DK-CLARIN FAGSPROGLIGT KORPUS

Dokumentation

Indhold

1 Indledning

DK-CLARIN fagsprogligt korpus er et tekstkorpus bestående af fagsproglige tekster fra perioden 2000-2010. Det omfatter tekster på i alt ca. 11 mio. løbende ord fordelt på 7 domæner nemlig sundhed og medicin, klima og miljø, økonomi, landbrug, it, bygge- og anlæg samt nanoteknologi.

Teksterne er blevet konverteret fra forskellige formater og er blevet forsynet med lemma (ordets grundform), ordklasse og termsandsynlighed (om ordet er et fagord).

Projektet er en del af det danske CLARIN-projekt (2008-2010), som har til formål at udvikle en dansk it-infrastruktur til brug for humanistiske forskere og er finansieret af Forsknings- og Innovationsstyrelsen.

Projektgruppen har bestået af: Jakob Halskov fra Dansk Sprognævn samt Anna Braasch, Dorte Haltrup Hansen og Sussi Olsen fra Center for Sprogteknologi, Københavns Universitet.

Projektet blev præsenteret på LREC 2010-konferencen på Malta med artiklen Quality Indicators of LSP Texts — Selection and Measurements Measuring the Terminological Usefulness of Documents for an LSP Corpus der omhandler automatisk kvalitetssikring i tekstindsamlingsprocessen.

2 Korpusopbygning

I henhold til specifikationerne skal DK-CLARIN fagsprogligt korpus bestå af 11 mio. løbende ord fordelt på et antal domæner. Teksterne skal stamme fra 2000-2010.

Den endelige størrelse og sammensætning af korpusset fremgår af følgende skema.

Domæne	**Bruttopulje** **Antal ord med alle tekster fra SKAT og Sundhed.dk**	**Nettokorpus** **Antal ord inkl. en enkelt eksperttekst fra SKAT og inkl. et udvalg af tekster fra Sundhed.dk**
Sundhed og medicin	7.382.444	4.095.608
Landbrug	2.374.924	2.374.924
Klima og miljø	1.460.644	1.460.644
Økonomi	3.588.743	1.349.067
IT	1.101.059	1.101.059
Byggeri og anlæg	577.392	577.392
Nanoteknologi	358.144	358.144
Total	**16.843.350**	**11.316.838**

Tabel 1: Korpussstørrelse

Arbejdspakken har indsamlet mellem 17 og 18 mio. ord, men da der er mange tekster der ikke kan konverteres eller processeres af forskellige årsager, er mange filer slettet i processeringen, hvilket medfører at der nu er færdigprocesseret tekster med i alt 16.843.350 løbende ord.

Imidlertid er der heri indregnet et meget stort antal ord fra hhv. SKAT og Sundhed.dk.

Til økonomi-domænet har vi fra SKAT indsamlet i alt 2,9 mio. løbende ord. Disse fordeler sig på en mindre del ekspert-lægmand-tekster (228.000 ord) og en række meget store ekspert-ekspert-tekster (2,7 mio. ord)af høj fagsproglig kvalitet. Hvis vi inkluderer alle disse ekspert-ekspert-tekster, vil såvel dette domæne som det samlede korpus blive meget ubalanceret. Derfor har vi valgt i en netto korpusopstilling kun at tælle en enkelt af de store ekspert-ekspert-filer med på knap ½ mio. ord. Det er baggrunden for tallet for økonomidomænet som i yderste højre kolonne i tabellen.

Til sundhedsdomænet har vi fået adgang til mange mio. løbende ord fra Sundhed.dk. Af dem har vi downloadet ca. 5 mio. De downloadede tekster er af god kvalitet og omhandler flere forskellige fagområder inden for sundhed, men vi mener at 5 mio. ord er for stor en andel fra samme afsender i forhold til domænets øvrige størrelse. Vi har derfor valgt at udelade ca. 3,3 mio. og medtage 2 mio. ord fra Sundhed.dk i vores netto korpusoptælling (højre kolonne).

Det samlede balancerede korpus er således på **11,3 mio.** løbende ord.

Når vi alligevel har valgt at downloade og processere de øvrige 3,3 mio. ord fra Sundhed.dk og de 2,5 mio. fra SKAT, er det fordi disse tekster som nævnt er af udmærket kvalitet, og en kommende bruger med særlig interesse for sundheds- eller økonomidomænet vil kunne have glæde af at have adgang til disse tekster. De figurerer derfor i en bruttopulje af tekster selvom vi ikke tæller dem med i vores nettokorpus.

2.1 Domæner i korpusset

Det endelige antal domæner blev udvalgt efter følgende kriterier

1) at domænerne gerne må have almen interesse og stor berøringsflade, så de kan bidrage til flest mulige formål. Der vil være tale om domæner der hyppigt bidrager til almensproget med vandring af termer fra fagsprog til almensprog, og domæner hvor der arbejdes meget med terminologi. En fordel ved sådanne domæner er at teksterne også er lettere tilgængelige.
2) at undgå domæner/fagområder med for meget kunstsprog, dvs. formler etc. Derved udelukker vi i høj grad fagområder som matematik og kemi.
3) at undgå domæner der ligger for tæt op ad almensprog, som fx sport, kultur og politik fra aviser. Dette valg er delvist foretaget for at afgrænse os i forhold til wp. 2.1, det almensproglige referencekorpus.

Det blev også besluttet at satse på nogle få store domæner samt nogle mindre. På forhånd var det forventet at klima og miljø-domænet ville blive et af de store, mens landbrug ville være mindre, men det viste sig at landbrug var et domæne hvor der var rigtig mange tilgængelige tekster med forskellige kommunikationsniveauer. Derfor endte landbrug med at være det næststørste domæne hvorimod klima og miljø endte nede på en tredjeplads, af nogenlunde samme størrelse som økonomi. Der er dog et vist sammenfald mellem tekster om miljø og landbrug hvor

domæneklassificeringen er vanskelig. Her har leverandøren været udslagsgivende for hvor en tekst blev klassificeret.

I metadataene til selve teksterne (også kaldet headeren) er angivet hvilket domæne den pågældende tekst tilhører. På http://dkclarin.ku.dk/om/beskrivelse_af_arbejdspakker/beskrivelse_af_arbejdspakke_2/beskrivelse_af_arbejdspakke_2.2/ontology.xml/ ligger en xml-fil med domænerne.

