrstoću granica između grupa, postoje čvrsto grupiranje (engl. *hard clustering*) i meko grupiranje (engl. *soft clustering*). U čvrstom grupiranju svaki element može pripadati samo jednoj (ili najviše jednoj) grupi, dok u mekom grupiranju postoje razine pripadnosti, odnosno, vjerojatnosti pripadanja pojedinoj grupi, tako da element može biti podijeljen između više grupa. U ovom je radu naglasak na čvrstom particijskom grupiranju. Ostali pristupi ukratko su opisani u potpoglavlju 4.2.8.

Grupiranje vrhova grafa obično se provodi uklanjanjem vrhova, uklanjanjem bridova ili mješavinom ta dva pristupa. Iako postupak ne mora eksplicitno slijediti te paradigme, takva se promjena implicitno događa. Moguće je upitati se (1) kada je opravdano ukloniti brid, (2) kada je opravdano ukloniti vrh i (3) o čemu sve treba voditi računa kako bi grupiranje dalo zadovoljavajuće rezultate.

Prvo je pitanje najjednostavnije; brid se uklanja ukoliko on u grafu (ili lokalno, u dijelu grafa) nema dovoljnu relativnu važnost (težinu) da bi ga trebalo ostaviti. Uklanjanjem bridova moguće je razdvajanje podgrafa iz kojeg je brid uklonjen na dva nepovezana podgrafa. Upravo to je grupiranje grafa: podjela na međusobno nepovezane podgrafe od kojih svaki predstavlja neki kontekst višeznačne riječi.

Drugo pitanje ima manje očit odgovor. Prema (Dorow et al., 2004), vrh je nepotreban ili nepoželjan onda kad je riječ koju predstavlja višeznačna. Naime, višeznačne riječi obično se pojavljuju u različitim kontekstima, pa u grafu spajaju podgrafe koji bi inače trebali biti nepovezani, različiti konteksti. Uklanjanjem višeznačnih vrhova konteksti se razdvajaju. Više o odluci o broju značenja pojedinog vrha nalazi se u potpoglavlju 4.2.2.

Treće pitanje vrlo je općenito i odgovor na njega ovisi o primjeni i potrebama primjenskog sustava. Obično je neželjeno imati puno nepovezanih vrhova ili neravnomjeran broj elemenata u svakoj grupi. Ostala svojstva grupa teško je karakterizirati kao pozitivna ili negativna. Ponekad je dobro imati manje i zbijenije (dobro povezane) grupe, gdje svaki vrh vrlo vjerojatno i semantički spada u svoju grupu. S druge strane, nekad je potrebno imati razgranate i manje povezane grupe, tako da svi mogući konteksti vezani uz jedno značenje budu grupirani zajedno. Svojstva grupa koja nastaju kao rezultat grupiranja ovise o pojedinom algoritmu, ali i parametrima algoritma. Algoritmi, odnosno metode grupiranja i njihova svojstva opisani su u nastavku poglavlja 4.2, a podešavanje parametara u poglavlju 4.3.

Za svaku je metodu u nastavku dan primjer njenog izvršavanja nad pokusnim grafom. Rezultati nisu reprezentativni već ilustrativni i ne donose pregled stvarnih performansi algoritama. Parametri za ilustraciju algoritama nisu optimizirani. Kao primjer

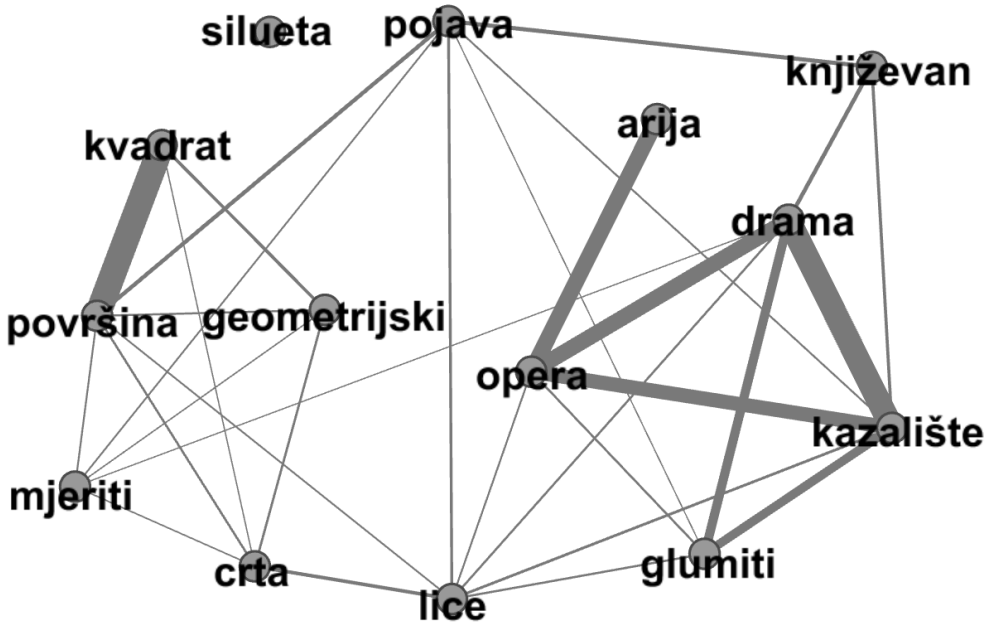
korištena je višeznačna riječ „lik“. Prema jednom od mogućih tumačenja, ona ima tri značenja (lik u fiktivnom djelu, geometrijski konstrukt te obris ili izgled osobe). U nastavku je tablica težina neusmjerenih bridova izračunata kao Diceov koeficijent (tablica 4.1) dobivena iz grafa supojavljivanja korištenog u implementaciji sustava te grafički prikaz grafa (slika 4.1). Deblja linija brida označava veću težinu. Riječi u grafu odabrane su ručno tako da predstavljaju tri predložena značenja.

Tablica 4.1: Težine bridova pokusnog grafa

Prva riječ	Druga riječ	Težina brida (10^{-4})	Prva riječ	Druga riječ	Težina brida (10^{-4})
kazalište	književan	9.4	površina	crta	5.5
kazalište	pojava	3.5	površina	kvadrat	124
kazalište	opera	65	površina	lice	2.9
kazalište	lice	7.7	arija	opera	73.8
mjeriti	geometrijski	1.3	crta	mjeriti	1.7
pojava	književan	14.6	crta	geometrijski	5.8
pojava	mjeriti	2.9	crta	kvadrat	0.8
drama	književan	14.5	crta	lice	11.7
drama	kazalište	114	kvadrat	geometrijski	9.9
drama	mjeriti	0.6	lice	pojava	8
drama	opera	72.3	lice	opera	3.5
drama	lice	5.1	glumiti	kazalište	3.8
drama	glumiti	39.6	glumiti	pojava	0.6
površina	mjeriti	3.3	glumiti	opera	5.3
površina	pojava	14	glumiti	lice	5
površina	geometrijski	4.3			

4.2.1. Algoritam B–MST

Minimalno razapinjuće stablo (engl. *minimum spanning tree*) čest je pojam kako u teoriji grafova, tako i u računarskoj znanosti (među ostalim). Gradi se iz povezanog grafa, kojeg karakterizira činjenica da se iz jednog vrha može doći do bilo kojeg vrha, bilo neposredno, bilo prelaskom preko drugih vrhova. Cilj je dobiti takav graf u kojem je broj bridova minimalan, $|V| - 1$, gdje je V skup vrhova, a da graf ne izgubi svojstvo povezanosti. Minimalno razapinjuće stablo je graf koji, pridržavajući se prethodna



Slika 4.1: Pokusni graf (debljina bridova označava stupanj sličnosti)

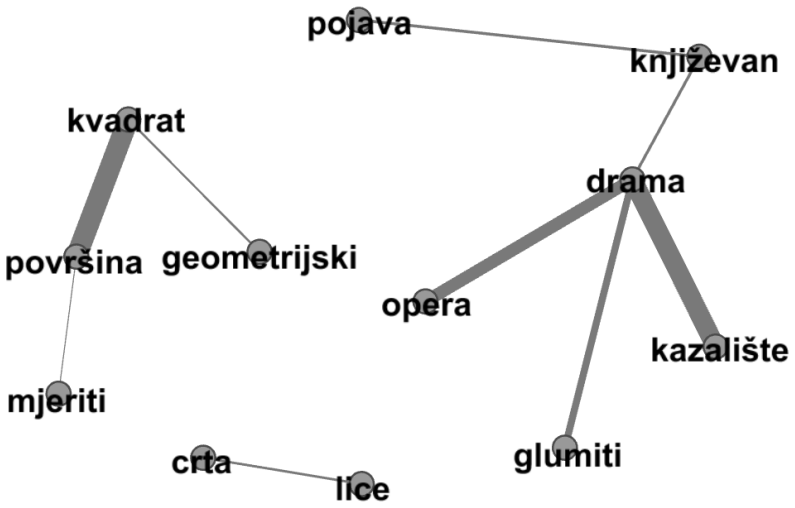
dva zahtjeva zadovoljava još jedan: ukupan zbroj težina bridova je najmanji mogući. Maksimalno razapinjuće stablo (engl. *maximum spanning tree*) je vrlo slično, osim što zadnji uvjet umjesto minimalnog zbroja težina zahtijeva maksimalan zbroj.

B–MST, Balansirano maksimalno razapinjuće stablo (engl. *Balanced Maximum Spanning Tree*, (Di Marco i Navigli, 2011), nadograđen u (Di Marco i Navigli, 2013)). Ideja je algoritmom za određivanje maksimalnog razapinjućeg stabla doći do skupa najznačajnijih bridova te pažljivo ukloniti one najmanje bitne među njima kako bi preostale čvrsto povezane grupe. Algoritam se provodi u nekoliko koraka. U prvom se koraku uklanjaju svi vrhovi kojima je stupanj jedan, odnosno, oni koji su povezani samo s jednim drugim vrhom, pa čine svojevrsnu „slijepu ulicu“. Takvi vrhovi dijelom pobijaju smisao pronalaženja skupa najvažnijih bridova jer ne postoji mogućnost izbora nekog drugog brida za vrh stupnja jedan. U drugom se koraku za tako smanjeni graf određuje maksimalno razapinjuće stablo. Nakon toga preostaje uklanjanje bridova. Jedini parametar algoritma je broj grupa koje će na kraju preostati (N). Ukoliko se prvobitno povezanom grafu ukloni N bridova, vrhovi će biti razdijeljeni na N grupa, što znači da je dovoljno ukloniti N bridova najmanje težine kako bismo zadovoljili početne zahtjeve. Na taj način radi originalni algoritam (MST). Razliku u B–MST čini balansiranost njegovih rezultata. To se odnosi na postizanje ravnoteže između broja vrhova u grupama koje nastaju kao rezultat primjene algoritma. Dodatno ograničenje osigurava upravo takvu ravnotežu među grupama vrhova. Kako bi spriječili nastajanje

vrlo malih grupa, ograničenje zabranjuje uklanjanje brida ukoliko će jedna od dvije tako dobivene nove grupe imati manje od $\frac{|V|}{2N}$ vrhova.

Dodatnu pažnju zaslužuje algoritam izgradnje razapinjućeg stabla. Korišten je Kruskalov algoritam, koji polazi od potpuno nepovezanog skupa vrhova. Polazeći od brida ekstremne težine (najveće ili najmanje), krećući se monotono prema suprotnom ekstremu, dodaju se bridovi između vrhova. Jedini je uvjet osigurati da se ne spajaju vrhovi koji su već spojeni posredno preko drugih vrhova. U slučaju dobre implementacije, složenost izvođenja je relativno niska, $O(|E| \log |V|)$ (Cormen et al., 2001).

Prednost ovog algoritma je mali broj parametara (samo jedan), ali je taj parametar ujedno i mana; potrebno je unaprijed odrediti broj grupa, odnosno značenja riječi, što je teško unaprijed odlučiti (odrediti broj značenja za općeniti slučaj). Jedna je mogućnost odrediti broj grupa nekim drugim algoritmom te predati dobiveni broj grupa kao parametar u B–MST. Druga je mogućnost iskoristiti uočenu korelaciju između frekvencije riječi u korpusu i broja njenih značenja te koristiti neku funkciju frekvencije riječi kao parametar.



Slika 4.2: Graf dobiven metodom B–MST s parametrom $k = 3$. Prvobitno nepovezan vrh *silueta* izbačen je jer ga nije bilo moguće spojiti u razapinjuće stablo, a vrh *arija* uklonjen je jer je bio stupnja 1.

4.2.2. Algoritam SquaT++

SquaT++ je algoritam temeljen na radu (Dorow et al., 2004), kasnije razvijen u (Navigli i Crisafulli, 2010) kao SquaT i dovršen u (Di Marco i Navigli, 2013). Prva verzija algoritma računala je važnost svakog pojedinog vrha (originalni naziv mjere je *curvature*, predstavljen u radu (Eckmann i Moses, 2002) van konteksta otkrivanja značenja) kao omjer broja trokuta (odnosno, ciklusa u grafu duljine tri vrha) u kojima vrh v sudjeluje i broja trokuta u kojima bi v mogao sudjelovati. Izabran je trokut jer broj trokuta može ukazivati na broj značenja pojedine riječi; ukoliko su dva vrha povezana bridom, nije jasno koliko je različitih značenja tog para riječi i koje je značenje prve riječi povezano s kojim značenjem druge. Međutim, ako par riječi istovremeno ima poveznice prema drugim riječima, do te mjere da im je većina susjeda zajedničko, vjerojatno je da obje riječi imaju samo po jedno značenje, odnosno, da se uglavnom pojavljuju u istim kontekstima. Vrhovi koji nemaju većinu susjeda zajedničku vjerojatnije se pojavljuju u više različitih konteksta i, prema tome, imaju više značenja. Vrhovi s malim brojem značenja mogu se promatrati kao *semantička sidra*, i daju vrijednu informaciju o kontekstu u kojem se nalaze, odnosno, jak su pokazatelj određenog konteksta.

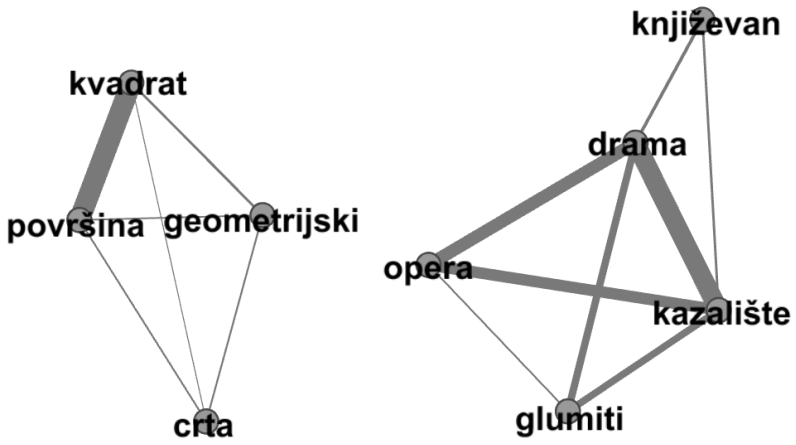
SquaT osim trokuta promatra pojavljivanje novog lika u grafu – četverokuta (što objašnjava ime, *Squares and Triangles*). Motivacija je ista kao i za trokut, a autori smatraju da korištenje podatka o broju četverokuta u kojima vrh sudjeluje i broju četverokuta u kojima bi mogao sudjelovati doprinosi kvaliteti rješenja. SquaT++ (*Squares, Triangles and more*) je konačna nadogradnja, u kojoj se uz trokute i četverokute promatraju i četverokuti s dijagonalom, odnosno podgrafovi s četiri vrha i pet bridova.

Ako je $Tri(v)$ omjer broja postojećih i broja mogućih trokuta, $Sqr(v)$ omjer broja omjer broja postojećih i broja mogućih četverokuta, a $Dia(v)$ omjer broja postojećih i broja mogućih četverokuta s dijagonalom za promatrani vrh v, SquaT++ je njihova linearna kombinacija: $SquaT{+}{+}(v) = \alpha\, Tri(v) + \beta Sqr(v) + \gamma Dia(v)$. Vrijednosti α, β i γ su parametri algoritma. Za $\beta = \gamma = 0$ algoritam je sveden na originalni oblik, a za $\gamma = 0$ na SquaT. Četvrti parametar je numerički prag σ ispod kojeg $SquaT{+}{+}(v)$ mora biti kako bi se iz grafa uklonio vrh v.

Bitno je dodati da je algoritam primjenjiv i na bridove grafa, jer je i za njih moguće odrediti broj ciklusa određene dužine u kojima oni sudjeluju ili bi mogli sudjelovati. Takva varijanta algoritma blaža je jer nema uklanjanja vrhova, pa je skup riječi koje preostanu u grafu veći, a istovremeno omogućuje lakše podešavanje parametara, uz manju računalnu složenost.

Iako je motivacija za korištenje ovog algoritma opravdana, a premise na kojima je

zasnovan ispravne, postoje nedostaci. Prvi je veliki broj parametara, koje je vrlo teško fino ugoditi. Osim toga, složenost algoritma vrlo je visoka te dostiže $O(n^4)$ zbog pronalaženja ciklusa duljine 4. Također, algoritam nije jednoznačno određen. Niti u jednom radu u kojem su korišteni ciklusi duljine veće od tri (SquaT i SquaT++) ti ciklusi nisu jasno definirani. Primjerice, ukoliko promatramo kvadrate, nije jasno ubrajaju li se u njih i ciklusi duljine četiri vrha, ali s više od četiri brida. Osim toga, nejasno je i na koji način je zamišljeno brojanje mogućih kvadrata i složenijih ciklusa, odnosno, koliko je bridova dopušteno dodati kako bi se ti potencijalni ciklusi zatvorili. Budući da je u ovom radu analiziran rad samostalno implementiranog algoritma, nema garancije optimalne implementacije, odnosno rekonstrukcije uspješnosti kakvu su imali autori.



