ODKRIVANJE KOREFERENČNOSTI V SLOVENSKEM JEZIKU NA OZNAČENIH BESEDILIH IZ COREF149

Slavko ŽITNIK
Fakulteta za računalništvo in informatiko Univerze v Ljubljani

Marko BAJEC
Fakulteta za računalništvo in informatiko Univerze v Ljubljani

Žitnik, S., in Bajec, M. (2018): Odkrivanje koreferenčnosti v slovenskem jeziku na označenih besedilih iz coref149. Slovenščina 2.0, 6 (1): 37–67.
DOI: http://dx.doi.org/10.4312/slo2.0.2017.1.37-67.

Odkrivanje koreferenčnosti je ena izmed treh ključnih nalog ekstrakcije informacij iz besedil, kamor spadata še prepoznavanje imenskih entitet in ekstrakcija povezav. Namen odkrivanja koreferenčnosti je prek celotnega besedila ustrezno združiti vse omenitve entitet v skupine, v katerih vsaka skupina predstavlja svojo entiteto. Metode za reševanje te naloge se za nekatere jezike z več govorci razvijajo že dalj časa, medtem ko za slovenski jezik še niso bile izdelane. V prispevku predstavljamo nov, ročno označen korpus za odkrivanje koreferenčnosti v slovenskem jeziku – korpus coref149. Za avtomatsko odkrivanje koreferenčnosti smo prilagodili sistem SkipCor, ki smo ga izdelali za angleški jezik. Sistem SkipCor je na slovenskem gradivu dosegel 76 % ocene CoNLL 2012. Ob tem smo analizirali še vplive posameznih tipov značilk in preverili, katere so pogoste napake. Pri analiziranju besedil smo razvili tudi programsko knjižnico s spletnim vmesnikom, prek katere je možno izvesti vse opisane analize in neposredno primerjati njihovo uspešnost. Rezultati analiz so obetavni in primerljivi z rezultati pri drugih, bolj razširjenih jezikih. S tem smo dokazali, da je avtomatsko odkrivanje koreferenčnosti v slovenskem jeziku lahko uspešno, v prihodnosti pa bi bilo potrebno izdelati še večji in kvalitetnejši korpus, v katerem bodo koreferenčno naslovljene vse posebnosti slovenskega jezika, kar bi omogočilo izgradnjo učinkovitih metod za avtomatsko reševanje koreferenčnih problemov.

Ključne besede: odkrivanje koreferenčnosti, slovenščina, ssj500k, coref149, algoritem SkipCor

1 UVOD

Odkrivanje koreferenčnosti (oz. razreševanje koreferenc, angl. *coreference resolution*, *entity linking*) pomeni gručenje omenitev oz. združevanje omenitev, ki se sklicujejo na isto entiteto. Nekateri uporabljajo tudi izraz ekstrakcija entitet (angl. *entity extraction*), pri čemer nalogo širijo in za gručo koreferenčnih omenitev določajo še njen tip. Npr.: »*[Janez]₁ in [Mojca]₂ sta poročena. [Spoznal]₁ [jo]₂ je v [Ljubljani]₃, ko [ji]₂ je kupil rože.*« Omenitve so označene z oglatimi oklepaji in se nanašajo na entitete. Entitete predstavljajo koreferenčne gruče omenitev, pri čemer npr. gruča *{Mojca, jo, ji}* združuje omenitve iz besedila, ki se nanašajo na isto entiteto, tj. osebo, ki jo poznamo pod imenom *Mojca*.

Od vseh treh glavnih nalog ekstrakcije informacij (angl. *information extraction*), tj. prepoznavanja imenskih entitet, ekstrakcije povezav in odkrivanja koreferenčnosi, so raziskovalci prav zadnjo raziskovali najmanj časa, zato so metode za odkrivanje koreferenčnosti zaenkrat še splošne in dosegajo slabše rezultate v primerjavi z ostalima dvema nalogama. Eden izmed razlogov za to je tudi ta, da besedilni kontekst pogosto ne ponuja vseh potrebnih informacij za njihovo razreševanje. Npr. iz besedila »*Janez si je ogledoval novega golfa. Predstavil ga je Marku. Nato ga je kupil.*« ne moremo nedvoumno razbrati, ali je golfa kupil Marko ali Janez.

Entiteta lahko predstavlja enega ali več objektov (npr. *[Marjan]*, *[Mojca in Lojze]*, *[učenci 5.b razreda]*). V besedilu so entitete omenjene z lastnim imenom, občnoimenskimi samostalniškimi besedami/besednimi zvezami ali zaimki. V splošnem ločimo imenske omenitve (npr. *Janez*), nominalne omenitve (npr. *moški s klobukom*) ali zaimenske omenitve (npr. *on*) (Luo 2007), v slovenskem jeziku pa je lahko omenitev izražena še s končnico glagolske oblike (Toporišič 2004: 607), kar odkrivanje koreferenčnosti še dodatno otežuje.

Pomembno je ločiti termin odkrivanja koreferenčnosti od odkrivanja anafor

(angl. *anaphora resolution*). Anafore so vrsta ponovitev (grš. anafora = obnova) in so predvsem znane iz poezije, ko se eden ali več zaporednih verzov začenja z isto ali istimi besedami. Medtem ko pri odkrivanju koreferenčnosti stremimo k združevanju omenitev v gruče, se pri odkrivanju anafor ukvarjamo le z odkrivanjem predhodne omenitve z enako referenco (Orasan in dr. 2008), torej v smislu, kot anaforično rabo zaimkov (ter ob njej še kataforično) prepoznavajo raziskovalci besediloslovja (npr. Korošec 1981: 179, 182–183). Npr. v besedilu »*Janez je brcnil žogo. Ta je odletela daleč stran.*« je zaimek *ta* anaforično navezan na referenco *žoga*. Odkrivanje koreferenčnosti je torej širše od anaforičnosti, vezano je na pomen, pri katerem je potrebno upoštevati tudi npr. antonomazijo ali zamenjavo imen (angl. *apposition*), ki se pogosto pojavlja v poročevalskih besedilih. Npr. v stavku »*Mojca, Janezova žena, dela v OBI-ju.*« Mojco še dodatno definiramo s pristavkom *Janezova žena* in to poimenovanje kot sklicevanje na Mojco uporabimo še kdaj v nadaljevanju. Anafore niso nujno pomensko vezane na entiteto, saj lahko v nekaterih primerih omenitvi označimo kot anaforični, vendar ne kot koreferenčni. Npr. v stavkih »*Marko je kupil [golfa]. Kupil [ga] je tudi Janez.*«. Omenitvi *golfa* in *ga* sta v tem primeru anaforični, saj pomenita isto stvar/koncept. Nista pa koreferenčni, saj se ne sklicujeta na isto entiteto (*Markov golf* je različen od *Janezovega golfa*).

Naloga odkrivanja koreferenčnosti je v grobem razdeljena na (A) identifikacijo omenitev in (B) gručenje omenitev. Raziskovalci se identifikaciji omenitev niso veliko posvečali in so omenitve pogosto identificirali s pravili s pomočjo razpoznanih samostalnikov v odvisnostnih drevesnicah (Lee in dr. 2011), ki označujejo sintaktično in semantično strukturo stavkov. Pri gručenju omenitev pa največ pristopov sloni na binarni klasifikaciji med dvema omenitvama, na podlagi katere se odločijo, ali sta omenitvi koreferenčni ali ne. Poleg tega z različnimi tehnikami podatke filtrirajo, tako da ni potrebno pregledovanje vseh n-parov.[2] Kljub temu da je bilo predstavljenih mnogo različnih tehnik z uporabo metod strojnega učenja in predlaganih veliko funkcij za iskanje značilk

(Bengtson in Roth 2008; Ng in Cardie 2002; Orasan in Evans 2007), se nekateri nenadzorovani sistemi ali sistemi, ki temeljijo na pravilih (Lee in dr. 2011), na vnaprej definirani domeni odrežejo približno enako dobro.

Najbolj znani dogodki, ki so spodbudili večje zanimanje za raziskave na področju odkrivanja koreferenčnosti, so konference MUC-7 (Chinchor 1998), ACE 2004 (Doddington in dr. 2004) ter seriji evalvacij CoNLL (Pradhan in dr. 2012) in SemEval (Recasens in dr. 2010).

2 SORODNA DELA

Večina tehnik za odkrivanje koreferenčnosti, kot smo že omenili, predstavlja problem kot binarno klasifikacijo, kar omogoča uporabo že znanih modelov strojnega učenja (Ng in Cardie 2002; Culotta in dr. 2007). Pri tem načinu algoritem za vsak par omenitev preveri, ali sta koreferenčni ali ne (tj. se sklicujeta na isto entiteto ali ne). Nasprotno pa nenadzorovane tehnike za odkrivanje koreferenčnosti temeljijo na upoštevanju zaporedij omenitev ter predhodno definiranih pravil ali hevristik (Haghighi in Klein 2009; Lee in dr. 2011), zaradi česar so težje za vzdrževanje in slabše prilagodljive za sorodne probleme ali nove domene.