2.2 Tekster fordelt på kommunikationstyper

I specifikationerne gøres der rede for de forskellige kommunikationstyper, og det blev fastlagt at tekster til dette korpus skulle være skrevet af eksperter (eller semieksperter) til eksperter, semi-ekspert eller lægmand.

Vi har bestræbt os på at opnå en vis bredde inden for hvert domæne således at der både skulle være rene eksperttekster og ekspert til lægmand-tekster, men dette har ikke været lige let at opnå. Nedenstående figur viser hvor mange løbende ord der er indsamlet fra de forskellige domæner fordelt på de forskellige kommunikationstyper. Tallene er sorteret på domæner:

Domæne	Kommunikationstype	Antal ord	Total Bruttopulje
Byggeri og anlæg	expert-advanced	57.775	
Byggeri og anlæg	expert-basic	173.711	
Byggeri og anlæg	expert-expert	345.906	**577.392**
It	expert-advanced	260.082	
It	expert-basic	840.977	**1.101.059**
Klima og miljø	expert-advanced	773.640	
Klima og miljø	expert-basic	553.223	
Klima og miljø	expert-expert	133.781	**1.460.644**
Landbrug	advanced-basic	18.759	
Landbrug	expert-advanced	373.781	
Landbrug	expert-basic	421.846	
Landbrug	expert-expert	1.579.297	**2.393.683**
Nanoteknologi	expert-advanced	54.964	
Nanoteknologi	expert-basic	303.180	**358.144**
Økonomi	expert-advanced	18.024	
Økonomi	expert-basic	228.686	
Økonomi	expert-expert	3.342.033	**3.588.743**

Sundhed og medicin	expert-advanced	278.095	
Sundhed og medicin	expert-basic	6.129.259	
Sundhed og medicin	expert-expert	975.090	**7.382.444**
Total		**16.862.109**	**16.862.109**[1]

Tabel 2: Antallet af ord fordelt på kommunikationstype og domæne, sorteret efter domæne.

I tabellen tales der om expert-basic hvilket svarer til ekspert-lægmand, hvor expert-advanced svarer til ekspert-semiekspert, typisk tekster til studerende etc.

Det er interessant at analysere tallene for hvilke kommunikationstyper vi har indsamlet for hvilke domæner. Det er måske typisk for domænerne at vi har fundet utroligt mange ekspert-ekspert-tekster om økonomi, mens den overvejende del af de medicintekster vi har fået adgang til, er ekspert-lægmand.

For domænerne nanoteknologi og it har det ikke været muligt for os at finde ekspert til ekspert-tekster på dansk. Al kommunikation mellem eksperter inden for de to domæner ser ud til at foregå på engelsk. Hvis det findes på dansk, har vi i hvert fald ikke fået adgang til dem.

I nedenstående tabel ses de samme tal, men denne gang sorteret på kommunikationstyper.

Domæne	Kommunikationstype	Antal ord	Total Bruttopulje
Landbrug	advanced-basic	18.759	**18.759**
Byggeri og anlaeg	expert-basic	173.711	
It	expert-basic	840.977	
Klima og miljoe	expert-basic	553.223	
Landbrug	expert-basic	421.846	
Nanoteknologi	expert-basic	303.180	
Oekonomi	expert-basic	228.686	
Sundhed og medicin	expert-basic	6.129.259	**8.650.882**
Byggeri og anlaeg	expert-advanced	57.775	
It	expert-advanced	260.082	
Klima og miljoe	expert-advanced	773.640	
Landbrug	expert-advanced	373.781	

[1] Dette tal er lidt højere end bruttotallet fra figur 1 fordi vi i denne sammentælling har medtaget nogle advanced-basic tekster fra landbrugsdomænet som vi ellers har valgt ikke at medregne i hverken bruttopulje eller nettokorpus fordi vi normalt kræver tekster med forfattere på ekspertniveau.

Nanoteknologi	expert-advanced	54.964	
Oekonomi	expert-advanced	18.024	
Sundhed og medicin	expert-advanced	278.095	**1.816.361**
Byggeri og anlaeg	expert-expert	345.906	
Klima og miljoe	expert-expert	133.781	
Landbrug	expert-expert	1.579.297	
Oekonomi	expert-expert	3.342.033	
Sundhed og medicin	expert-expert	975.090	**6.376.107**
Total		**16.862.109**	**16.862.109**

Tabel 3: Antallet af ord fordelt på kommunikationstype og domæne, sorteret på kommunikationstype.

I denne tabel bliver det tydeligt at vores korpus samlet har en rimelig balance mellem ekspert-ekspert-tekster og ekspert-lægmand-tekster. Da vi til det endelige nettokorpus har fravalgt lidt flere ekspert-lægmand-tekster end ekspert-ekspert-tekster, bliver disse to tal endnu mere lige. Der er overraskende mange ekspert-advanced som for en stor del dækker over tekster skrevet til studerende eller til administrative medarbejdere inden for et fagområde, folk som ganske vist har en viden om emnet, men ikke på ekspertniveau.

3 Tekstindsamling (JH)

Første skridt i tekstindsamlingen bestod i at finde potentielle tekstleverandører for de syv forskellige fagområder dækket af korpusset. Tekstleverandørerne blev primært lokaliseret ved hjælp af internetsøgninger og personlige kontakter. Første kontakt med de potentielle tekstleverandører bestod typisk i udsendelse af e-mails med et udkast til en brugsaftale og/eller telefonisk henvendelse. Aftaledokumenterne kan ses i bilag X. For hver potentiel leverandør registreredes der kontaktoplysninger, aftaleindgåelse, leverancestatus med mere i en simpel MySQL-database.

Efter modtagelse af et underskrevet aftaledokument, kunne den konkrete tekstindsamling begynde. Indsamlingsteknikkerne varierede alt efter teksternes formater, tilgængelighed og beskaffenhed. I den følgende tabel gives der eksempler på alle de forskellige teksttyper og -formater som er blevet behandlet i projektet. Afsnittet om tekstprocessering beskriver selve konverteringen af de forskellige tekstformater til XML. Den mest anvendte indsamlingsmetode var manuel download fra websted.

