Skladenjski razčlenjevalnik za slovenščino

Kaja Dobrovoljc,[1] Simon Krek,[2] Jan Rupnik[3]

[1]Trojina, zavod za uporabno slovenistiko
Partizanska cesta 5, 4220 Škofja Loka
kaja.dobrovoljc@trojina.si
[2] Laboratorij za umetno inteligenco, Institut Jožef Stefan
Jamova cesta 39, 1000 Ljubljana
simon.krek@ijs.si
[3] Odsek za tehnologije znanja, Institut Jožef Stefan
Jamova cesta 39, 1000 Ljubljana
jan.rupnik@ijs.si

Povzetek

V prispevku opišemo skladenjski razčlenjevalnik za slovenščino, ki je osnovan na razčlenjevalniku MSTParser (Minimum-Spanning Tree Parser) in je bil izdelan v okviru projekta Sporazumevanje v slovenskem jeziku. Naučen je na učnem korpusu ssj500k, ki vsebuje 11.411 ročno preverjenih povedi, razčlenjenih po sistemu odvisnostne drevesnice JOS. Pri sistemu JOS je natančnost razčlenjevalnika 90,43 % za napovedane povezave in 87,52 % za napovedane in označene povezave. Prispevek predstavi zasnovo razčlenjevalnika, učni korpus ssj500k ter sistem odvisnostne drevesnice JOS. Jedro prispevka je podrobna analiza natančnosti razčlenjevanja po skladenjskih oznakah, na koncu prispevka pa predstavimo možnosti za izboljšanje razčlenjevalnika na podlagi analize. Razčlenjevalnik je prosto dostopen pod licenco Apache License V2.0.

Dependency Parser for Slovene

This paper introduces the dependency parser for Slovene based on the MSTParser (Minimum-Spanning Tree Parser), which was developed as part of the Communication in Slovene project. It was trained on the ssj500k training corpus, containing 11.411 manually annotated sentences, parsed in accordance with the JOS Dependency Treebank system. For the JOS system, the parser's accuracy measures 90.43% for unlabelled dependencies and 87.52% for labelled dependencies. The paper presents the design of the parser, the ssj500k training corpus and the JOS Dependency Treebank system. The core of this paper consists of a detailed analysis of the parser's accuracy in relation to dependency labels, which also serves as the basis for the final part of the paper, in which we present possibilities for further improvement. The parser is freely available under the Apache License v2.0.

1. Uvod

Skladenjsko razčlenjevanje predstavlja enega od temeljnih jezikovnotehnoloških postopkov obdelave besedil, ki omogoča in podpira kompleksnejše jezikovne tehnologije, kot so strojno prevajanje, luščenje informacij, govorno komuniciranje, avtomatsko povzemanje, odgovarjanje na vprašanja itd.

Pri obravnavi skladenjskega razčlenjevanja je treba ločiti med uporabljenim (teoretskim) jeziko(slo)vnim modelom in metodo, ki jo uporablja razčlenjevalnik. Jezikoslovni modeli, uporabljeni za različne jezike, so zelo raznoliki, načeloma jih lahko razdelimo v dve večji skupini: sistem odvisnostnih drevesnic ter frazna gramatika, pri čemer velja, da je prva uporabnejša za jezike, ki imajo prost besedni red (npr. slovanski jeziki), druga pa za jezike s staln(ejš)im besednim redom (npr. angleščina). Obstajajo tudi hibridni modeli, ki beležijo informacije tako o odvisnostnih razmerjih kot o notranji sestavi zvez.

Po drugi strani so strojne metode, uporabljene pri razčlenjevanju, tipično treh vrst: razčlenjevanje na podlagi vnaprej pripravljenih pravil[1], statistično razčlenjevanje ter razčlenjevanje s hibridnimi postopki, pri katerih je uporabljena kombinacija obeh metod. Razčlenjevalniki, ki delujejo na podlagi pravil, za svoje delovanje potrebujejo množico ročno napisanih pravil, s katerimi interpretirajo slovničnost ali neslovničnost analiziranih struktur. Na nasprotni strani so statistični razčlenjevalniki, ki ne potrebujejo vnaprej pripravljenih informacij o jeziku, potrebujejo pa učno množico oz. učni korpus, iz katerega izdelajo statistični model, s katerim interpretirajo nova, neznana besedila. Obe metodi imata svoje prednosti in slabosti. Kot prednost statistične metode je mogoče omeniti predvsem manjši časovni in finančni vložek, ki je potreben za hitro doseganje razmeroma dobrih rezultatov, kar je spodbudilo nastanek množice statističnih razčlenjevalnikov v 90-ih letih prejšnjega stoletja.

V prispevku opisujemo statistični razčlenjevalnik, ki je nastal na podlagi razčlenjevalnika MSTParser, ki kot statistično metodo uporablja iskanje minimalnega vpetega drevesa v usmerjenih grafih in ga podrobneje opišemo v nadaljevanju.

2. Odvisnostni model

Pri obravnavanem razčlenjevalniku je bil uporabljen odvisnostni model, razvit v okviru projekta Jezikoslovno označevanje slovenščine (JOS) (Ledinek in Erjavec, 2009; Erjavec et al., 2010). Sistem obsega 10 oznak, ki jih glede na strukturnoskladenjsko raven delimo na tri skupine.