Eden izmed začetnih pristopov za odkrivanje koreferenčnosti s pomočjo metod nadzorovanega učenja je temeljil na odločitvenih drevesih in dvanajstih lokalnih funkcijah značilk (Soon in dr. 2001). Ta pristop je izboljšal rezultate prejšnjih metod, ki so temeljile le na ročno definiranih pravilih. Kljub temu da je bil zelo enostaven, njegovih rezultatov poznejši raziskovalci niso mogli izboljšati le z uporabo bolj naprednih klasifikatorjev, kot so SVM (Rahman in Ng 2009), modeli največje entropije (Luo in dr. 2004) ali markovska omrežja (Huang in dr. 2009). Zaradi tega so pokazali, da je jedro uspeha v definiciji inovativnih, jezikoslovno bogatih in globalnih atributskih funkcij (Ng in Cardie 2002; Bengtson in Roth 2008). Bengtson in Roth (2008) sta jih celo sistematično razdelila v kategorije in ponazorila njihovo pomembnost, s čimer sta pokazala, da je z uporabo primerno zasnovanih atributskih funkcij

odkrivanje koreferenčnosti možno značilno izboljšati. Zadnje izboljšave pa so bile dosežene z uporabo globokih nevronskih mrež (Clark in Manning 2016). Tudi ta pristop ne deluje le nad surovim besedilom, vendar na vhodu poleg vektorskih vložitev potrebuje še dodatne značilke.

Nenadzorovani pristopi ne potrebujejo učnih podatkov za izgradnjo modelov, a kljub temu dosegajo odlične rezultate. Večina predlaganih modelov pridobi osnovne omejitve iz splošnih besedil, izboljša ročne hevristike na podlagi pregleda označenih besedil ali uporabi ročno definirana pravila. Haghighi in Klein (2009) sta predlagala modularni nenadzorovani pristop, ki je sestavljen iz treh delov. Prvi identificira sintaktične poti med omenitvami, nato oceni semantično ujemanje med njimi in izbere referenčne omenitve, ki služijo kot seme za nadaljnje razvrščanje omenitev. Lee in dr. (2011) so izboljšali Raghunathanov sistem (Raghunathan in dr. 2010), ki temelji na večkratnih procesiranjih podatkovne množice. Predlagali so trinajst zaporednih procesiranj, ki so jih razvrstili padajoče po njihovi natančnosti. Vsako procesiranje podatkovne množice uporabi nekaj ročno definiranih pravil, ki temeljijo na sintaktičnih odvisnostnih drevesnicah, imenskih entitetah, različnih hevristikah in lastnostih podatkovne množice. V primerjavi z odkrivanjem koreferenčnosti le med pari omenitev je bilo predlaganih tudi nekaj nenadzorovanih sistemov, ki odkrivajo koreferenčnost nad seznami omenitev (Bejan in dr. 2009; Bejan in Harabagiu 2010; Charniak 2001).

Na področju faktorskih grafov so McCallum in dr. (2005) za odkrivanje koreferenčnosti predlagali tri modele pogojnih naključnih polj (CRF, angl. *conditional random fields*). Prvi model je splošen, zato je učenje in napovedovanje z njim kompleksno ter počasno. Drugi model deluje med pari omenitev, tretji model pa predstavlja pare omenitev kot opazovana stanja v modelu. Wellner in dr. (2004) so uporabili prvi McCallumov model in namesto odkrivanja koreferenčnosti združevali citate, kar je podoben problem odkrivanju koreferenčnosti. Pristop, ki je najbolj podoben algoritmu, ki smo ga uporabili za lastno raziskavo, je uporaba posebne vrste linearnoverižnih

modelov CRF z dodanimi faktorji nad nezaporednimi stanji (angl. *skip-chain CRF*) (Finkel in dr. 2005), vendar je njihovo učenje in napovedovanje v primerjavi z osnovnim modelom še vedno počasnejše. Culotta in dr. (2007) so za odkrivanje koreferenčnosti predlagali uporabo grafičnih modelov prvega reda na podmnožicah omenitev v stilu hierarhičnega razvrščanja, torej njihov sistem deluje nad potencialnimi entitetami (tj. množicami omenitev) in ne ločeno nad omenitvami. Ram in Devi (2012) sta predlagala sistem, ki uporablja dva tipa modelov CRF. Najprej s splošnim modelom CRF identificira omenitve s pomočjo odvisnostne drevesnice, nato pa za odkrivanje koreferenčnosti uporabi linearnoverižne modele CRF. Čeprav sta Ram in Devi uporabila le linearne modele, sta kot vhod vzela pare omenitev in ne celega zaporedja omenitev.

Odkrivanje koreferenčnosti je že zelo dobro raziskano za angleški jezik, medtem ko za slovanske jezike ne obstaja mnogo del na tem področju, in to predvsem zaradi pomanjkanja ročno označenih korpusov ali pobud za njihovo gradnjo. Analize in korpuse označenih besedil tako najdemo le za ruski (Ju in dr. 2014), poljski (Ogrodniczuk in Kopeć 2011) in hrvaški jezik (Glavaš in Šnajder 2015), pri čemer učna množica za hrvaški jezik ni na voljo, saj je bila izdelana v sodelovanju z zasebnim sektorjem. Za slovenski jezik smo zasledili le opis posebnosti pri odkrivanju koreferenčnosti in analizo določenih hevristik kot pomoč za druge naloge procesiranja naravnega jezika (Holozan 2015).

Namen tega prispevka ni primerjava obstoječih pristopov ali optimizacija parametrov in značilk za dosego najboljšega rezultata, temveč prikaz zmožnosti odkrivanja koreferenčnosti na novem, ročno označenem korpusu slovenskega jezika. Za analizo smo uporabili algoritem SkipCor (Žitnik in dr. 2014), ki za angleški jezik dosega primerljive rezultate z najboljšimi modeli, ki smo jih povzeli zgoraj. Algoritem SkipCor za delovanje uporablja linearnoverižne modele CRF. Koreferenčnosti odkriva tako, da vsak dokument pretvori v osnovno zaporedje omenitev in dodatna zaporedja z izpuščenimi omenitvami, s čimer lahko z enostavnimi modeli odkrije koreferenčnost na daljših razdaljah.

3 SLOVENSKI KOREFERENČNI KORPUS COREF149

Za slovenski jezik korpus z označenimi koreferencami še ni obstajal, zato smo ga zgradili sami (Žitnik 2018). Pred tem je obstajalo le nekaj smernic, kakšen korpus bi za tako nalogo morali imeti (Holozan 2015). Jezikoslovci (Bucik 2001) so se sicer že ukvarjali tudi s problemom, kako izbrati primere za označevanje koreferenc, in so pripravili korpus, ki pa zaradi premajhnega števila besedil in neoznačenih koreferenčnih zaimkov ni primeren za statistično analizo.

Korpus coref149 (Žitnik 2018) smo označili nad izbranimi besedili korpusa ssj500k v1.4 (Krek 2015), ki že vsebuje oblikoskladenjske oznake in ima označene imenske entitete. Korpus ssj500k vsebuje večje število besedil (1.677), členjenih na med seboj nepovezane odstavke iz različnih besedil, zato jih nismo mogli uporabiti kot osnovno enoto, tj. kot dokument, v katerem bi označili koreferenčnost. Zaradi tega smo kot osnovno enoto izbrali posamezen odstavek. Vseh odstavkov je v korpusu ssj500k 8.137, vendar je veliko izmed njih zelo kratkih in so posledično neprimerni za označevanje koreferenčnosti. Da smo pridobili smiselno množico besedil, v katerih bi izvedli analizo koreferenčnosti, smo zato izbrali le odstavke, ki vsebujejo vsaj 100 besed in imajo vsaj 6 označenih imenskih entitet. Ker pa imenske oznake niso označene v celotnem korpusu ssj500k, je bil izplen končnih besedil za našo analizo še dodatno manjši. Korpus coref149 tako sestavlja 149 odstavkov, 1.060 povedi in 26.960 pojavnic.

3.1 Postopek in pravila označevanja

Ideja o označitvi slovenskega besedila za odkrivanje koreferenčnosti se je rodila ob prvi izvedbi predmeta *Obdelava naravnega jezika* na Fakulteti za računalništvo in informatiko Univerze v Ljubljani v študijskem letu 2016/2017. Študenti naj bi označili posamezne dokumente, pri čemer bi vsako besedilo označila dva študenta, na podlagi česar bi lahko izračunali ujemanje označevalcev in izbrali končne oznake. Zaradi majhne udeležbe študentov pri

označevanju nismo "pokrili" pripravljenega korpusa niti z rezultati enega študenta. Zato smo z uporabo orodja WebAnno (Yimam 2013) označeni korpus izdelali sami. Prvo označevanje smo izvedli spomladi 2017, drugo pa jeseni 2017, ko smo ponovno pregledali celoten korpus in vnesli popravke.