Eksempler på teksttyper og tekstformater			
Leverandør	*Teksttype*	*Indsamlingsmetode*	*Tekstformat*
Region Hovedstaden	Specialerapporter	Manuel download fra websted	PDF

Det jordbrugsvidenskabelige fakultet	Faglige rapporter	Manuel download fra websted	PDF
sundhed.dk, laegehaandbogen.dk	Formidlende sygdoms- og symptombeskrivelser	Automatisk download fra websted med wget	(X)HTML
Libris	It-hæfter for lægfolk	Personlig afhentning på USB-stick	TXT

Tabel 4: Teksttyper og -formater

De downloadede originale tekster blev dernæst uploadet til serveren Ida på CST ved hjælp af et perl-script. Parallelt med uploadningen registrerer scriptet metadata om teksterne (deres domæne, leverandøren, stien til filen osv.) i en MySQL-database som både vedligeholdes lokalt på DSN og på Ida på CST. Dermed optræder de uploadede filer automatisk i headergeneratoren hvorfra headerkoderne kan åbne dem og kode headere (læs mere afsnit 1.5).

Indsamlede tekstmængder opgjort efter tekstformat	
Tekstformat	**Antal w-elementer**
HTML	8.829.537
PDF	6.826.713
TXT	1.273.202
ODT	71.664
DOC	13.259

Tabel 5: Tekster opgjort efter format

4 Tekstprocessering

4.1 Tekstkonvertering

Der blev taget udgangspunkt i tre forskellige tekstformater, nemlig PDF, HTML og TXT. Især PDF voldte mange vanskeligheder at konvertere til XML. Selv med kommercielle konverteringsredskaber som Adobes egen ”Save as XML”-funktion i Acrobat, stod det hurtigt klart at semantikken i XML’en nærmest varierede fra tekst til tekst og derfor i mange tilfælde var svær at forudsige, fortolke og arbejde videre med. Desuden opstod der en del fejl i segmenteringen af teksten (især når denne indeholdt flere kolonner og/eller mange tabeller og figurer). Endelig opstod der uforklarlige

mellemrumstegn midt i visse ord som i det nedenstående eksempel hvor ordet ”overflade” er blevet delt.

<Sect><P>…vekselvirkninger mellem nanoaggregaterne og mellem aggregaterne og den *overfl ade*, de sidder på. </P>

Et stort antal alternative PDF-konverteringsværktøjer blev afprøvet, både ikke-kommercielle værktøjer som pdftotext, pdfedit, pdftohtml -xml og kommercielle værktøjer som ABBYY Finereader. I sidste ende viste pdftotext sig at være den bedste overall-løsning. Det blev derfor besluttet at anvende dette værktøj til at konvertere samtlige PDF-tekster til TXT og derpå konvertere dette format til XML (primært ved at genkende tekstafsnit med et simpelt script).

Der er dog stadig forskellige problemer med de konverterede pdf-filer. Sidenumre og sidehoveder/-fødder optræder jævnligt inde i den løbende tekst ligesom tabeller og figurer kan ødelægge tekstens sammenhæng. Henvendelser til forskellige udenlandske samarbejdspartnere viser at andre har samme problem, og at en bedre løsning end den vi har valgt, endnu ikke er fundet.

Velstrukturerede HTML-tekster viste sig i de fleste tilfælde relativt enkle at konvertere til XML, idet værktøjet TagSoup (http://home.ccil.org/~cowan/XML/tagsoup/) muliggør nem konvertering fra (muligvis invalid) HTML til valid XHTML som igen kan konverteres til XML ved hjælp af et XSLT-script og en XSLT-fortolker som Saxon (http://saxon.sourceforge.net/). I nogle tilfælde var det endda muligt på denne facon automatisk at trække alle relevante metadata om teksten ud af det originale HTML-dokument således at tekstheaderne ikke skulle kodes manuelt (det gjaldt fx teksterne fra sundhed.dk).

4.2 Processering til i Clarin Basic Format (cbf), samt validering

Alle tekster gennemløber en proces hvor de tokeniseres, dvs. opdeles i enkelt ord eller lignende enheder og konverteres til det fælles CLARIN format. Desuden bliver tekstens metadata valideret for at se om syntaksen overholdes. Denne proces er beskrevet i detaljer i dokumentet ”Dokumentation af processeringspipeline” af Dorte Haltrup Hansen, http://dkclarin.ku.dk/om/beskrivelse_af_arbejdspakker/beskrivelse_af_arbejdspakke_2/beskrivelse_af_arbejdspakke_2.2/PipelineProces_final_30082011.doc/

4.3 Headergenerering (JH)

Alle tekster i det fagsproglige korpus er udstyret med en header som indeholder en række metadata om tekstens ophav, tilblivelse osv. Headeren følger TEI-P5-standarden, og alle headere er enten genereret automatisk eller semi-automatisk (ved hjælp af et stykke specialudviklet software, en såkaldt headergenerator). De følgende to afsnit vil kort beskrive og give eksempler på henholdsvis automatisk og semi-automatisk headergenerering.

