Samodejno indeksiranje povzetkov

Boštjan Jerko

Inštitut za biomedicinsko informatiko, Medicinska fakulteta, Univerza v Ljubljani
Vrazov trg 2, 1000 Ljubljana
bostjan.jerko@mf.uni-lj.si

Povzetek

Z uporabo baze Biomedicina Slovenica, ki vključuje vsa besedila avtorjev na področju medicine in sorodnih ved, si bomo pogledali kako lahko poteka samodejno dodajanje ključnih besed posameznim povzetkom. Osnova za izbiro ključnih besed so ključne besede iz MeSH-a. Poskušali bom prikazati možnost za uporabo računalniške aplikacije kot pomoč pri dodeljevanju ključnih besed zapisom v bibliografski bazi.

Automatic abstract indexing

We will use Biomedicina Slovenica database to see how automatic indexing can be done using descriptors from MeSH. Biomedicina Slovenica contains abstracts of authors in medicine and related sciences. I will try to show how to use computer application as a helping tool with tagging data in bibliographic database using descriptors.

1. Uvod

Na inštitutu za biomedicinsko informatiko urejamo bibliografsko bazo "Biomedicina Slovenica" (*http://www.mf.uni-lj.si/ibmi/info-viri/index.html*) vseh slovenskih avtorjev na področju biomedicine. Za vsebinsko obdelavo zapisov uporabljamo angleške ključne besede hierarhično urejenega tezavra "Medical Subject Headings". Vsak zapis vsebuje od enega do pet ključnih besed I, ki opisujejo najpomembnejše vidike članka in nato še ključne besede II (od 2 do 12) za manj pomembne vsebine (Dornik in Adamič, 2004).

Ključne besede člankom, ki so obravnavani v Biomedicini Slovenici, določa strokovnjak na področju biomedicine (zdravnica). Je pa postopek precej časovno zahteven, saj je včasih potrebno pregledati ne le izvleček članka, ampak tudi celoten članek in se iz obsežnega spiska ključnih besed (22.568 zapisov) odločiti, katera ključna beseda bi bila primerna za opis teme članka.

Ogledali si bomo postopek, ki sem ga razvil za "polsamodejno" določanje ključnih besed izvlečkom člankov, ki so zbrani v Biomedicini Slovenici.

Prikazal bom, kako sem preizkušal metodo in predloge za njihovo izboljšanje. Podal bom tudi predlog, ki bi omogočal boljšo uporabo računalnika pri določanju ključnih besed izvlečkov članka in postopke, ki so za to potrebni.

Ogledali si bomo tudi uspešnost metode.

2. Biomedicina Slovenica

Bibliografsko bazo Biomedicino Slovenico vzdržuje Inštitut za Biomedicinsko informatiko že več kot 20 let. V tem času se je nabralo več kot 100.000 zapisov objav (tako doma kot v tujini) vseh slovenskih avtorjev na področju biomedicine. Objave morajo zadoščati določenim kriterijem, da so vnešene v bazo.

V tako obsežni bazi je potrebno uporabiti postopke za označevanje zapisov, tako da je iskanje poenostavljeno. V bibliografskih bazah je običajna uporaba ključnih besed ali t.i. deskriptorjev. Ker je Biomedicina Slovenica bibliografska baza s področja biomedicine, je bilo potrebno uporabiti deskriptorje, ki izhajajo s tega področja. Ameriška National Library of Medicine uporablja za svojo bazo MEDLINE hierarhično urejen tezaver MeSH (Medical Subject Heading), ki je precej obsežen. Vseh zapisov je 22.568. Ta tezaver smo uporabili tudi mi za naše deskriptorje, ki so tako zapisani v angleškem jeziku.

Za vsak zapis v Biomedicini Slovenici pa je potrebno določiti dve ravni deskriptorjev:

- deskriptorje prvega reda, ki podrobno določajo temo članka in jih je od ena do pet,
- deskriptorje drugega reda, ki splošneje določajo temo članka in jih je od dva do dvanajst

Primer oblike zapisa iz Biomedicine Slovenice v XML obliki:

```
<record>
<id>26620</id>
<title>Naslov članka</title>
<translated_title>Naslov članka v angleščini
</translated_title>
<eng_abstract>angleški izvleček</eng_abstract>
<slo_abstract>slovenski izvleček</slo_abstract>
<descriptors1>
<des>ključna beseda ravni 1</des>
<qual>kvalifikator</qual>
</descriptors1>
<descriptors1>
<des>klučna beseda ravni 1</des>
</descriptors1>
<descriptors2>
<des>ključna beseda ravni 2</des>
```

</descriptors2>
</record>

Že iz števila vseh zapisov v MeSH-u je razvidno, da je "ročno" dodeljevanje deskriptorjev včasih mukotrpno, saj mora strokovnjak pregledati izvleček članka, včasih tudi celoten članek in na podlagi lastnih izkušenj določiti, kateri od 22.568 možnih deskriptorjev bi bil primeren za opis vsebine članka.