Povezave prvega nivoja (oznake *del*, *dol*, *vez*, *skup*, *prir*) označujejo razmerja znotraj besednih zvez – med jedrnim in nejedrnim delom povedka, predlogom in jedrom besedne zveze, (stavčnimi oz. nestavčnimi) prilastki in odnosnico, modalnim glagolom in dopolnilom ter med povedkom in povedkovim določilom.

[1] Tak razčlenjevalnik za slovenščino je denimo integriran v slovnični pregledovalnik BesAna podjetja Amebis: http://besana.amebis.si/preverjanje/.

Povezave drugega nivoja (oznake *ena, dve, tri, štiri*) sovpadajo s tradicionalnim pojmovanjem osebka, predmeta in prislovnih določil ter se uporabljajo tako za označevanje stavčnočlenskih vlog znotraj stavka kot za označevanje odvisnikov.

Povezava tretjega nivoja je pravzaprav ena sama (*modra*), uporablja pa se za povezovanje hierarhično najvišjih pojavnic (najpogosteje jedrni del povedka glavnega stavka in priredij) ter skladenjsko manj predvidljivih ali oddaljenih struktur, ki bi sicer ostale nepovezane, pa tudi za vsa ločila, ki niso povezana s katero izmed ostalih povezav.

Posamezne oznake in njihova razmerja do skladenjskih kategorij, kakršne poznamo iz tradicionalnih opisov slovenskega jezikoslovja in zaradi robustne narave avtomatskega označevanja niso neposredno prenosljive v predstavljeni model, so podrobneje opisani v specifikacijah za označevalce učnega korpusa[2].

3. Uporabljeni viri in orodja

3.1. MSTParser

3.1.1. Razčlenjevalnik

V pričujočem razdelku bomo povzeli delovanje skladenjskega razčlenjevalnika Minimum-Spanning-Tree Parser (MSTP) (McDonald, Lerman in Pereira, 2006). Najprej uvedimo nekaj osnovnih pojmov. Naj bo besedilo $B = (x_1, \dots, x_m)$ zaporedje povedi, kjer vsaka poved $x_i = (t_{i,1}, \dots, t_{i,n_i})$ predstavlja zaporedje pojavnic $t_{i,j}$, kjer n_i označuje število pojavnic v i-ti povedi. Odvisnostno drevo y za dano poved $x = (t_1, \dots, t_k)$ je definirano kot usmerjen povezan acikličen označen graf $y = (V_x, A_y, L_y)$, kjer $V_x = (v_1, \dots, v_k)$ predstavlja množico označenih vozlišč (prirejenih pojavnicam), $A_x = ((izvor_1, cilj_1), \dots, (izvor_{|A_y|}, \dots, cilj_{|A_y|}))$ predstavlja vektor usmerjenih povezav med pari vozlišč, kjer sta $izvor$ in $cilj$ zaporedji indeksov. $L_y = (l_1, \dots, l_{|A_y|})$ predstavlja množico oznak, prirejenih usmerjenim povezavam. Vsak l_i je element množice možnih oznak povezav: {"dol", "del", "vez", "prir", "skup", "ena", "dve", "tri", "štiri", "modra"}. Zaradi pogojev acikličnosti in povezanosti sledi, da je y drevo. S povezavo $(izvor_i, cilj_i)$ predstavimo skladenjsko odvisnost pojavnice, prirejene vozlišču $cilj_i$, od pojavnice, prirejene vozlišču $izvor_i$. Zaradi tehničnih razlogov dodamo grafu še tehnično vozlišče v_0, ki v množici povezav lahko nastopa samo kot izvor (metaelement). Naloga odvisnostnega razčlenjevanja je za dano poved x poiskati najustreznejše odvisnostno drevo y. Najprej bomo povzeli, kako se v modelu MSTP izračuna ustreznost drevesa y za dano poved x. Predstavili bomo rešitev za iskanje neoznačenih odvisnostnih dreves in kasneje omenili razširitev z označenimi povezavami. S pomočjo funkcije ustreznosti lahko razčlenjevanje prevedemo na problem iskanja najustreznejšega drevesa med vsemi možnimi odvisnostnimi drevesi, ki jih lahko priredimo dani povedi. Sedaj poenostavimo oznake in povezavo $(izvor_i, cilj_i)$ označimo kot (i, j).