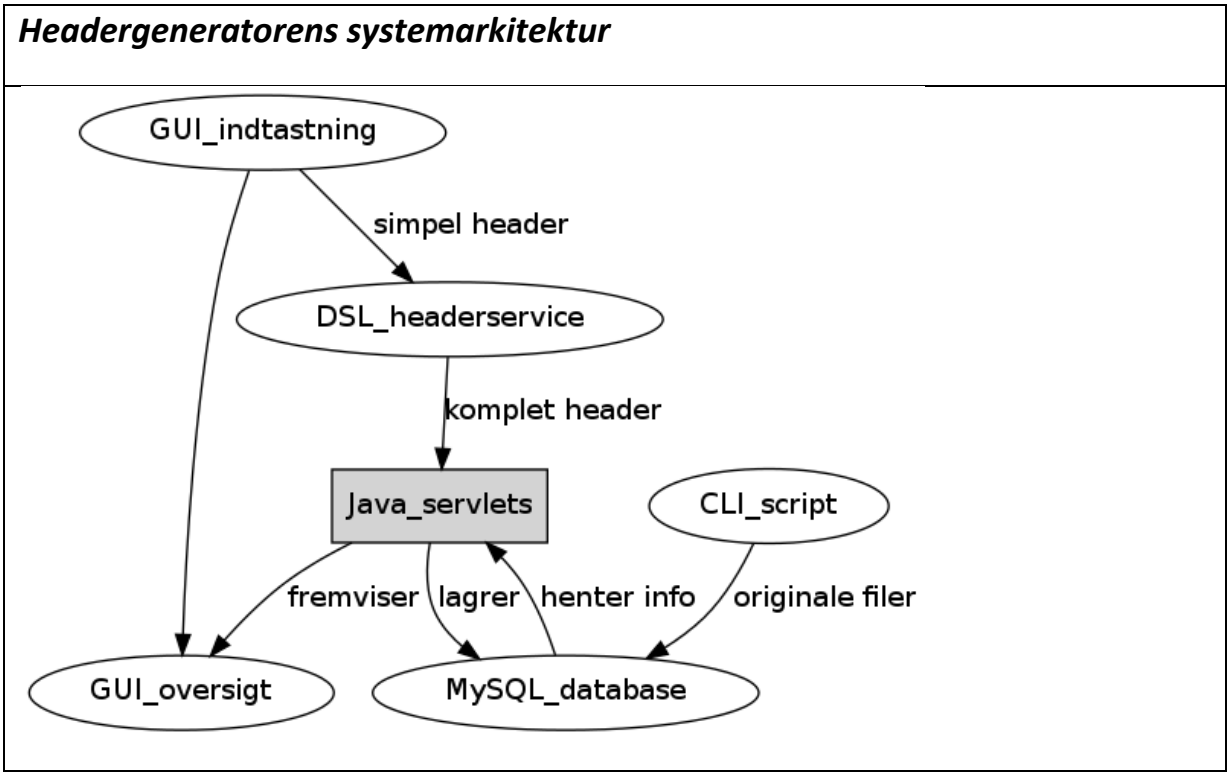
Automatisk headergenerering

For enkelte tekstleverandører var de originale tekster af en sådan beskaffenhed og struktur at det viste sig muligt automatisk at identificere alle relevante metadata og generere tekstheadere fuldautomatisk. Et eksempel på en sådan tekstleverandør er sundhed.dk hvor teksternes HTML-format er så struktureret og regelmæssigt at det kan omformes til XHTML og bearbejdes med XSLT

således at informationer om forfattere, udgivelsesdato, titel osv. automatisk kan udtrækkes og indsættes i tekstheaderen.

Semi-automatisk headergenerering

For flertallet af tekstleverandørerne forelå de originale tekster imidlertid i formater som PDF hvor det ikke var muligt at udtrække metadata automatisk. I disse tilfælde måtte metadata kodes manuelt, men for at reducere arbejdsbyrden blev en såkaldt headergenerator udviklet af DSN og hostet hos CST. Headergeneratorens overordnede systemarkitektur er skitseret i nedenstående figur som også illustrerer hvordan der kommunikeres med en webservice der hostes på DSL.

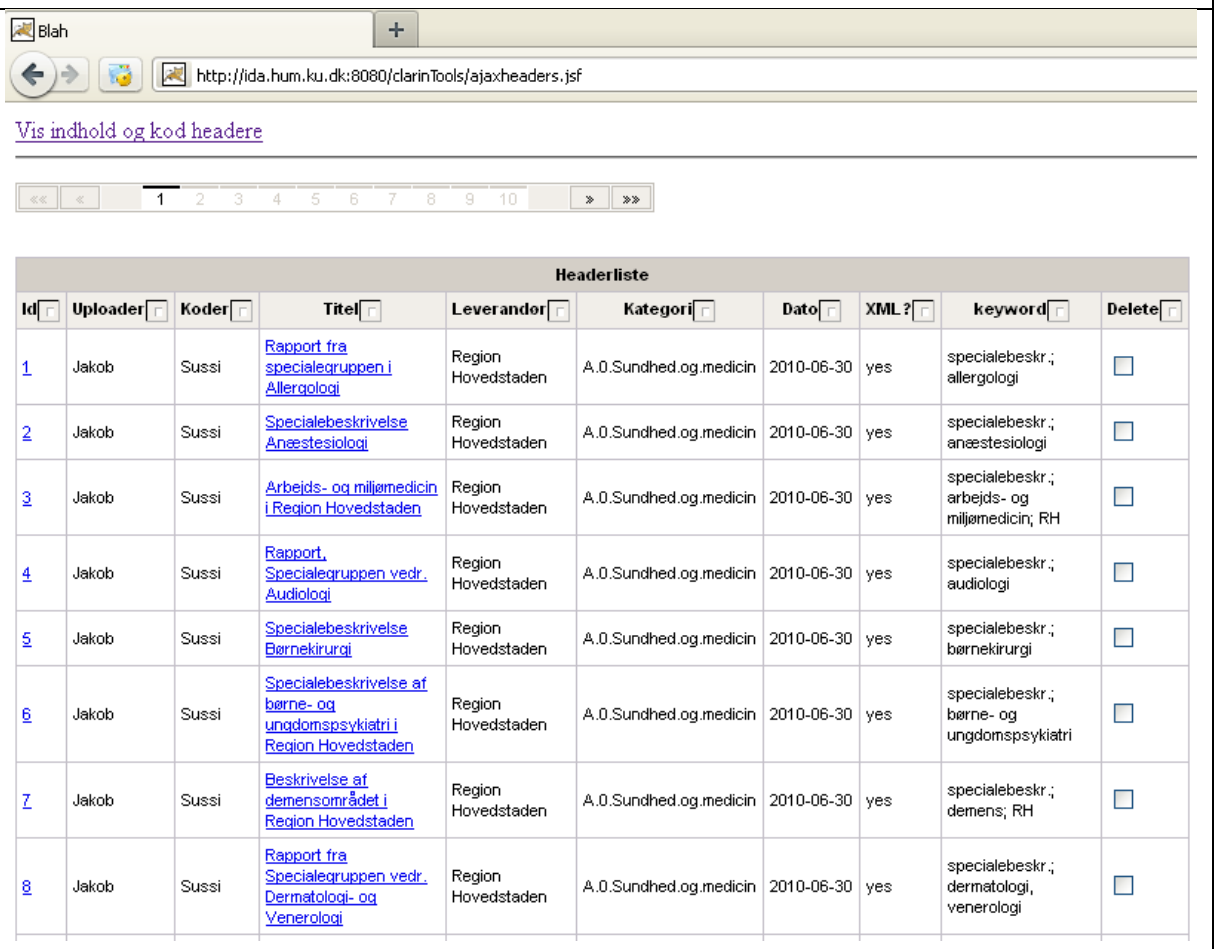


Figur 1 Headergeneratorens systemarkitektur

Headergeneratoren er en java-webapplikation der anvender Java servlets til dels at kommunikere med en MySQL-database hvori tekster og headere lagres og dels at generere brugergrænsefladen (ved hjælp af JSF – Java Server Faces). Endelig er der servletter som afsender requests til en webservice på DSL (med simple headere) og modtager responses fra samme webservice (med de komplette headere). De originale filer uploades til MySQL-databasen på CST via et CLI-script som køres på DSN.