Slika 4.3: Graf dobiven metodom SquaT++ za vrhove, parametri su $\alpha = 0.3$, $\beta = 0.3$, $\gamma = 0.4$, $\sigma = 0.3$. Nekoliko vrhova je izbačeno, ali su svi bridovi među preostalim vrhovima ostali netaknuti. Algoritam dovodi do vrlo povezanih manjih grupa.

4.2.3. Algoritam „pokvarenog telefona“

Algoritam „pokvarenog telefona“ (engl. *Chinese Whispers*, (Biemann, 2006)) nazvan je prema popularnoj dječjoj igri u kojoj igrači šapću pojam koji su čuli od jednog suigrača drugom suigraču (tako da se greška može pojaviti u svakom koraku, s humorističnim ishodom). Ime opisuje postupak, u kojem susjedni vrhovi utječu na pojedini

vrh (i njegov odabir grupe), što se kroz određeni broj iteracija ponavlja. Umjesto dijeljenja početnog grafa (*top–down* pristup), ovdje algoritam polazi od grafa u kojem svaki vrh pripada svojoj grupi te se grupe iterativno spajaju (*bottom–up*).

U početku je graf podijeljen na $|V|$ grupa. U svakoj se iteraciji za svaki $v \in V$, gdje su v analizirani nasumičnim redoslijedom, određuje grupa kojoj je najbliži. Odspajanjem v od ostatka grafa moguće je pronaći takvu grupu za koju je zbroj bridova koji sadrže v maksimalan. Formalno, za svaki $v \in V$, nasumičnim redoslijedom, za svaku iteraciju odabir grupe c odvija se na sljedeći način:

$$cluster(v) = \underset{c}{\operatorname{argmax}} \sum_{\substack{(v,v') \in E \\ cluster(v')=c}} w(v, v')$$

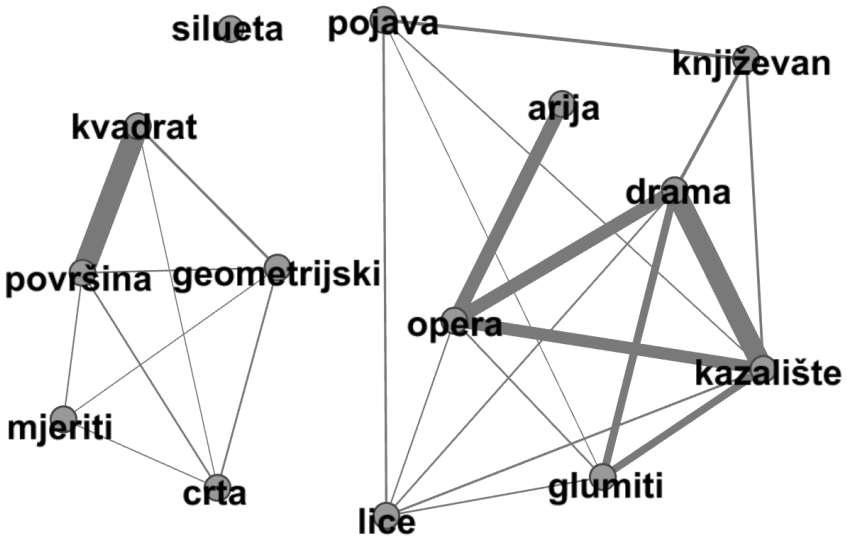
U trenutku kad iteracija ne donese promjene u odnosu na onu prethodnu, algoritam staje, kao i u slučaju da je broj iteracija postigao zadani maksimum. Iako algoritam ne traži definiranje tog parametra, postoji mogućnost zaglavljivanja u beskonačnoj petlji, odnosno, titranja oko konačnog rješenja, pa je korisno dodati takvo ograničenje. Algoritam je nedeterministički s obzirom na konačno rješenje, jer dva različita izvršavanja algoritma mogu dati različite rezultate. Ipak, autori smatraju da u dovoljno velikom grafu ti problemi ne igraju veliku ulogu. Broj situacija u kojima je moguće dodijeliti više grupa istom vrhu nije čest, pa je dovoljan vrlo mali broj iteracija kako bi algoritam došao do konačnog rješenja. U pokusima izvedenima za potrebe ovog rada niti jednom nije došlo do beskonačne petlje, a izvršavanje nad istim grafom rezultiralo je istim ili gotovo istim grupiranjima.

Prednost algoritma je u činjenici da ne zahtijeva nikakve parametre, već je općenit. Također, vrijeme izvršavanja je u većini slučajeva vrlo kratko.

4.2.4. Algoritam HyperLex

Algoritam HyperLex predstavio je Véronis (2004). Umjesto uklanjanja bridova ili vrhova, ideja je redom odabirati bitne čvorove (engl. *hubs*), odnosno vrhove koji su najbolji predstavnici određenog konteksta. Izvodi se u nekoliko koraka. Prvo je potrebno izgraditi listu vrhova L sortiranu prema broju pojavljivanja odgovarajuće riječi u korpusu, od najčešće do najrjeđe. Kako bi vrh bio izabran za čvor, mora zadovoljavati dva uvjeta:

$$\frac{deg(v)}{max_{v' \in V} deg(v')} \geq \sigma_1$$



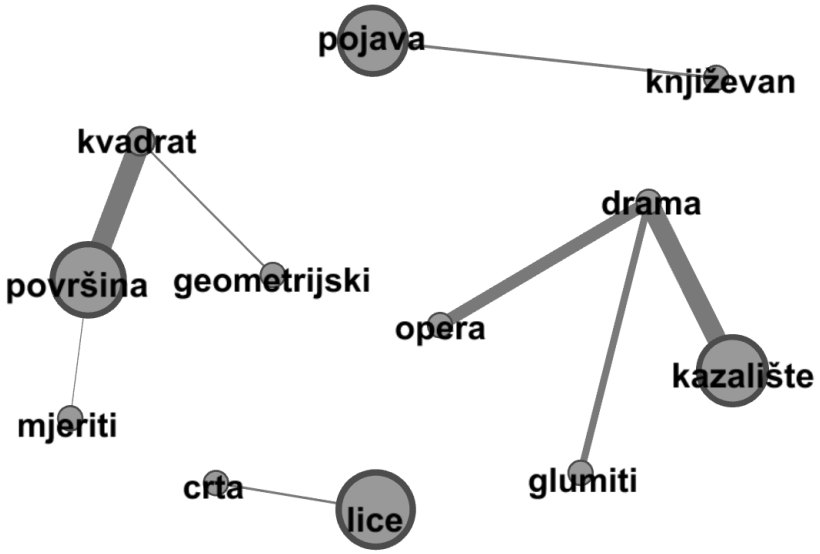
Slika 4.4: Graf dobiven algoritmom „pokvarenog telefona“. Svi vrhovi početnog grafa su ostali, a uklonjene su samo veze između vrhova koji pripadaju različitim grupama.

$$\frac{\sum\limits_{(v,v')\in E} w(v,v')}{deg(v)} \geq \sigma_2$$

$deg(v)$ označava stupanj vrha v, a $w(v, v')$ težinu brida između vrhova v i v'. Parametri algoritma su σ_1 i σ_2.

Nakon što je vrh iz L odabran za čvor, uklanja se iz L, zajedno sa svim svojim susjedima. Razlog tome je izbjegavanje situacije u kojoj bi dva susjedna vrha bila odabrana za čvorove zbog toga što se predstavnici različitih konteksta ne bi smjeli previše supojavljivati. Algoritam staje u trenutku kad prvi element iz L ne zadovoljava uvjete ili kad se L isprazni.

Odabrani čvorovi predstavljaju kontekste, ali vrhovi grafa još nisu grupirani. Postupak grupiranja relativno je jednostavan nakon što su čvorovi poznati. U početni graf iz kojeg su čvorovi odabrani dodaje se još jedan vrh, v'', koji predstavlja promatranu višeznačnu riječ. Novi je vrh spojen s odabranim čvorovima grafa bridovima beskonačne težine. Nakon toga, potrebno je samo sagraditi maksimalno razapinjuće stablo te na kraju ukloniti v''. Budući da u početnom grafu nema bridova beskonačne težine, postupak jamči da će u početku izgradnje maksimalnog razapinjućeg stabla odabrani čvorovi biti povezani preko v''. Svi ostali vrhovi biti će povezani s v'' preko točno jednog čvora. Uklanjanjem v'' stablo se rastavlja na broj grupa jednak broju čvorova.



Slika 4.5: Graf dobiven metodom HyperLex, parametri su $\sigma_1 = 0.25$, $\sigma_2 = 0.001$. Veći vrhovi predstavljaju odabrane čvorove. Cijelo razapinjuće stablo jednako je prikazanom grafu, uz dodatak jednog vrha (*lik*) i bridova između tog vrha i odabranih čvorova. Vrh *arija* nestao je jer odlučeno izbacivati vrhove stupnja 1 pri izgradnji maksimalnog razapinjućeg stabla.

4.2.5. Algoritam PageRank

PageRank (Page et al., 1999) algoritam osmišljen je kako bi mjerio važnost pojedine web–stranice, s ciljem boljeg rangiranja rezultata pretrage, tako da važnije stranice budu više rangirane. Najpoznatiji je po svojoj primjeni u rangiranju rezultata pretrage tražilice Google.[1] Ukoliko je mreža web stranica prikazana kao graf, taj graf nije težinski, a bridovi su mu usmjereni; brid pokazuje na postojanje poveznice iz jednog vrha (stranice) u drugi vrh (stranicu). Najvažnije su one stranice na koje postoje poveznice s velikog broja drugih stranica. Na važnost stranica utječe važnost stranica koje na nju pokazuju. Određivanje važnosti stranica iz tog je razloga težak zadatak. Priča koja ilustrira ponašanje i opravdava metodu PageRanka je sljedeća: osoba pregledava web stranice tako što nasumično odabire poveznicu na stranici na kojoj se trenutno nalazi. U svakom trenutku, uz vjerojatnost $1 - d$ (d se naziva faktorom prigušenja, engl. *damping factor*, $0 \leq d \leq 1$) prestaje s pretraživanjem i kreće od početka, prelaskom na stranicu koja nije nužno povezana sa stranicom na kojoj se nalazi. Gledano u okviru grafa definiranog ranije u ovome radu, vjerojatnost kretanja od početka, od vrha v, je $\frac{1-d}{|V|}$. Vjerojatnost prelaska na vrh v nasumičnim odabirom poveznice s

[1] `https://www.google.com/`

prethodne stranice je $d \sum_{(v',v)\in E} \frac{1}{|L_{v'}|}$, gdje je $L_{v'}$ skup poveznica na stranici v'. Jedna verzija formule za izračun PageRanka je

$$PR(v) = \frac{1-d}{|V|} + d \sum_{(v',v)\in E} \frac{PR(v')}{|L_{v'}|}$$

Važno je primijetiti da se ovdje radi o usmjerenom grafu te za bridove vrijedi $(v', v) \not\equiv (v, v')$.

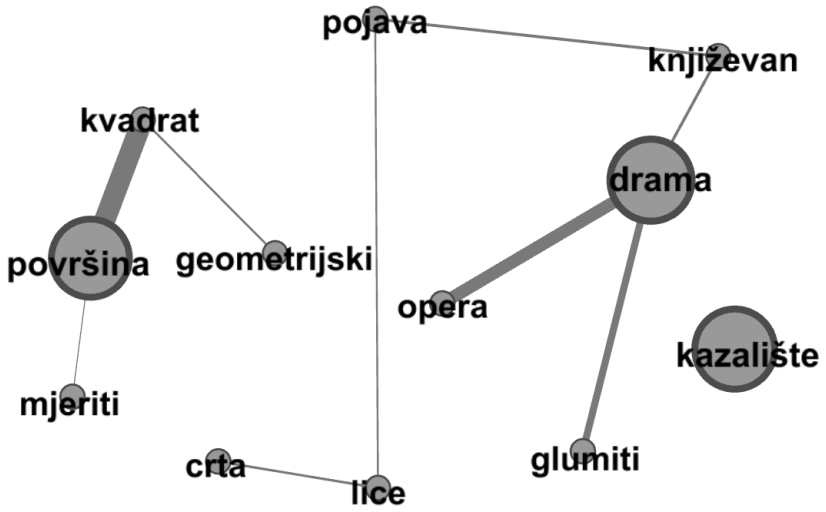
Rješenje je za velike količine podataka (primjerice, s količinama kojima barata Google) u domeni linearne algebre, dok je prikaz u obliku grafova samo ilustrativan. Na manje količine podataka, kao što su filtrirani grafovi supojavljivanja, moguće je elegantno primijeniti PageRank.

Agirre et al. (2006) predlažu primjenu PageRanka u određivanju značenja riječi. Iako je algoritam prethodno korišten u nenadziranom razrješavanju višeznačnosti, citirani je rad ovdje glavni oslonac u implementaciji i analizi PageRanka. Glavni je zadatak pretvoriti neusmjereni težinski graf u oblik koji je pogodan za rad PageRanka.

$$PR(v) = (1-d) + d \sum_{(v,v')\in E} \frac{w(v,v')}{\sum_{(v',v'')\in E} w(v',v'')} PR(v')$$

Umjesto vjerojatnosti prelaska, drugi dio formule prikazuje snagu s kojom v' pokazuje na v, pomnoženu s važnosti vrha v'. Snaga pokazivanja je omjer težine brida (v, v') i zbroja težina svih bridova koji su susjedni s v'. Ako je iznos tog omjera blizu 1, znači da je povezanost para (v, v') vrlo visoka i da v' vrlo snažno pokazuje na v. Bitno je primijetiti da se ovdje ne radi o simetričnoj mjeri, jer nije poznato ništa o susjedima v i težini brida (v, v') u odnosu na ostale bridove koji su incidentni s v. Iz toga je vidljivo da je korištenjem neusmjerenog težinskog grafa ovom formulom simuliran usmjereni graf koji nije težinski.

Algoritam se izvršava iterativno, tako da se u svakom koraku za izračun $PR(v)$ koriste vrijednosti $PR(v')$ iz prethodnog koraka. Za prvi korak su vrijednosti obično inicijalizirane na $\frac{1}{|V|}$. Izvršavanje se provodi do numeričke konvergencije vrijednosti ili nakon unaprijed zadanog broja koraka. Nakon izračuna vrijednosti, potrebno je odabrati čvorove. Iako je moguće odabrati određeni broj vrhova s najvišim iznosom $PR(v)$, obično je za općenitost rada algoritma bolje postaviti čvrstu donju granicu σ, ispod koje vrhovi neće biti odabrani kao čvorovi. Grupiranje se odvija na isti način kao u potpoglavlju 4.2.4, izgradnjom maksimalnog razapinjućeg stabla, nakon dodavanja novog vrha, povezanog s čvorovima bridovima beskonačne težine.



Slika 4.6: Graf dobiven metodom PageRank s parametrima $d = 0.85$, $\sigma = 1.7$. Veći vrhovi predstavljaju odabrane čvorove. Zanimljivo je primijetiti kako je nekoliko od bridova najveće težine uklonjeno. U ovom primjeru takav postupak nije doveo do boljeg grupiranja, ali postoje situacije gdje je takav postupak semantički opravdan te za njih PageRank daje bolje rezultate.

4.2.6. Algoritam HITS

Algoritam HITS (Gibson et al., 1998), punim imenom Hypertext Induced Topic Search, sličan je oblikom i namjenom algoritmu PageRank, opisanom u potpoglavlju 4.2.5. Originalna verzija također radi s usmjerenim grafovima bez težina na bridovima. Ključna je razlika između HITS-a i PageRanka u tome što HITS vrhove procjenjuje u dvije dimenzije, za razliku od PageRanka, koji dodjeljuje samo jednu vrijednost (važnost vrha, odnosno, iznos $PR(v)$). Algoritam HITS pretpostavlja da postoje dvije kvalitete koje vrh može zadovoljavati: koliko je vrh dobar kao čvor,[2] te koliko je vrh dobar kao autoritet. Čvor (engl. *hub*) je vrh grafa koji pokazuje na puno drugih vrhova. Autoritet (engl. *authority*) je vrh na kojeg pokazuje puno drugih vrhova. Vrh je dobar čvor ako pokazuje na dobre autoritete, a autoritet je dobar ako na njega pokazuju dobri autoriteti. Jasno je da je i ovdje riječ o iterativnom algoritmu jer su vrijednosti kružno povezane i ovisne jedna o drugoj.

Prema trenutnom saznanju, još nitko nije primijenio algoritam HITS na otkrivanje skupa značenja riječi, iako, kao i kod PageRanka, postoje upotrebe u razrješavanju

[2]Riječ „hub“ obično se prevodi kao „koncentrator“ (Halonja i Mihaljević, 2003), ali je, s obzirom da je u potpoglavlju 4.2.4 preveden kao „čvor“ te da takav prijevod ima jasnije značenje, ovdje zadržan isti prijevod. Pri tome „čvor“ nije sinonim za vrh grafa, kako se često može pronaći u literaturi.

višeznačnosti. Transformacija težinskog neusmjerenog grafa u usmjereni netežinski obavljena je na sličan način kao kod PageRanka.

$$hubs(v) = \sum_{(v,v') \in E} \frac{w(v,v')}{\sum_{(v,v') \in E} w(v,v')} auth(v')$$

$$auth(v) = \sum_{(v,v') \in E} \frac{w(v,v')}{\sum_{(v',v'') \in E} w(v',v'')} hubs(v')$$

Bitno je primijetiti da je razlika između formule za $hubs(v)$ i $auth(v)$ u nazivniku razlomka, koji određuje smjer brida. U slučaju formule za $hits(v)$, razlomak pokazuje koliko je veza između v i v' jaka tako što određuje omjer težine njihovog incidentnog brida i zbroja težina svih bridova incidentnih s v. Prema tome, želimo vidjeti koliko je jaka veza s kojom v pokazuje na v', dok nije bitno koliko su jake ostale veze drugih vrhova s v'. Brid je zato usmjeren iz v u v'.