Pri označevanju smo se omejili na omenitve entitet, ki predstavljajo:

- osebo ali skupino oseb (*[Marijan Schiffrer]*, *[poslanec SKD]*, *[zamejci]*),
- organizacijo (*[Biotehniška fakulteta]*, *[BTF]*, *[Kmetijski poskusni center Jable]*) ali
- zamljepisno ime (*[Jable]*, *[Rodica]*, *[Maribor]*).

Označili smo vse možne omenitve, ki se sklicujejo na entiteto zgornjega tipa in so izrecno omenjene v besedilu. Če entiteta v celotnem dokumentu ni bila omenjena poimensko oz. je bil sklic nanjo le z zaimkom, je nismo označili (npr. omenitve, ki kažejo na pripovedovalca v prvoosebnem besedilu). Če se je neka entiteta pojavila še v drugi, razširjeni entiteti, sta obe entiteti predstavljeni kot popolnoma svoji entiteti (npr. oseba, ki nastopa v besedilu sama, in skupina oseb, ki vključuje tudi to osebo). V korpusu smo označili tudi, ali med omenitvama obstaja koreferenčnost, nismo pa definirali različnih tipov koreferenčnih povezav. Npr. *[nacionalni TV hiši [ZDF]₂ in [ARD]₃]₁* smo označili s tremi omenitvami, saj *ZDF* in *ARD* pomenita vsaka svojo entiteto, celoten primer pa novo – združeno entiteto, ki predstavlja obe televizijski hiši skupaj.

Pri označevanju omenitev, ki so v besedilu dopolnjene z levim prilastkom (angl. *predicate complement*, *premodifier*), smo dali oznako za dve omenitvi, ki se prekrivata in kažeta na isto entiteto. Npr. v besedilu *[strokovnega direktorja [KC]₂ [Zorana Arneža]₁]₁* smo označimo dve omenitvi, ki se nanašata na osebo *Zoran Arnež*. Tretja omenitev se sklicuje na *Klinični center*. Nasprotno pa v primeru pristavkov in desnih samostalniških prilastkov do tega ne prihaja, saj je omenitev navadno dopolnjena z dodatnim besedilom, ki je pri pristavku

ločena še z ločilom. Primera tovrstnih imenitev sta *[Mag. Franc Avberšek]₁ , [direktor [velenjskega podjetja]₂]₁* in *[Olimpijskega komiteja [Slovenije]₂]₁ ([OKS]₁)]*, pri čemer *velenjsko podjetje* in *Slovenija* nista povezani z ostalima dvema omenitvama.

Poleg omejitve na omenitve, ki so v besedilu izrecno zapisane, smo posebno pozornost namenili še elipsi (izpustu) osebka. Če je bila omenitev na entiteto v vlogi osebka v povedi izpuščena (Holozan 2015: 62), smo kot omenitev označili povedkovo vez, iz katere se lahko razbere koreferenčnost. Primeri povedkovih vezi so *[je] lepa*, *[je bila] lepa*, *[postal je] učitelj* ali *[mora] delati*. V povedi smo označili le prvo povedkovo določilo oz. povedek. Če se je v povedi nahajal še drugi tip omenitve na entiteto, smo v tej povedi označili le to omenitev in ne dodatno še povedka. Dobesedni navedek smo obravnavali kot novo poved.

3.2 Opis korpusa

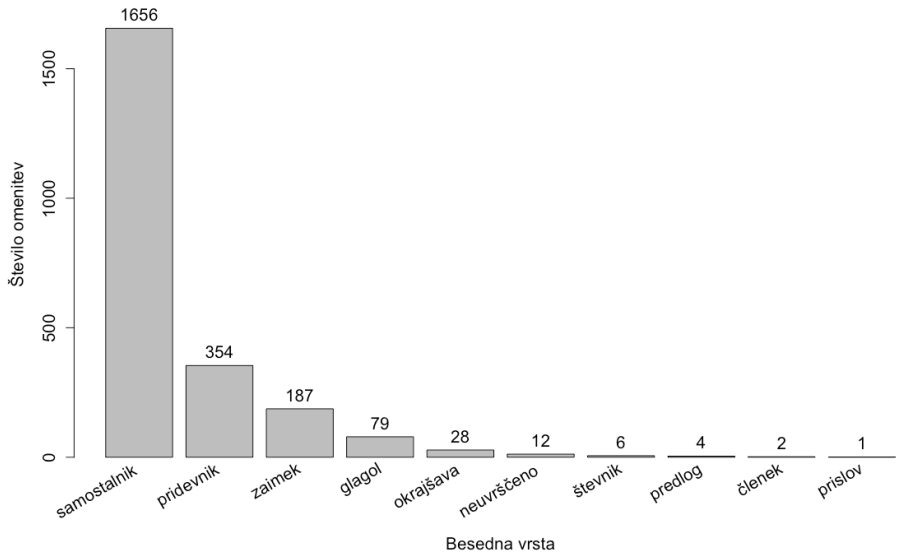
V Tabeli Tabela 1 so predstavljene osnovne lastnosti koreferenčnega korpusa coref149. Korpus vsebuje precejšnje število trivialnih entitet (angl. *singleton entity*), tj. entitet, ki so sestavljene le iz ene omenitve. Število antonomazij smo pridobili avtomatsko, in sicer sodijo mednje vsi primeri, pri katerih se med dvema koreferenčnima omenitvama pojavi le en znak. Znaki, ki ločujejo antonomazicijske pare omenitev, so v korpusu naslednji: '-', '*v*', '*(*', '*/*' in ','. Kot poseben primer navajamo prekrivne omenitve in prekrivne omenitve iste entitete. Gre za posamezne pare omenitev, med katerimi obstaja presek z vsaj eno pojavnico. Pri slednjih gre za omenitve, ki so dopolnjene z dodatnim opisom spredaj.

Lastnost	**Število**
Entitete	1.277
Omenitve	2.329

Trivialne entitete	831
Antonomazije	40
Prekrivne omenitve	215
Prekrivne omenitve iste entitete	97

Tabela 1: Lastnosti korpusa coref149.

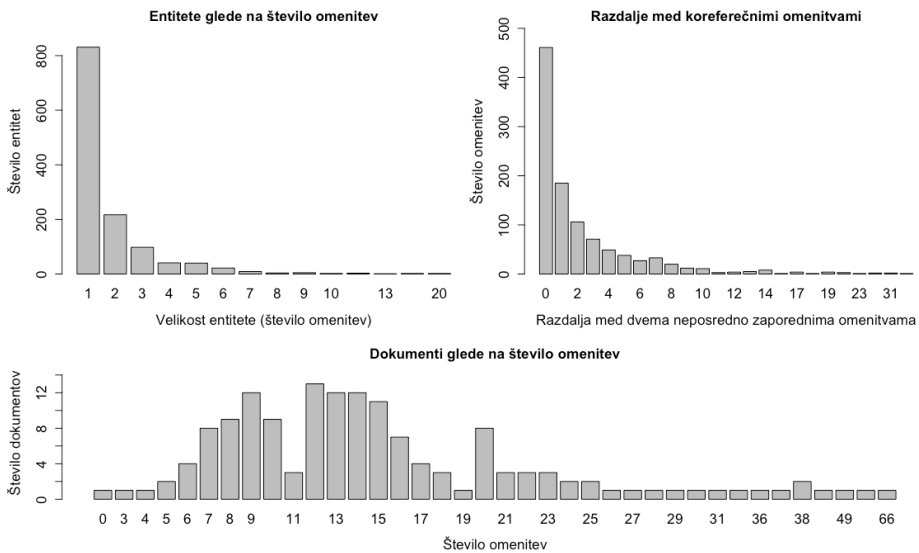
Na Sliki Slika 1 prikazujemo porazdelitev omenitev glede na njihovo besedno vrsto (ta izhaja iz oblikoskladenjskih oznak korpusa ssj500k). Besedne vrste, ki smo jih priredili posameznim omenitvam, smo določili na podlagi besedne vrste prve pojavnice v omenitvi (za jezikoslovno analizo bi bilo bolje, če bi besedno vrsto določili na podlagi jedra besedne zveze). Pri podatkih lahko opazimo, da prednjačijo samostalniki, ki so povečini hkrati tudi imenske entitete. Sledijo jim pridevniki, ki se v tolikšni meri pojavijo predvsem zaradi pridevnikov, ki se nahajajo na začetku omenitev kot levi prilastki v samostalniških zvezah. Primeri takšnih so *[Slovensko cesto]*, *[strokovni direktor]*, *[Microsoftov]* ali *[Srednji Ameriki]*. Pričakovano nato sledijo zaimki. Kot smo omenili v prejšnjem poglavju, je osebek v slovenščini lahko tudi izpuščen, zato smo v teh primerih označevali dele povedka. Tako v kategoriji glagolov nastopajo omenitve kot npr. *[je služil]*, *[je]*, *[so]*, *[Delal je]*. Ostale kategorije predstavljajo le specifične primere besed, ki se pojavljajo na začetku omenitev – npr. pri okrajšavah se poleg okrajšav pojavijo tudi omenitve oseb z akademskimi nazivi (*[dr. Lahovnik]*) ali okrajšanim osebnim imenom (*[B. Bonnaud]*).