Brugergrænsefladen omfatter to dele, nemlig en indtastningsdel (hvor metadata for en given tekst kodes manuelt i et skema), og en oversigtsdel som viser hvilke tekster databasen indeholder, hvilke som mangler headere osv.

Headergeneratorens brugergrænseflade - oversigtsdelen



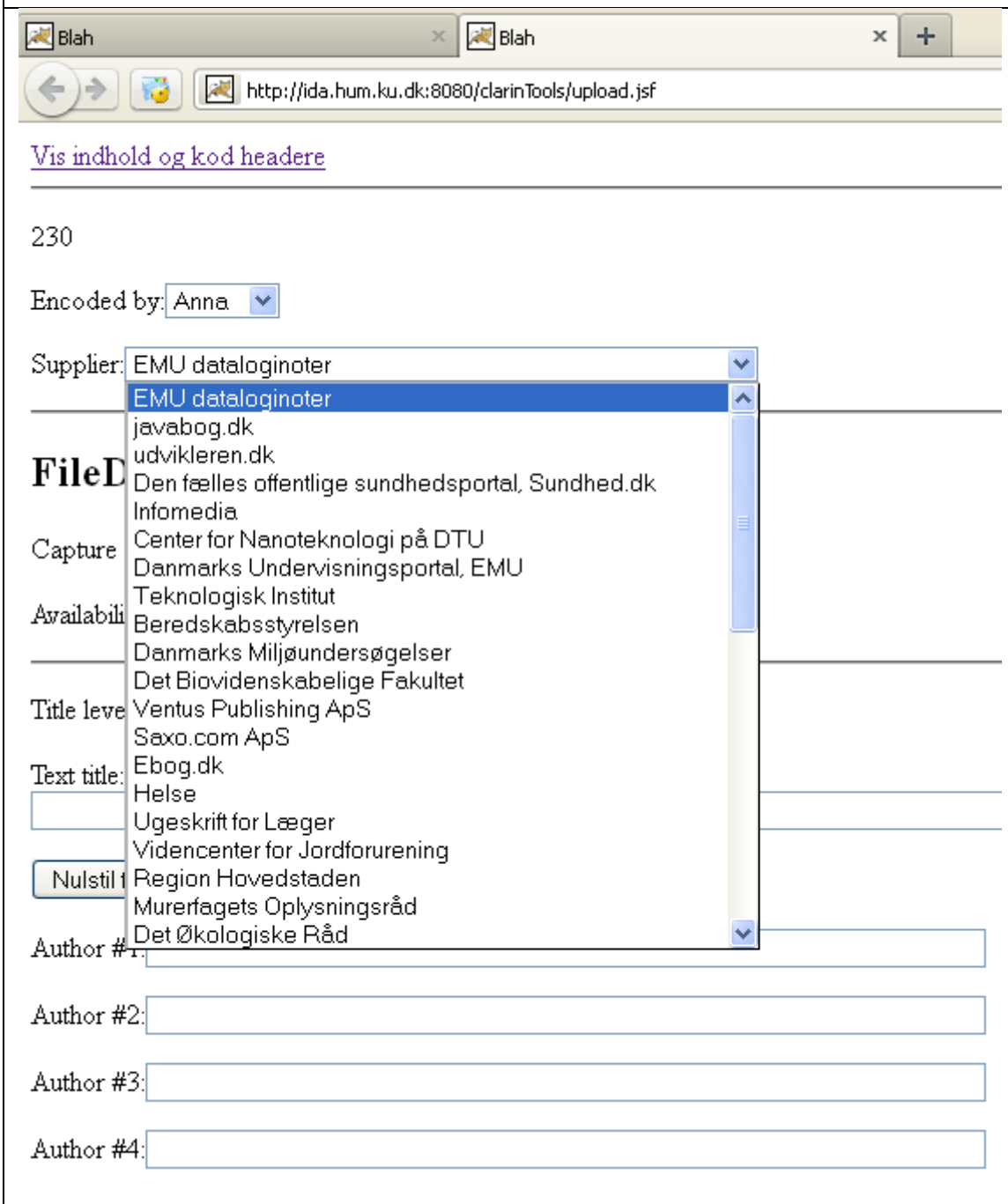
Headerliste

Id	Uploader	Koder	Titel	Leverandør	Kategori	Dato	XML?	keyword	Delete
1	Jakob	Sussi	Rapport fra specialegruppen i Allergologi	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; allergologi	☐
2	Jakob	Sussi	Specialebeskrivelse Anæstesiologi	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; anæstesiologi	☐
3	Jakob	Sussi	Arbejds- og miljømedicin i Region Hovedstaden	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; arbejds- og miljømedicin; RH	☐
4	Jakob	Sussi	Rapport, Specialegruppen vedr. Audiologi	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; audiologi	☐
5	Jakob	Sussi	Specialebeskrivelse Børnekirurgi	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; børnekirurgi	☐
6	Jakob	Sussi	Specialebeskrivelse af børne- og ungdomspsykiatri i Region Hovedstaden	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; børne- og ungdomspsykiatri	☐
7	Jakob	Sussi	Beskrivelse af demensområdet i Region Hovedstaden	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; demens; RH	☐
8	Jakob	Sussi	Rapport fra Specialegruppen vedr. Dermatologi- og Venerologi	Region Hovedstaden	A.0.Sundhed.og.medicin	2010-06-30	yes	specialebeskr.; dermatologi, venerologi	☐