Tako sem razvil aplikacijo, ki iz podanega izvlečka in naslova članka predlaga deskriptorje, primerne za obravnavani članek.

3. MeSH

Medical Subject Heading (*http://www.nlm.nih.gov/mesh/meshhome.html*) vsebuje 22.568 zapisov in je hierahično urejen. Na prvi ravni ima 15 zapisov, ki so označeni takole:

- Anatomy [A]
- Organisms [B]
- Diseases [C]
- Chemicals and Drugs [D]
- Analytical, Diagnostic and Therapeutic Techniques and Equipment [E]
- Psychiatry and Psychology [F]
- Biological Sciences [G]
- Physical Sciences [H]
- Anthropology, Education, Sociology and Social Phenomena [I]
- Technology and Food and Beverages [J]
- Humanities [K]
- Information Science [L]
- Persons [M]
- Health Care [N]
- Geographic Locations [Z]

Nato pa se število zapisov po ravneh drastično povečuje:

Tabela 1: Število zapisov

Raven	Št. zapisov
1	15
2	170
3	1308
4	6218

V bazi Biomedicina Slovenica so zapisi označeni z deskriptorji v angleškem jeziku. Razlogi za to so predvsem zgodovinski - na začetku gradnje baze slovenskih deskriptorjev še ni bilo.
Na inštitutu za Biomedicinsko informatiko se trudimo s prevajanjem deskriptorjev v slovenščino in njihovo vključevanje v medjezično bazo MeSH-a (S Nelson, 2004).

Ker sem naredil preizkus le na teh štirih nivojih, bom predstavitev MeSH-a tukaj zaključil.

4. Samodejno dodeljevanje deskriptorjev

Aplikacijo je potrebno na začetku pripraviti, tako da naredimo povezavo med možnimi deskriptorji in besedami oziroma besednimi zvezami. Temu postopku pravimo učenje. Za učenje moramo seveda imeti pripravljeno množico zapisov, ki imajo deskriptorje že dodeljene. Tako sem iz 1364 zapisov, ki sem jih imel na razpolago, pripravil učno množico s 1033 zapisi.

Zapisi v učni množice so imeli naslednjo obliko:

- naslov članka,
- izvleček v slovenščini,
- deskriptor prve ravni,
- deskriptor druge ravni.

Za postopek obravnave izvlečkov pa je potrebno zapise najprej pripraviti v postopku predobdelave.

4.1. Predobdelava zapisov

Najprej sem odstranil prazne besede in vse besede naslovov in izvlečkov v učni množici krnil, tako da sem dobil vektor ("spisek") besed, ki sem jih uporabil za normalizacijo posameznih zapisov v učni množici. Ker sem predpostavljal, da so besede iz naslova članka bolj povezane s temo, ki jo obravnava članek, sem le-tem dal večjo težo kot besedam iz abstrakta. V postopku testiranja se je izkazalo, da je tri primerna utež za besede iz naslova.

4.1.1. Prazne besede

Spisek praznih besed sem pripravil ročno, tako da sem na domači strani Nova beseda (*http://bos.zrc-sazu.si/s_beseda.html*) poiskal 10 najpogostejših besed v slovenščini. Nato sem poiskal vse besede, ki se enakomerno pojavljajo preko vseh dokumentov, ker nam le-te ne dajo dodatne informacije za določanje deskriptorjev zapisom. Na koncu pa je bilo potrebno odstraniti tudi znake, kot so: '", -", "!", ".", ",", ";", ":", "(", ")", "

4.1.2. Krnjenje

Glede potrebe po krnjenju besede se še vedno krešejo mnenja. Nekateri so prepričani, da je krnjenje nepotrebno in da ne prinese vidnejšega prispevka k uspehu metode, medtem ko so drugi prepričani, da je krnjenje potrebno. Seveda pa so polemike predvsem za krnilnike za angleški jezik. Za slovenščino takšnih polemik nisem zasledil, morda zato, ker obstajata le dva širši javnosti poznana krnilnika, ali pa zato, ker je v slovenščini več pregibnih besed in je taka polemika odveč. Uradno sta prvi krnilnik za slovenski jezik objavila Mirko Popovič in Peter Willett (Popovic in Willett, 1992).