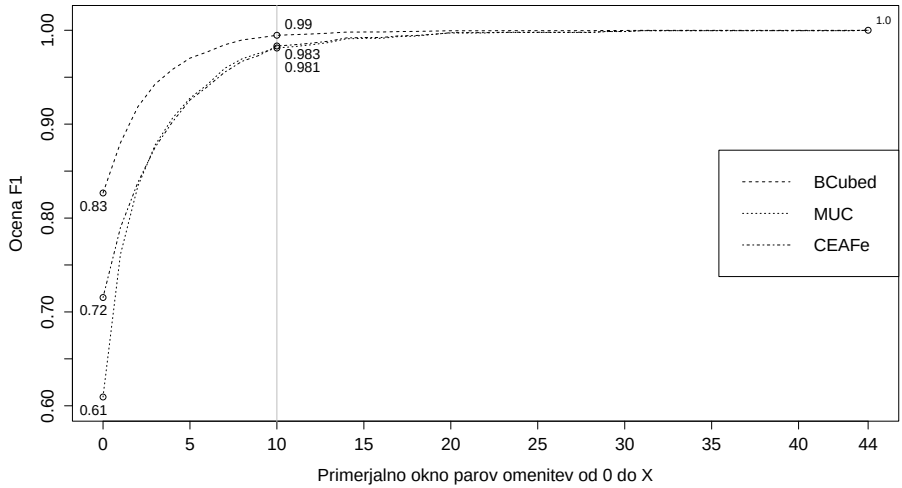
Slika 1: Porazdelitev omenitev glede na besedno vrsto prve pojavnice v omenitvi v korpusu coref149.

Slika **Slika 2** prikazuje različne porazdelitve lastnosti korpusa coref149. Kot smo videli že v Tabeli Tabela 1, je v korpusu največ trivialnih entitet, pri čemer so dovolj dobro zastopane tudi entitete, ki vsebujejo do pet (40 entitet) ali šest (22 entitet) omenitev. Poročanje o razdalji med dvema zaporednima koreferenčnima omenitvama je pomembno, če algoritem, ki bi ga uporabili za odkrivanje koreferenčnosti, uporablja le pare v določenem oknu v zaporedjih koreferenčnih omenitev. Na podlagi tega okna lahko ugotovimo njegovo največjo potencialno uspešnost. Razdalja 0 pomeni, da si morata biti koreferenčni omenitvi neposredno zaporedni, razdalja 1 pomeni, da je med njima neka tretja omenitev, in tako analogno dalje. V primeru idealnega algoritma s primerjanjem parov omenitev, ki bi združeval koreferenčne omejitve do določene razdalje, bi dosegli uspešnosti, ki so prikazane na Sliki Slika 3. Algoritem deluje tako, da vse omenitve v besedilu predstavlja kot zaporedje omenitev in nato primerja vse možne pare omenitev med seboj – če sta omenitvi označeni kot koreferečni, jih tako označi tudi algoritem, sicer ne.

Če bi torej primerjali vse možne pare, bi algoritem dosegel 100% uspešnost. Ker je vseh možnih parov veliko in to lahko vpliva na hitrost izvajanja algoritma, se sprehodimo čez zaporedje omenitev in gledamo le pare v določenem oknu omenitev (razdalje). Npr. če gledamo omenitve do razdalje 2, bomo za vsako omenitev pogledali le, ali so koreferenčne s katero od treh omenitev desno od nje (razdalja pomeni število vmesnih omenitev). Iz obeh slik opazimo, da lahko do vključno razdalje 10 dosežemo večino parov koreferenčnosti: 96 % (1013/1052). Analiza dokumentov glede na vsebnost števila omenitev pa pokaže, da večina dokumentov vsebuje od 5 do 25 omenitev, pri čemer eden izmed dokumentov (odstavek *ssj12.50*) ne vsebuje nobene omenitve. Dokumenti, ki vsebujejo manj kot 6 omenitev, se pojavijo, ker ne vsebujejo nobene entitete takšnega tipa, kot smo ga označevali (npr. *[Satellite M30X]* ali *[SD(TM) Card]*).



Slika 2: Porazdelitve osnovnih lastnosti korpusa coref149.



Slika 3: Rezultati odkrivanja koreferenčnosti z uporabo idealnega algoritma s primerjanjem parov omenitev.[1]

4 ODKRIVANJE KOREFERENČNOSTI V SLOVENSKEM JEZIKU

4.1 Algoritmi

Kot smo že navedli, smo za odkrivanje koreferenčnosti uporabili algoritem SkipCor (Žitnik in dr. 2014), ki za odkrivanje koreferenčnosti uporablja statistični algoritem pogojnih naključnih polj. Algoritem za vsak dokument zgradi zaporedje omenitev in nad tem zaporedjem označuje, katere omenitve so med seboj koreferenčne. Ker je uporabljena linearnoverižna struktura modela (angl. *linear-chain*), lahko algoritem ugotavlja le koreferenčnosti med neposredno zaporednimi omenitvami v zaporedju. Zaradi tega za vsak dokument na vhodu podamo več zaporedij: osnovno zaporedje, zaporedje z vsako drugo omenitvijo, zaporedje z vsako tretjo omenitvijo, …, zaporedje z vsako *n*-to omenitvijo. Z uporabo takega postopka lahko odkrivamo

[1] Vse omenitve smo predstavili kot zaporedje omenitev in jih nato primerjali z drugimi do določene razdalje (znotraj okna omenitev). Zelo dobro uspešnost idealni algoritem doseže že s primerjanjem do razdalje 10. Uporabljene metrike za ocenjevanje koreferenčnosti so podrobneje opisane v razdelku 4.1.2.

koreferenčne omenitve na razdalji do *n*, pri čemer se lahko vzporedno izvede napovedovanje z več modeli, tako da pregledovanje $O(n^2)$ parov omenitev, ki ga lahko izvajamo s tradicionalnimi algoritmi strojnega učenja, ni potrebno. Na ta način evalviramo tudi uspešnost napovedovanja med vsemi možnimi pari omenitev, pri čemer tvorimo zaporedja omenitev dolžine dva in uporabimo isti algoritem – SkipCor pari.

Uspešnost algoritma smo primerjali še z osnovnimi modeli. *Trivialni* model vsako omenitev uvrsti v trivialno entiteto. Model *vse v enem* vse omenitve v dokumentu uvrsti v isto entiteto, tako da je rezultat odkrivanja koreferenčnosti v tem primeru natanko ena entiteta na dokument. Model *točnega ujemanja* pa je model, ki primerja vse možne pare omenitev na določeni razdalji in jih vedno pravilno uvrsti glede na označene podatke.

4.1.1 ZNAČILKE

V modelih, ki jih uporabljamo, izbira ustreznih in informativnih značilk zelo vpliva na uspešnost učnih algoritmov. Kot parameter smo podali predloge funkcij, t. i. funkcije značilk (angl. *feature function*), ki s prehodom čez učni korpus zgenerirajo končne značilke. Funkcija značilk je predpis, ki na vhodu prejme omenitev in učni korpus ter kot izhod vrne značilko. Npr. če definiramo funkcijo značilk, ki generira značilke, ki pomenijo prve tri črke omenitve, in če je podana omenitev *Janez*, bi ta funkcija vrnila značilko *Jan*. Enako značilko bodo dobile tudi vse omenitve, ki se začnejo na *Jan-*. Generiranje značilk prek funkcij lahko ustvari mnogo različnih značilk, od katerih se lahko nekatere pojavilo zelo malokrat, zato pri učenju uporabimo le tiste, ki se v učni množici pojavijo vsaj petkrat. V literaturi je bilo na področju odkrivanja koreferenčnosti predlaganih že kar nekaj predlog funkcij značilk (Ng 2008; Soon in dr. 2001; Bengtson in Roth 2008; Broscheit in dr. 2010; Attardi in dr. 2010; Fernandes

in dr. 2012), ki jih lahko razdelimo v naslednje štiri kategorije:[2]

Oblika. Značilke kategorije *oblika* se nanašajo na lastnosti oblike pojavnic v omenitvi. Primeri značilk, ki sodijo v to kategorijo, so lema omenitve, pisanje omenitve ali besede pred njo z veliko začetnico, ujemanje para omenitev v predponi ali priponi omenitve, nahajanje omenitve znotraj dobesednega navedka, ujemanje podniza med omenitvama, ujemanje omenitve kot kratice ali okrajšave z drugo omenitvijo, podobnost zapisa omenitve (razdelitev rezultata metrike podobnosti JaroWinkler v tri skupine) in pojavitev omenitve kot antonomazija druge omenitve.

