Gradnja in označevanje korpusov

Tomaž Erjavec
Korpusno jezikoslovje
UNG
2008/2009

28. 11. 2008

Pregled predavanja

1. kako do korpusov
2. jezikoslovno označevanje

I. Kako do korpusov?

- uporaba že narejenih korpusov
- z zbiranjem besedil
- zajem s spleta

Obstoječi korpusi

- že narejeni korpusi:
  ELRA ($$), LDC ($), corpora@lists.uib.no
  vendar ne kaj dosti za slovenščino..
- korpusi slovenskega jezika:
  na teh predavanjih,
  dopisni seznam sdjt-l@ijs.si, vendar se je potrebno predhodno včlaniti v SDJT
- če ustreznega korpusa ni, ga moramo narediti sami…

Zbiranje besedil

- tematsko usmerjeno zbiranje besedil:
  - kakšna besedila nas zanimajo
  - kako jih najdemo
- datoteke Word od prijaznih avtorjev ali posrednikov:
  - osebni stiki
- iskanje in shranjevanjem besedil s spleta:
  - portali
  - Najdi.si, Google

Problemi

- zagotovitev zadosti velikega (reprezentativenega) korpusa za področje, ki nas zanima
- prepričati nosilce avtorskih pravic, da nam odstopijo besedila
- kako je to težko, je zelo odvisno od tega, kakšna je predvidena diseminacija korpusa:
  - uporaba samo za lastne raziskave /ustni dogovor/

Osnovna obdelava besedil

- dobljena besedila tipično shranimo kot navadno besedilo (zgubimo formatiranje, z izjemo odstavkov)
- pazimo, da vsa besedila shranimo v enakem kodnem naboru
- besedila po možnosti opremimo z metapodatki:
  naslov, avtor, datum, taksonomija, …
- uporaba:
  - konkordančnik, ki podpira uvoz navadnega besedila, npr. WordSmith

Jezikoslovna obdelava besedil

- Koraki pri osnovni jezikoslovni obdelavi korpusa:
  - tokenizacija:
    besedilo razdelimo na pojavnice
  - segmentacija:
    besedilo razdelimo na povedi
  - lematizacija:
    vsaki besedi pripišemo njeno osnovno obliko
  - oblikoslovno označevanje:
    vsaki besedi pripišemo njeno oblikoslovno oznako
- za slovenski jezik (še) ne obstajajo prosto dostopni programi, ki bi opravili gornje naloge

Spletni servis http://nl2.ijs.si/analyze/

- podpira lematizacijo oz. oblikoskladenjsko označevanje manjših besedil
- na izbiro več
  - programov:
    - RDR - samo lematizacija
    - CLOG - oblikoskladenjsko označevanje + lematizacija
  - jezikov
    - RDR - samo slovenščina
    - CLOG - tudi en, cs, ro, hu, et
  - načinov izpisa
    - samo leme
    - besedne oblike + leme
    - besedne oblike + leme v XML

Primer označenega besedila v XML

```
<div>
 <p>
  <s>
   <w lemma="kadrovski" ana="Agufpa">kadrovske</w>
   <w lemma="štipendija" ana="Ncfpa">štipendije</w>
   <c>,</c>
   <w lemma="ki" ana="Cs">ki</w>
   <w lemma="on" ana="Pp3mpa--c">jih</w>
   <w lemma="praviloma" ana="Run">praviloma</w>
   <w lemma="podeljevati" ana="Vmpr3p">podeljujejo</w>
   <w lemma="podjetje" ana="Ncnpn">podjetja</w>
   <c type="TERM">.</c>
  </s>
 …
```

Kaj z jezikoslovno obdelanimi besedili?