S druge strane, u formuli za $auth(v)$ bitno je vidjeti jačinu veza iz čvorova koji pokazuju na autoritet v, odnosno, modelirati brid usmjeren iz pojedinog v' u promatrani vrh v. Zbog toga se računa omjer težine brida (v, v') i zbroja težina svih ostalih bridova koji su incidentni s v'.

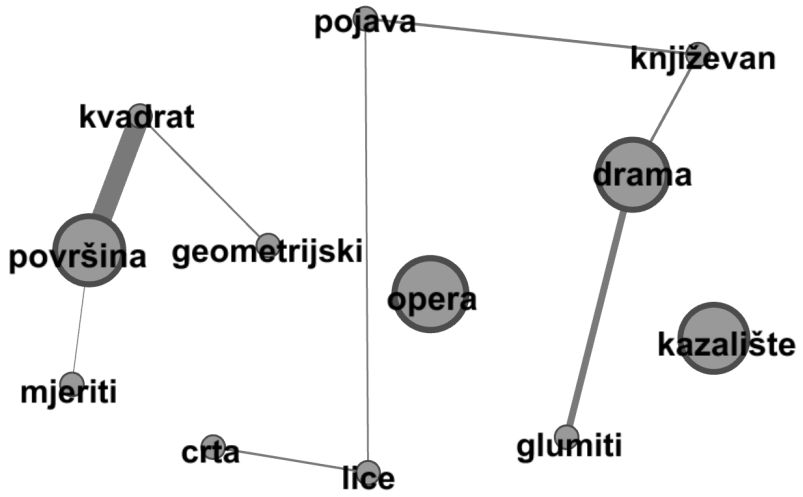
Nakon svake iteracije provodi se normalizacija vektora $\mathbf{hubs}$ i $\mathbf{auth}$. U svakoj je iteraciji prvo izračunata vrijednost vektora $\mathbf{hubs}$, koja se zatim koristi u računanju vektora $\mathbf{auth}$. Početne vrijednosti $hubs(v)$ i $auth(v)$ postavljeni su na 1 za sve $v \in V$. Nakon zadanog broja iteracija, algoritam se zaustavlja. Vrlo slično kao i kod PageRanka, biraju se vrhovi koji će predstavljati kontekste višeznačnih riječi. O važnosti vrha v kao predstavnika konteksta govori iznos $auth(v)$ jer on koncentrira informacije,[3] dok čvorovi ne nose informacije već ih „prosljeđuju“ do autoriteta.

Nakon odabranog skupa vrhova koji predstavljaju kontekste različitih značenja, grupiranje se obavlja na isti način kao u potpoglavljima 4.2.4 i 4.2.5, izgradnjom maksimalnog razapinjućeg stabla s odabranim čvorovima kao početnim stablom.

4.2.7. Algoritam MCL

Markovljev algoritam grupiranja (engl. *Markov clustering algorithm*, MCL) predstavljen je u (van Dongen, 2000). Zasniva se na pretpostavci da će nasumičnom šetnjom (slijednim posjećivanjem susjednih vrhova grafa nasumičnim redoslijedom, takvim da

[3] Što je još jedan razlog za izbjegavanje prijevoda „koncentrator“.



Slika 4.7: Graf dobiven metodom HITS s parametrom $\sigma = 0.2$. Veći vrhovi predstavljaju odabrane autoritete. Slično kao kod PageRanka, nekoliko od bridova najveće težine uklonjeno je. Općenito, rezultati PageRanka i HITS–a često su slični zbog srodnosti postupaka.

je viša vjerojatnost puta preko brida koji ima veću težinu) kroz graf put šetnje najvjerojatnije neće izaći iz čvrsto povezanih grupa (koje želimo pronaći) prije nego posjeti velik broj članova te grupe. Algoritam simulira protok (podataka, šetača) kroz graf, tako da su naglašena mjesta jakog strujanja, a mjesta slabog strujanja zanemarena. Autor pritom pod pojmom protok (originalno *flow*) ne misli na težine bridova, već na vjerojatni broj nasumičnih šetnji kroz određeni dio grafa; težine bridova naziva strujanjem (engl. *current*).

Graf se u postupku pretvara u Markovljev graf, u kojem se iz svakog vrha može doći u susjedni s određenom vjerojatnošću, tako da zbroj vjerojatnosti izlaza iz pojedinog vrha bude jednak 1. Markovljeva matrica, stohastička matrica koja predstavlja vjerojatnosti prijelaza (alternativan prikaz Markovljeva grafa) može se metodama linearne algebre modificirati i u koracima dovesti do grafa s željenim svojstvima (grupiranog u čvrsto povezane grupe). Korak proširenja i homogenizacije protoka naziva se ekspanzija i za njeno računanje koristi se obično matrično množenje. Nakon toga slijedi korak inflacije, gdje se protok sužava, tako da postane jači tamo gdje je strujanje gušće, a slabiji tamo gdje je strujanje slabije. Inflacija se obavlja slijednim Hadamardovim potenciranjem[4] i skaliranjem dijagonale. Ekspanzijom se elementi iste grupe međusobno povezuju, dok se inflacijom pojačava njihova povezanost i smanjuje povezanost

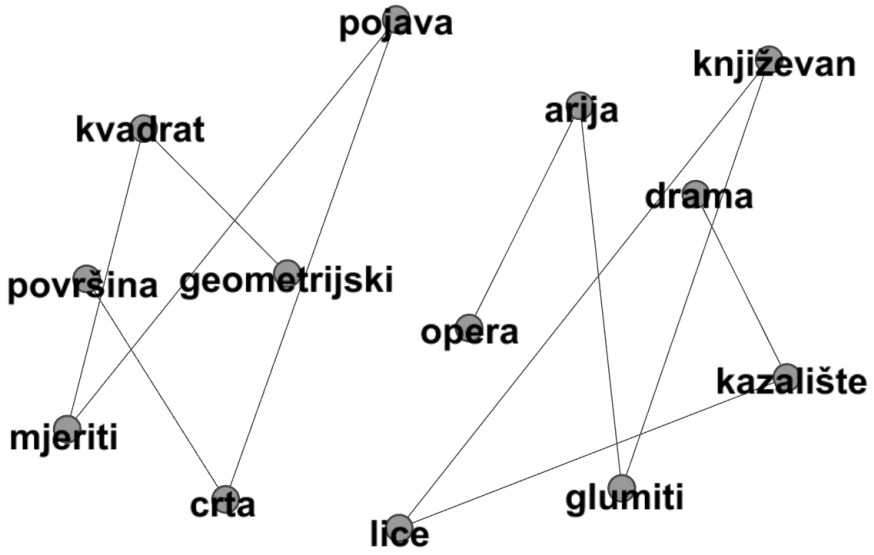
[4]Hadamardov umnožak (također i Schurov umnožak, *entrywise product*, $A \circ B = C$) je množenje dvaju matrica po elementima, tako da je rezultat matrica za koju vrijedi $c_{i,j} = a_{i,j} \cdot b_{i,j}$.

s drugim grupama. [5]

Jednostavnost postupka leži u jednostavnosti navedenih koraka. Ne koriste se kompleksne funkcije za izgradnju, spajanje ili razdjeljivanje grupa. Također, struktura grafa nije eksplicitno korištena, tako da nema operacija visoke složenosti za pojedini vrh grafa. Prostorna i vremenska složenost algoritma su $O(|V| \cdot k^2)$, gdje je k parametar koji određuje koliko je resursa alocirano za pojedini vrh, odnosno, za koliko se drugih čvorova pamti povezanost s promatranim vrhom. Drugim riječima, k je broj elemenata pojedinog stupca matrice različitih od nule. Prema riječima autora, čak se i koristeći male vrijednosti k mogu dobiti vrlo kvalitetni rezultati.

Tri su korištena parametra algoritma. Prvi je snaga inflacije, koja diktira granuliranost grupiranja; manja vrijednost daje veće grupe, dok veća vrijednost daje manje i zbijenije grupe. Drugi je broj unaprijed određenong načina raspolaganja resursima (uklanjanja manje bitnih informacija). Manji iznos smanjuje vrijeme izvođenja, ali može dovesti do manje kvalitetnih rezultata. Treći parametar je prethodno opisani iznos k.

Prema radu (Dorow et al., 2004), Markovljev algoritam grupiranja daje znatno bolje rezultate od originalnog *curvature* algoritma (opisanog u potpoglavlju 4.2.2).



Slika 4.8: Graf dobiven metodom MCL. Algoritam mijenja početne vrijednosti težina bridova.

[5] Animacija izvođenja algoritma može se vidjeti na stranici `http://micans.org/mcl/ani/mcl-animation.html`.

4.2.8. Ostale metode

Radi potpunosti, u nastavku je kratki osvrt na ostale metode grupiranja koje koriste strukturu grafa ili dovoljno sličnu zamjenu za strukturu grafa. Navedeni postupci nisu korišteni u izradi ovog rada. U stranoj se literaturi vrlo često može naići na primjere rada s višeznačnim riječima uz upotrebu strukture grafa, pri čemu graf nije izgrađen iz korpusa, već iz poznatih semantičkih odnosa, koristeći izvore semantičkog znanja. Najčešće se radi o korištenju strukture WordNeta (Sigman i Cecchi, 2002) i njenoj analizi. Budući da takve metode koriste potpuno drugačije pretpostavke, resurse i, općenito, pristup, one neće biti detaljnije opisane. Dovoljno je reći da je cilj takvih metoda odrediti udaljenosti (ili sličnosti) između riječi i njihovih značenja, ne samo onih neposredno povezanih, već i onih koje su povezane samo preko drugih riječi. Tako nastali težinski grafovi obično su bitno različiti od prethodno prikazanih težinskih grafova – svako značenje višeznačnih riječi već je poznato i predstavljeno zasebnim vrhom u grafu, a struktura grafa koristi se samo za razrješavanje višeznačnosti.

Mnogo bliže metode su one temeljene na strukturi grafa dobivenoj iz supojavljivanja u korpusu, pri čemu grupiranje nije particijsko s čvrstim granicama, već hijerarhijsko ili meko.

U radu (Hope i Keller, 2013) predstavljen je algoritam MaxMax, koji izvodi particijsko grupiranje mekih granica. Provodi se u tri koraka. Najprije se težinski graf transformira u usmjereni bez težina tako da vrh v ima brid prema v' ako i samo ako je za v' brid (v, v') u težinskom grafu brid s najvećom težinom. Nakon toga se svi vrhovi označavaju kao korijeni grafa te se slijedno svim potomcima nekog korijena uklanja ta oznaka. Na kraju je graf podijeljen na nekoliko podgrafova, od kojih svaki ima po jedan korijen, iz kojeg se usmjerenim bridovima može doći do svih ostalih vrhova tog podgrafa. Neki podgrafovi dijele vrhove (jedan vrh je potomak više od jednog korijena). Svaki od podgrafova predstavlja jedno značenje, a dijeljeni vrhovi pripadaju u više značenja. Prema autorima, metoda je mjerljiva s drugim modernim metodama u otkrivanju značenja.

Jurgens (2011) predstavlja jednostavnu metodu otkrivanja značenja temeljenu na hijerarhijskom grupiranju bridova grafa. Nakon izgradnje grafa supojavljivanja (ne uzimajući u obzir težine), vrhovi, koji u ovom trenutku još uvijek predstavljaju riječi, filtriraju se prema broju pojavljivanja u korpusu. Nakon toga se za svaki par susjednih bridova (e_{ij}, e_{ik}) određuje sličnost prema formuli $sim(e_{ij}, e_{ik}) = \frac{n_j \cap n_k}{n_j \cup n_k}$, gdje je n_j skup bridova vrha j, a n_k skup bridova vrha k (i je zajednički vrh). Na temelju tako

izračunatih sličnosti, gradi se dendrogram,[6] koji se zatim može odsjeći na proizvoljnoj visini, što rezultira proizvoljnom granulacijom elemenata. Grupirani bridovi čine zajednice (engl. *communities*) koje predstavljaju pojedini kontekst višeznačne riječi.

Hijerarhijsko grupiranje korišteno je i u (Klapaftis i Manandhar, 2010). Kao i u prethodno opisanom radu, ne grupiraju se riječi; ovdje se grupiranje obavlja nad kontekstima višeznačne riječi. Zbog toga su vrhovi grafa konteksti, a bridovi između njih označavaju sličnost. Vrhovi su prvo grupirani u dendrogram korigiran s obzirom na vjerojatnost svakog konteksta. Nakon toga su, odsijecanjem dendrograma na određenoj visini, određena značenja. Koristeći označen skup primjera, pronađeno je preslikavanje otkrivenih značenja na unaprijed poznat skup pravih značenja.

Brody i Lapata (2009) predstavljaju rad u kojem su konteksti modelirani multinomijalnim razdiobama značenja, gdje je svako značenje modelirano razdiobama riječi u korpusu. Taj probabilistički pristup koristi proširen skup značajki na kojima uči i, prema autorima, daje rezultate barem jednake kvalitete kao ostale moderne metode. Yao i Van Durme (2011) nadograđuju model u hijerarhijski Dirichleteov proces (engl. *hierarchical Dirichlet process*, HDP), koji ne mijenja bitno kvalitetu rezultata, ali uklanja potrebne parametre i sam pronalazi optimalan broj značenja za svaku riječ. Ove metode već imaju značajan odmak od strukture grafa slične onoj opisanoj u ovom radu.

4.3. Podešavanje parametara algoritama

Parametri koje je potrebno podesiti su granice korištene u predfiltriranju (kao i odabir kriterija filtriranja), parametri postfiltriranja i, najvažnije, parametri algoritama grupiranja. Budući da je cilj ostvariti optimalan rad sustava neovisan o slobodnoj procjeni ili proizvoljnom odabiru parametara, korišten je dio skupa za testiranje. Odabrana je petina skupa (detaljnije opisanog u poglavlju 5.1), tako da sadrži po tri riječi iz svakog frekvencijskog pojasa i po tri riječi od svih korištenih vrsta riječi. Kako bi evaluacija i dalje bila korektno obavljena na potpuno neviđenim podacima te riječi su izbačene iz skupa za evaluaciju. Za svaku riječ označavači su napravili vlastito grupiranje, nakon čega je određeno prosječno grupiranje te su evaluirani rezultati rada algoritama. Evaluacija je provedena na isti način kao i u konačnoj evaluaciji sustava (poglavlje 5.1). Za svaki parametar određene su gornja i donja granica te korak kojim se parame-

[6] Stablasta struktura podataka čiji su listovi grupirani elementi, a grane se račvaju od korijena prema listovima tako da račvanja blizu korijena označavaju slabu međusobnu sličnost, a račvanja blizu listova visoku sličnost.

tar povećava u svakoj iteraciji. Granice su određene na temelju prethodnih pokusa i uz konzultaciju s literaturom u kojoj su algoritmi opisani. Koraci su određeni tako da daju dovoljnu razinu preciznosti, a istovremeno ne unose previše kompleksnosti u model. Također, budući da vrijeme izvođenja ovisi o broju vrijednosti koje se moraju ispitati za svaki parametar, koraci su dovoljno veliki kako bi vrijeme izvođenja bilo razumno.

Parametri predfiltriranja i postfiltriranja nisu jedinstveni; za riječi koje se ne pojavljuju često u korpusu potrebno je smanjiti pragove jer inače skup vrhova nije dovoljno velik za kvalitetno grupiranje, ponekad je i prazan skup. Zbog toga se optimizacijom parametara direktno podešavaju sve vrijednosti osim w_1, koji se dinamički mijenja (polazeći od pronađenog optimuma) za po 5% svoje vrijednosti sve dok u negrupiranom grafu bude između 20 i 80 vrhova. Taj je broj određen empirijski i s ciljem interpretabilnosti i čitljivosti rezultata, pritom zadržavajući njihovu kvalitetu.

Za svaku kombinaciju parametara i svaku riječ proveden je postupak grupiranja. Ocjena kombinacije parametara je aritmetička sredina rezultata evaluacije za svaku riječ. Kao mjera točnosti rezultata korišten je Randov indeks koji uspoređuje zlatni standard (dobiven na temelju svih označavanja) i grupirani graf. Razlog zbog kojeg je odabran Randov indeks je činjenica da se pokazao najboljim u procjeni parametara koji su mogli biti i ručno optimizirani prema zlatnom standardu (primjerice, B–MST koristi samo parametar k, koji bi trebao biti što bliže prosječnom broju grupa koje su napravili označavači). Randov indeks opisan je u poglavlju 5.1. Budući da korištena mjera kvalitete grupiranja (kao i mnoge druge, slične mjere) može dati previše optimistične rezultate u slučaju ekstremnih grupiranja (i u slučaju da je previše i u slučaju da je premalo grupa), kao i za grupiranja koja rezultiraju slabo balansiranim grupama, sve kombinacije parametara za koje prosječan broj i veličina grupa nije zadovoljavajuć nisu uzimane u obzir; zanemarene su kombinacije parametara koje su u prosjeku dale manje od tri i više od šest grupa.