Leksika. Leksikalne značilke izhajajo predvsem iz oblikoskladenjskih oznak. Pogojna naključna polja podobno kot logistična regresija samodejno upoštevajo odvisnosti med značilkami, zato smo oblikoskladenjske oznake za slovenski jezik razbili na posamezne dele in iz njih tvorili značilke. Tako smo definirali uporabo besedne vrste omenitve ali dveh členov pred in za omenitvijo, vrsto oblikoskladenjske oznake, število, sklon, osebo, število pri svojilnem zaimku in zapis oblikoskladenjske oznake.

Pomen. Pomenske značilke se nanašajo na identifikacijo pomena omenitve. Te značilke so ujemanje v imenski entiteti dveh zaporednih omenitev, par imenskih entitet dveh zaporednih omenitev, imenska entiteta trenutne omenitve, identifikacija živosti, spola pri tretji osebi edninskih svojilnih zaimkov in spola. Slednje tri oznake so del oblikoskladenjskih oznak.

Razdalja. V nekaterih primerih je pomembno vedeti, kako oddaljeni sta omenitvi, kar lahko nakazuje na njuno koreferenčnost. Algoritem SkipCor implicitno že upošteva razdaljo v vsakem zgrajenem modelu. Poleg tega definira še značilke, kot so razdalja v stavkih med omenitvama (število stavkov, ki se med omenitvama pojavijo v izvornem besedilu), razdalja v členih med omenitvama (število pojavnic, ki se med omenitvama pojavijo v izvornem

[2] Natančna definicija značilk se lahko razbere iz izvorne kode iz razreda `SloCoreferenceFeatureFunctionPackages`.

besedilu) in indikatorska značilka nahajanja zaporednih omenitev v istem stavku.

4.1.2 OCENE USPEŠNOSTI

V času razvoja metod za odkrivanje koreferenčnosti je bilo predlaganih kar nekaj ocen uspešnosti, a še vedno ni jasno odločeno, katero je najbolje uporabljati. Prve ocene (Chinchor 1991; Chinchor in Sundheim 1993) so temeljile na grafovskih značilnostih rezultatov ali na primerjavi parov omenitev. Kljub temu, da je bilo število odkritih koreferenčnosti majhno, so bile njihove ocene visoke in so slabo ločevale dobre sisteme od slabih. V svoji analizi smo uporabili novejše metrike, ki so izpeljanke ocene F_1 z različnimi pristopi računanja natančnosti in priklica:

MUC. Glavni namen razvoja ocene MUC (Vilain in dr. 1995) je bil doseči boljšo razlago rezultatov pri odkrivanju koreferenčnosti. Ime je dobila po istoimenski konferenci – MUC (angl. *Message Understanding Conference*). Ocena temelji na povezavah (angl. *link-based metric*) med omenitvami in je bila v dosedanjih raziskavah uporabljena največkrat. Natančnost ocene izračunamo tako, da preštejemo, kolikšno bi bilo najmanjše število povezav, ki bi jih morali dodati k rezultatu, da bi bile entitete razpoznane pravilno. Nasprotno priklic ocene meri, koliko povezav bi morali odstraniti, da ne bi obstajala nobena več. Ocena MUC torej bolje oceni sisteme, ki entitetam priredijo veliko omenitev, medtem ko popolnoma ignorira razpoznavanje entitet, ki so sestavljene le iz ene omenitve – trivialne entitete. Npr. če sistem v dokumentu razpozna le eno entiteto (tj. vse omenitve so povezane med seboj), bo dosegel 100% priklic in precej visoko natančnost.

BCubed. Zaradi slabosti ocene MUC so raziskovalci predlagali oceno BCubed (Bagga in Baldwin 1998), ki se osredotoča na posamezne omenitve in meri preseke med napovedanimi gručami omenitev ter pravimi gručami. Naj bo k prava entiteta in r razpoznana entiteta, ki je sestavljena iz omenitev m. Priklic

omenitve *m* izračunamo kot $|k \cap r|/|k|$ in natančnost kot $|k \cap r|/|r|$. Prednost te ocene je, da upošteva tudi entitete z le eno omenitvijo in daje višjo težo razdruževanju ali združevanju entitet z več omenitvami.

CEAF. S pogosto uporabljano oceno CEAF (Luo 2005) (angl. *Constrained Entity-Alignment F-Measure*) so raziskovalci želeli doseči boljšo razlago rezultatov oz. razlik med sistemi. Ocena pomeni delež pravilno razpoznanih entitet. Za namene naše evalvacije smo uporabili oceno, ki temelji na ocenjevanju entitet (obstaja tudi različica, ki temelji na omenitvah) in želi razpoznani entiteti najti čim boljši par v množici pravih entitet. Pri oceni izračunamo priklic kot (popolna podobnost) / ($|k|$) in natančnost kot (popolna podobnost) / ($|r|$), kjer za problem največjega dvojnega ujemanja (angl. *maximum bipartite matching*) uporabljamo Kuhn-Munkresov algoritem.

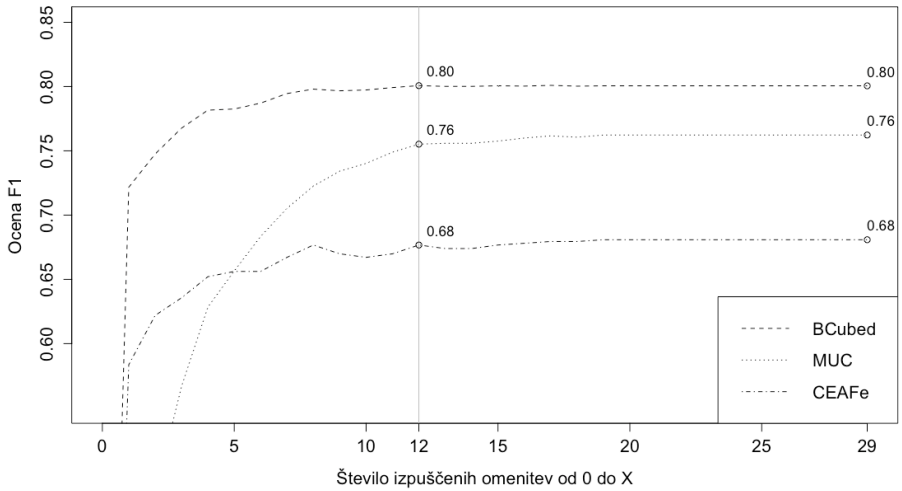
4.2 Rezultati

Kot smo pojasnili že v razdelku 3, smo pri analizi[3] uporabili podatkovno množico coref149, ki smo jo pridobili iz korpusa ssj500k. Korpus smo razdelili v učni in testni del v razmerju 70 : 30 in za namene ponovljivosti fiksirali začetno vrednost naključnega generatorja. Za učenje modelov smo uporabili le značilke, ki se v učni množici prožijo vsaj petkrat.

Algoritem SkipCor zahteva nastavitev parametra, ki definira število modelov, ki se bodo zgradili za različna zaporedja omenitev. Na Sliki Slika 3 je razvidno, da bi idealni algoritem dosegel že zelo veliko natančnost pri zaporedjih omenitev od 0 do 10 izpuščenih omenitev. Rezultati na Sliki 4 so podobni, na podlagi česar smo se odločili v vseh nadaljnjih analizah uporabljati zaporedja omenitev z do 12 izpuščenimi omenitvami. Na Sliki 4 se parameter evalvira le do vrednosti 29, ker je to največja razdalja med dvema neposredno

[3] Izvorna koda z vsemi nastavitvami za zagon analiz je dostopna v razredu `SSJ500kCoreference`.

koreferenčnima omenitvama v učni množici in bi ob nadaljevanju z večjimi razdaljami rezultati ostali enaki.