Model MSTP je predstavljen kot N-dimenzionalni vektor uteži $w = (w_1, \dots, w_N)$, skupaj z N funkcijami lastnosti $(f_1(\cdot), \dots, f_N(\cdot))$. Vsaka funkcija lastnosti preslika par povedi in usmerjene povezave, $(x, (v_i, v_j))$, v množico $\{0,1\}$. Naloga funkcij lastnosti je zaznava različnih značilnosti dane povezave med parom pojavnic. Naj bo $tag(x) = (pos_1, \dots pos_k)$ zaporedje oblikoskladenjskih oznak, prirejeno povedi x. Navedimo nekaj primerov funkcij lastnosti:

$$f_1(x = (t_1, \dots, t_k), (i,j)) = \begin{cases} 1; \text{če: } t_i = "strgan", t_j = "čevelj" \\ 0; sicer \end{cases},$$

$$f_2(x = (t_1, \dots, t_k), (i,j)) = \begin{cases} 1; \text{če: } t_i = "je",\ pos_i = "Gp-ste-n" \\ 0; sicer \end{cases},$$

$$f_3(x = (t_1, \dots, t_k), (i,j)) = \begin{cases} 1; \text{če: } pos_i = "Rsr", pos_{i-1} = "Kav", pos_j = "Dr", pos_{j+1} = "Somer" \\ 0; sicer \end{cases}.$$

Kvaliteto odvisnostnega drevesa v MSTP izrazimo kot vsoto kvalitet povezav: $score(x, y) = \sum_{i=1}^{|y_A|} s(x, A_y(i))$, kjer kvaliteto i-te povezave izračunamo kot uteženo vsoto vektorja lastnosti: $s\left(x, A_y(i)\right) = \sum_{\ell=1}^{N} w_\ell \cdot f_\ell(x, A_y(i))$. Uteži w_ℓ so rezultat strojnega učenja in predstavljajo pomembnost funkcij lastnosti f_ℓ za merjenje kvalitete povezav. Tehnično poročilo (Mcdonald, Crammer in Pereira, 2005) vsebuje podroben opis vseh tipov funkcij lastnosti, ki so uporabljene v sistemu MSTP. Če na kratko povzamemo, avtorji uporabijo indikatorje različnih kombinacij: beseda izvora (f_1, f_2), beseda cilja (f_1), oblikoslovna oznaka izvora (f_2, f_3), cilja (f_3) in besed poleg izvora ali cilja (f_3).

3.1.2. Razčlenjevanje

Razčlenjevanje nove povedi poteka v dveh fazah, kjer privzamemo, da imamo na voljo naučen model MSTP, tj. vektor uteži $w = (w_1, \dots, w_N)$. Vhodni povedi $x = (t_1, \dots, t_k)$ priredimo poln usmerjen graf $G_x = (\mathrm{V} = \{0,1, \dots, k\}, \mathrm{A})$, kjer V vsebuje $k+1$ vozlišč (vključno s tehničnim vozliščem) , in A vsebuje naslednje povezave: $A = \{(i,j) | i > 0, j > 0, i \neq j\} \cup \{(0,j) | j > 0\}$. Vsaki usmerjeni povezavi $(i,j) \in A$ priredimo vektor lastnosti: $F_{i,j} \rightarrow (f_1(x, (i,j)), \dots, f_N(x, (i,j)))$ in nato kvaliteto povezave: $S_{i,j} = \sum_{k=1}^{N} F_{i,j}(k) \cdot w_k$. Graf G_x skupaj s koeficienti $S = \{S_{i,j} | (i,j) \in A\}$ predstavlja utežen usmerjen graf. Zaradi linearnosti funkcije $score(x, y)$ je optimalno odvisnostno drevo natanko maksimalno vpeto drevo uteženega grafa (G_x, S). Za iskanje maksimalnega vpetega drevesa obstaja učinkovita rešitev s časovno zahtevnostjo $O(k^2)$ (Chu in Liu, 1965; Tarjan, 1977), kjer je k število pojavnic v povedi x.

3.1.3. Učenje

Naj bo $U = \{(x_1, y_1), \dots, (x_m, y_m)\}$ množica parov povedi in odvisnostnih dreves. Učenje odvisnostnega razčlenjevalnika predstavlja reševanje naslednjega optimizacijskega problema:

$$\min_{w \in R^N} \|w\|$$
$$p.p.$$
$$score(x_i, y_i) - score(x_i, y') \geq L(y_i, y'),\ \ \forall y' \in dt(x_i),$$
$$\forall i = 1{:}m,$$

kjer $dt(x)$ predstavlja množico vseh dopustnih neoznačenih odvisnostnih dreves in $L(y, y')$ prešteje število napak drevesa y' glede na dano pravilno drevo y. Minimizacija norme vektorja uteži sili učenje k preprostejšim modelom, s čimer preprečuje pretirano prileganje modela k učni množici. Optimizacijski pogoji omejujejo prostor iskanja modelov na modele, ki konsistentno ocenjujejo kvaliteto odvisnostnih dreves na učni množici. Avtorji (McDonald, Lerman in Pereira, 2006) za iskanje rešitve uporabijo iterativno metodo k-best MIRA (Margin Infused Relaxed Algorithm).

[2] http://www.slovenscina.eu/Media/Kazalniki/Kazalnik2/SSJ_Kazalnik_2_Specifikacije-ucni-korpus_v1.pdf

3.1.4. Označena odvisnostna drevesa

Iskanje označenih odvisnostnih dreves poteka v dveh fazah: v prvi fazi se povedi priredi neoznačeno odvisnostno drevo, za tem pa se za to drevo uporabi markovski model prvega reda za iskanje najverjetnejšega zaporedja oznak povezav. Razčlenjevanje celotnega učnega korpusa (11.411 oblikoslovno označenih stavkov) na procesorju Intel Core i7 3.07GHz CPU traja 31 minut.