Figur 2: Headergeneratorens grænseflade

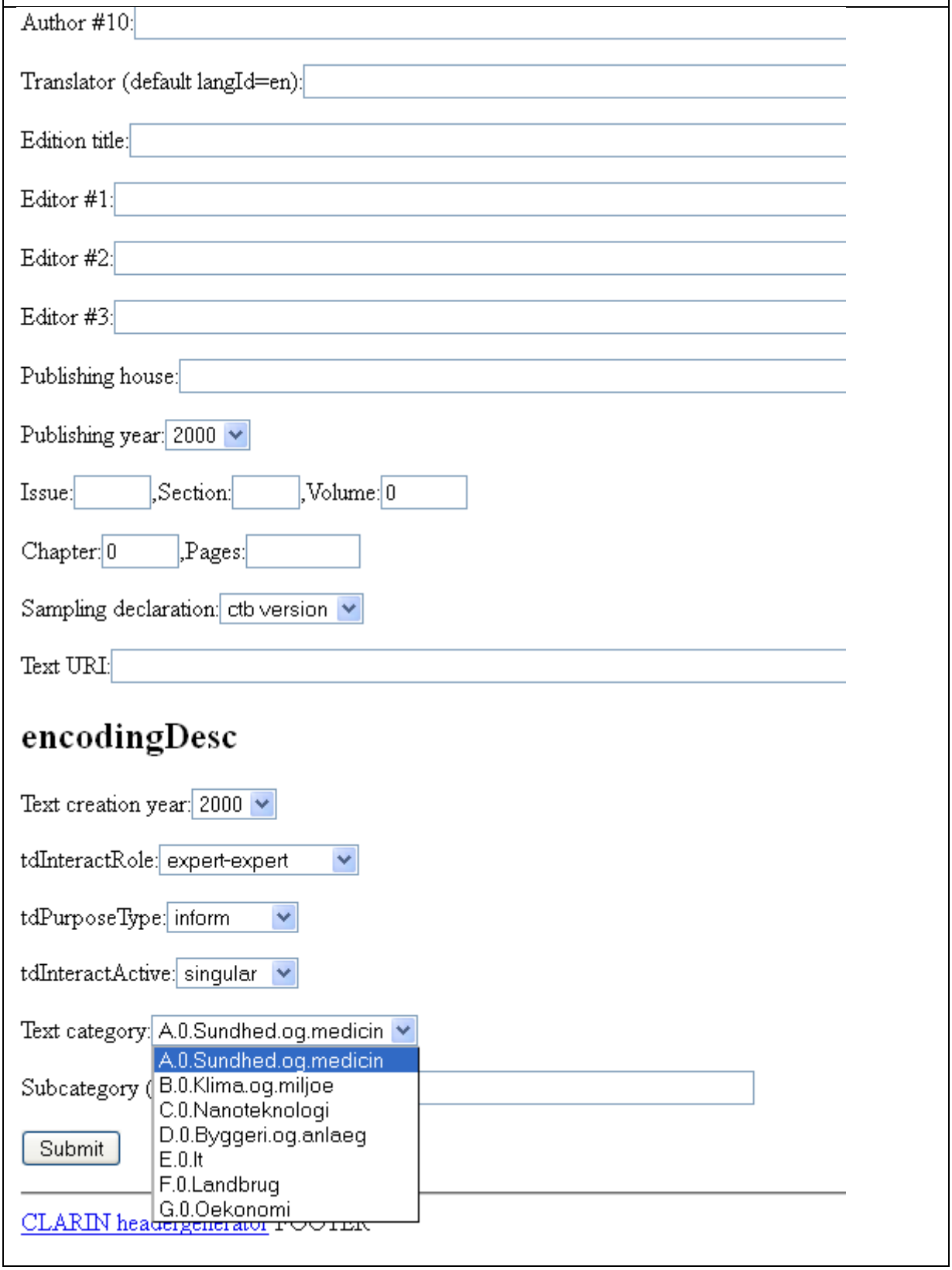
Fra oversigtsdelen kan brugeren navigere til indtastningsdelen ved at klikke på en tekst-id hvortil der endnu ikke er kodet nogen header.

Headergeneratorens brugergrænseflade – indtastningsdelen 1



Figur 3: Headergeneratorens brugergrænseflade – indtastning del 1

Headergeneratorens brugergrænseflade – indtastningsdelen 2



Figur 5: Headergeneratorens brugergrænseflade – indtastning del 2

4.4 Erfaringer med kodning af headere i headergeneratoren

Semiautomatisk headergenerering vha. headergeneratoren er blevet brugt til ca. 660 tekster.

Der blev udarbejdet en vejledning 'Kodning af headere' til brug for de personer der skulle kode headere. Dokumentet findes på http://dkclarin.ku.dk/om/beskrivelse_af_arbejdspakker/beskrivelse_af_arbejdspakke_2/beskrivelse_af_arbejdspakke_2.2/Kodning_af_headere.doc

Headergeneratoren er i sig selv let at benytte, det vanskelige ved headerkodningen er finde de nødvendige oplysninger i teksten. Ikke alle udgivelser har en kolofon, men har oplysningerne liggende forskellige steder i teksten. Flere gange har vi måttet finde oplysninger om forfatter, udgiver, værtspublikationens titel, årstal el. lign. på hjemmesiden hvorfra teksten er hentet. Og nogle få tekster har vi måttet lade udgå fordi det ikke var muligt at få oplyst fx årstallet for udgivelsen.

Headergeneratoren gør det enkelt og tidsbesparende at kode flere tekster fra samme udgiver eller fra samme publikationsserie lige efter hinanden da værdierne fra sidste kodning optræder næste gang der åbnes en ny kodningsformular - så længe der ikke er gået for lang tid uden aktivitet.

Det er vanskeligt at fastslå tidsforbruget for en headerkodning da dette afhænger af hvor enkelt det er at finde oplysningerne i teksten, og af hvor mange tekster af samme slags der skal kodes. En header for en ny type tekster kan tage 20-25 minutter at kode mens en enkel tekst inde i en række af ensartede tekster fra fx samme udgivelsesserie kan kodes på 2-4 minutter.

5 Annoteringer

Det fagsproglige korpus er annoteret med part of speech-tags, lemmaer og termstatus.