Tablica 4.2 i tablica 4.3 prikazuju optimalne parametre. Vrijednosti parametara za SquaT++ algoritam vjerojatno nisu optimalne jer je vrijeme izvođenja optimizacije parametara predugo, stoga su parametri filtriranja i parametri algoritma optimizirani odvojeno.

4.4. Razgraničavanje značenja

Rad na ovom projektu započeo je nakon niza pokusa s ciljem razrješavanja višeznačnosti uz pomoć rječničkih opisa značenja i strukture grafa supojavljivanja. Iako su korišteni skup značenja i pripadajući opisi bili neadekvatni, pokazali su da je struktura

Tablica 4.2: Odabrani optimalni parametri filtriranja

Algoritam	w_1	w_2	w_3	f_1	f_2	d_1
B–MST	0.0075	0	0.1	1 000 000	200	2
SquaT++ (vrhovi)	0.005	0	0.1	1 000 000	200	2
„Pokvareni telefon“	0.0025	0.001	0.1	200 000	200	2
HyperLex	0.005	0.02	0.1	800 000	200	2
PageRank	0.0025	0.001	0.1	1 000 000	200	2
HITS	0.01	0	0.1	700 000	200	3
MCL	0.0075	0	0.05	1 500 000	200	2

Tablica 4.3: Odabrani optimalni parametri algoritama grupiranja

Algoritam	Odabrani parametri
B–MST	k = 4
SquaT++ (vrhovi)	α = 0.1, β = 0.4, γ = 0.5, σ = 0.3
„Pokvareni telefon“	nema parametara
HyperLex	σ_1 = 0.15, σ_2 = 0.006
PageRank	d = 0.85, σ = 1.6
HITS	σ = 0.135
MCL	I = 1.25, R = 7, k = 7

grafa temeljenog na supojavljivanjima koristan resurs, što je i nadahnulo ovaj rad. Procedura je bila nadogradnja Leskovog algoritma (Lesk, 1986), a razliku je činio odmak od binarnog vrednovanja riječi koje se pojavljuju u kontekstu višeznačnice. Naime, originalni algoritam za svaki opis značenja u rječniku računa udio riječi iz konteksta višeznačne riječi koje se pojavljuju u promatranom opisu. Onaj opis koji ima najveći iznos tog omjera (drugim riječima, najveći presjek riječi iz opisa i riječi iz konteksta, relativno s obzirom na duljinu opisa) odabran je kao značenje višeznačnice u danom kontekstu. Modificirani algoritam koristi istu intuiciju, ali u obzir uzima težine bridova grafa supojavljivanja prema formuli

$$S(G_i, C) = \frac{\sum\limits_{v \in G_i, v' \in C} w(v, v')}{|G_i|}$$

gdje je C skup riječi iz konteksta višeznačnice, a G_i i–ti opis promatrane višeznačne riječi.

Ispitivanja su provedena na skupu označenom značenjima iz rječničke baze (taj skup nije korišten u okviru izrade ovog rada). Leskov algoritam postigao je točnost od

oko 35%, dok je modificirani algoritam postigao točnost od oko 50%. Slaba točnost Leskovog algoritma najviše je potvrda prevelike granuliranosti značenja u rječničkoj bazi i opisa koji vrlo slabo predstavljaju kontekst, odnosno uobičajenu upotrebu pojedinog značenja. No čak ni rezultati modificiranog algoritma nisu zadovoljavajući. Upotreba drugačijeg načina usrednjavanja (osim prikazane aritmetičke sredine) nije dovela do poboljšanja rezultata. Zbog toga je sljedeći korak bio razvoj sustava za otkrivanje značenja. No, kao što je već nekoliko puta spomenuto, otkrivanje značenja samo po sebi nema smisla; iako u okvirima rada nije predstavljena upotreba otkrivanja i razgraničavanja značenja u nekom primjenskom sustavu, u nastavku je prikazan prijedlog jednostavne i općenite metode razgraničavanja značenja koja koristi prethodno otkriven skup značenja.

U trenutku kad započinje postupak otkrivanja kojem značenju pripada višeznačna riječ u promatranom kontekstu, na raspolaganju je skup grupa vrhova grafa i skup vrhova koji odgovaraju riječima iz konteksta. Skup grupa za neku višeznačnu riječ mora biti jednak u svakom pokretanju algoritma i za svaki promatrani kontekst, u suprotnom slučaju klasifikacija nema smisla jer je nemoguće (ili barem vrlo teško) usporediti značenja pridodijeljena različitim kontekstima. Kako grupe predstavljaju kontekste, potrebno je usporediti s kojom se grupom promatrani kontekst najviše preklapa. Za svaku riječ iz konteksta koja se nalazi u grafu supojavljivanja postoje dva slučaja. Ili je riječ već u točno jednoj grupi ili se ne nalazi niti u jednoj grupi. Prva situacija je trivijalna i ne zahtijeva objašnjenje. Pitanje je kako postupiti u drugom slučaju. Jedna je mogućnost potpuno zanemariti takve riječi. No, ovisno o grupi za pojedinu višeznačnu riječ, takvo rješenje ne mora biti optimalno. S druge strane, postoje metode uobičajene u metodama grupiranja (Alpaydin, 2004), kad se jedna po jedna riječ dodaje u grupu, čineći tako novu, veću grupu. Algoritam jednostruke povezanosti (metoda najbližeg susjeda, engl. *single–link clustering*, *nearest–neighbor algorithm*) pronalazi element neke od postojećih grupa s kojim je novi element najpovezaniji i dodaje novi element u grupu iz koje je pronađeni, najbliži element.

$$cluster(v) = \underset{c}{\operatorname{argmax}} \left(\max_{v' \in c} w(v, v')\right)$$

Na ovaj način pronalazi se najjača veza između nove riječi i bilo koje druge riječi koja je unutar neke grupe. Ukoliko su obje riječi jednoznačne, a težine bridova dobro oponašaju semantičku povezanost između dvije riječi, nema razloga sumnjati u kvalitetu rezultata dobivenih ovom metodom. Ipak, to nije uvijek tako.

Metoda potpune povezanosti (metoda najdaljeg susjeda, engl. *complete–link clustering*, *farthest–neighbor algorithm*) pronalazi grupu u kojoj je najmanja sličnost svih

njenih elemenata i novog elementa najveća.

$$cluster(v) = \underset{c}{\operatorname{argmax}} \left(\min_{v' \in c} w(v, v')\right)$$

Ova je metoda suprotna krajnost prethodnoj. Temelji se na vrlo pesimističnim pretpostavkama o kvaliteti postojećih grupa.

Metoda prosječne povezanosti (engl. *average–linkage clustering*) neki je oblik kompromisa između dvije prethodne metode. Novi se član dodaje u grupu s čijim članovima ima najveću prosječnu povezanost.

$$cluster(v) = \underset{c}{\operatorname{argmax}} \frac{\sum_{v' \in c} w(v, v')}{|c|}$$

Performanse pojedinih metoda i odluka o najboljoj nalaze se u poglavlju 5.2.

5. Evaluacija i rezultati

Kako bi bila obavljena kvalitetna evaluacija ovog sustava, potrebno je riješiti nekoliko problema. Prvi je problem vezan uz skup podataka za testiranje – budući da prethodno nije razvijan sustav za otkrivanje i razgraničavanje značenja riječi za hrvatski jezik, ne postoji niti javno dostupan skup za testiranje. Postupak izgradnje takvog skupa opisan je u poglavlju 5.1. Sljedeći je problem pitanje prikladnih metoda evaluacije za konkretne probleme višeznačnosti riječi. Na kraju dolazi problem evaluacije zadatka razrješavanja višeznačnosti koristeći prethodno određene grupe značenja – zadatak procjene klasifikacije na grupe koje moguće nisu potpuno točno određene.

5.1. Načini evaluacije otkrivanja značenja riječi

Strani radovi često koriste neki od javno dostupnih skupova za evaluaciju, najčešće vezan uz SemEval[1] radionice. SemEval, originalno pod nazivom Senseval, nastao je iz potrebe za kvalitetnim sustavom za evaluaciju sustava za razrješavanje višeznačnosti, kao i zbog nedostatka ručno označenog korpusa. SemEval se razvio kasnije (2007. godine) te se ne bavi isključivo evaluacijom problema vezanih uz višeznačnost, već i evaluacijom drugih zadataka u semantičkoj analizi. U svakoj iteraciji dodaje se podrška za nove jezike, ali hrvatski jezik nažalost još nije uvršten među njih. Natjecatelji, odnosno sudionici zajedno oblikuju zadatke. Zatim stručnjaci označavaju testni skup, nakon čega natjecatelji (bez znanja o oznakama) korištenjem svojih sustava generiraju rješenja. Oznake stručnjaka i generirana rješenja se uspoređuju, nakon čega se objavljuju rezultati.

Agirre i Soroa (2007) predlažu dvije metode evaluacije otkrivanja značenja riječi. Prva je usporedba dobivenih grupa i skupa primjera, koji je ručno označen postojećim značenjima. Drugi je otkrivanje na koji se način grupe preslikavaju na unaprijed poznat skup značenja te evaluacija kroz razrješavanje višeznačnosti. U oba je slučaja

[1] `http://alt.qcri.org/semeval2014/`

potrebno imati već odabran točan skup značenja i skup primjera označen tim značenjima. Manandhar i Klapaftis (2009) predlažu nove mjere evaluacije (kako bi rezultati evaluacije bili informativniji), ali ne odmiču se od ideje testiranja grupiranja uz korištenje primjera označenih unaprijed poznatim skupom značenja. Budući da je zadatak izgradnje takvog skupa za testiranje iznimno opsežan, a korištene metode ne daju direktnu evaluaciju grupiranja, odluka je izgraditi manje zahtjevan skup i osmisliti način evaluacije koji evaluiraju upravo grupiranje.

5.1.1. Ispitni skup podataka

Riječi nad kojima se obavlja evaluacija odabrane su tako da svaku vrstu riječi (imenice, pridjeve i glagole) predstavlja po 15 različitih višeznačnica. Također, riječi su podijeljene u pojaseve učestalosti tako da trećina riječi (po pet imenica, pet pridjeva i pet glagola) bude iz pojedinog pojasa. Pojas visokofrekventnih riječi (f_H) čine sve riječi koje se pojavljuju više od 50 000 puta, pojas srednjefrekventnih (f_M) sve riječi koje se pojavljuju manje od 50 000, ali više od 1000 puta, dok su relativno rijetke $f_L$riječi one koje se pojavljuju manje od 1000 puta. Ukupno, to čini skup od 45 višeznačnih riječi. Svaka odabrana riječ u rječničkoj bazi ima barem dva različita značenja, a uvjet odabira bio je da uz to postoje barem dva značenja koja se na engleski jezik prevode isključivo različitim riječima, koje međusobno nisu sinonimi niti im se značenja preklapaju. Razlog tog uvjeta je smanjenje moguće pristranosti zbog skupa značenja iz rječničke baze; uvjet osigurava neosporivost višeznačnosti svake od riječi. Primjerice, riječ „sat“ moguće je prevesti kao *clock*, *hour* i *class* (između ostaloga), gdje svaki prijevod predstavlja drugo značenje, a niti jedan prijevod nema presjeka značenja. Skup odabranih višeznačnih riječi prikazan je u tablici 5.1.

Za svaku riječ iz skupa analizirana su značenja iz rječničke baze. Za svako od navedenih značenja ručno su odabrane riječi iz njihovih opisa, tako da svaki opis (odnosno, različito značenje) predstavlja barem jedna, a obično od tri do pet riječi. Odabrane riječi sortirane su abecedno tako da poredak nema utjecaja na označavače. Skupini od 10 označavača dane su upute (B). Prema tim uputama, zadatak je bio dosjetiti se svih značenja višeznačne riječi te zatim grupirati listu odabranih riječi tako da svaka riječ iz pojedine grupe semantički asocira na određeno značenje. Preklapanja riječi su pri tome bila dopuštena, tako da ista odabrana riječ može biti član više različitih grupa. Također, nije bilo nužno iskoristiti sve riječi. Razina granulacije značenja i snage semantičke veze potrebne da bi riječ bila grupirana u pojedno značenje nije bila zadana, već je bila na odabir označavačima, tako da skup značenja (subjektivno) djeluje pri-

Tablica 5.1: Odabrane višeznačnice za ispitni skup podataka. Pojas visokofrekventnih riječi označen je s f_H, pojas srednjefrekventnih s f_M, a pojas niskofrekventnih s f_L.

	Imenice	Pridjevi	Glagoli
f_H	put	siguran	značiti
	sat	prav	dodati
	dan	poseban	održati
	sud	domaći	iznositi
	zemlja	vrijedan	stajati
f_M	smjena	pošten	žaliti
	marka	skroman	skupiti
	instrument	oštar	položiti
	faktor	zvučan	spustiti
	vatra	prijelazan	spremati
f_L	košuljica	ispucan	prolomiti
	čunj	vražji	zaviti
	špaga	potplaćen	odvaliti
	balavac	uvrnut	plesti
	tikva	blijed	sklanjati

rodno. Ukoliko bi se označavač dosjetio značenja koje nije predstavljeno niti jednom riječi, uputa nalaže prijaviti takvu riječ i sporno značenje, ali ne dodavati nove riječi u skup odabranih riječi niti raditi posebno grupiranje za sporno značenje. Budući da je do pojave spornih značenja došlo u manje od 1% slučajeva, ona su zanemarena jer ne čine bitnu razliku u rezultatima, a znatno kompliciraju postupak evaluacije. Primjeri grupiranja prikazani su u tablici 5.2.

Bitno je spomenuti da označavačima nije bilo dopušteno koristiti se ikakvim leksičkim resursima, tražiti značenja pomoću Interneta ili literature, niti se međusobno konzultirati. Cilj je bio dobiti skup značenja koja su očita i prirodna označavačima. Veći broj označavača služi kako bi se izbjegle greške zbog mogućnosti da pojedini označavač zaboravi neko značenje. Također, većim brojem označavača lakše je usrednjiti označavanja.

Dobrovoljci označavači su osobe iz različitih krajeva Republike Hrvatske kojima je hrvatski materinji jezik, različitih dobnih skupina, različitih razina i usmjerenja obrazovanja. Popis označavača naveden je u zahvali na početku rada, a u nastavku nisu korištena njihova imena, već su anonimni i predstavljeni slovima.

Tablica 5.2: Primjeri grupiranja troje označavača.

Višeznačna riječ: špaga (označavač a)			
društven	figura	figura	konopac
koristan	gimnastika	komad	koristan
poznanstvo	komad	tanak	napraviti
privilegija	napraviti	tijelo	tanak
veza	noga		
	pokret		
	tijelo		

Višeznačna riječ: špaga (označavač b)		
figura	komad	društven
gimnastika	konopac	poznanstvo
komad	tanak	veza
napraviti		život
noga		
pokret		
tijelo		

Višeznačna riječ: špaga (označavač c)			
komad	figura	društven	koristan
koristan	gimnastika	koristan	napraviti
konopac	napraviti	poznanstvo	život
tanak	noga	privilegija	
	pokret	veza	
	tijelo	život	

5.1.2. Evaluacija grupiranja riječi

Nastavljajući na problematiku s početka ovog poglavlja, u ovom je radu prikazan model evaluacije otkrivanja značenja riječi bez potrebe za primjenskim sustavom čije se performanse evaluiraju. Pokrivene su dvije vrste grupiranja koja se evaluiraju i uspoređuju; opisi iz rječničke baze i označavanja primjeri su mekog grupiranja jer granice nisu potpuno definirane i isti element može (različitim stupnjem pripadnosti) pripadati u više grupa, dok je izlaz grupiranja razvijenog sustava primjer čvrstog grupiranja, gdje svaki element pripada najviše jednoj grupi. U oba slučaja moguće je da pojedini element ne pripada niti jednoj grupi. Ovdje se radi o metodama evaluacije particijskog grupiranja, a ne hijerarhijskog grupiranja jer pojedina grupa u niti jednom slučaju nije eksplicitno podgrupa neke druge grupe. U svakom je slučaju cilj imati opravdanu metodu koja uspoređuje dva grupiranja (bilo istog ili različitog tipa) mjerom sličnosti ili različitosti. Ta se metoda zatim može primjenjivati na bilo koji par grupiranja. Čvrsto grupiranje se u slučaju usporedbe s mekim pretvara u poseban oblik mekog grupiranja, gdje postoje samo dva stupnja pripadnosti (element ili pripada ili ne pripada grupi), presjek bilo koje dvije grupe prazan je skup. U nastavku su također predstavljene metode usporedbe rezultata isključivo čvrstog grupiranja.

Dodatan problem bila je izrada zlatnog standarda (engl. *gold standard*), koji usrednjuje rezultate više grupiranja istih elemenata, tako da konačan rezultat bude što bolja aproksimacija sredine između rezultata grupiranja. Naime, takav zadatak nije trivijalan niti jednoznačno određen. U dostupnoj literaturi nisam naišao na odgovarajuće rješenje tog problema, zbog čega je posebno detaljno opisana korištena metoda, uz objašnjenja svakog koraka i opravdanje ispravnosti metode.