3.2. Učni korpus ssj500k

Učni korpus ssj500k je bil izdelan v okviru projekta Sporazumevanje v slovenskem jeziku[3] (SSJ) in temelji na obeh učnih korpusih, izdelanih v okviru projekta JOS. Sestavljen je iz celotnega korpusa jos100k ter dodatnih 400.000 besed iz enomilijonskega korpusa jos1M. Vsi jezikoslovni metapodatki (oznake, leme, tokenizacija) so bili v korpusu ssj500k še enkrat ročno pregledani, povečana je bila množica skladenjsko označenih in ročno pregledanih povedi. V delu, ki ga zajema korpus jos100k, so bile dodane informacije o lastnih imenih za potrebe strojnih prepoznavalnikov imenskih entitet. Za razliko od korpusov jos100k in jos1M je bila v korpusu ssj500k v celoti ročno pregledana in popravljena tudi stavčna segmentacija in tokenizacija, kar omogoča tudi preverjanje uspešnosti označevalnikov in razčlenjevalnikov pri teh dveh postopkih. Številčni podatki o elementih v korpusu ssj500k so v Tabeli 1.

oznaka	opis	ssj500k
	besedilo	1.677
<p>	odstavek	8.137
<s>	stavek oz. poved	27.829
<w>	beseda	500.295
<c>	ločilo/simbol	85.953
<w> + <c>	pojavnica	586.248
<links>	element s skladenjskimi povezavami	11.411
<link>	skladenjska povezava	235.865
<chunks>	element s povezavami na imenske entitete	2.178
<chunk>	imenska entiteta	4.398

Tabela 1: Število elementov v učnem korpusu ssj500k

Učni korpus ssj500k je prosto dostopen na spletnih straneh projekta SSJ[4] pod licenco Creative Commons Priznanje avtorstva-Nekomercialno 3.0.[5]

4. Skladenjski korpus in natančnost razčlenjevanja

Skladenjsko označeni del ssj500k je izdelan po sistemu odvisnostne drevesnice JOS (Ledinek, 2010; Ledinek in Erjavec, 2009; Erjavec et al., 2010).[6] Učna množica za skladenjski razčlenjevalnik obsega 11.411 ročno preverjenih povedi ali 200.320 besed, kar predstavlja približno dve petini učnega korpusa ssj500k.

Skladenjski del korpusa ssj500k je nastajal v treh fazah. V prvi je bil v projektu JOS vzporedno s pripravo specifikacij za skladenjsko označevanje oblikovan testni nabor 500 povedi. Ta je služil kot učna množica za učenje prvega statističnega modela, s katerim je bil označen korpus jos100k. Avtomatsko pripisane skladenjske povezave so nato v programu Označevalnik stavkov[7] ročno pregledali in popravili označevalci, po dva za vsako poved oziroma trije, kadar je med prvima dvema označevalcema prihajalo do razlik. V tretji fazi je bila v okviru projekta Sporazumevanje v slovenskem jeziku učna množica povečana in celoten krog polavtomatskega razčlenjevanja ponovljen za novih 100.000 besed, vključno z analizo in popravki celotnega skladenjskega dela korpusa ssj500k.

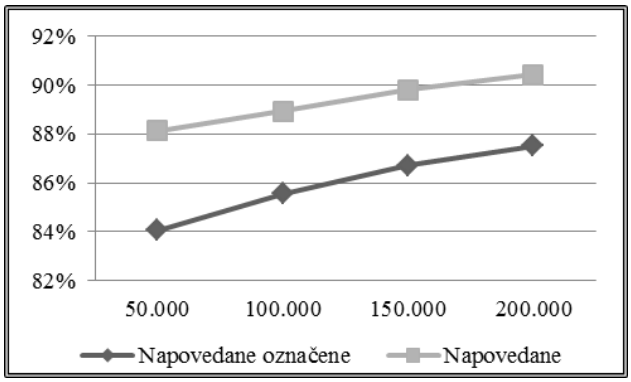
Tabela 1 prikazuje pogostost posameznih tipov skladenjskih razmerij in njihovo povprečno dolžino, tj. absolutno razdaljo med položajem izvorne in ciljne pojavnice. Navidezno je najdaljša korenska (*modra*) povezava, saj korenski element zavzema mesto ničte pojavnice.

št.	Oznaka	Frekvenca	povprečna dolžina
1	modra	66.155	16,2
2	del	16.134	2,4
3	dol	79.627	1,8
4	ena	11.690	3,5
5	dve	15.639	2,9
6	tri	5.779	2,7
7	štiri	14.246	4,0
8	prir	6.482	5,0
9	vez	19.288	2,6
10	skup	825	1,2

Tabela 2: Tipi skladenjskih povezav glede na pogostost v učnem korpusu ssj500k in njihova povprečna dolžina

Z večanjem učne množice je natančnost označevanja postopoma naraščala (slika 1), pri čemer najvišja natančnost skladenjskega razčlenjevanja z učenjem na učnem korpusu ssj500k (~200.000 besed), merjena z 10-kratnim prečnim preverjanjem, meri 90,43 % za natančnost napovedanih povezav (tj. pravilno določeno mesto povezave) oziroma 87,52 % za natančnost napovedanih označenih povezav (tj. pravilno določena mesto in tip povezave).