Moje mnenje je, da je krnjenje za slovenščino potrebno ravno zaradi pregibnosti besed.

V postopku izdelave krnilnika sem izdelal nov krnilnik za slovenščino, ki je narejen v programskem jeziku za izdelavo krnilnikov "Snowball" (Porter, 2004), ki ga je razvil M. Porter, avtor najbolj znanega in verjetno tudi uporabljanega krnilnika za angleščino. Frakes in Baeza-Yates (W.B. Frakes in Ricardo Baeza-Yates, 1992) sta razdelila krnilnike v več skupin:

- različnost končnic,
- krnilnik n-teric,
- krnilnik priponk.

Snowball deluje na preprostem postopku odstranjevanja različnih končnic glede na določene pogoje. Na strani *http://snowball.tartarus.org* so navedeni primeri pravil za krnilnike različnih svetovnih jezikov. Za slovenščino sem napisal pravila za odstranjevanje končnic in tako narejen krnilnik uporabil v postopku predpriprave zapisov za aplikacijo za samodejno dodeljevanje deskriptorjev. Pravila temeljijo na dolžini krnjene besede in njenih končnic. Tako na primer krnim besedo, ki je dolga 9 ali več znakov in ima končnico -ovski.

4.2. Uporabljena metoda

Za postopek samodejnega dodeljevanja deskriptorjev abstraktom sem uporabil metodo poimenovano TFIDF. Metoda se v literaturi pogosto pojavlja (Park in Zhang, 2003),(Koster, 2004),(Han in Karypis, 2000),(Fabrizio Sebastiani, 2002). Kratica TFIDF pomeni "Term Frequency Inverse Document Frequency". Že ime nam pove, da delamo s številom pojavljanja besed v dokumentih (term frequency), medtem ko nam inverzna frekvenca dokumenta (inverse document frequency) pove, v koliko dokumentih se pojavlja določen krn.

4.2.1. Učenje

Opišimo postopek TFIDF (Jerko, v delu).

Vzemimo učno množico dokumentov. Iz te učne množice dokumentov ustvarimo novo množico, kjer so člani enolični izrazi iz vseh učnih dokumentov. Dobimo torej spisek enoličnih izrazov, ki se pojavljajo v posameznih učnih dokumentih. V vsakem dokumentu nato preštejemo število posameznih enoličnih izrazov iz tega spiska. Števila enoličnih izrazov nato utežimo tako, da za vsakega od izrazov ugotovimo, v koliko dokumentih se pojavlja. Tako dobimo vektor TFIDF. Vse dokumente lahko zapišemo v obliki TFIDF, tako da zapišemo dva podatka:

- za vsak dokument število enoličnih izrazov, kar v angleščini imenujemo "Term Frequency",
- za vsak izraz, v koliko dokumentih se pojavlja, kar je v angleščini "Inverse Document Frequency"

Tako naredimo množico vektorjev TFIDF dolžine M (število vseh enoličnih izrazov v dokumentih), po en vektor za vsak dokument v učni množici dokumentov, ki jih je N.

V naslednjem koraku določimo vektor TFIDF dokumentu, za katerega želimo ugotoviti, kateremu dokumentu iz učne množice je dokument najbolj podoben. Ker delamo z vektorji, potrebujemo postopek za primerjavo dveh vektorjev. Ena od takih metod je kosinusna funkcija, ki določa podobnost med dvema vektorjema, tako da določi kosinus kota med dvema normaliziranima vektorjema.