Budući da su skupovi riječi koje su grupirali označavači različiti od riječi koje grupira sustav (s jedne strane, zbog umjetnog načina dobivanja riječi za prvi skup, a s druge, zbog postupka filtriranja riječi iz grafa), nije moguće direktno evaluirati grupiranje. Zbog toga je evaluacija ograničena na riječi koje su grupirali označavači. S obzirom da je presjek skupova relativno malen (oko 7% riječi koje su grupirali označavači nalazi se u filtriranom skupu riječi, koji je ulazni parametar za algoritme grupiranja), u grupe dobivene sustavom dodaju se riječi koje nedostaju, tako da prosječna povezanost dodane riječi i cijele grupe u koju se riječ dodaje bude maksimalna. Postupak je, dakle, sličan onom opisanom u poglavlju 4.4. Mali presjek skupova ne označava slabu kvalitetu filtriranja riječi niti loš rad sustava (riječi koje su grupirali označavači nisu nužno imalo „ispravnije“ od onih koje prođu filtriranje), ali može ukazivati na nešto manju pouzdanost mjera evaluacije.

5.1.3. Evaluacija mekog grupiranja

Budući da su označavači mogli grupirati bilo koju riječ proizvoljan broj puta s bilo kojom drugom ponuđenom riječi, svaki označeni dokument (dokument predstavlja grupiranje jednog označavača za jednu višeznačnicu) moguće je predstaviti nizom vektora, tako da svaka riječ ima svoj pridruženi vektor koji govori koliko je vjerojatno da je pojedina riječ grupirana u istu grupu kao i neka druga riječ. Vektori su normalizirani tako da je ukupan zbroj vjerojatnosti u svakom vektoru jednak 1. Ukoliko su vektori prikazani kao reci kvadratne matrice, broj u i–tom retku i j–tom stupcu govori kolika je vrijednost da je riječ predstavljena retkom i grupirana s riječi predstavljenom stupcem j. Elementi dijagonale postavljeni su na nulu jer nema smisla provjeravati koliko je često neka riječ u istoj grupi sa samom sobom.

Ukoliko postoje dvije matrice istih dimenzija, pri čemu isti reci i stupci odgovaraju istim riječima, moguće je procijeniti koliko dobro jedna matrica aproksimira drugu ili kolika je razlika između vektora matrice.

Korištene su dvije mjere, Jensen–Shannonova divergencija (engl. *Jensen–Shannon divergence*) i srednja apsolutna pogreška (engl. *mean absolute error*), ali je za definiciju Jensen–Shannonove divergencije potrebno prethodno definirati i Kullback–Leiblerovu divergenciju (engl. *Kullback–Leibler divergence*).

Kullback–Leiblerova divergencija, $D_{KL}(P||Q)$, gdje su P i Q vjerojatnosne razdiobe, daje podatak o tome koliko je informacije izgubljeno aproksimirajući P pomoću Q. Pri tome je P obično ispravna razdioba podataka (u ovom slučaju, oznaka označavača), a Q njena aproksimacija. Mjera nije simetrična, tako da $D_{KL}(P||Q) \neq D_{KL}(Q||P)$. Ukoliko su razdiobe jednake, iznos divergencije je nula, a ukoliko nisu, iznos je veći od nule (ne postoji gornja granica). U ovom se slučaju divergencija računa prema formuli

$$D_{KL}(P||Q) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij} \cdot \log \frac{p_{ij}}{q_{ij}}}{n} \tag{5.1}$$

pri čemu je p_i vektor i–te riječi. U slučaju kad je $p_{ij} = 0$, iznos tog člana u zbrajanju jednak je nuli.

Jensen–Shannonova divergencija mjera je temeljena na Kullback–Leiblerovoj divergenciji, s razlikom u tome što je simetrična i što joj je iznos uvijek konačan. Ukoliko je s M označena matrica za koju vrijedi $M = \frac{1}{2}(P + Q)$, divergencija se računa prema formuli

$$D_{JS}(P||Q) = \frac{1}{2}(D_{KL}(P||M) + D_{KL}(Q||M)) \tag{5.2}$$

Ova divergencija poprima vrijednosti između 0 i 1, gdje nula označava potpuno jednake matrice, a jedan potpuno različite.

Srednja apsolutna pogreška jednostavna je mjera razlike između dvije vjerojatnosne razdiobe (vektora). Ukoliko su p_i i q_i vektori koji opisuju i–tu riječ, pogreška iznosi

$$MAE = \frac{1}{n^2}\sum_{i=1}^{n}|p_i - q_i| = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}|p_{ij} - q_{ij}| \tag{5.3}$$

Ovim se metodama mogu usporediti označavanja različitih označavača, odrediti razlike između pojedinačnog označavanja i zlatnog standarda, ali ih je moguće koristiti i u evaluaciji grupa čvrstih granica. U tom se slučaju vektori grade na isti način, ali su sve vrijednosti vektora različite od nule međusobno jednake i ovisne isključivo o veličini grupe u kojoj se riječ nalazi. Iako je i taj način evaluacije dobar, više informacija mogu pružiti specijalizirane mjere usporedbe čvrstih grupiranja.

5.1.4. Evaluacija čvrstog grupiranja

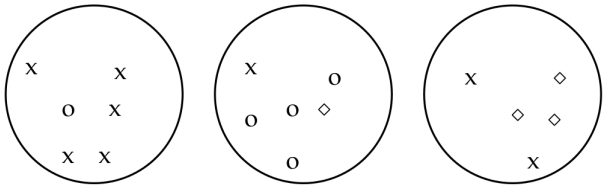
Iako je zadatak grupiranja sličan zadatku klasifikacije, bitna je razlika u tome što kod grupiranja nema apriornog znanja o klasama, njihovom broju, ni njihovim obilježjima. Elementi se grupiraju u klase, ali je problem preslikati tako dobivene grupe u one za koje je poznato da postoje (primjerice, jer su tako odredili označavači).

Za evaluaciju binarne klasifikacije često se koriste mjere temeljene na izračunatim vrijednostima TP, TN, FP i FN. Redom, te oznake predstavljaju *true positive* (broj primjera ispravno klasificiranih kao pozitivni), *true negative* (broj primjera ispravno klasificiranih kao negativni), *false positive* (broj primjera neispravno klasificiranih kao pozitivni) i *false negative* (broj primjera neispravno klasificiranih kao negativni). Pri tome su dvije klase predstavljene kao pozitivna i negativna (ili binarno, kao 0 i 1).

Kod višeklasne klasifikacije vrijednosti su promatrane pojedinačno za svaku klasu. Tako za klasu i vrijednost TP_i označava broj primjera ispravno klasificiranih u klasu i, FP_i broj primjera koji su neispravno klasificirani u klasu i, FN_i broj primjera koji pripadaju klasi i, ali nisu tako klasificirani, dok su u TN_i svi ostali primjeri, koji nisu trebali niti jesu klasificirani u i. No i s takvim oznakama, u slučaju evaluacije grupiranja potrebno je imati grupe koje odgovaraju klasama ili barem znati na koji se način grupe preslikavaju u klase.

U ovome su radu korištene definicije TP, TN, FP i FN iz (Manning et al., 2008). Prema njihovim definicijama, nema potrebe za zasebnim promatranjem svake klase. Razlika je i u tome što se ne ocjenjuju primjeri, već parovi elemenata koji su grupirani. Budući da je teško odrediti vanjski kriterij procjene pojedinog elementa, koristi se činjenica da grupiranje dobro odgovara točnoj klasifikaciji ako je velik broj elemenata koji su u istoj grupi također zajedno i u klasi. Prema tome, svaki par elemenata koji su grupirani može pripadati skupini TP, TN, FP ili FN. U skupini TP su oni koji su u istoj grupi i istoj klasi. Skupini TN pripadaju oni koji nisu u istoj grupi niti istoj klasi, FP su oni koji su u istoj grupi, ali različitim klasama, a FN oni koji su u različitim grupama i istoj klasi. Kod ovakvih definicija, zbroj $TP + TN + FP + FN$ ne iznosi n, već $\binom{n}{2}$.

Slika 5.1 prikazuje jednostavan slučaj za evaluaciju, gdje kružnice označavaju grupiranje, a različiti znakovi (simboli elemenata) označene klase. Vrijednosti su $TP = 20$, $TN = 72$, $FP = 20$, $FN = 24$.



Slika 5.1: Ilustrativan prikaz grupiranja elemenata, preuzet iz (Manning et al., 2008)

Nakon što su određene vrijednosti TP, TN, FP i FN, lako je izračunati uobičajene mjere kvalitete grupiranja.

Preciznost (engl. *precision*) omjer je između broja parova koji su ispravno grupirani zajedno i ukupnog broja parova koji su zajedno grupirani. U općenitom slučaju, preciznost daje podatak o tome koliko je točno klasificiranih elemenata u skupu svih pozitivno klasificiranih elemenata.

$$P = \frac{TP}{TP + FP} \tag{5.4}$$

Međutim, preciznost nije dovoljan pokazatelj kvalitete grupiranja. U slučaju kad grupiranje rezultira velikim brojem malih grupa, preciznost može biti vrlo visoka. Dovoljno je da postoji $n - 1$ grupa te da su u jedinoj grupi koja ima dva elementa oni elementi koji su u istim klasama da bi preciznost iznosila 100%. Zbog toga preciznost često dolazi u paru s odzivom (engl. *recall*), omjerom broja parova koji su ispravno grupirani zajedno i broja svih parova koji su klasificirani zajedno.

$$R = \frac{TP}{TP + FN} \tag{5.5}$$

Niti odziv sam po sebi nije dobar pokazatelj kvalitete grupiranja. U slučaju da je grupiranje rezultiralo samo jednom grupom, koja sadrži sve elemente, iznos odziva je 100%, iako je jasno da je takvo grupiranje u velikom broju slučajeva beskorisno. Kako bi preciznost i odziv bili prikazani jednom brojkom, koristi se F–mjera (engl. *F–score*) kao njihova harmonijska sredina.

$$F_\beta = \frac{(1 + \beta^2)PR}{\beta^2 P + R} \tag{5.6}$$

Parametar β koristi se za naglašavanje preciznosti ili odziva (ovisno o potrebi), a za vrijednost $\beta = 1$ preciznost i odziv jednako su bitni. Korištenje F–mjere s dobro odabranim parametrom β daje dobru informaciju o kvaliteti grupiranja. No, postoje i druge mjere koje daju dodatne informacije o grupiranju, a koje F–mjera zanemaruje.

Randov indeks govori koliki je ukupni udio točnih odluka u ukupnom broju parova svih elemenata. Ukoliko su klase sličnih veličina, Randov indeks dobra je i jednostavna mjera kvalitete grupiranja.

$$Rand = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.7}$$

Čistoća (engl. *purity*) donosi informaciju o ujednačenosti elemenata pojedine grupe s obzirom na njihovu pravu klasifikaciju. Za svaku je grupu određen broj elemenata svake klase koju ona sadrži. Čistoća iznosi prosječan broj elemenata u presjeku grupe i klase s kojom je taj presjek najveći.

$$pur = \frac{\sum_{i}^{N} \max_{j} |c_i \cap k_j|}{N} \tag{5.8}$$

pri čemu je c_i i–ta grupa, k_j j–ta klasa, a N ukupan broj grupa. Iako čistoća daje dobar uvid u kvalitetu pojedine grupe, moguće je da bude preoptimistična. Primjerice, za slučaj gdje je svaki element u svojoj grupi ($n = N$), čistoća iznosi 100%, iako ovakvo grupiranje nije nimalo kvalitetno.

Zbog toga je korištena i zadnja mjera, normalizirana uzajamna informacija (engl. *normalized mutual information*). Ona je omjer između uzajamne informacije skupa grupa i skupa klasa te aritmetičke sredine između entropije skupa grupa i entropije skupa klasa. Bez pretjeranog ulaženja u domenu teorije informacije, uzajamna informacija govori o tome koliko pojedina grupa c_i dijeli informacije s klasom k_j. Drugim

riječima, uzajamna informacija govori o tome koliko se znanje o klasama povećava uz dostupne informacije o grupama.

$$\begin{aligned} I(C;K) &= \sum_i \sum_j P(c_i \cap k_j) \log \frac{P(c_i \cap k_j)}{P(c_i)P(k_j)} \\ &= \sum_i \sum_j \frac{|c_i \cap k_j|}{N} \log \frac{N|c_i \cap k_j|}{|c_i||k_j|} \end{aligned} \tag{5.9}$$

Entropija skupa grupa govori o tome koliko je taj skup nesiguran, odnosno, koliko različitih vrijednosti može poprimiti.

$$H(C) = -\sum_i P(c_i) \log P(c_i) = -\sum_i \frac{|c_i|}{N} \log \frac{|c_i|}{N} \tag{5.10}$$

$$H(K) = -\sum_j P(k_j) \log P(k_j) = -\sum_j \frac{|k_j|}{|K|} \log \frac{|k_j|}{|K|} \tag{5.11}$$

Normalizirana uzajamna informacija računa se prema

$$NMI(C;K) = \frac{I(C;K)}{\frac{1}{2}(H(C) + H(K))} \tag{5.12}$$

Sve mjere imaju raspon od 0 do 1, gdje 0 označava loše grupiranje (u usporedbi s promatranim klasama), a 1 označava situaciju u kojoj su grupe identične klasama. U tablici 5.3 prikazane su izračunate vrijednosti za prethodno prikazani primjer.

Tablica 5.3: Vrijednosti mjera za primjer sa slike 5.1

Mjera	Vrijednost
P	0.5
R	0.45
F_1	0.48
$Rand$	0.68
pur	0.71
NMI	0.36

5.1.5. Stvaranje zlatnog standarda

Budući da je zadatak označavanja i grupiranja značenja subjektivan i podložan greškama, nema smisla koristiti grupiranje pojedinog označavača kao točno i jedino ispravno grupiranje. Točne klase, dakle, treba odrediti na neki drugi način. Jedna od mogućnosti je generirati sva moguća grupiranja i odabrati ono s kojim se označavači najviše slažu. S obzirom da je takvih grupiranja previše, čak i za male skupove grupiranih elemenata (radi se o Stirlingovom broju druge vrste), ovaj je pristup odbačen.

Zanimljiva je činjenica da je u literaturi vrlo rijetko rješavan problem izrade zlatnog standarda za grupiranje, pogotovo grupiranja pri kojem su označavači imali slobodu klasifikacije pojedinog elementa u više od jedne grupe ili u niti jednu grupu. Do kraja pisanja ovog rada nisam pronašao niti jedan relevantan rad koji bi se bavio tim problemom. Zbog toga je korišten sljedeći pristup: za svakog označavača i svaku višeznačnicu koju je označavao izgrađena je normalizirana matrica riječ–riječ Q opisana u potpoglavlju 5.1.3. Sve matrice koje odgovaraju pojedinoj višeznačnici zbrojene su i podijeljene s brojem označavača. Na taj su način zadržana sva svojstva početne matrice (vektori odgovaraju riječima, ukupan zbroj vjerojatnosti grupiranja riječi jednak je jedan), ali su vrijednosti usrednjene. Matrica je zatim transformirana tako da vrijednosti nove matrice Q' budu jednake $q'_{ij} = q'_{ji} = \frac{2}{q_{ij} + q_{ji}}$. Time je određena matrica udaljenosti između dva pojedina elementa (riječi). Udaljenost je recipročna vrijednost aritmetičke sredine sličnosti između parova (i, j) i (j, i). Pod pojmom sličnosti para ovdje se misli na (nesimetričnu) vjerojatnost grupiranja drugog elementa u istu grupu u kojoj je prvi element. Pokusi su pokazali da određivanje udaljenosti između vektora koji predstavljaju pojedine elemente ne rezultira dobrim grupiranjem.

Nakon što su dostupni iznosi udaljenosti između svih grupiranih elemenata, korišten je poznati (Alpaydin, 2004) algoritam hijerarhijskog aglomerativnog grupiranja (engl. *hierarchical agglomerative clustering*, HAC). Počevši od grupiranja u kojem je svaki element u svojoj grupi ($|K| = n$), algoritam u svakoj iteraciji spaja grupe među kojima postoji najmanja prosječna udaljenost (prosjek između svih parova elemenata iz jedne i elemenata iz druge grupe). Uz zadani argument k, koji odgovara željenom konačnom broju grupa, algoritam se zaustavlja nakon $n - k - 1$ koraka. Parametar k je ovdje određen kao prosječni broj značenja za pojedinu višeznačnu riječ, prema oznakama označavača. Konačno, rezultat je kvalitetno particijsko i čvrsto grupiranje riječi dobiveno na temelju grupiranja označavača, tzv. zlatni standard.

Primjer grupiranja prikazan je u tablici 5.4, a sličnosti označavanja pojedinih označavača sa zlatnim standardom (točnije, matricom sličnosti iz koje nastaje zlatni stan-

Tablica 5.4: Primjer grupiranja iz zlatnog standarda

Višeznačna riječ: špaga		
konopac	noga	veza
život	figura	koristan
komad	tijelo	privilegija
tanak	napraviti	društven
	gimnastika	poznanstvo
	pokret	

Tablica 5.5: Razlike između grupiranja označavača i prosječnog grupiranja

Označavač	D_{JS}	MAE
a	0.0273	0.0132
b	0.0323	0.0156
c	0.0349	0.0146
d	0.0357	0.0151
e	0.0358	0.0149
f	0.0367	0.0151
g	0.0388	0.0162
h	0.0396	0.0167
i	0.0406	0.016
j	0.0534	0.0207

dard) prikazane su u tablici 5.5. Vidljivo je da su neki označavači grupirali vrlo slično zlatnom standardu, dok su neki grupirali manje slično. Iz toga je bitno odrediti međusobno slaganje označavača, kako bi rezultati onih označavača koji su grupirali najmanje slično od ostalih bili uklonjeni. Iako nije nužno da su takva označavanja neispravna, njihovim uklanjanjem smanjuje se devijacija, pa je moguće odrediti pouzdaniji zlatni standard, nazvan jaki zlatni standard.

Iako je uobičajeno koristiti mjere poput Cohenove κ, one nisu nužno primjenjive na usporedbe dvaju grupiranja, pa su korištene mjere opisane u potpoglavlju 5.1.3.