- lahko se jih naloži v SketchEngine (zahtevan točno določen format)
- od “namiznih” konkordančnikov žal nobeden ne podpira označenih besedil
- možnost uporabe skozi lastno programje:
  - Perl, Python
- lahko prosite mene, da jih naložim v konkordančnik CQP
- kako to izgleda:
  http://nl2.ijs.si/dsi.html
  http://nl2.ijs.si/index-bi.html
  http://nl.ijs.si/jos/cqp/
  http://nl.ijs.si/jaslo/cqp/

“Web as Corpus”

- na medmrežju je več in več besedil
- zakaj torej ne uporabiti kar teh besedil za izdelavo korpusov?
- potencialni problemi:
  - vseh zvrsti besedil ni na medmrežnju, ali pa jih je (pre)malo
  - besedila so lahko slabo napisana
- prednosti:
  - besedila so neposredno dostopna na medmrežju
  - s pomočjo avtomatskih metod je možno hitro zbrati korpuse, ki so za več velikostnih razredov večji, kot pa tisti, ki bi jih lahko zbrali ročno

Avtomatski zajem s spleta

- BootCat - del servisov SketchEngine (z njim narejeni SKE korpusi *WaC)
- programi dostopni tudi za lastno rabo
- Postopek:
  1. uporabnik vnese seznam ključnih besed
  2. BootCat uporabi Google, da najde spletne strani, ki te besede vsebujejo
  3. strani pobere, jih očisti, poenoti zapis in (za nekatere jezike) jezikoslovno označi
  4. (ključne besede iz dobljenega korpus doda k ključnim besedam iz 1. in ponovi 1.-4.)
  5. korpus doda h korpusom dostopnim preko SKE

BootCat - izbira ključnih besed

- ustreznost dobljenega korpusa je zelo odvisna od vnešenih besed
- morajo biti tipične za področje, ki nas zanima
- ne smejo biti enake kot besede v drugih jezikih

II. Jezikoslovno označevanje

1. tehnike označevanje
2. tokenizacija in segmentacija
3. oblikoslovno označevanje
4. lematizacija

Označevanje

- besedilo analiziramo na določeni jezikovni ravni
- rezultat analize zapišemo v korpus, t.j. korpus označimo
- tak korpus je nato primeren za nadaljno, bolj poglobljeno obravnavo
- ljudje lahko iščejo (tudi) po pripisanih oznakah oz. te oznake pogledajo
  - radikalni korpusni pristop to zavrača (Sinclair): analize, opravljene nad označenimi korpusi so pod vplivov nekih vnaprejšnjih teorij
- računalniki lahko oznake uporabijo za nadaljnje procesiranje

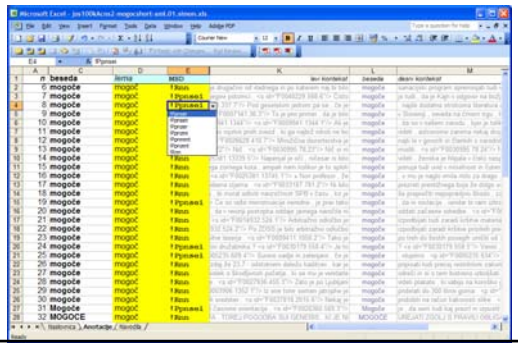
Označevalne tehnike

- ročno označevanje
- označevanje z ročno napisanimi pravili
- označevanje z avtomatsko naučenimi pravili (modeli)

Ročno označevanje

- s pomočjo urejevalnika ekspert (jezikoslovec) označuje korpus
- potrebna je natančna definicija “gramatike”, t.j. nabora dovoljenih kategorij oz. relacij
- problem posebej akuten, ko je označevalcev več: izdelava priročnika, vzporedno označevanje
- za nekatera področja (semantično označevanje) je ujemanje med različnimi označevalci < 70%
- dobrodošlo je preverjanje: formalno, vsebinsko
- drago in zamudno, vendar potreben prvi korak, da dobimo testni oz. učni korpus

Primer: oblikoslovne oznake za “mogoče” v korpusu JOS100k



Strojno označevanje

1. z ročno napisanimi pravili
2. s strojnim učenjem, na osnovi ročno označene učne množice
3. s strojnim učenjem, na osnovi neoznačene učne množice (statistične metode)

(pogosta kombinacija strojnega in ročnega označevanja, v več korakih)