Zapišimo postopek TFIDF matematično:

Imamo množico N dokumentov, ki jo označimo D (enačba 1). Nato iz množice dokumentov zapišemo množico enoličnih izrazov X (enačba 2). Za vsakega od teh N dokumentov zapišemo, koliko različnih izrazov vsebuje posamezen dokument (enačba 3).

$$D = \{d_1, d_2, ..., d_j, ..., d_N\} \tag{1}$$

$$X = \{x_1, x_2, ..., x_i, ...x_M\} \tag{2}$$

Sedaj lahko za posamezen dokument množice D zapišemo število izrazov, ki so se pojavili v določenem dokumentu. Torej lahko zapišemo enačbo 3 za vsak dokument iz množice D:

$$d_j^{tf} = \{tf_1, tf_2, ..., tf_M\}; j = 1, 2, 3, , ..., N \tag{3}$$

Podatke iz enačbe 3 sedaj utežimo z inverzno frekvenco dokumenta, ki je določena kot $log(\frac{N}{df_i^{tf}})$ za vsak izraz iz množice X (enačba 2).

Torej imamo končno enačbo 4:

$$TFIDF = \tag{4}$$
$$= \{tf_1 log(\frac{N}{d_1^{tf}}), ..., tf_i log(\frac{N}{d_i^{tf}}), ..., tf_M log(\frac{N}{d_M^{tf}})\}$$

V enačbi 4 je N število obravnavanih dokumentov, d_i^{tf} je število krnov v določenem dokumentu, tf_i je število pojavljanj določenega krna v dokumentih (v koliko dokumentih se pojavi določen izraz), M pa predstavlja dolžino vektorja, ki predstavlja normirani dokument (torej zapis dokumenta z določeno dolžino).

Vse zapise iz učne množice sem zapisal z enačbo 4, tako sem dobil 1033 (število učnih zapisov) vektorjev dolžine M. Pri tem sem krnom iz naslovov člankov, ki so bili tudi del zapisa v učni množici, dodelil večji pomen, tako da sem jih utežil z utežjo tri.

Ker je določen deskriptor lahko opisoval več zapisov, sem vse vektorje, ki jih opisuje deskriptor, združil z uporabo enačbe 5. Torej uporabimo preprosto računanje povprečja elementov vektorja in tako določimo centroid vektorjem, ki jih druži skupen deskriptor:

$$C = \frac{1}{|S|} \sum_{tfidf \in S} tfidf_i \tag{5}$$

Na koncu postopka učenja naj bi dobili toliko centroidov, kot je deskriptorjev. Seveda se to zgodi le, če smo v učni množici zbrali zapise, ki skupaj vsebujejo vse deskriptorje.

4.2.2. Testiranje

V postopku testiranja sem uporabil preostalih 331 od 1364 zapisov v korpusu. Vsak dokument od 331 je bilo potrebno primerjati z vsakim centroidom, nato pa sem razvrstil dodeljene deskriptorje po podobnosti s centroidi od najbolj podobnega do najmanj podobnega. Ker imamo na razpolago vektorje, je bilo potrebno uporabiti ustrezno cenilko, ki primerja vektorje. Uporabil sem kosinusno funkcijo (enačba 6).

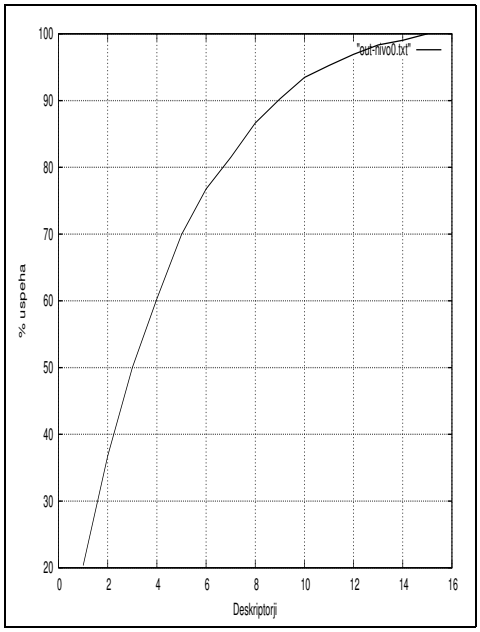
$$cos(tfidf_i, tfidf_j) = \frac{tfidf_i \cdot tfidf_j}{||tfidf_i||_2 \star ||tfidf_j||_2} \qquad (6)$$

Z uporabo kosinusne funkcije primerjamo dva vektorja in kot rezultat dobimo realno število, ki nam predstavlja podobnost med tema vektorjema.