Primjenom HAC algoritma na skupinu označavača, koristeći rezultate iz prethodne tablice (5.6) kao sličnosti između njihovih označavanja, odabrani su označavači `c`, `d`, `e`, `f` i `i` (podebljani u tablici) kao skupina najsličnijih. Njihova usrednjena grupiranja temelj su za izgradnju jakog zlatnog standarda.

Tablica 5.6: Međusobne razlike između grupiranja označavača, D_{JS}

	a	b	c	d	e	f	g	h	i	j
a	–	0.019	0.021	0.021	0.023	0.019	0.023	0.026	0.023	0.029
b	0.019	–	0.025	0.025	0.027	0.026	0.027	0.026	0.028	0.035
c	0.021	0.025	**–**	**0.02**	**0.018**	**0.02**	0.027	0.03	**0.018**	0.027
d	0.021	0.025	**0.02**	**–**	**0.018**	**0.022**	0.025	0.026	**0.019**	0.028
e	0.023	0.027	**0.018**	**0.018**	**–**	**0.018**	0.025	0.026	**0.02**	0.03
f	0.019	0.026	**0.02**	**0.022**	**0.018**	**–**	0.024	0.028	**0.019**	0.027
g	0.023	0.027	0.027	0.025	0.025	0.024	–	0.008	0.027	0.03
h	0.026	0.026	0.03	0.026	0.026	0.028	0.008	–	0.029	0.034
i	0.023	0.028	**0.018**	**0.019**	**0.02**	**0.019**	0.027	0.029	**-**	0.027
j	0.029	0.035	0.027	0.028	0.03	0.027	0.03	0.034	0.027	–

5.1.6. Rezultati otkrivanja značenja riječi

U tablicama 5.7 i 5.8 prikazani su rezultati rada algoritama (koristeći optimalne parametre). Budući da ne postoje usporedivi rezultati drugih sustava, kao gornja granica rada sustava prikazane su ocjene opisa iz rječničke baze. Riječi iz svakog pojedinog opisa čine grupu, koja je na isti način uspoređena sa zlatnim standardom. Prva tablica prikazuje performanse koristeći zlatni standard, a druga koristeći jaki zlatni standard.

Tablica 5.7: Uspješnost rada algoritama s optimalnim parametrima na zlatnom standardu

Algoritam	F_1	$Rand$	pur	NMI
B–MST	31.2 ± 10.4	59.5 ± 9.8	49.7 ± 14.5	$\mathbf{35.5 \pm 14.5}$
SquaT++ (vrhovi)	$\mathbf{39.5 \pm 9.1}$	41.8 ± 17.6	34.5 ± 14.7	0.3 ± 0.5
„Pokvareni telefon“	34.6 ± 12.4	54.3 ± 16.1	51.8 ± 14.9	9.7 ± 15.1
HyperLex	36.1 ± 9.5	46.4 ± 15.4	44.8 ± 15.5	4.3 ± 7.7
PageRank	31.3 ± 12	59.7 ± 10.3	$\mathbf{53.9 \pm 16}$	10.5 ± 13.8
HITS	29.5 ± 12.5	56.9 ± 14.2	50.1 ± 15.7	17 ± 17.9
MCL	34.9 ± 12.7	$\mathbf{59.8 \pm 12.4}$	49.6 ± 16.8	27.8 ± 19.3
Rječnik	51.3 ± 21.7	80 ± 10.7	82.2 ± 12.5	67.2 ± 18

Iz rezultata je vidljivo da nije postignut rezultat jednako dobar kao rezultat rječničke baze. S druge strane, vidljivo je da je model u stanju dati dobre rezultate. Velik dio krivice za slabiji uspjeh modela dolazi od vrlo trivijalnog algoritma filtriranja vrhova prije grupiranja; osim što postoji šum, problem je i u izostanku nekih manje

Tablica 5.8: Uspješnost rada algoritama s optimalnim parametrima na jakom zlatnom standardu

Algoritam	F_1	$Rand$	pur	NMI
B–MST	30 ± 11.8	60 ± 9	48.5 ± 13.4	$\mathbf{33.6 \pm 12.1}$
SquaT++ (vrhovi)	37.7 ± 8.8	41 ± 18.4	33.2 ± 13.6	0.6 ± 1
„Pokvareni telefon“	30.5 ± 11.1	52.1 ± 15.3	48.3 ± 12.1	9.7 ± 15.3
HyperLex	35.4 ± 10.6	46.8 ± 16.5	43.9 ± 14.9	5 ± 7.8
PageRank	$\mathbf{50.5 \pm 13.1}$	$\mathbf{60.3 \pm 10.2}$	$\mathbf{52.3 \pm 15.9}$	13.8 ± 14.8
HITS	28.4 ± 10.6	56.6 ± 15.1	48.5 ± 14.1	16.1 ± 15.9
MCL	31.7 ± 10.9	58.8 ± 11.9	47.3 ± 14.7	23.6 ± 12.9
Rječnik	56 ± 19.8	79.2 ± 10.1	81.3 ± 10.3	65.4 ± 18.6

zastupljenih značenja. Šum se odnosi na postojanje vrhova koji se jednostavnim metodama teško mogu ukloniti (primjerice, dio grupa čine čvrsto povezani vrhovi koji predstavljaju imenovane entitete koji se često pojavljuju uz višeznačnicu). O problemu takvih, manjih i zbijenijih grupa koje semantički ne predstavljaju neko značenje govori uspješnost algoritma B–MST; to je jedini algoritam koji eksplicitno kontrolira broj i veličinu grupa koje nastaju, pa vjerojatno iz tog razloga, usprkos svojoj jednostavnosti daje dobre rezultate. Izostanak manje učestalih značenja drugi je problem. Teško je pronaći značenja koja se pojavljuju relativno rijetko jer početni skup vrhova ne sadrži dovoljno riječi karakterističnih za to značenje da bi se te riječi istaknule kao zasebna grupa.

Osim navedenih, moguća su još barem dva razloga zašto rječnički skup ima bolje rezultate od onih dobivenih predstavljenim modelom. Prvi su prenesena značenja koja se na temelju konteksta teško mogu razlikovati od doslovnih značenja; za rješavanje takvog problema vjerojatno je potrebna dublja semantička analiza od one predstavljene u ovom radu. Osim toga, moguća je pristranost označavača značenjima iz rječnika. Iako označavači nisu vidjeli taj skup značenja, same riječi koje su grupirali ipak dolaze iz opisa značenja, tako da ih je prisustvo pojedine riječi možda podsjetilo na značenje koje inače ne bi smatrali relevantnim.

Ipak, vidljivo je da algoritmi grupiranja igraju veliku ulogu u kvaliteti rezultata. PageRank se ističe kao vjerojatno najbolji od korištenih algoritama, iako B–MST, MCL i algoritam „pokvarenog telefona“ imaju svoje prednosti (i usporedivih su rezultata). Uz nužne preinake i poboljšanja, moguće je da najbolji algoritmi ipak daju rezultate usporedive, ako ne i bolje od onih iz rječničkog skupa, što je i cilj automatskog ot-

krivanja značenja. Općenito, algoritmi koji imaju manje parametara daju prosječno bolje rezultate od algoritama kojima je potrebno zadati veći broj parametara. To može, ali i ne mora ukazivati na prostor za poboljšanje u metodama optimizacije korištenih parametara.

Tablice 5.7 i 5.8 ne pokazuju značajnu razliku u rezultatima između zlatnog standarda i jakog zlatnog standarda, s izuzetkom F_1 mjere. Razlika koju je bitno primijetiti je u algoritmima koji su ostvarili najbolje rezultate; u slučaju jakog zlatnog standarda PageRank je uvjerljivo najbolji algoritam, dok je u slučaju običnog standarda to vodstvo manje očito.

U nastavku su tablice 5.9 i 5.10, u kojima su prikazani rezultati rada algoritama, prikazani odvojeno za tri korištene vrste riječi i tri korištena frekvencijska pojasa. Mjera koja procjenjuje sve algoritme je Randov indeks jer se u pokusima pokazao kao dobar model ocjenjivanja grupiranja. Naime, njegovi rezultati najsličniji su ljudskim procjenama kvalitete grupiranja (temeljeno na nekoliko pokusa provedenih u sklopu ovog rada), a istovremeno je teorijski opravdan, bez očitih nedostataka za korišteni skup za evaluaciju.

Tablica 5.9: Iznos Randovog indeksa za različite vrste riječi (koristeći jaki zlatni standard)

Algoritam	Imenice	Pridjevi	Glagoli
B–MST	61.3 ± 7.5	59.2 ± 11.7	59.4 ± 7.9
SquaT++ (vrhovi)	40.7 ± 18.1	44.5 ± 19.6	37.8 ± 18.4
„Pokvareni telefon“	50.6 ± 17.1	53.5 ± 13.9	52.4 ± 16.1
HyperLex	46.4 ± 18	45.4 ± 13.4	48.5 ± 19
PageRank	57.2 ± 6.3	$\mathbf{63.9 \pm 12.4}$	$\mathbf{59.7 \pm 10.5}$
HITS	58.6 ± 14	56.8 ± 14.6	54.5 ± 17.5
MCL	$\mathbf{61.6 \pm 10.4}$	56.9 ± 13.7	58 ± 11.7
Rječnik	79.5 ± 10.3	75.7 ± 11.9	82.3 ± 7.6

Iz tablice 5.9 vidljivo je da se kao najbolji algoritam pokazao PageRank (iako mu je B–MST blizu, pogotovo uzevši u obzir malu devijaciju rezultata). Suprotno očekivanom, ne postoji značajna razlika u performansama između različitih vrsta riječi. Iako su glagoli uobičajeno teški za semantičku analizu, rezultati su samo malo slabiji od rezultata za imenice i pridjeve. Zanimljivo je da rječnička baza ima najslabiji rezultat za pridjeve, dok automatski model pridjevima najbolje otkriva značenja.

Pomalo iznenađuju i rezultati promatrani za različite pojase frekvencija. Očekivano je da je najlakše otkriti značenja najčešćih riječi jer je za njih dostupan velik broj pri-

Tablica 5.10: Iznos Randovog indeksa za različite frevencijske pojase (koristeći jaki zlatni standard)

Algoritam	f_H	f_M	f_L
B–MST	63.1 ± 9.1	57.1 ± 7.7	59.7 ± 9.8
SquaT++ (vrhovi)	25.4 ± 5.1	35.5 ± 7.8	62.1 ± 14.1
„Pokvareni telefon“	35.7 ± 11.6	61 ± 10.4	59.7 ± 7.9
HyperLex	53.5 ± 19.4	36.2 ± 13.4	50.1 ± 11.3
PageRank	$\mathbf{63.8 \pm 9.5}$	$\mathbf{63.3 \pm 8.4}$	53.6 ± 9.8
HITS	59.9 ± 14.7	51.4 ± 17	58.6 ± 13.2
MCL	53.6 ± 14.2	57.6 ± 8.1	$\mathbf{65.2 \pm 10.2}$
Rječnik	84.1 ± 5.7	75.9 ± 6.8	77.4 ± 14.4

mjera. Rezultati za značenja iz rječničke baze podržavaju tu pretpostavku, ali rezultati sustava su gotovo jednaki za sve pojase frekvencija, a neočekivano najbolji za najrjeđe riječi. Takvi rezultati imaju najmanje dva moguća razloga. Prvi je slabo filtriranje, gdje jednostavan algoritam lakše zanemaruje kontekste za riječi koje se supojavljuju s mnogo drugih riječi (jer su i same vrlo učestale), pa je teže prikupiti manje zastupljena značenja. Drugi razlog je veći broj značenja češćih riječi; prema rječničkoj bazi, riječi koje se češće pojavljuju imaju znatno veći broj značenja od rjeđih riječi.

Treba primijetiti da PageRank daje relativno slab rezultat na pojasu niske frekvencije. Usprkos tome, tih riječi je zanemarivo malo u korpusu u odnosu na one češće, pa je PageRank u općenitom slučaju optimalan izbor algoritma.

5.2. Evaluacija i rezultati razgraničavanja značenja

Evaluacija razrješavanja višeznačnosti obično se može svesti na evaluaciju višeklasne klasifikacije; postoje klase (značenja) i postoje primjeri kojima treba odrediti klasu (točno značenje). Međutim, budući da ovdje nema točnog skupa klasa, evaluacija je provedena na sljedeći način.

Odabrano je devet višeznačnih riječi (po tri iz svakog frekvencijskog pojasa i po tri od svake vrste riječi) iz prethodno opisanog skupa za evaluaciju. Za svaku je riječ nasumično odabrano po pet primjera iz korpusa (rečenica u kojima se pojavljuju). Svakom je primjeru dodijeljena jedna od grupa, prema postupku opisanom u poglavlju 4.4. S druge strane, iste je primjere troje označavača označilo jednom od klasa iz zlatnog standarda. Određena je ispravna klasa (ovdje nije bilo velikih problema jer je

broj klasa mali te su jasne granice među njima). Označeni skup za učenje priložen je u dodatku A.

Budući da je grupiranje već evaluirano, slabije grupiranje ne treba i ovdje kažnjavati. Ocjena kvalitete razgraničavanja značenja jednaka je omjeru broja riječi iz presjeka odabrane grupe i ispravne klase i ukupnog broja riječi ispravne klase. Rezultati su prikazani u tablici 5.11, tako da svaka metoda dodavanja riječi iz konteksta u grupe u zasebnom stupcu (jednostruka, potpuna i prosječna povezanost).

Za usporedbu je korišten Leskov algoritam, koji koristi skup značenja iz rječnika, a riječi iz opisa pojedinog značenja kao grupu. Također, ručno je odabrano dominantno značenje, odnosno, značenje za koje su označavači odredili da je najčešće u jeziku. Tablica prikazuje točnost klasifikacije ukoliko je apriorno odabrano to značenje. Treća vrijednost s kojom su uspoređeni rezultati sustava je automatski odabrana klasa prema prosječnoj frekvenciji riječi koje ju čine; klasa koja ima najvišu prosječnu frekvenciju riječi odabrana je u svim slučajevima te je određena točnost takve klasifikacije.

Tablica 5.11: Uspješnost rada metoda za razgraničavanje značenja

	Jednostruka	Prosječna	Potpuna
B–MST	45.9 ± 33.1	50.1 ± 33.2	9.1 ± 18.2
SquaT++ (vrhovi)	81.4 ± 31.4	81.4 ± 31.4	62.4 ± 45.4
„Pokvareni telefon“	60.6 ± 33.2	46.3 ± 40.6	2.8 ± 12.2
HyperLex	76.3 ± 30	64.6 ± 37.1	53.5 ± 41.2
PageRank	47.4 ± 30.8	33.9 ± 34.4	9.6 ± 18.2
HITS	53.2 ± 40.9	49.1 ± 42	38.5 ± 43.4
MCL	52.1 ± 26	45.5 ± 29.2	5.1 ± 9.3
Lesk	49.5 ± 45.6	45.2 ± 45.6	17.2 ± 32.3
dominantno značenje		68.8 ± 22.6	
najčešće riječi		31.1 ± 28.5	

Iz tablice 5.11 vidljivo je da različiti algoritmi grupiranja daju različite rezultate za primjere iz ispitnog skupa. Najbolji rezultat postigao je SquaT++, ali je, imajući u vidu nešto slabije rezultate u potpoglavlju 5.1.6 i neuobičajeno visoke vrijednosti za metodu potpune povezanosti, moguće iskrivljenje rezultata zbog slabe uravnoteženosti veličina grupa ili vrlo visoke devijacije broja grupa. Pouzdani algoritmi MCL, algoritam „pokvarenog telefona“ i HITS postigli su rezultate više od onih dobivenih Leskovim algoritmom, što pokazuje uspješnost modela čak i uz manjkavost podsustava za otkrivanje značenja.

Rezultati pokazuju da je najbolja metoda povezivanja riječi iz konteksta s grupama značenja metoda jednostruke povezanosti, dok je u svakom slučaju preporučljivo izbjegavati metodu potpune povezanosti.

Visoke devijacije učestale su u području razrješavanja višeznačnosti, odnosno, razgraničavanja značenja. Zbog manjeg skupa za ispitivanje nisu prikazani rezultati za pojedini frekvencijski pojas ili vrstu riječi. Metoda odabira dominantnog značenja temelji se na ručnom grupiranju značenja i označavanju primjera, pa su i očekivani relativno visoki rezultati.

Poboljšanjem rezultata sustava za otkrivanje značenja, poboljšali bi se i rezultati razgraničavanja značenja, što znači da bi i vrlo jednostavna metoda za razgraničavanje (kao ona prikazana u radu) mogla biti od velike koristi u raznim primjenskim sustavima.

6. Zaključak

Poznavanje pravog značenja višeznačnih riječi u nekom kontekstu važno je za široki spektar primjena u području obrade prirodnog jezika. Kako bi bilo moguće odrediti pravo značenje riječi, potrebno je poznavati skup značenja te riječi, kao i primjere čestih konteksta u kojima se pojedino značenje pojavljuje. Takve je resurse teško i skupo proizvesti, a oni dostupni za neke jezike obično su slabije kvalitete. Zbog toga dolazi do potrebe za automatskim otkrivanjem skupa značenja te njihovih karakterističnih konteksta.