5.1 Part og speech-tags og lemmaer

Hver annotation ligger i en såkaldt spangroup ved siden af teksten, og hvert led i de forskellige annotationslag refererer tilbage til de unikke id'er for hvert token, t1, t2 etc. Nedenfor ses et eksempel fra en spangroup først med et stykke tokeniseret tekst:

```
<spanGrp ana="#CstClarinDaTokeniser">
                    …
<span xml:id="t2534" from="#i151.1">Windows</span>
                    <span xml:id="t2535" from="#i151.3">vil</span>
                    <span xml:id="t2536" from="#i151.5">nu</span>
                    <span xml:id="t2537" from="#i151.7">advare</span>
                    <span xml:id="t2538" from="#i151.9">dig</span>
                    <span xml:id="t2539" from="#i151.11">om</span>
                    <span xml:id="t2540" from="#i151.12">,</span>
                    <span xml:id="t2541" from="#i151.14">at</span>
                    <span xml:id="t2542" from="#i151.16">du</span>
                    <span xml:id="t2543" from="#i151.18">ikke</span>
                    <span xml:id="t2544" from="#i151.20">har</span>
                    <span xml:id="t2545" from="#i151.22">et</span>
```

```
<span xml:id="t2546" from="#i151.24">aktivt</span>
<span xml:id="t2547" from="#i151.26">antivirusprogram</span>
<span xml:id="t2548" from="#i151.28">installeret</span>
<span xml:id="t2549" from="#i151.30">på</span>
<span xml:id="t2550" from="#i151.32">din</span>
<span xml:id="t2551" from="#i151.34">computer</span>
<span xml:id="t2552" from="#i151.35">.</span>
…

</spanGrp>
```

Nedenfor vises den samme tekst med pos-tags. Pos-taggingen er udført med CST's pos-tagger (http://cst.dk/online/pos_tagger/),

```
<spanGrp ana="#csttaggerXML">
…
<span xml:id="p2534" from="#t2534">EGEN</span>
<span xml:id="p2535" from="#t2535">V_PRES</span>
<span xml:id="p2536" from="#t2536">ADV</span>
<span xml:id="p2537" from="#t2537">V_INF</span>
<span xml:id="p2538" from="#t2538">PRON_PERS</span>
<span xml:id="p2539" from="#t2539">PRÆP</span>
<span xml:id="p2540" from="#t2540">TEGN</span>
<span xml:id="p2541" from="#t2541">UKONJ</span>
<span xml:id="p2542" from="#t2542">PRON_PERS</span>
<span xml:id="p2543" from="#t2543">ADV</span>
<span xml:id="p2544" from="#t2544">V_PRES</span>
<span xml:id="p2545" from="#t2545">PRON_UBST</span>
<span xml:id="p2546" from="#t2546">ADJ</span>
<span xml:id="p2547" from="#t2547">N_INDEF_SING</span>
<span xml:id="p2548" from="#t2548">V_PARTC_PAST</span>
<span xml:id="p2549" from="#t2549">PRÆP</span>
<span xml:id="p2550" from="#t2550">PRON_POSS</span>
<span xml:id="p2551" from="#t2551">N_INDEF_SING</span>
<span xml:id="p2552" from="#t2552">TEGN</span>
…
</spanGrp>
```

og endelig vises teksten med lemmaer. Lemmatiseringen er udført med CST's lemmatiser (http://cst.dk/online/lemmatiser/)

```
<spanGrp ana="#cstlemma">
…
<span xml:id="l2534" from="#t2534" >Windows</span>
<span xml:id="l2535" from="#t2535" >ville</span>
<span xml:id="l2536" from="#t2536" >nu</span>
<span xml:id="l2537" from="#t2537" >advare</span>
<span xml:id="l2538" from="#t2538" >du</span>
<span xml:id="l2539" from="#t2539" >om</span>
<span xml:id="l2540" from="#t2540" >,</span>
<span xml:id="l2541" from="#t2541" >at</span>
<span xml:id="l2542" from="#t2542" >du</span>
<span xml:id="l2543" from="#t2543" >ikke</span>
<span xml:id="l2544" from="#t2544" >have</span>
```

```
                <span xml:id="l2545" from="#t2545" >en</span>
                <span xml:id="l2546" from="#t2546" >aktiv</span>
                <span xml:id="l2547" from="#t2547" >antivirusprogram</span>
                <span xml:id="l2548" from="#t2548" >installere</span>
                <span xml:id="l2549" from="#t2549" >på</span>
                <span xml:id="l2550" from="#t2550" >din</span>
                <span xml:id="l2551" from="#t2551" >computer</span>
                <span xml:id="l2552" from="#t2552" >.</span>
…
</spanGrp>
```

5.2 Termstatus

Ordenes termstatus er beregnet med Dansk Sprognævns termtagger. Samtlige tokens i teksten får en 'termhood'-værdi. Denne beregnes på basis af frekvensoplysninger i et almensprogligt referencekorpus sammenholdt med frekvensen af den givne token lokalt i inputdokumentet. Værdierne beregnes med den statistiske metode log-likelihood. Man kan læse mere om termtaggeren og dens algoritmer i dokumentet "Dokumentation af processeringspipeline" (http://dkclarin.ku.dk/om/beskrivelse_af_arbejdspakker/beskrivelse_af_arbejdspakke_2/beskrivelse_af_arbejdspakke_2.2/PipelineProces_final_30082011.doc/).

```
<spanGrp ana="#DsnClarinDaTermTaggerLogLikelihood">
…
<span xml:id="th2534" from="#t2534" >2884.522</span>
<span xml:id="th2535" from="#t2535" >61.166</span>
<span xml:id="th2536" from="#t2536" >0</span>
<span xml:id="th2537" from="#t2537" >5.423</span>
<span xml:id="th2538" from="#t2538" >441.963</span>
<span xml:id="th2539" from="#t2539" >0</span>
<span xml:id="th2540" from="#t2540" >0</span>
<span xml:id="th2541" from="#t2541" >0</span>
<span xml:id="th2542" from="#t2542" >0</span>
<span xml:id="th2543" from="#t2543" >-3.204</span>
<span xml:id="th2544" from="#t2544" >-52.246</span>
<span xml:id="th2545" from="#t2545" >0</span>
<span xml:id="th2546" from="#t2546" >3.955</span>
<span xml:id="th2547" from="#t2547" >101.805</span>
<span xml:id="th2548" from="#t2548" >81.206</span>
<span xml:id="th2549" from="#t2549" >0</span>
<span xml:id="th2550" from="#t2550" >1318.490</span>
<span xml:id="th2551" from="#t2551" >674.797</span>
<span xml:id="th2552" from="#t2552" >0</span>
…
</spanGrp>
```

Afhængigt af hvilken terminologisk skole man tilhører, så er der mange definitioner på hvad en term eller et fagudtryk er. Her følger to:

Terms signify concepts belonging to a specific subject field (Madsen, 1991).

The terminological unit represents a concept, uniquely and completely, taken out of any textual context. The existence of this one-to-one relationship between a linguistic expression and an extra-linguistic object is a situation which particularly concerns the terminological units (Bourigault, 1992).