[3] http://www.slovenscina.eu/tehnologije/ucni-korpus
[4] http://www.slovenscina.eu/tehnologije/ucni-korpus, http://razclenjevalnik.slovenscina.eu/
[5] http://creativecommons.org/licenses/by-nc/3.0/deed.sl
[6] Sistem uporablja deset oznak, za razliko od sistemov SDT (Erjavec in Ledinek, 2006) in PDT (http://ufal.mff.cuni.cz/pdt2.0/), ki uporabljata 28 oznak v precej kompleksnejšem sistemu.
[7] Program je dostopen na strani projekta SSJ: http://www.slovenscina.eu/Vsebine/Sl/Kazalniki/K10.aspx



Slika 1: Natančnost glede na velikost učne množice

V okviru analize je bila izvedena tudi primerjava glede na stopnjo uporabe oblikoskladenjskih informacij pri učenju razčlenjevalnika. Natančnost razčlenjevanja z upoštevanjem celotnih oblikoskladenjskih oznak, upoštevanjem zgolj vrhnjih kategorij (podatka o besedni vrsti) oz. neupoštevanjem oblikoskladenjskih oznak je predstavljena v Tabeli 3.

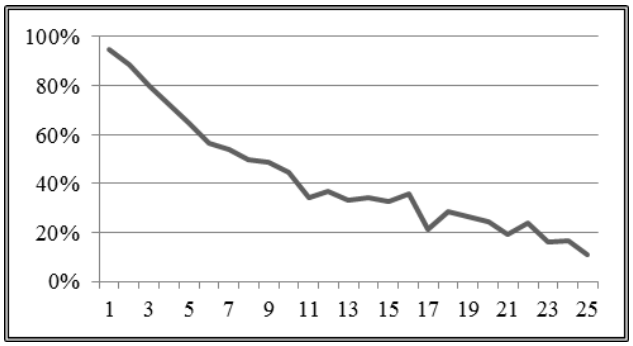
	Celotne oznake	Besedne vrste	Brez oznak
Napovedane povezave	90,43 %	88,86%	73,71 %
Napovedane označene povezave	87,52 %	84,41%	65,34 %

Tabela 3: Natančnost skladenjskega razčlenjevalnika z upoštevanjem oblikoskladenjskih oznak.

Kot izhodišče za nadaljnjo podrobnejšo analizo rezultatov skladenjskega razčlenjevanja je bila upoštevana prva množica, torej rezultat označevanja korpusa z ročno pregledanimi celotnimi oblikoskladenjskimi oznakami, v njej pa smo se osredotočili na opazovanje natančnosti razčlenjevanja glede na zgradbo povedi in posebnosti posamezne skupine skladenjskih razmerij.

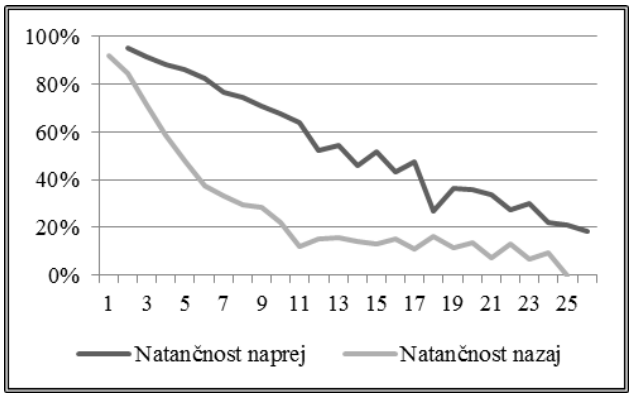
Natančnost označevanja z dolžino povedi pričakovano upada, od povprečno 95,7-odstotne natančnosti za povedi z 1–5 pojavnicami do 84,4-odstotne natančnosti za povedi z več kot 50 pojavnicami. Pri podrobnejšem pregledu povedi z manj kot 30 pojavnicami vidimo, da natančnost strmo pada do dolžine 10, nato je krivulja položnejša. Povprečna poved skladenjskega korpusa vsebuje 18 besed oz. 21 pojavnic (vključno z ločili).

V teoriji bi moral biti model, ki temelji na iskanju globalno optimalnih rešitev na grafu, kakršen je model MSTP, enako uspešen ne glede na dolžino povezave med izvorno in ciljno pojavnico, toda realno natančnost tudi z dolžino povezav upada (slika 2), saj so daljše povezave običajno tudi bolj dvoumne (McDonald in Nivre, 2007), denimo pri nizih prislovnih določil, ki so glede na pomen lahko povezani v eno besedno zvezo ali pa cilj več različnih (razmeroma dolgih) povezav.