U ovom je radu predstavljen model za otkrivanje značenja temeljen na strukturi grafa supojavljivanja induciran iz tekstnog korpusa. Kroz pokuse s nizom algoritama za grupiranje vrhova grafa prikazane su njihove prednosti i mane za zadatak grupiranja konteksta, odnosno značenja višeznačnih riječi. Dobiveni skup značenja uspoređen je sa skupom značenja iz rječničke baze; iako su rezultati slabiji, očigledan je potencijal modela. Osim toga, predstavljen je i jednostavan model za razgraničavanje između otkrivenih značenja, također temeljen na strukturi grafa supojavljivanja. Model se pokazao boljim od trivijalne metode, ali i Leskovog algoritma, jednog od temeljnih algoritama u području razrješavanja višeznačnosti.

Najveći prostor za poboljšanja u budućem radu je u postupcima filtriranja riječi za skup koji sadrži predstavnike svih značenja. Riječi koje su grupirane nisu potpuno pročišćene od nevažnih riječi (primjerice, imenovanih entiteta), a istovremeno neka značenja nemaju dovoljno predstavnika da bi se mogla istaknuti kao zasebno značenje. Također, prostor za poboljšanje postoji i kod otkrivanja manje doslovnih (prenesenih) značenja; pri tome relativno jednostavan model temeljen na distrubucijskoj semantici nije dovoljno dobar i vjerojatno je potrebna dublja analiza semantike teksta.

LITERATURA

Željko Agić, Nikola Ljubešić, i Danijela Merkler. Lemmatization and morphosyntactic tagging of croatian and serbian. U *Proceedings of ACL*, 2013.

Eneko Agirre i Philip Glenny Edmonds. *Word sense disambiguation: Algorithms and applications*, svezak 33. Springer Science+ Business Media, 2007.

Eneko Agirre i Aitor Soroa. Semeval-2007 task 02: Evaluating word sense induction and discrimination systems. U *Proceedings of the 4th International Workshop on Semantic Evaluations*, stranice 7–12. Association for Computational Linguistics, 2007.

Eneko Agirre, David Martínez, Oier López de Lacalle, i Aitor Soroa. Two graph-based algorithms for state-of-the-art wsd. U *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, stranice 585–593. Association for Computational Linguistics, 2006.

Ethem Alpaydin. *Introduction to machine learning*. The MIT Press, 2004.

Satanjeev Banerjee i Ted Pedersen. Extended gloss overlaps as a measure of semantic relatedness. U *IJCAI*, svezak 3, stranice 805–810, 2003.

Chris Biemann. Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems. U *Proceedings of the first workshop on graph based methods for natural language processing*, stranice 73–80. Association for Computational Linguistics, 2006.

David M Blei, Andrew Y Ng, i Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

Samuel Brody i Mirella Lapata. Bayesian word sense induction. U *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, stranice 103–111. Association for Computational Linguistics, 2009.

Samuel Brody, Roberto Navigli, i Mirella Lapata. Ensemble methods for unsupervised wsd. U *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, stranice 97–104. Association for Computational Linguistics, 2006.

Peter F Brown, Stephen A Della Pietra, Vincent J Della Pietra, i Robert L Mercer. Word-sense disambiguation using statistical methods. U *Proceedings of the 29th annual meeting on Association for Computational Linguistics*, stranice 264–270. Association for Computational Linguistics, 1991.

Marine Carpuat i Dekai Wu. Improving statistical machine translation using word sense disambiguation. U *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, stranice 61–72, 2007.

Sung-Hyuk Cha. Comprehensive survey on distance/similarity measures between probability density functions. *City*, 1(2):1, 2007.

Thomas H. Cormen, Clifford Stein, Ronald L. Rivest, i Charles E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd izdanju, 2001. ISBN 0070131511.

Scott C. Deerwester, Susan T Dumais, Thomas K. Landauer, George W. Furnas, i Richard A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990.

Antonio Di Marco i Roberto Navigli. Clustering web search results with maximum spanning trees. U *AI* IA 2011: Artificial Intelligence Around Man and Beyond*, stranice 201–212. Springer, 2011.

Antonio Di Marco i Roberto Navigli. Clustering and diversifying web search results with graph-based word sense induction. *Computational Linguistics*, 39(3):709–754, 2013.

Mona Diab i Philip Resnik. An unsupervised method for word sense tagging using parallel corpora. U *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, stranice 255–262. Association for Computational Linguistics, 2002.

Beate Dorow i Dominic Widdows. Discovering corpus-specific word senses. U *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 2*, stranice 79–82. Association for Computational Linguistics, 2003.

Beate Dorow, Dominic Widdows, Katarina Ling, Jean-Pierre Eckmann, Danilo Sergi, i Elisha Moses. Using curvature and markov clustering in graphs for lexical acquisition and word sense discrimination. *arXiv preprint cond-mat/0403693*, 2004.

Jean-Pierre Eckmann i Elisha Moses. Curvature of co-links uncovers hidden thematic layers in the world wide web. *Proceedings of the national academy of sciences*, 99 (9):5825–5829, 2002.

Philip Edmonds. Senseval: The evaluation of word sense disambiguation systems. *ELRA Newsletter*, 7(3):5–14, 2002.

Stefan Evert. Corpora and collocations. *Corpus Linguistics. An International Handbook*, 2, 2008.

Olivier Ferret. Discovering word senses from a network of lexical cooccurrences. U *Proceedings of the 20th international conference on Computational Linguistics*, stranica 1326. Association for Computational Linguistics, 2004.

David Gibson, Jon Kleinberg, i Prabhakar Raghavan. Inferring web communities from link topology. U *Proceedings of the ninth ACM conference on Hypertext and hypermedia: links, objects, time and space—structure in hypermedia systems: links, objects, time and space—structure in hypermedia systems*, stranice 225–234. ACM, 1998.

Antun Halonja i Milica Mihaljević. Nazivlje računalnih mreža. *Rasprave Instituta za hrvatski jezik i jezikoslovlje*, 29(1):87–101, 2003.

Zellig S Harris. Distributional structure. *Word*, 1954.

David Hope i Bill Keller. Maxmax: a graph-based soft clustering algorithm applied to word sense induction. U *Computational Linguistics and Intelligent Text Processing*, stranice 368–381. Springer, 2013.

Vedrana Janković, Jan Šnajder, i Bojana Dalbelo Bašić. Random indexing distributional semantic models for croatian language. U *Text, Speech and Dialogue*, stranice 411–418. Springer, 2011.

David Jurgens. Word sense induction by community detection. U *Proceedings of TextGraphs-6: Graph-based Methods for Natural Language Processing*, stranice 24–28. Association for Computational Linguistics, 2011.

Abraham Kaplan. *An experimental study of ambiguity and context*. Rand Corporation, 1950.

Ioannis P Klapaftis i Suresh Manandhar. Word sense induction & disambiguation using hierarchical random graphs. U *Proceedings of the 2010 conference on empirical methods in natural language processing*, stranice 745–755. Association for Computational Linguistics, 2010.

Jey Han Lau, Paul Cook, Diana McCarthy, David Newman, i Timothy Baldwin. Word sense induction for novel sense detection. U *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, stranice 591–601. Association for Computational Linguistics, 2012.

Michael Lesk. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. U *Proceedings of the 5th annual international conference on Systems documentation*, stranice 24–26. ACM, 1986.

Nikola Ljubešić i Tomaž Erjavec. hrwac and slwac: Compiling web corpora for croatian and slovene. U *Text, Speech and Dialogue*, stranice 395–402. Springer, 2011.

Swaminathan Madhu i Dean W Lytle. A figure of merit technique for the resolution of non-grammatical ambiguity. *Mechanical translation*, 8(2):9–13, 1965.

Suresh Manandhar i Ioannis P Klapaftis. Semeval-2010 task 14: evaluation setting for word sense induction & disambiguation systems. U *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, stranice 117–122. Association for Computational Linguistics, 2009.

Christopher D. Manning, Prabhakar Raghavan, i Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

Roberto Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10, 2009.

Roberto Navigli i Giuseppe Crisafulli. Inducing word senses to improve web search result clustering. U *Proceedings of the 2010 conference on empirical methods in*

natural language processing, stranice 116–126. Association for Computational Linguistics, 2010.

Roberto Navigli i Mirella Lapata. Graph connectivity measures for unsupervised word sense disambiguation. U *Proceedings of the 20th international joint conference on Artifical intelligence*, stranice 1683–1688. Morgan Kaufmann Publishers Inc., 2007.

Roberto Navigli i Mirella Lapata. An experimental study of graph connectivity for unsupervised word sense disambiguation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(4):678–692, 2010.

Hwee Tou Ng i Hian Beng Lee. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. U *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, stranice 40–47. Association for Computational Linguistics, 1996.

Lawrence Page, Sergey Brin, Rajeev Motwani, i Terry Winograd. The pagerank citation ranking: Bringing order to the web. 1999.

Rebecca J Passonneau, Ansaf Salleb-Aouissi, Vikas Bhardwaj, i Nancy Ide. Word sense annotation of polysemous words by multiple annotators. U *LREC*, 2010.

Ted Pedersen i Rebecca Bruce. Distinguishing word senses in untagged text. U *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, svezak 2, stranice 197–207, 1997.

Philip Resnik i David Yarowsky. A perspective on word sense disambiguation methods and their evaluation. U *Proceedings of the ACL SIGLEX workshop on tagging text with lexical semantics: Why, what, and how*, stranice 79–86, 1997.

Magnus Sahlgren. An introduction to random indexing. U *Methods and applications of semantic indexing workshop at the 7th international conference on terminology and knowledge engineering, TKE*, svezak 5, 2005.

Hinrich Schütze. Automatic word sense discrimination. *Computational linguistics*, 24 (1):97–123, 1998.

Hinrich Schütze i Jan O Pedersen. Information retrieval based on word senses. 1995.

Mariano Sigman i Guillermo A Cecchi. Global organization of the wordnet lexicon. *Proceedings of the National Academy of Sciences*, 99(3):1742–1747, 2002.

Jan Šnajder, B Dalbelo Bašić, i Marko Tadić. Automatic acquisition of inflectional lexica for morphological normalisation. *Information Processing & Management*, 44 (5):1720–1731, 2008.

Jan Šnajder, Sebastian Padó, i Željko Agić. Building and evaluating a distributional memory for croatian. U *51st Annual Meeting of the Association for Computational Linguistics*, stranica in press, 2013.

Tim Van de Cruys i Marianna Apidianaki. Latent semantic word sense induction and disambiguation. U *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, stranice 1476–1485. Association for Computational Linguistics, 2011.

Stijn Marinus van Dongen. Graph clustering by flow simulation. 2000.

Jean Véronis. A study of polysemy judgements and inter-annotator agreement. U *Programme and advanced papers of the Senseval workshop*, stranice 2–4, 1998.

Jean Véronis. Hyperlex: lexical cartography for information retrieval. *Computer Speech & Language*, 18(3):223–252, 2004.

Warren Weaver. Translation. *Machine translation of languages*, 14:15–23, 1955.

Dominic Widdows i Beate Dorow. A graph model for unsupervised lexical acquisition. U *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, stranice 1–7. Association for Computational Linguistics, 2002.

Xuchen Yao i Benjamin Van Durme. Nonparametric bayesian word sense induction. U *Proceedings of TextGraphs-6: Graph-based Methods for Natural Language Processing*, stranice 10–14. Association for Computational Linguistics, 2011.

David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. U *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, stranice 189–196. Association for Computational Linguistics, 1995.

Chung Yong i Shou King Foo. A case study on inter-annotator agreement for word sense disambiguation. 1999.

Ying Zhao, George Karypis, i Usama Fayyad. Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10(2):141–168, 2005.

George Kingsley Zipf. Human behavior and the principle of least effort. 1949.

Model za otkrivanje i razgraničavanje značenja višeznačnih riječi hrvatskoga jezika

Sažetak

Automatsko otkrivanje značenja višeznačnih riječi korak je prema izgradnji kvalitetnog sustava za razrješavanje višeznačnosti u situacijama kada nije dostupan adekvatan skup značenja ili dovoljan broj njihovih konteksta. Otkrivanje značenja ovdje se obavlja nenadzirano, korištenjem grafa supojavljivanja koji modelira semantičke odnose između riječi na temelju njihovih distribucija u korpusu. U okviru ovog rada predstavljen je pregled nekoliko algoritama za grupiranje vrhova grafa, analizirane su njihove prednosti i mane te ponuđena usporedba njihovih rezultata. Graf je grupiran na kontekste višeznačne riječi, gdje svaki kontekst predstavlja jedno značenje. Također je ponuđen i jednostavan model za razrješavanje višeznačnosti koristeći prethodno određen skup značenja. Na kraju, predstavljeni su postupci evaluacije grupiranja posebno namijenjeni evaluaciji otkrivanja značenja koji ne zahtijevaju vrednovanje kroz primjenski sustav.

Ključne riječi: obrada prirodnog jezika, višeznačnost riječi, otkrivanje značenja riječi, razgraničavanje značenja riječi, graf supojavljivanja

Word Sense Induction and Discrimination Model for Croatian words

Abstract

Automatic word sense induction is a step towards obtaining a good–quality word sense disambiguation system. It is to be used when no adequate sense inventories or their example contexts is available. Word sense induction is done using unsupervised methods, in this particular case by using a cooccurrence graphs which model semantic relations between words based on their distributions throughout the document corpora. This thesis presents a number of graph–vertice clustering algorithms, the analysis of their perks and flaws, along with the comparison of their results. The graph was clustered to represent different contexts of an ambiguous word, each context describing a usage of a single word sense. A simple word disambiguation model which uses the inducted set of senses if then presented. Methods aimed specifically to evaluate a word sense induction without any need for an indirect evaluation through application system are presented.

Keywords: natural language processing, word sense ambiguity, word sense induction, word sense discrimination, cooccurence graph

Dodatak A
Skup podataka za evaluaciju razgraničavanja značenja

word=sud
sentence=Rončević je inače u Sabor došao nakon nepravomoćne presude Županijskog suda u Zagrebu zbog nepovoljne kupnje kamiona .
senseCluster=istražan općinski zasjedati zgrada državni pravnik

word=sud
sentence=Na nedjeljnom izboru najbolju djevojku među 20 finalistica odabrat će stručni ocjenjivački sud sastavljen od istaknutih osobe iz modnog i medijskog svijeta kao što su : predstavnik agencije FORD Robert Knapp , predstavnici modnih agencija MODELS 1 iz Londona , GROUP iz Barcelone , WHY NOT iz Milana , FORD SUPERMODEL Hrvatske 2005 . Antonija Bralić , manekenka Jasmina Hdaga , modna dizajnerica Loredana Bahorić , modni dizajner i stilist Robert Sever , modna dizajnerica Jasmina Haddad , fotografkinja Mare Milin , modni urednik magazina ELLE Saša Joka , glavna urednica magazina COSMOPOLITAN Slavica Josipović i urednica u tjedniku GRAZIA Renata Rašović .
senseCluster=iskaz ocjena dijeliti mišljenje poricati tvrditi

word=sud
sentence=Kao pravna država prvi smo zainteresirani za sudbinu tih dokumenata i uskoro ćemo ponovno imati Savjet za suradnju s Međunarodnim kaznenim sudom i analizirati što možemo dodatno učiniti .
senseCluster=istražan općinski zasjedati zgrada državni pravnik

word=sud

sentence=E sad , ta prljava kampanja međusobnih optužbi i podmetanja traje otkad Pirate Bay postoji (ima tome sada već šest godina) , no otkako je firma stavljena na prodaju te otkako su udružene američko - švedske akcije krenule javno napadati website i vucarati ga po sudovima , postajalo je sve žešće i žešće .
senseCluster=istražan općinski zasjedati zgrada državni pravnik

word=sud
sentence=Naime , presuda Općinskog suda u Karlovcu donesena u studenom 2007 . godine postala je pravomoćna te će oni ubrzo , čim se sredi " papirologija " , biti upućeni u jednu od hrvatskih kaznionica .
senseCluster=istražan općinski zasjedati zgrada državni pravnik

word=vrijedan
sentence=U druga dva ogleda su do minimalnih , ali vrijednih pobjeda , stigle igračice Francuske , odnosno , Mađarske .
senseCluster=dragocjen cijena skupocjen visok

word=vrijedan
sentence=Vrijedan kompleks Šepurine Premijerka Jadranka Kosor na ovotjednoj je sjednici Vlade najavila kako je za prodaju spremno desetak bivših MORH-ovih objekata , i već idućeg tjedna povući će se prvi potezi da bi se nekretnine što prije prodale : raspisat će se natječaj za sudske vještake koji bi trebali procijeniti vrijednost nekretnina .
senseCluster=dragocjen cijena skupocjen visok

word=vrijedan
sentence=Drophead Coupe vrijedan 370.000 eura je vrhunac ponude kabrioleta , ali i automobilske ponude uopće .
senseCluster=dragocjen cijena skupocjen visok

word=vrijedan
sentence=- Kao i svima u našoj branši , i nama je teško u ovo krizno vrijeme , osobito zbog tečaja , no upornim radom i uz pomoć vrijednih i stručnih radnika nekako se uspijevamo održati na tržištu - kaže Marija Kežman , čija djeca također rade u tvrtki .
senseCluster=marljiv radinost

word=vrijedan
sentence=NEPOZNATI muškarac drsko je u nedjelju kasno navečer pokrao 17-godišnju djevojku u Zadru , otevši joj mobitel vrijedan oko 2.000 kuna , priopćila je danas zadarska policija .
senseCluster=dragocjen cijena skupocjen visok

word=iznositi
sentence=Razmak između polica iznosi 90 cm , a glavni prolaz 120 cm .
senseCluster=imati izraziti

word=iznositi
sentence=Što se tiče politike , ministar financija Andrej Bajuk izjavio kako bi na mjesto guvernera radije postavio osobu bližu političkoj desnici , a iz Janšinih krugova kao inkriminirajuću činjenicu iznosi se Gasparijev mandat na mjestu viceguvernera u Narodnoj banci SFRJ .
senseCluster=imati izraziti