Slika 4: Rezultati algoritma SkipCor z uporabo vseh značilk v odvisnosti od parametra izpuščenih omenitev.[4]

Tabela Tabela 2: Rezultati odkrivanja koreferenčnosti v korpusu coref149 z osnovnimi in predlaganimi modeli. prikazuje rezultate odkrivanja koreferenčnosti v korpusu coref149. Poleg že zgoraj omenjenih metrik tu podajamo še oceno *CoNLL 2012*, ki je povprečenje treh ostalih metrik. Prve tri vrstice predstavljajo rezultate z osnovnimi modeli, ki jih navajamo zaradi lažje interpretacije rezultatov. Pri metriki MUC opazimo, da ne upošteva trivialnih entitet, medtem ko doseže popoln priklic ob klasifikaciji vseh omenitev v eno entiteto. Pri BCubed je enako: v zadnjem primeru dosežemo popoln priklic, medtem ko ob razvrstitvi vseh omenitev v trivialne entitete dosežemo popolno natančnost. Pri CEAFe pa opazimo, da s trivialnimi entitetami dosežemo

[4] Na podlagi primerjave z rezultati, ki jih dobimo, če primerjamo vse možne pare omenitev, opazimo, da enako dobre rezultate dosežemo že pri razdalji 12, ki smo jo zato uporabili pri nadaljnji analizi. Ocene uspešnosti so podrobneje opisane v razdelku 4.1.2.

približno polovično uspešnost, medtem ko z eno entiteto za vse omenitve dosežemo približno tretjino uspešnosti. Model *točno ujemanje* doseže rezultate blizu 100 %, ker pravilno klasificira vse pare koreferenčnosti, ki se pojavijo na razdalji 29 (naj pa spomnimo, da nekaj koreferenčnosti kljub temu ostane neodkrite, saj se v testni množici nekaj neposredno koreferenčnih omenitev pojavi na daljši razdalji). V drugem delu tabele primerjamo rezultate modelov SkipCor in SkipCor pari. Oba dosežeta primerljive rezultate, ki so po uspešnosti podobni rezultatom odkrivanja koreferenčnosti v ostalih jezikih (gl. spodaj). Iz primerjave z osnovnimi modeli lahko sklepamo, da algoritem koreferenčnost odkriva ustrezno.

Algoritem	**MUC**	**BCubed**	**CEAFe**	**CoNLL 2012**
Trivialni	00,0; 00,0; 00,0	100; 51,9; 68,3	50,8; 50,8; 50,8	39,7
Vse v enem	52,3; 100; 68,7	25,2; 100; 40,3	33,0; 33,0; 33,0	47,3
Točno ujemanje	100; 99,2; 99,6	100; 99,8; 99,9	99,6; 99,6; 99,6	99,7
SkipCor pari	**78,1**; 71,6; 74,7	**83,8**; 77,5; **80,5**	**71,5**; **71,5**; **71,5**	**75,6**
SkipCor	74,8; **77,7**; **76,2**	75,7; **84,9**; 80,1	68,1; 68,1; 68,1	74,8

Tabela 2: Rezultati odkrivanja koreferenčnosti v korpusu coref149 z osnovnimi in predlaganimi modeli.[5]

Ker je podatkovna množica, ki smo jo hkrati z analizo koreferenčnosti slovenskih besedil predstavili v tem delu, edina nam znana, ne moremo izvesti

[5] Ocene uspešnosti so podrobneje opisane v razdelku 4.1.2. V vsakem polju vrednosti zaporedno predstavljajo natančnost, priklic in oceno F. Odebeljeno so označene najboljše ocene vsake izmed metrik.

širše ali bolj poglobljene primerjalne analize. Kljub temu lahko kot zanimivost podamo rezultate odkrivanja koreferenčnosti v angleškem jeziku. Glede na to, da nismo uporabili dodatnih zunanjih korpusov in odkrivali koreferenčnosti med že identificiranimi omenitvami, so rezultati prek kategorije *Gold-standard Closed* primerljivi z drugimi. Isti algoritem – SkipCor, ki smo ga uporabili v tem delu, na angleškem koreferenčnem korpusu CoNLL 2012 (Pradhan in dr. 2012) z razširjenimi in jezikovno odvisnimi značilkami dosega naslednje rezultate: 72,7; 69,8; 41,7[6] (Žitnik in dr. 2014). Trenutno najboljši model z uporabo globokih nevronskih mrež na isti podatkovni množici dosega naslednje rezultate: 74,23; 62,95; 58,70 (Clark in Manning 2016). Če primerjamo oboje rezultate na angleškem besedilu, opazimo, da ta model doseže precej večjo oceno CEAFe, vendar manjšo oceno BCubed. Zanimivo pa je, da v angleškem jeziku zelo dobre rezultate dosega algoritem, ki temelji na zaporedni uporabi filtrov pravil ter na množici CoNLL 2011 (Pradhan in dr. 2011), in sicer: 63,9; 70.0; 48.3 (Lee in dr. 2011). Na podlagi teh primerjav lahko ugotovimo, da bi bilo možno za slovenski jezik izdelati prav tako dobre modele, kot so izdelani za angleški jezik, le da bi potrebovali večji in bolj bogato označen korpus (tj. korpus, označen tudi s tipom koreferenčnih povezav).

V Tabeli Tabela 3 primerjamo pomembnosti posameznih skupin značilk za odkrivanje koreferenčnosti. Ob uporabi posameznih skupin smo pričakovano najboljše rezultate dosegli z uporabo skupine značilk tipa *oblika*, saj omenitve same po sebi nosijo veliko koreferenčne podobnosti. To je posledica tega, da se na nekatere entitete pogosto sklicujemo z enakimi omenitvami. K temu lahko dodamo še podobne omenitve in omenitve, ki se ujemajo prek podnizov. Značilke tipa *leksika* in *pomen* dosežejo posamezno približno enako uspešnost, medtem ko se značilke tipa *razdalja* odrežejo najslabše. Najboljše rezultate sodežemo s kombinacijo značilk, medtem ko lahko opazimo, da ni bistvene razlike med uporabo vseh ali le značilk tipa *oblika* in *leksika* skupaj.

[6] Vrednosti predstavljajo F_1 ocene MUC, BCubed in CEAFe.

Skupina značilk	**MUC**	**BCubed**	**CEAFe**	**CoNLL 2012**
Oblika (O)	**72,7**;**63,8**;**68,0**	80,6; **77,4**; **79,0**	**69,2**;**69,2**;**69,2**	**72,0**
Leksika (L)	60,8; 25,9; 36,3	86,2; 60,9; 71,4	56,4; 56,4; 56,4	54,7
Pomen (P)	63,9; 10,9; 18,6	**95,9**; 55,0; 70,0	54,7; 54,7; 54,7	47,7
Razdalja (R)	22,6; 1,9; 3,5	95,6; 52,6; 67,9	51,1; 51,1; 51,1	40,9
O in L	74,3; 77,2; 75,7	74,4; **84,9**; 79,3	**68,2**;**68,2**;**68,2**	74,4
O; L in P	72,8; 75,5; 74,1	74,7; 83,9; 79,0	67,4; 67,4; 67,4	73,5
O; L; P in R	**74,8**; **77,7**; **76,2**	**75,7**;**84,9**;**80,1**	68,1; 68,1; 68,1	**74,8**

Tabela 3: Analiza uporabe različnih skupin značilk za odkrivanje koreferenčnosti z uporabo algoritma SkipCor.[7]

V Tabeli Tabela 4 podajamo še rezultate po odstranitvi posebnosti označenega korpusa coref149. Posebnosti sta označevanje prekrivnih omenitev istih entitet in povedkov. Npr. za omenitvi *[strokovnega direktorja KC [Zorana Arneža]$_1$]$_1$* smo za namene te analize upoštevali le daljšo omenitev, krajšo (tj. *Zorana Arneža*) pa smo odstranili. Sklepali bi lahko, da bodo v tem primeru rezultati občutno slabši, saj izgledajo takšni pari omenitev enostavni za klasifikacijo v primerjavi z drugimi. Analiza je pokazala, da je uspešnost v tem primeru slabša, vendar le za približno 3 %. Pri povedkih smo odstranili še omenitve, ki so označena povedkova določila. Glede na to, da smo pokazali, da je najbolj

[7] Ocene uspešnosti so podrobneje opisane v razdelku 4.1.2. V vsakem polju vrednosti zaporedno predstavljajo natančnost, priklic in oceno F. Odebeljeno so označene najboljše ocene vsake izmed metrik.

uspešna skupina značilk tipa *oblika* in da so pojavnice omenitev, ki smo jih upoštevali pri analizi v Tabeli Tabela 4, vedno različne od zaimkov ali imen, bi lahko sklepali, da bo po njihovi odstranitvi rezultat boljši od najboljšega v Tabeli Tabela 4. A očitno ni tako, razlog pa je verjetno v tem, da deli oblikoskladenjskih oznak nosijo dovolj informacij, da je možno pravilno napovedovati tudi takšne koreferenčne povezave.

Lastnost	**MUC**	**BCubed**	**CEAFe**	**CoNLL 2012**
Prekrivne omenitve	70,9; 70,0; 70,4	77,3; 81,4; 79,3	68,2; 68,2; 68,2	72,6
Povedki	**74,4**;**75,3**;**74,9**	76,6; **84,4**; 80,3	68,9; 68,9; 68,9	**74,7**
Prekrivne omenitve in povedki	72,9; 71,0; 71,9	**79,4**;82,5;**80,9**	**70,1**; **70,1**; **70,1**	74,3

Tabela 4: Rezultati algoritma SkipCor nad podatki z odstranjenimi posebnostmi korpusa coref149.[8]

4.2.1 ANALIZA NAPAK

Ob ročnem pregledu odkritih koreferenčnosti smo opazili dva tipa pogostih napak:

A: V primeru prekrivnih omenitev različnih entitet v korpusu ssj500k nastopajo oznake imenskih entitet le za en tip. V besedilu *[Raziskovalno postajo za živinorejo [Rodica]$_1$]$_2$* nastopata omenitvi, ki se nanašata na dve različni entiteti, vendar jih kljub temu pri odkrivanju koreferenčnosti pogosto klasificiramo v isto entiteto. K temu dodatno pripomorejo še oznake imenskih

[8] Ocene uspešnosti so podrobneje opisane v razdelku 4.1.2. V vsakem polju vrednosti zaporedno predstavljajo natančnost, priklic in oceno F. Odebeljeno so označene najboljše ocene vsake izmed metrik.