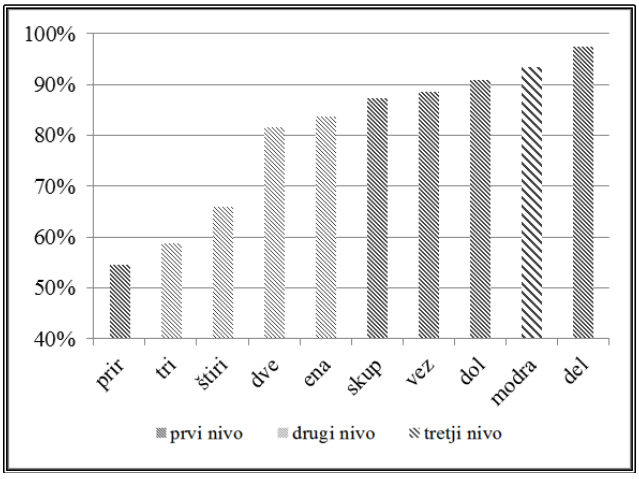
Slika 2: Natančnost glede na razdaljo med izvorno in ciljno pojavnico

Precejšna razlika se pojavlja v natančnosti povezovanja glede na smer povezave, torej glede na to, ali ciljna pojavnica stoji za izvorno ali pred njo (slika 3), natančnost označevanja pri povezavah naprej je namreč v povprečju kar za 16,2 odstotne točke višja kot pri povezavah nazaj.



Slika 3: Natančnost glede na smer in dolžino povezave med izvorno in ciljno pojavnico

5. Analiza po oznakah



Slika 4: Natančnost označevanja glede na tip povezave.

Na sliki 4 prikazujemo natančnost napovedanih označenih povezav glede na tip povezave. Natančnost je z izjemo povezave prir najvišja pri povezavah prvega nivoja

(*del*, *dol*, *vez*, *skup*) in pri povezavi tretjega nivoja (*modra*), nižja pa je pri povezavah drugega nivoja (*ena*, *dve*, *tri*, *štiri*) in povezavi *prir*.

5.1. Povezave prvega nivoja

Kot je razvidno iz slike 4, je za to skupino (besednozveznih) povezav natančnost razčlenjevanja najvišja, še posebej pri povezovanju delov zloženega povedka (povezava *del*, 97,5 %). Nekoliko nižja je natančnost pri povezovanju jedra in določila drugih besednih zvez (povezava *dol*; 90,8 %), ki pa se zelo razlikuje glede na posamezen tip besedne zveze: razčlenjevalnik je izredno natančen pri povezovanju predložnih zvez (98,3 %; X-*dol*-D[8], tj. povezava *dol* s katerekoli pojavnice na predlog) ter pri povezovanju (pridevniškega, zaimkovnega ali številskega) prilastka s samostalniškim jedrom (97,1 %; S-*dol*-[PZK]), nekoliko manj pa pri povezovanju drugih tipov nestavčnih modifikacij. Povprečne rezultate dosega pri označevanju samostalniških zvez s samostalniškim prilastkom (88,5 %; S-*dol*-S), pridevniških povedkovih določil (89,7 %; G-*dol*-P), prislovnih zvez s prislovnim prilastkom (87,9 %; R-*dol*-R) ter pri povezovanju modalnih glagolov in nedoločnika oz. namenilnika (89,2 %; G-*dol*-G), medtem ko je najmanj natančen pri povezovanju prilastkovih odvisnikov (55,7 %; [^G]-*dol*-G). Podobno nizko stopnjo natančnosti dosega pri povezovanju nepridevniških povedkovih določil (52,4 %; G-*dol*-[^GP]), toda za razliko od vseh ostalih povezav prvega nivoja je pri slednjih vendarle razmeroma uspešen z vidika pravilno napovedanih (a ne tudi označenih) povezav (natančnost naraste na 92,0 %; v 94 % napak razčlenjevalnik namreč namesto povedkovega določila oz. povezave *dol* pripiše osebek oz. povezavo *ena*). Zanimivo je, da so povezave prvega nivoja tudi tiste, kjer so rezultati označevanja besedila brez oblikoskladenjskih oznak najboljši, denimo 81,7 % pri povezavi *del*, 75,5 % pri povezavi *skup*, 68,9 % pri povezavi *dol* in 65,4 % pri povezavi *vez* oziroma skupaj v povprečju 21 odstotnih točk slabši od razčlenjevanja oblikoskladenjsko označenega besedila.

Ne glede na stopnjo označenosti korpusa je povezava *prir* tista, ki med vsemi tipi povezav dosega daleč najslabše rezultate (54,6% natančnost). Razčlenjevalnik uspešno prepozna priredni veznik ter jedro drugega dela priredja (81,0 % oz. 89,4 % za vezniške besede brez upoštevanja ločil; [^G]-*vez*-G), ne prepozna pa povezave med jedrom prvega in drugega dela priredja. Označevanje vezniških besed je še bolj natančno na stavčni ravni (93,6 %; G-*vez*-X), še posebej pri povezavah na veznike (95,4 %; G-*vez*-V).

5.2. Povezave drugega nivoja

Povezave drugega nivoja skupno dosegajo razmeroma nizko, 74,6-odstotno, natančnost označenih napovedanih povezav, toda v slabi polovici primerov so napake posledica napačno pripisane oznake in ne napačno napovedane povezave. Natančnost napovedanih povezav tako meri 85,6 %. Ker povezave drugega nivoja vsebujejo podatke o glagolski vezljivosti, omenjena podatka pri analizi med spadata med pomembnejše.