word=iznositi
sentence=Drugi paket , Optimax , namijenjen je korisnicima čiji prosječni mjesečni troškovi telefoniranja iznose više od 200 kuna , a korisnik dobiva prvih 100 minuta razgovora u nacionalnoj nepokretnoj mreži po tarifi od jedne lipe po minuti .
senseCluster=platiti cijena stajati novac

word=iznositi
sentence=Sudac na tamošnjem sudu odlučio je osuditi magarca na 24 sata zatvora , a njegovom vlasniku naplatiti kaznu od 50 egipatskih funti što iznosi 45 kuna .
senseCluster=platiti cijena stajati novac

word=iznositi
sentence=Vrijednost projekta iznosi 101.460,00 EUR (745.731,00 HRK) , od čega je Varaždinska županija osigurala 26.379,60 EUR dok sukladno pravilima natječaja , EU financira preostalih 75.080,40 EUR .
senseCluster=platiti cijena stajati novac

word=faktor
sentence=Gradski čelnici postaju alat s kojim investitori postižu svoj cilj umjesto da

budu njihov korektivni faktor .
senseCluster=važan sredstvo uzrok pridonositi čimbenik

word=faktor
sentence=Prema studiji , klimatske promjene , ribarenje , zagađenje i drugi ljudski faktori izvršili su snažan utjecaj na gotovo pola svjetskih mora .
senseCluster=važan sredstvo uzrok pridonositi čimbenik

word=faktor
sentence=Napomenuo je da je HDZ pobijedio na prošlim parlamentarnim izborima , da u rukama ima svu vlast , a , kako je rekao , građani trebaju procijeniti treba li Hrvatskoj korektivni faktor vlasti , odnosno " hoće li sva jaja trpati u istu košaru " .
senseCluster=važan sredstvo uzrok pridonositi čimbenik

word=faktor
sentence=Humor se uglavnom sastoji od promjene smjera i faktora iznenađenja .
senseCluster=važan sredstvo uzrok pridonositi čimbenik

word=faktor
sentence=Niz je razloga za to , od specifičnog ustroja unutar obitelji do egzistencijalnih faktora koji prisiljavaju te mlade osobe na bavljenje djelatnostima koje će u materijalnom smislu osigurati opstanak njima samima i njihovim obiteljima .
senseCluster=važan sredstvo uzrok pridonositi čimbenik

word=zvučan
sentence=Zanemarimo li izostanak adekvatne zvučne kulise , sve je drugo u pravom sportskom stilu .
senseCluster=kuka kutija stijena krak

word=zvučan
sentence=Bolja zvučna izolacija i strujno oblikovani sustavi protoka zraka odnosno ispušnih plinova doprinose redukciji buke , inače karakterističnoj za roadstere .
senseCluster=kuka kutija stijena krak

word=zvučan

sentence=Zvučni nazivi međutim , ne znače mnogo ako nisu popraćeni kvalitetnim sirovinama i odgovarajućim kemijskim sastavom .
senseCluster=ime čuven poznat

word=zvučan
sentence=Ova godina također ima dobar line up koji prema riječima organizatora još nije zaključen pa se očekuje još zvučnih imena .
senseCluster=ime čuven poznat

word=zvučan
sentence=Ljetos su na nezaštićenom prijelazu kod Poznanovca poginule tri vrlo mlade djevojke , nakon čega je počela županijska akcija ugrađivanja svjetlosnih i zvučnih signalnih uređaja uz mahom rampama nezaštićene prijelaze . Očito prekasno .
senseCluster=kuka kutija stijena krak

word=spustiti
sentence=Čak se ni tada , budući da su sami procesori bili namijenjeni gornjem dijelu tržišta , cijena DDR3 memorije nije osobito spustila .
senseCluster=sniziti cijena

word=spustiti
sentence=Oštećenja na mostu dogodila su se 5 . ožujka kad je bura puhala brzinom od 80 kilometara na sat i temperatura se spustila na nula stupnjeva .
senseCluster=sniziti cijena

word=spustiti
sentence=Shvativši da je u nepoznatom krajoliku , a ne kod Malog jezera , Karamana ili ostalih njemu poznatih dijelova Kopaonika , najvjerojatnije je odbacio skije i pokušao se uz pomoć štapova spustiti niz potok .
senseCluster=mjesto uspeti dignuti planina sići

word=spustiti
sentence=Nakon mršava dva boda u prva dva kola , jučer je uslijedio novi šok za klub s rekordnim brojem naslova u Bundesligi – novi prvoligaš Mainz svladao ga je s 2 : 1 i spustio ga na mizerno 12 . mjesto , s dobrim izgledima da nakon nedjelje bude i niže .
senseCluster=mjesto uspeti dignuti planina sići

word=spustiti
sentence=Druga članica konzorcija koji pokušava spasiti posao u Crnoj Gori - Tehnika - zaronila je 2,16 posto i pala na 1810 kuna , a tijekom dana spustila se i do 1774 kune .
senseCluster=sniziti cijena

word=balavac
sentence=Obnovljene fasade , betonske podloge i druga uočljiva mjesta , za balavce sa sprejevima u ruci pravi su mamac .
senseCluster=nezreo brada koža ponašati nepristojan malen nedorasti dijete

word=balavac
sentence=Kaži mi koji to balavac zna šta je protestant ?
senseCluster=nezreo brada koža ponašati nepristojan malen nedorasti dijete

word=balavac
sentence=Prema takvim djelatnicama s iskustvom balavci koji imaju političke moći morali imati barem poštovanje , ako ništa drugo .
senseCluster=nezreo brada koža ponašati nepristojan malen nedorasti dijete

word=balavac
sentence=Ali nije da sam neki balavac .
senseCluster=nezreo brada koža ponašati nepristojan malen nedorasti dijete

word=balavac
sentence=" Andrija Getoš , otac Gordane Getoš Magdić , rekao mi je : ’ Ako ona bude optužena , ti si pokojni . ’ Getoševi su me tretirali kao balavca , a iz njihovih je iskaza vidljivo da nisu u stanju koncizno i logično lagati " , rekao je odvjetnik Arambašić .
senseCluster=nezreo brada koža ponašati nepristojan malen nedorasti dijete

word=uvrnut
sentence=Način na koji ulazite u moj i život moje sestre , oduzimate nam mir i kršite našu privatnost je u najmanju ruku degutantan i uvrnut .
senseCluster=čudan sulud smušen naopak

word=uvrnut
sentence=Gledatelja je pak zbunila uvrnuta kronologija - prvo je išlo javljanje uživo s Plesa , potom nemontirana snimka iz atenske zračne luke , pa iz aviona , gdje mikrofon dugo nije radio , ukratko - previše truda , premalo efekta .
senseCluster=čudan sulud smušen naopak

word=uvrnut
sentence=" Je li album mračan ? Sigurno . Je li uvrnut ? Naravno . No , povrh svega , on je prekrasan " , opisuju ga u RollingStoneu .
senseCluster=razlikovati

word=uvrnut
sentence=Vodite rašuna da je kondom postavljen baš u vagini i da nije uvrnut .
senseCluster=košulja savijanje podviti rukav

word=uvrnut
sentence=Iako je Crni zub zapravo popravljao samog sebe , nema tog glazbenog čika zube koji bi bolje sredio caklinu čudne šume uvrnutih pjesama nego što je to napravio sam Koja , pretvorivši ih u pravu plesnu džunglu !
senseCluster=razlikovati

word=plesti
sentence=Ti isti analitičari sada pred ove predsjedničke izbore pletu istu priču o mudrosti birača .
senseCluster=blizina tajan spremati smišljati zamka šteta

word=plesti
sentence=Joško Dujmović za tu prigodu pokazao je kako se pletu mreže od pruća , dok je Ante Čosić iz Dobropoljane " pjumbivao " konope i radio mornarske čvorove .
senseCluster=savijati košara nit grančica konac igla vuna sastavljati

word=plesti
sentence=Prije dolaska u zatvor nisam ni znala plesti .
senseCluster=savijati košara nit grančica konac igla vuna sastavljati

word=plesti

sentence=Predstavit će se i nekoliko kulturnih umjetničkih društava , Maslina iz Turnja , Zlatna luka iz Sukošana i KUD Bokolje iz Dobropoljane koji pokazati kako se pletu sprtve i mriže na tradicionalan način .
senseCluster=savijati košara nit grančica konac igla vuna sastavljati

word=plesti
sentence=O razlozima zašto vlasnici odgađaju stečaj koji je neminovan i zavlače radnike , pletu se različite priče , od one o povratu duga kamatarima i pokušaju izvlačenju novca , do onih osobne prirode – činjenice da bračni par naviknut na bogatstvo teško može prihvatiti propast i gubitak kontrole nad tvrtkom .
senseCluster=blizina tajan spremati smišljati zamka šteta

Tablica A.1: Skup riječi i njihovih značenja prema zlatnom standardu

Riječ	Skup značenja
sud	• istražan, općinski, zasjedati, zgrada, državni, pravnik • opseg, posuda • iskaz, ocjena, dijeliti, mišljenje, poricati, tvrditi • logički, negativan, relacija, modalitet
vrijedan	• dostojan, zaslužiti, povjerenje, poštovanje • marljiv, radinost • dragocjen, cijena, skupocjen, visok
iznositi	• platiti, cijena, stajati, novac • cipela, istrošiti, odjeća, rabljen • premjestiti, odstraniti • imati, izraziti
faktor	• poslovanje, upravitelj, posrednik, struka, pogon, financiranje, osoba, tvornica, snaga, radnik • važan, sredstvo, uzrok, pridonositi, čimbenik • broj, rezultat, množiti, grafički
zvučan	• kuka, kutija, stijena, krak • titranje, jasan, glasnica, odzvanjati, zvonak, odjekivati, glas, artikulirati • ime, čuven, poznat
spustiti	• smjestiti, protuargument • skroman, realan, ambicija • sniziti, cijena • mjesto, uspeti, dignuti, planina, sići • padobran, avion, zemlja
balavac	• puž, rod, golać, vrsta, gol • nezreo, brada, koža, ponašati, nepristojan, malen, nedorasti, dijete • riba, grgeč • nos, curiti, ubrus, sluzav
uvrnut	• čudan, sulud, smušen, naopak • košulja, savijanje, podviti, rukav • razlikovati
plesti	• jezik, govoriti, miješati, logičan, jasnoća, tuđi, stvar, nejasan, nerazgovjetan • savijati, košara, nit, grančica, konac, igla, vuna, sastavljati • rasti, biljka, obavijati, motati • blizina, tajan, spremati, smišljati, zamka, šteta

Dodatak B
Upute za označavanje podataka za evaluaciju otkrivanja značenja

Grupiranje značenja riječi
UPUTE ZA OZNAČAVANJE

Marko Bekavac

2. travnja 2014.

Dragi dobrovoljci,

hvala Vam na pomoći u izvedbi mog diplomskog rada pod naslovom „Model za otkrivanje i razgraničavanje značenja višeznačnih riječi hrvatskoga jezika“. Među ostalim, moj je zadatak na temelju velikog tekstnog korpusa otkriti sva značenja višeznačnih riječi (primjerice, riječ *mjesto* može se odnositi na naselje, točku na površini, položaj u redoslijedu, radno mjesto i slično), bez korištenja unaprijed definiranog popisa značenja. Budući da nije dostupan skup označenih primjera pomoću kojih mogu izvršiti procjenu rada svog sustava, potrebno ga je izraditi.

Označavanje se provodi na sljedeći način:

Za svaku višeznačnicu zadan je pripadajući skup riječi. Potrebno je riječi iz tog skupa grupirati tako da svaka grupa asocira na pojedino značenje višeznačnice. Moguće je da istu riječ dijele dvije ili više grupa, odnosno, moguće je da je pojedina riječ povezana s više od jednog značenja višeznačnice. Primjerice, riječ *pozicija* vezana je uz barem dva značenja riječi *mjesto* - točka na površini, kao i položaj u redoslijedu.

Također, moguće je i da pojedina riječ ne asocira na niti jedno od značenja. Primjerice, riječ *stol* nije vezana uz nijedno od navedenih značenja riječi *mjesto*.

Riječi iz ponuđenih skupova poredane su abecedno te su svedene na normalizirani

oblik, tako da budu što neutralnije i što manje utječu na grupiranje; moguće je da uobičajena upotreba riječi ne odgovara normaliziranom obliku. Primjerice, iako je uobičajeno reći „radno mjesto“, u skupu predloženih riječi će stajati riječ „radni“.

Riječi se grupiraju tako što se za svako značenje u zasebni stupac upisuje znak „x“ ukoliko riječ pripada tom značenju. U nastavku je primjer označavanja (slika B.1).

	A	B	C	D	E
1	mjesto				
2					
3	dio		x		
4	grad	x			
5	ograničen		x		
6	položaj		x		x
7	posao				x
8	površina		x		
9	prostor		x		
10	prvi			x	
11	radni				x
12	radnja	x			
13	redoslijed			x	
14	selo	x			
15	služba				x
16	stol				
17	točka		x		
18	trgovački	x			

Slika B.1: Primjer označavanja.

Stupac B sadrži riječi vezane uz značenje mjesta kao naselja, stupac C uz značenje mjesta kao lokacije, stupac D uz značenje mjesta kao položaja u redoslijedu itd.

Moja je preporuka razmisliti o svim značenjima dane višeznačnice. Nakon toga, pročitati riječi i eventualno dodati značenja kojih ste se nakon toga sjetili. Zatim, jedno po jedno značenje, grupirati riječi. Bitno je da se pokušate sjetiti svih Vama poznatih značenja, ali slučajno izostavljanje manjeg broja značenja nije problem. Moguće je da pojedina riječ nema više od jednog značenja, pa je potrebno samo označiti riječi vezane uz to značenje.

Preporučam i razbijanje posla označavanja na više kraćih dijelova. Predugo označavanje zamorno je i naporno, ali i smanjuje kvalitetu rezultata.

Vrlo je važno obaviti označavanje neovisno, bez savjetovanja s drugim osobama i bez upotrebe pomagala, poput rječnika ili Interneta. Cilj nije testiranje ljudi, niti je cilj dobiti egzaktne rezultate, potrebno mi je osobno mišljenje o broju značenja i riječima koje im pripadaju.

Ukoliko primijetite da postoji neko značenje višeznačnice koje nije predstavljeno niti jednom od ponuđenih riječi, molim da **ne** dopisujete ništa u tablice, već da na-

pravite zabilješku o tome, po mogućnosti s kratim opisom značenja koje nedostaje, riječima koje bi asocirale na to značenje ili nekim uobičajenim kontekstom u kojem se to značenje pojavljuje.

Molim da mi označene tablice pošaljete na adresu `marko.bekavac2@fer.hr`. Velika zahvala svima koji sudjeluju!

Dodatak C
Upute za korištenje programskog ostvarenja

Prije korištenja programskog ostvarenja provjeriti poglavlje 3.5.

Osnovno ostvarenje ima dvije funkcionalnosti: izgradnju skupa značenja za zadanu riječ i razgraničavanje između tih značenja. Druge funkcionalnosti nisu izdvojene, ali su dostupne u programskom kodu (i većim dijelom prikazane u klasi `hr.fer.takelab.wsid.execute.Test`). Glavne funkcionalnosti dostupne su u klasi `hr.fer.takelab.wsid.execute.Execute`, u `main` metodi. Također, načinjene su i izvršne datoteke (`.jar`). U svakom je slučaju način pokretanja jednak.

Izgradnja skupa značenja obavlja se na način opisan u tablici C.1. Razgraničavanje značenja obavlja se na način opisan u tablici C.2.

Budući da se svakim pokretanjem struktura grafa mora učitati s diska, programsko rješenje ne izvršava se trenutno. Uobičajeno vrijeme izvršavanja je oko dvije minute. Kad bi graf bio učitan u radnu memoriju (primjerice, u slučaju implementacije programskog rješenja kao web–aplikacije), vrijeme izvršavanja bilo bi znatno kraće.

Tablica C.1: Parametri programskog ostvarenja za otkrivanje značenja

Parametar	Opis
`-wsi`	Odabir funkcionalnosti otkrivanja značenja.
`-a`	Odabir korištenog algoritma (uz upotrebu optimalnih parametara). Mogućnosti su `bmst`, `squat`, `chinese`, `hyperlex`, `pagerank`, `hits` i `mcl`.
`-w`	Odabir višeznačne riječi za koju je potrebno otkriti skup značenja.
`-c`	Putanja datoteke izgrađenog i serijaliziranog težinskog grafa supojavljivanja. Trenutno se radi o strukturi `HashMap<hr.fer.takelab.wsid.data.LemmaPair, Double>`, tako da takav graf mora biti izgrađen koristeći priloženi programski kod. Uz programski kod priložen je korišteni graf (datoteka `pairs_3.ser`).
`-o`	Putanja datoteke u koju se zapisuje skup značenja. Iako se i ovdje radi o strukturi ovisnoj o implementaciji (klasa `HashMap<hr.fer.takelab.wsid.graph.Clusters`), otkrivena značenja ispisana su na standardni izlaz.

Tablica C.2: Parametri programskog ostvarenja za razgraničavanje značenja

Parametar	Opis
`-wsd`	Odabir funkcionalnosti razgraničavanja značenja.
`-c`	Putanja datoteke izgrađenog i serijaliziranog težinskog grafa supojavljivanja, pogledati tablicu C.1.
`-i`	Putanja datoteke sa zapisanim skupom značenja (rezultat prve funkcionalnosti programskog ostvarenja).
`-s`	Rečenica u kojoj se nalazi višeznačna riječ.