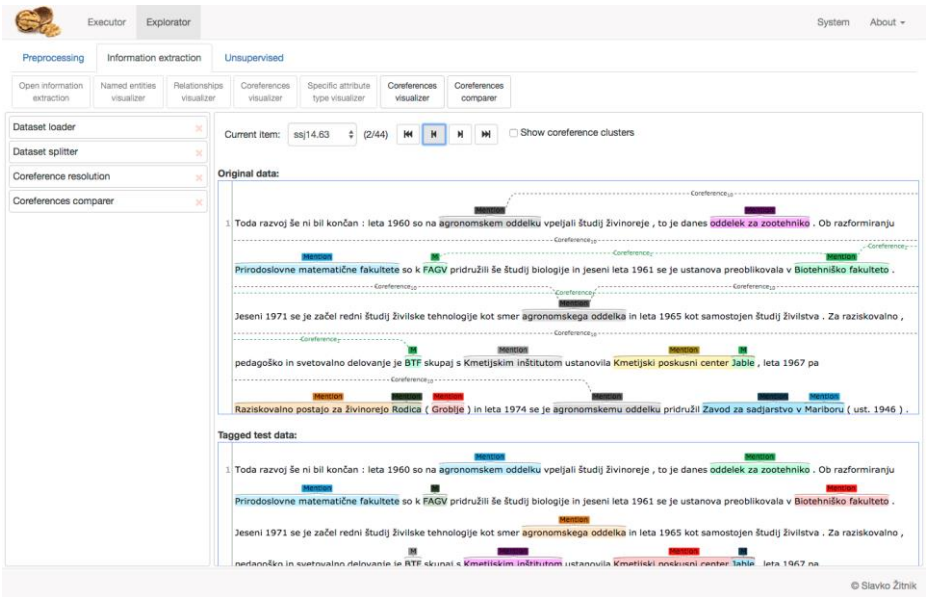
entitet, saj so vse besede daljše omenitve označene kot *stvarno* ime, čeprav bi beseda *Rodica* v zgornji zvezi morala nositi poleg te še dodatno oznako *zemljepisno* ime. Enako velja tudi za primer, pri katerem vse omenitve algoritem razreši v isto entiteto: *[KRŠKE]*$_1$, *[Krško]*$_1$, *[krški]*$_1$, *[Kmečke zadruge Krško]*$_2$.

B: Nekatere entitete v rezultatu nastopajo z veliko omenitvami. Npr. omenitve *[Njena]*$_1$, *[lokalna policija]*$_1$, *[njeni]*$_2$, *[Kfor]*$_3$, *[Njihova]*$_3$, *[Nato]*$_4$, *[Nata]*$_4$, *[Jugoslovanske oborožene sile]*$_5$ algoritem razreši kot eno entiteto. Do tega pride predvsem zaradi podobnih zaimkov, ki nastopajo kot omenitve in prek katerih se sklicujemo na različne entitete. V tem primeru so to *njena*, *njeni* ali *njihova*. V fazi gručenja rezultatov v algoritmu SkipCor so neposredno upoštevali klasifikacijo modela med omenitvami in iterativno združevali vse omenitve, ki so bile prepoznane kot koreferenčne. Če bi dodatno izboljšali fazo gručenja, bi lahko del napak tega tipa odpravili.

Raziskovalca Kummerfeld in Klein (2013) sta definirala klasifikacijo napak in predlagala sistem za evalvacijo, ki ugotovi število transformacij, ki so potrebne, da vrnjen rezultat algoritma preoblikujejo v popolnoma pravilen rezultat. Definirala sta pet transformacij: posodobi razpon (angl. *alter span*), razčleni (angl. *split*), izbriši (angl. *remove*), predstavi (angl. *introduce*) in združi (angl. *merge*). Odkrivanje koreferenčnosti sta predstavila kot celostno nalogo, tj. vključno z identifikacijo omenitev. Zaradi tega so v našem primeru smiselne samo transformacije *razčleni* in *združi*, pri čemer naš algoritem napravi 94 napak tipa razčleni in 80 napak tipa združi. Operacije *razčleni* razčlenijo entiteto z dodatnimi omenitvami na več entitet, kar je potrebno pri tipih napak, ki smo jih predstavili zgoraj. Operacije *združi* pa sestavijo več delno razpoznanih entitet v eno skupno. V zgornjem primeru se tako entiteti, ki se nanaša na *Kfor*, doda še entiteta z omenitvijo *vojaške sile*, ti dve namreč prej nista bili razrešeni skupaj. Predlagana pravila smo izvedli zaporedno, tako smo v našem primeru najprej razčlenili vse mešane entitete in jih nato združili. Število razčlenitev je večje kot število združitev, iz česar lahko sklepamo, da naš

algoritem naredi več napak z razreševanjem nepovezanih omenitev v isto entiteto.

Ob analizi odkrivanja koreferenčnosti v slovenskem jeziku smo hkrati izdelali še spletno aplikacijo *nutIE* (Slika 5), s katero lahko uvozimo podatke, zgradimo modele in pregledujemo ter primerjamo rezultate z označenimi. Zasnova orodja je bila že predstavljena (Žitnik in dr. 2017) in vključuje programsko knjižnico z metodami za ekstrakcijo informacij ter je prosto dostopna na spletu.[9]



Slika 5: Orodje za pregledovanje analize koreferenčnosti *nutIE*.

5 SKLEP

Pri prvem na slovenskem jeziku preizkušenem avtomatskem odkrivanju koreferenčnosti, ki smo ga opisali v prispevku, smo pri oceni CoNLL 2012 dosegli uspešnost 76 %. S tem smo pokazali, da lahko v slovenščini koreferenčnost odkrivamo enako uspešno kot v ostalih, po številu govorcev

[9] Programska knjižnica z zalednim delom: https://bitbucket.org/szitnik/nutie-core in spletni vmesnik: https://bitbucket.org/szitnik/nutie-web.

večjih jezikih. Za še boljše rezultate in bolj poglobljene analize bi bilo treba v prihodnje zgraditi še večji ter bolj reprezentativen korpus in ga opremiti z bogatejšimi koreferenčnimi oznakami (prim. Tabelo 5 v Žitnik in dr. 2014). Šele natančnejšim analizam bi lahko sledil predlog bolj informativnih značilk, poleg tega pa bi morali upoštevati tudi značilnosti slovenskega jezika, kot so nanašanje na entitete prek stavčnih prilastkov, prislovnih zaimkov ali izpuščenih osebkov. Pri slednjih bi lahko namesto označevanja drugih oblik omenitev preskusili tudi vrivanje eksplicitnega (umetnega) zaimka v izvorno besedilo v fazi predprocesiranja besedila, čeprav na ta način verjetno ne bi dosegli izboljšanja, saj se vse kategorialne lastnosti zaimkov kažejo tudi v povedkovih določilih, ki smo jih označevali.

LITERATURA

Attardi, G., Rossi, S. D., in Simi, M. (2010): TANL-1: coreference resolution by parse analysis and similarity clustering. *Proceedings of the 5th International Workshop on Semantic Evaluation*: 108-111. Uppsala.

Bagga, A., in Baldwin, B. (1998): Algorithms for scoring coreference chains. *The first international conference on language resources and evaluation workshop on linguistics coreference*: 563-566.

Bejan, C., Titsworth, M., Hickl, A., in Harabagiu, S. (2009): Nonparametric bayesian models for unsupervised event coreference resolution. *Advances in Neural Information Processing Systems*: 73-81.

Bejan, C. A., in Harabagiu, S. (2010): Unsupervised event coreference resolution with rich linguistic features. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*: 1412-1422.

Bengtson, E., in Roth, D. (2008): Understanding the value of features for coreference resolution. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*: 294-303. Waikiki.

Broscheit, S., Poesio, M., Ponzetto, S. P., Rodriguez, K. J., Romano, L.,

Uryupina, O., ... in Zanoli, R. (2010): BART: A multilingual anaphora resolution system. *Proceedings of the 5th international workshop on semantic evaluation*: 104-107. Uppsala.

Bucik, K. (2001): Strukture kohezije: strukture koreferenc in tematske progresije: Diplomsko delo. Ljubljana: Univerza v Ljubljani, Filozofska fakulteta.