Med povezavami drugega nivoja razčlenjevalnik najbolj natančno označuje povezavo *ena* (83,7 %), ki z vidika tradicionalne slovnice ustreza povezavi med povedkom in osebkom. Razčlenjevalnik je najbolj uspešen pri določanju samostalniškega (87,5 %; G-*ena*-S) oziroma zaimkovnega osebka (82,1 %; G-*ena*-Z), manj pa pri prepoznavanju osebkov drugih besednih vrst in pa pri povezovanju osebkovnega odvisnika (60,0 %; G-*ena*-G). V povprečju se v približno dveh tretjinah napak moti v samem mestu povezave, v preostali tretjini pa sicer pravilno povezanemu paru pojavnic pripiše napačen tip povezave (namesto povezave *ena* največkrat zmotno pripiše povezavo za povedkovo določilo [*dol*] ali predmet [*dve*]).

Razčlenjevalnik je podobno uspešen tudi pri označevanju povezave *dve* med povedkom in predmetom stavka (81,5 %), bolj pri povezavah na samostalnik (84,1 %; G-*dve*-S) in zaimek (90,8 %; G-*dve*-Z), manj pa pri povezovanju predmetov drugih besednih vrst in predmetnega odvisnika (65,4 %; G-*dve*-G) ter pri predmetnih povezavah s pridevniškega povedkovega določila (51,6 %; P-*dve*-X). Za razliko od povezave *ena* je pri povezavi *dve* razmerje med napakami izvora in napakami tipa precej enakovredno, pri čemer razčlenjevalnik pri predmetnih povezavah z glagola na samostalnik, zaimek, številnik in prislov pri več kot polovici napak določi pravilno mesto povezave, a napačen tip. V dveh tretjinah takih primerov izbere oznako *štiri*, kar je glede na pomanjkanje leksikonskih informacij o vezljivosti glagolov pričakovano, zlasti pri predložnih besednih zvezah.

Poleg povezave *prir* sta med povezavami z najnižjo stopnjo natančnosti povezavi *tri* in *štiri* (58,6 % oz. 66,0 %), ki ustrezata povezavi s povedka na prislovno določilo. Označevanje prislovnih povezav je razmeroma uspešno le pri povezavah z glagola na prislov (81,8 % za povezavo *tri* in 79,5 % za *štiri*), pri vseh ostalih besednih vrstah in pri prislovnih povezavah s povedkovega določila pa je natančnost nižja. Daleč najnižja je natančnost pri prislovnih odvisnikih (0,6 % oz. 16 %; G-*tri*/*štiri*-G), kar je bistveno slabši rezultat kot pri drugih vrstah odvisnikov.

Ker imata povezavi *tri* in *štiri* razmeroma visok delež napak tipa in je ločevanje med njima z vidika avtomatskega razdvoumljanja precej zahtevno, smo ju v eni od analiz združili v en sam tip povezave. Število napak zaradi napačno pripisane oznake pravilno povezanima pojavnicama se je sicer zmanjšalo, natančnost napovedanih označenih povezav pa je ostala nespremenjena. Za 6,3 odstotnih točk se je povečala zgolj natančnost označenih povezav *tri* in *štiri* (s 63,8 % na 70,2 %), označevanje ostalih dveh povezav drugega nivoja (*ena* in *dve*) pa se je celo poslabšalo (za 0,9 oz. 0,5 odstotne točke). Tako se z združitvijo obeh prislovnih povezav natančnost napovedanih povezav na celotni množici ni bistveno izboljšala (s 87,52 % na 87,98 %), natančnost napovedanih označenih povezav pa se je zaradi drugih sprememb celo rahlo poslabšala (z 90,43 % na 90,41 %),

5.3. Povezave tretjega nivoja

Skupna natančnost označevanja povezave *modra* je razmeroma visoka (93,4 %), pri čemer je treba upoštevati, da ta vrednost združuje tako natančnost korenskih povezav za besedne pojavnice (87,8 %) kot povezav na

[8] Pri poimenovanju besednih vrst uporabljano vrhnje kategorije po sistemu JOS (http://nl.ijs.si/jos/msd/html-sl/index.html).

ločila (99,0 %). Pri povezavah na besede je natančnost določanja pravilnega mesta povezave nadpovprečno visoka pri členku, okrajšavi, glagolu, vezniku in številniku, precej pod povprečjem pa pri prislovu, samostalniku, predlogu, pridevniku in zaimku. Razumljivo se pri tej povezavi pojavljajo samo napake izvora, ne pa tudi tipa (vedno *modra*). Spodbudna je natančnost pri prepoznavanju hierarhično najvišje vloge povedka v glavnih stavkih in priredjih (93,4 %; metaelement-*modra*-G).

6. Zaključek

Kot vsi statistični modeli tudi statistični razčlenjevalnik daje dobre rezultate pri označevanju tipičnega v jeziku, slabše pa pri označevanju manj pogostih struktur. Natančnost pada tudi z dolžino povedi in povezav. Podrobnejša analiza rezultatov za posamezne skupine povezav kaže, da je razčlenjevalnik nadpovprečno uspešen pri povezovanju besednih zvez (še posebej glagolskih, samostalniških in predložnih), pri določanju vezniških besed v stavkih in pri povezovanju pojavnic na korenski element. Povprečne rezultate glede na skupno natančnost dosega pri povezovanju povedkovih določil, modalnih glagolskih zvez ter pri prepoznavanju stavčnega osebka in predmeta, podpovprečen pa je pri povezovanju besednozveznih priredij in prislovnih določil. Podpovprečna je tudi natančnost označevanja prilastkovih, osebkovih in predmetnih odvisnikov, še posebej nizka pa je natančnost označevanja prislovnih odvisnikov.