Ročno napisana pravila

- programi, ki delajo s pomočjo ročno napisanih pravil (jezikoslovec/računalničar)
- programi zahtevajo zelo formalna pravila, omejena glede na izbrano teorijo/formalizem/implementacijo
- klasični način (1970-1990) pisanja programov za jezikoslovno analizo
- problemi:
  - **pokritje**:
    tipično imajo ti programi majhen besedni zaklad in poznajo omejeno število možnih konstrukcij
  - **krhkost**:
    programi popolnoma odpovejo pri “nenavadnih” besedilih
  - **dvoumnost**:
    programi se ne morejo odločit kadar je več (teoretično) možnih analiz: “Time flies like an arrow”

Strojno učenje

- vodeno strojno učenje (supervised learning)
- program se uči na osnovi ročno označenih podatkov (korpusa)
- prednosti glede na ročno pisana pravila:
  - večje pokritje besedišča in možnih vzorcev (ob primerno velikem učnem korpusi)
  - robustnost: programi "se znajdejo" tudi pri nenavadnih besedilih
  - pri dvoumnosti se program odloči za najbolj verjetno možnost
- slabosti:
  - potreba po velikih (dragih!) ročno označenih korpusih
  - naučena pravila so tipično težko razumljiva, in jih je težko ročno popravljati
  - če je vhodno besedilo zelo različno od učnega korpusa, so rezultati slabi

Ravni označevanja

- označujemo lahko praktično karkoli kar je koristno za neko aplikacijo
- delitev po ravneh jezikoslovne obravnave:
  - oblikoslovje
  - leksika
  - skladnja
  - semantika
- primeri:
  1. predobdelava: tokenizacija in segmentacija
  2. oblikoslovno označevanje
  3. lematizacija

Tokenizacija

- razdelitev besedila na pojavnice (besede in ločila)
- prvi korak jezikoslovne analize
- vsi konkordančniki potrebujejo korpus razdeljen na pojavnice (indeksiranje)
- v osnovi enostavno:
  - besedilo: "Biba leze, biba gre."
  - razdelimo besedilo po presledkih: "|Biba|leze,|biba|gre.|"
  - nato odščipnemo ločila: "|Biba|leze|,|biba|gre|.|"

Problemi s tokenizacijo

- ločila so lahko del pojavnice:
  - vrstilni števniki: “4.”
  - krajšave: “npr.”
  - te probleme se rešuje s jezikovno odvisnimi pravili in seznami
- napake v besedilu:
  - “.. biba gre.Biba mala..”
    vendar “http://nl.ijs.si/”
- deljaji:
  - “obiskovalec, če iščete besede, ki poime-
    nujejo pojave informatike …”
  - vendar: “obiskovalec, če iščete rumeno-
    zelen otroški voziček …”

Segmentacija

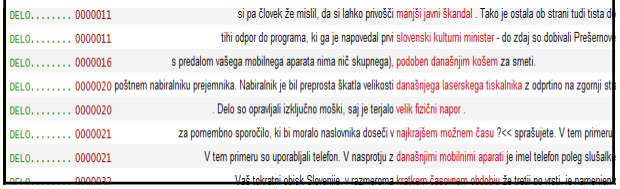
- razdelitev besedila na povedi
- takoj za tokenizacijo najbolj osnovna stopnja obdelave
- v večini primerov zopet enostavno:
  konec povedi je med pojavnicama, ki ju opisujeta naslednja regularna izraza:
  [.!?] [A-Z].*
- vendar zopet problemi:
  - “g. Žnidaršič”, “Mesta, kot npr. Velenje.”
  - “»Takoj nehaj!« mu je ukazal.”

Oblikoslovno označevanje

- vsaki besedi v besedilu pripišemo njene oblikoslovne lastnosti, npr. samostalnik moškega spola ednine, v orodniku
- vse oblikoslovne lastnosti besedne oblike so po navadi združene v oznako, npr.
  `Somei →`
  `samostalnik vrsta=občno_ime spol=moški število=ednina sklon=imenovalnik`
- za slovenski jezik ločimo skoraj 2000 različnih oznak
- oblikoslovne oznake so koristne, ker npr. omogočajo iskanje po oblikoslovnih lastnostih, npr. besedni vrsti