5. Analiza rezultatov

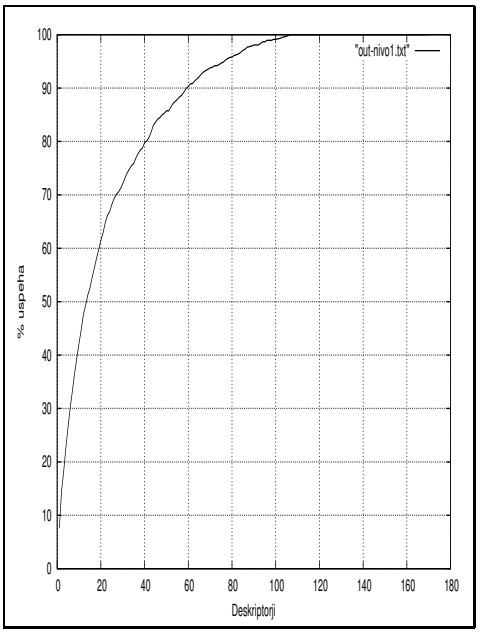
Ker je v MeSH-u 22.568 možnih deskriptorjev, bi bilo testiranje z vsemi deskriptorji iz MeSH-a problematično. Tako sem preizkus naredil z upoštevanjem hierarhične strukture MeSH-a in preizkusil metodo na različnih ravneh v MeSH-u. Na prvi ravni sem imel na razpolago 15 možnih deskriptorjev. Uspešnost metode sem preizkusil s primerjavo dodeljenih deskriptorjev v postopku testiranja z deskriptorji, ki jih je dodelil strokovnjak. Pri 100 % uspešnosti metode dodeljevanja deskriptorjev so testnemu zapisu pravilno dodeljeni vsi deskriptorji.

Testnim dokumentom je bilo dodeljeno v povprečju 2,17 deskriptorjev, vendar pa sem ugotovil, da je enemu od dokumentov v testni množici strokovnjak dodelil sedem deskriptorjev. Torej sem mejo uspešnosti postavil na sedem dodeljenih deskriptorjev.



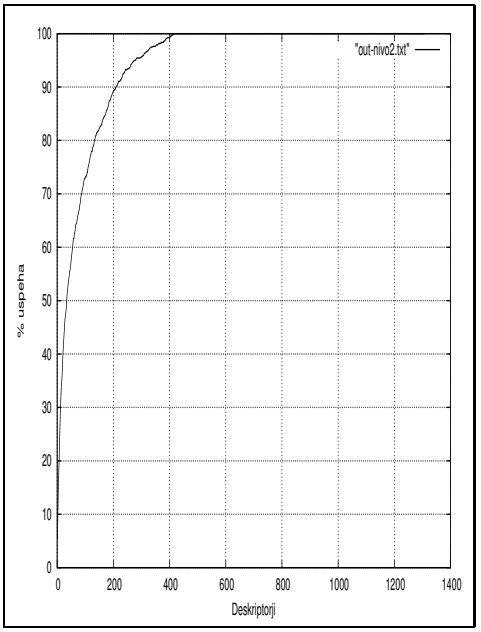
Slika 1: Uspešnost razvrščanja metode na ravni 1

Kot je razvidno na sliki 1, je uspešnost metode na ravni 1 pri sedmih dodeljenih deskriptorjih 82 %.



Slika 2: Uspešnost razvrščanja metode na ravni 2

Na ravni 2 imamo možnih 170 deskriptorjev in ker je strokovnjak enemu od dokumentov v testni množici dodelil 10 deskriptorjev, si poglejmo, kakšna je uspešnost metode pri desetih najprimernejših deskriptorjih. V tem primeru je uspešnost metode 40 %, vendar pa zanesljivost dodeljenih deskriptorjev hitro narašča in doseže 100 % pri okoli 100 deskriptorjih.