Vsekakor obstaja še veliko možnosti za izboljšavo rezultatov. Na podlagi podatkov na sliki 1 nadaljnje širjenje učne množice v tej fazi razvoja razčlenjevalnika ni smotrno, saj natančnost narašča razmeroma počasi v primerjavi s potrebno količino vloženega ročnega dela. Pred vsebinskimi izboljšavami razčlenjevalnika je smiselno temeljito preveriti morebitne napake in neskladja pri ročnem označevanju, bodisi s še enim krogom usmerjenega ročnega pregledovanja bodisi s pomočjo specializiranih programov za avtomatsko odkrivanje napak pri odvisnostnem razčlenjevanju (Boyd, Dickinson in Meurers, 2008; Hall in Novák, 2011).

Z vidika izboljšav v delovanju razčlenjevalnika je perspektivno predvsem aktivno učenje (*active learning*) posameznih tipov povedi, besednih zvez ali povezav (Mirroshandel in Nasr, 2011), poleg tega je smiselno razmišljati v smeri hibridnega razčlenjevalnika, ki bi poleg statističnih modelov vseboval tudi ročno napisana skladenjska pravila oz. kombinacijo dveh samostojnih razčlenjevalnikov. Uspešnost takih hibridnih modelov se v tujih raziskavah že potrjuje (Foth in Menzel, 2006; Sennrich et al., 2009). V kontekst hevrističnega usmerjanja statističnih modelov se umeščajo tudi podatki o vezljivosti, denimo iz leksikalne baze za slovenščino (Gantar in Krek, 2011), s katerim bi lahko bistveno zmanjšali delež napak tipa predvsem pri povezavah drugega nivoja, kjer ima razčlenjevalnik težave pri razdvoumljanju predmetnih oz. prislovnodoločilnih povezav.

7. Literatura

Boyd, A., Dickinson, M., Meurers, D. (2008). On Detecting Errors in Dependency Treebanks. Research on Language and Computation, 6(2), 113-137.

Chu, Y., Liu, T. (1965). On the shortest arborescence of a directed graph. *Science Sinica*, 14, 1396-1400.

Erjavec, T., Ledinek, N. (2006). Slovenska odvisnostna drevesnica: prvi rezultati. V *Zbornik 5. slovenske in 1. mednarodne konference Jezikovne tehnologije IS-LTC* (str. 162-167). Ljubljana: Institut Jožef Stefan.

Erjavec, T., Fišer, D., Krek, S., Ledinek, N. (2010). The JOS Linguistically Tagged Corpus of Slovene. V *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC)* (str. 1806-1809). Pariz: ELRA.

Foth, K.A., Menzel, W. (2006). Hybrid parsing: using probabilistic models as predictors for a symbolic parser. V *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL* (str. 321-328). Sydney: ACL.

Gantar, P., Krek, S. (2011) Slovene lexical database. V *Natural language processing, multilinguality: sixth international conference* (str. 72-80). Modra, Slovaška: Slovenská akadémia vied.

Hall, K.B., Novak, V. (2011). Corrective Dependency Parsing. *Text, Speech and Language Technology: Trends in Parsing Technologies*, 1(43), 151-167.

Ledinek, N. (2010). Slovenska skladnja v skladenjsko označenih korpusih slovenščine. Doktorska disertacija.

Ledinek, N., Erjavec, T. (2009). Odvisnostno površinskoskladenjsko označevanje slovenščine: specifikacije in označeni korpusi. V *Zbornik Simpozija Obdobja: Infrastruktura slovenščine in slovenistike* (str. 2019-224). Ljubljana: Znanstvena založba Filozofske fakultete.

McDonald, R., Crammer, K., Pereira, F. (2005). Spanning Tree Methods for Discriminative Training of Dependency Parsers, UPenn CIS Technical Report: MS-CIS-05-11.

McDonald, R., Nivre, J. (2007). Characterizing the Errors of Data-Driven Dependency Parsing Models. V *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (str. 168-170). Praga: ACL.

McDonald, R., Lerman, K., Pereira, F. (2006). Multilingual Dependency Parsing with a Two-Stage Discriminative Parser. V *Tenth Conference on Computational Natural Language Learning (CoNLL-X)*.

Mirroshandel, S. A., Nasr, A. (2011). Active Learning for Dependency Parsing Using Partially Annotated Sentences. V *Proceedings of International Conference on Parsing Technologies (IWPT)*.

Sennrich, R., Schneider, G., Volk, M., Warin, M. (2009). A New Hybrid Dependency Parser for German. V *Proceedings of the Biennial GSCL Conference*.

Tarjan, R.E. (1977) Finding Optimum Branchings. *Networks*, 7, 25-35.