FidaPLUS

- pri FidaPLUS iščemo po oblikoslovni oznaki po “kanalu” 2 (#2)
- npr. zaporedje dveh pridevnikov in samostalnika: #2P*_#2P*_#2S*



Primer izpisov oblikoslovnih oznak v iKorpusu

Oblikoslovno označevanje

- posamezna besedna oblika ima lahko več možnih oblikoslovnih oznak, npr. “agenciji” <-- Sozed, Sozem, Sozdi, Sozdt
- katera je prava, je odvisno od uporabe v povedi: “Za ti dve agenciji$_{[Sozdt]}$…V tej agenciji$_{[Sozem]}$ …”
- Dvoumnost je lahko tudi na nivoju besedne vrste: “Hotel$_{[G]}$ je popust pri plačilu računa za hotel$_{[S]}$.”

Avtomatsko oblikoslovno označevanje

- večina označevalnikov se nauči modela iz ročno označenega korpusa
- iz korpusa izlušči leksikon z besednimi oblikami in možnimi oblikoslovnimi oznakami le-teh
- nauči se tudi verjetnosti pojavitve oblikoslovnih oznak glede na lokalni kontekst (trojčki)
- nekateri vsebujejo modul za označevanje neznanih besed
- za slovenski jezik je točnost približno 90% za celotno oznako, 97% za besedno vrsto

Lematizacija

- lema besede je njena osnovna oblika, npr.
  miza, mize, mizi, mizo, mizama,… → *miza*
  hoditi, hodim, hodiš, hodi, … hodil → *hoditi*
  nočem → *?*
  čl. → *?*
- lema nima jezikoslovnega pomena, pač pa je po konvenciji "neoznačena" oblika besede
- kot pri oblikoslovnem označevanju, je lema v splošnem določena šele skozi kontekst:
  hotela → *hotel* ali *hoteti*
  sedel → *sedeti* ali *sesti*

FidaPLUS

- pri FidaPLUS iščemo po lemi po kanalu 1 (#1)
- npr. vse pojavitve leme "človek": #1človek

Čeprav je za nami že druga sezona, mi mnogo ljudi pravi: Ja, nisem še prišel, ampak
značilni stavki generacije, ki so ji večeri namenjeni. Človek je malo utrujen, nekaj dobrega poje, pa še
Baticaloju je v enem samem valu umrlo več kot 3000 ljudi . Na palmah med ruševinami posedajo vrane, ogr
. Naj vas spomnimo le, da naj bodo to ljudje , ki živijo in delujejo na območju Mestne občine
puščajo trajen pečat v našem okolju. Naj bodo to ljudje z različnih področij, tisti, za katere lahko rečete
tudi nekoč v zgodovini, ki bo beležila dogodke in ljudi našega časa.
a sama,srečna in predvsem zelo zadovoljna skupinica ljudi ,ki jih družijo vsakdani v veselju,skrbi,
in da jo je potrebno negovati ter gojiti odnose med ljudmi ,ki so se odločili živeti skupaj. Potem mi
tako davnih časih,ki pa v spominih naših prekmurskih ljudi ne bodo nikoli tonili v pozabo.
,ki ji jih vsi radi izpolnjujejo in s kopico ljudi okoli sebe,ki jo imajo radi in ji dajejo
ih in 60-ih letih preteklega stoletja so se ljudje preseljevali iz podeželja v mesta. Razvoj mest j

FidaPlus

- Pozor: “neznane” besede v korpusu niso lematizirane (ali oblikoslovno označene)
- #1tajkun

- tajkun*

Avtomatska lematizacija

- ročno pisana pravila, ali strojno naučen model
- v splošnem potrebuje program za lematizacijo tudi oblikoslovno oznako
  hotela$_{[Ggnd\text{-}ez]}$ → hoteti
  hotela$_{[Somer]}$ → hotel

Nadaljnje ravni označevanja

- skladenjsko označevanja
- označevanje imen
- označevanje terminov
- pomensko označevanje (hrošč^1, hrošč^2, …)
- večjezični korpusi:
  stavčna poravnava vzporednih korpusov,
  poravnava prevodnih ekvivalentov
- govorjeni korpusi: poravnava transkripcije s signalom
- itd.

Skladenjsko označevanje