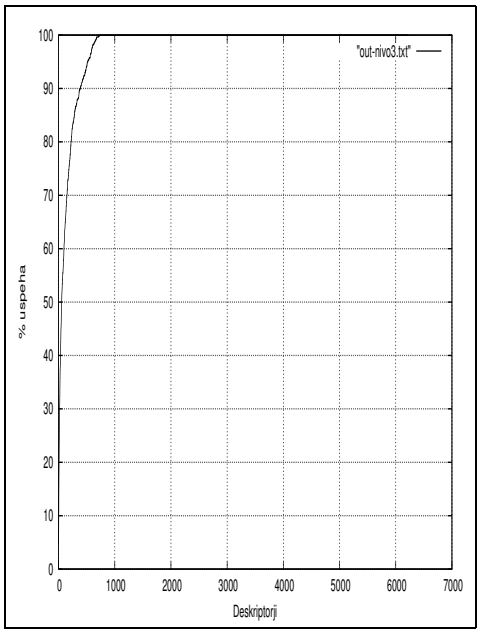


Slika 3: Uspešnost razvrščanja metode na ravni 3

Na ravni 3 imamo možnih 1308 deskriptorjev, kar je precej obsežno število in lahko pričakujemo precej slabo določanje pravilnega deskriptorja. Če si pogledamo diagram 3, vidimo, da je pri 12 dodeljenih deskriptorjih, kot jih je največ dodelil strokovnjak, uspešnost metode 30 %. Za-

nimivo pa je, da metoda doseže 100 % pri 416 predlaganih deskriptorjih, oziroma je bil vsaj v enem primeru pravilni deskriptor na 416. mestu.

Nazadnje si lahko pogledamo še raven 4, kjer imamo največ 11 deskriptorjev in je uspešnost metode pri tem številu možnih deskriptorjev 22 %. 100 % uspešnost pa metoda doseže pri 727 dodeljenih deskriptorjih, kar je pri 6218 možnih deskriptorjih zelo dober uspeh.



Slika 4: Uspešnost razvrščanja metode na ravni 4

Kot lahko vidimo, je metoda pri tako veliki učni množici (1064 dokumentov) in na četrti ravni pri enajstih deskriptorjih dokaj slaba, tako da bi to metodo težko uporabili v popolnoma samodejnem sistemu. Sistem pa nam lahko dobro služi, če ga uporabimo za zoževanje možnega števila deskriptorjev. Seveda je tukaj tudi problem premajhne velikosti učne množice (1064 dokumentov) pri tako velikem številu deskriptorjev.

Vedeti pa je potrebno, da lahko metodo še naprej učimo z zapisi, ki smo jim dodelili pravilne deskriptorje, tako kot je opisano v postopku učenja. Torej se bo uspešnost metode stalno izboljševala.

6. Nadaljnje delo

Ta trenutek me zanima predvsem izdelava uporabniku prijaznega prikaza možnih deskriptorjev. Eden od možnih prikazov je tudi takšen, kot ga uporablja NLM na svoji domači strani za MeSH (*http://www.nlm.nih.gov/cgi/mesh/2004/MB_cgi*), kjer so deskriptorji urejeni v drevesno strukturo.

Naslednji postopek, ki bi ga želel preizkusiti, je določanje deskriptorjev po drevesni strukturi. Tako bi metoda najprej razvrstila možne deskriptorje na ravni 1 in nato na vseh 15 razredih ravni 1 nadaljevala glede na težo možnega deskriptorja. Torej najprej obdelava v veji najbolj verjetnega deskriptorja na ravni 1, nato obdelava veje naslednjega možnega deskriptorja ...

7. Literatura

E Dornik in Š Adamič. 2004. Biomedicina slovenica. *http://www.mf.uni-lj.si/ibmi/info-viri/index.html*, 9 februar.

Fabrizio Sebastiani. 2002. Machine Learning in Automated Text Categorization. *ACM Computing Surveys, Vol. 34, No. 1*, marec.

E H Han in G Karypis. 2000. Centroid-based document classification: Analysis and experimental results. *Principles of Data Mining and Knowledge Discovery*.

B Jerko. v delu. Samodejno razvrščanje izvlečkov v slovenskem jeziku. Magistrsko delo.

C H A Koster. 2004. Document classification. *http://www.cs.kun.nl/ kees/ir2/papers/h03.pdf*, 23 september.

S Park in B Zhang. 2003. Large scale unstructured document classification using unlabeled data and syntactic information. *PAKDD 2003, LNAI 2637*.

M Popovic in P Willett. 1992. The effectiveness of stemming for natural-language access to slovene textual data. *Journal of the American Society for Information Science, 43(5)*.

M Porter. 2004. Snowball. *http://snowball.tartarus.org*, 9 februar.

M Schopen et al S Nelson. 2004. An interlingual database of mesh translations. *http://www.nlm.nih.gov/mesh/intlmesh.html*, avgust.

W.B. Frakes in Ricardo Baeza-Yates. 1992. *Information Retrieval*. Prentice Hall.