Charniak, E. (2001): Unsupervised learning of name structure from coreference data. *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*: 1-7.

Chinchor, N. (1991): MUC-3 evaluation metrics. *Proceedings of the 3rd conference on Message understanding*: 17-24. Pennsylvania.

Chinchor, N., in Sundheim, B. (1993): MUC-5 evaluation metrics. *Proceedings of the 5th conference on Message understanding*: 69-78.

Chinchor, N. A. (1998): Overview of MUC-7/MET-2. *Proceedings of the 7th Message Understanding Conference*. 1-5. San Diego.

Clark, K., & Manning, C. D. (2016): Deep reinforcement learning for mention-ranking coreference models. *Proceedings of EMNLP 2016*: 1-7. Austin.

Culotta, A., Wick, M., in McCallum, A. (2007): First-order probabilistic models for coreference resolution. *Proceedings of The Conference of the North American Chapter of the Association for Computational Linguistics*: 81-88.

Doddington, G. R., Mitchell, A., Przybocki, M. A., Ramshaw, L. A., Strassel, S., in Weischedel, R. M. (2004): The Automatic Content Extraction (ACE) Program-Tasks, Data, and Evaluation. *Proceedings of LREC*: 837-840. Pariz.

Fernandes, E. R., Dos Santos, C. N., in Milidiú, R. L. (2012): Latent structure perceptron with feature induction for unrestricted coreference resolution. *Joint Conference on EMNLP and CoNLL-Shared Task*: 41-

48. Jeju.

Finkel, J. R., Grenager, T., in Manning, C. (2005): Incorporating non-local information into information extraction systems by gibbs sampling. *Proceedings of the 43rd annual meeting on association for computational linguistics*: 363-370.

Glavaš, G., in Šnajder, J. (2015): Resolving Entity Coreference in Croatian with a Constrained Mention-Pair Model. *The 5th Workshop on Balto-Slavic Natural Language Processing*: 17-23.

Haghighi, A., in Klein, D. (2009): Simple coreference resolution with rich syntactic and semantic features. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*: 1152-1161.

Holozan, P. (2015): Sistem za razreševanje koreferenc pri analizi slovenskih besedil in možnosti njegove uporabe. *Slovenščina 2.0*, 3 (1): 60–89.

Huang, S., Zhang, Y., Zhou, J., in Chen, J. (2009): Coreference resolution using markov logic networks. *Advances in Computational Linguistics*, 41 (1), 157-168.

Ju, T. S., Roytberg, A., Ladygina, A. A., Vasilyeva, M. D., Azerkovich, I. L., Kurzukov, M., ... in Grishina, Y. (2014): RU-EVAL-2014: Evaluating anaphora and coreference resolution for Russian. *Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue"*: 681-694.

Korošec, T. (1981): Besediloslovna vprašanja slovenščine. *XVII. seminar slovenskega jezika, literature in kulture*: 173-186. Ljubljana: Filozofska fakulteta.

Krek, S., Erjavec, T., Dobrovoljc, K., Holz, N., Ledinek, N., Može, S. (2015): Training corpus ssj500k 1.4. *Slovenian language resource repository CLARIN.SI*. Dostopno prek: http://hdl.handle.net/11356/1052 (1. marec 2018).

Kummerfeld, J. K., in Klein, D. (2013): Error-driven analysis of challenges in

coreference resolution. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*: 265-277. Seattle.

Lee, H., Peirsman, Y., Chang, A., Chambers, N., Surdeanu, M., in Jurafsky, D. (2011): Stanford's multi-pass sieve coreference resolution system at the CoNLL-2011 shared task. *Proceedings of the fifteenth conference on computational natural language learning: Shared task*: 28-34.

Luo, X., Ittycheriah, A., Jing, H., Kambhatla, N., in Roukos, S. (2004): A mention-synchronous coreference resolution algorithm based on the bell tree. *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*: 136-143.

Luo, X. (2005): On coreference resolution performance metrics. *Proceedings of the conference on human language technology and empirical methods in natural language processing*: 25-32. Vancouver.

Luo, X. (2007): Coreference or not: A twin model for coreference resolution. In Human Language Technologies 2007. *Proceedings of The Conference of the North American Chapter of the Association for Computational Linguistics*: 73-80. Rochester.

McCallum, A., in Wellner, B. (2005): Conditional models of identity uncertainty with application to noun coreference. *Advances in neural information processing systems*, 905-912.

Ng, V., in Cardie, C. (2002): Improving machine learning approaches to coreference resolution. *Proceedings of the 40th annual meeting on association for computational linguistics*: 104-111.

Ng, V. (2008): Unsupervised models for coreference resolution. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*: 640-649. Waikiki.

Ogrodniczuk, M., in Kopeć, M. (2011): End-to-end coreference resolution baseline system for Polish. *Proceedings of the Fifth Language & Technology Conference: Human Language Technologies as a*

Challenge for Computer Science and Linguistics: 167-171. Poznań.

Orasan, C., Cristea, D., Mitkov, R., in Branco, A. H. (2008): Anaphora Resolution Exercise: an Overview. *Proceedings of LREC*: 1-5. Marrakech.

Orasan, C., in Evans, R. J. (2007): NP animacy identification for anaphora resolution. *Journal of Artificial Intelligence Research*, 29 (1): 79-103.

Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., in Xue, N. (2011): Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*: 1-27.

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., in Zhang, Y. (2012): CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. *Joint Conference on EMNLP and CoNLL-Shared Task*: 1-40. Jeju.

Raghunathan, K., Lee, H., Rangarajan, S., Chambers, N., Surdeanu, M., Jurafsky, D., in Manning, C. (2010): A multi-pass sieve for coreference resolution. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*: 492-501.

Rahman, A., in Ng, V. (2009): Supervised models for coreference resolution. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*: 968-977.

Ram, R. V. S., in Devi, S. L. (2012): Coreference resolution using tree CRFs. *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics*: 285-296.

Recasens, M., Màrquez, L., Sapena, E., Martí, M. A., Taulé, M., Hoste, V., ... in Versley, Y. (2010): Semeval-2010 task 1: Coreference resolution in multiple languages. *Proceedings of the 5th International Workshop on Semantic Evaluation*: 1-8.

Soon, W. M., Ng, H. T., in Lim, D. C. Y. (2001): A machine learning approach

to coreference resolution of noun phrases. *Computational linguistics*, 27(4), 521-544.

Toporišič, J. (2004): Slovenska slovnica. Maribor: Obzorja.

Yimam, S. M., Gurevych, I., de Castilho, R. E., in Biemann, C. (2013): WebAnno: A flexible, web-based and visually supported system for distributed annotations. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL 2013)*: 1-6. Sofia.

Vilain, M., Burger, J., Aberdeen, J., Connolly, D., in Hirschman, L. (1995): A model-theoretic coreference scoring scheme. *Proceedings of the 6th conference on Message understanding*: 45-52.

Wellner, B., McCallum, A., Peng, F., in Hay, M. (2004): An integrated, conditional model of information extraction and coreference with application to citation matching. *Proceedings of the 20th conference on Uncertainty in artificial intelligence*: 593-601.

Žitnik, S., Šubelj, L., in Bajec, M. (2014): SkipCor: Skip-mention coreference resolution using linear-chain conditional random fields. *PloS one*, 9(6), e100101.

Žitnik, S., Draskovic, D., Nikolić, B., in Bajec, M. (2017): nutIE—A modern open source natural language processing toolkit. *Proceedings of the 25th Telecommunication Forum (TELFOR)*: 1-4. Belgrade.

Žitnik, S. (2018): Coreference training corpus for Slovene - coref149. *Slovenian language resource repository CLARIN.SI*. Dostopno prek: http://hdl.handle.net/11356/1182 (19. marec 2018).

COREFERENCE RESOLUTION FOR SLOVENE ON ANNOTATED DATA FROM COREF149

Coreference resolution is one of the three main tasks of the information extraction from text. Its goal is to classify all mentions of entities in a text discourse into groups where each group would represent a separate entity. Coreference resolution methods for larger languages are being developed for quite some time, while none has been proposed for the Slovene language yet.

In this paper we present a new manually annotated Slovene corpus for coreference resolution - coref149. We adapt our english-based automatic coreference resolution system SkipCor to the Slovene language and achieve 76% CoNLL 2012 score. We analyse the influences of developed feature functions and check types of the most frequent errors. During the text analysis we have also developed a software library with a web interface, which offers to run all the analysis we describe in this paper and to browse their predictions. The resuls are promising and comparable to the results of coreference analysis for other larger languages. We show that it is possible to implement algorithms for automatic coreference resolution for the Slovene language. Therefore we propose to prepare a larger and better quality corpus featuring all the specifics of the language, which would enable the implementation of generally useful methods for coreference resolution.

Keywords: coreference resolution, Slovene, ssj500k, coref149, SkipCor algorithm