Forskellen på ord og termer er altså at termer, i teorien, er tekstmarkører for domænespecifikke begreber som opretholder deres semantiske indhold helt uafhængigt af den kontekst hvori de forekommer; de er med andre ord monoseme og kontekstuafhængige (bortset fra den kontekst selve domænet skaber). I praksis kan det imidlertid være mere vanskeligt at afgøre om en given tekststreng har termstatus eller ej, for eksempel deles mange termer mellem flere forskellige domæner. Desuden afhænger den opfattede termstatus også af læserens domænespecifikke viden. Lægfolk eller semi-eksperter vil således være tilbøjelige til at klassificere flere tekststrenge som termer end "rigtige" eksperter med omfattende indsigt i domænets ontologi. Derfor er det også vanskeligt formelt at evaluere hvor godt en termtaggers output i grunden er. I afsnit 1.7.3.2 forsøger vi alligevel at give en uformel evaluering af termtaggerens præcision.

Under alle omstændigheder er det vigtigt at understrege at termhood-værdierne i ovenstående eksempel ikke siger noget i sig selv og heller ikke kan sammenholdes på tværs af teksterne i korpusset. Værdierne er alene en rangordning af ordforrådet i den aktuelle tekst hvor tekststrenge med de højeste positive værdier er markant overrepræsenteret i den aktuelle tekst i forhold til det almensproglige referencekorpus (og dermed antageligvis har større termhood), mens tekststrenge med værdier omkring 0 er neutrale, og tekststrenge med negative værdier er markant underrepræsenteret i den aktuelle tekst (og er dermed antageligvis mere karakteristiske for almensprog end for det pågældende fagsprog).

6 Kvalitetssikring

Kvalitetssikring vedrører mange aspekter af korpusarbejdet. Vi har arbejdet med automatisk kvalitetssikring af indsamlede tekster, derudover har vi lavet manuelle stikprøvekontroller af et meget stort antal filer vedr. tekstkonverteringen og den videre processering til det endelige format. Endelig har vi planer om at lave en lille validering af annotationslagene, dvs. foretage en stikprøvekontrol af de forskellige annoteringer.

6.1 Automatisk kvalitetssikring af tekster

Arbejdspakkegruppen har forsøgsvis en procedure for automatiske kvalitetstjek. Tjekkene inddeles i to grupper: de der måler læsbarhed og graden af formelt sprog, og de der måler fagsproglighedsgraden i teksterne.

Gruppen præsenterede som nævnt metoden på LREC 2010 konferencen og fik mange interesserede henvendelser og ideer til videreudvikling. Desværre har tidspres bevirket at vi ikke har kunnet videreudvikle og implementere metoden i den endelige processeringspipeline. Da vi kun har hentet tekster fra hjemmesider som vi på forhånd havde kvalitetstjekket, har det nok ikke haft en afgørende betydning for kvaliteten af de indsamlede tekster, men det er helt klart at det til et fremtidigt korpus vil være tidsbesparende hvis man automatisk kan kvalitetstjekke downloadede tekster inden disse processeres og indlemmes i korpus. Evt. også inden man beder om tilladelse til at bruge teksterne. Tekster der ikke opfylder kvalitetskravene, bliver jo alligevel slettet og vil ikke figurere nogen steder.

6.2 Manuel stikprøvekontrol

I forbindelse med konvertering og processering af teksterne er der foretaget manuel stikprøvekontrol på en lang række filer. Kontrollerne afslørede at især konverteringen af pdf-filer er et problem som også beskrevet i afsnit 1.4.1.

De hyppigste problemer er at der optræder blanktegn inde i ord, at sidehoved og –fod optræder inde i den løbende tekst, samt at der ved komplicerede pdf-filer med mange tabeller og illustrationer bliver rod i kolonnerne fordi konverteringsprogrammet misfortolker hvor teksten fortsætter, og tekstens sammenhæng dermed går tabt. Da næsten 7 mio. ord kommer fra pdf-tekster, er det ikke umiddelbart en løsning helt at udelade pdf-teksterne. Vi har valgt at gøre brugere opmærksomme på at sådanne problemer kan forekomme i teksterne.

6.3 Validering og evaluering af annoteringerne

Arbejdspakkegruppen mener at der foruden de ovennævnte tjek og kontroller bør udføres en validering/ evaluering af annoteringerne i korpus, både af PoS-tags- og lemma-tilskrivningen og af termtag-tilskrivningen. Da der imidlertid ikke er afsat tid i projektet til større valideringer, er der tale om validering og evaluering i mindre målestok.

6.3.1 Validering af lemma- og pos-annoteringer

Hvis man tager udgangspunkt i en række større internationale korpusprojekter som CST har lavet lignende valideringer for, ville det ideelle antal validerede ord være en promille af hele korpusset svarende til 11.300 ord. En validering af den størrelsesorden er imidlertid meget resursekrævende, og ligesom det ofte er tilfældet i store internationale projekter, er det i dette projekt nødvendigt at skære drastisk ned på antallet af ord. Vi har valgt at dividere med 10 og gennemføre et tjek af 1130 ord.

Valideringen er foregået ved at sætninger med til sammen ca. 1130 ord er blevet udvalgt tilfældigt fra korpus. Der er blevet valideret på hele sætninger for at bedre at kunne vurdere en pos-tag og et lemmas korrekthed i sammenhængen. Der blev dog udvalgt flere sætninger end hvad der svarer til de 1130 ord, da hver sætning jo også indeholder fra et til flere tegn som ikke tælles som ord. Det samlede uddrag er på knap 1300 tokens. Hvert ord er blevet tjekket for de to annoteringstyper. Fejl er blevet optalt, men ændringsforslag er ikke angivet da evt. fejl ikke vil blive rettet i korpus, men kun ville være til nytte ved en forbedring af værktøjerne. Men valideringen giver et fingerpeg om annoteringernes pålidelighed i form af en procentsats.

Det endelige antal ord der blev valideret var 1280. Heraf blev der fundet 104 pos-tagfejl , svarende til 8,1% fejl, og afhængig af hvordan man optæller lemmatiseringsfejl blev der fundet 26 lemmatiseringsfejl svarende til 2 %.

Resultatet er glimrende for lemmatisering, men knap så godt for pos-taggingen. I det følgende vil resultaterne blive analyseret.

Fejl i postagging

En af fejlkilderne for pos-taggeren er forkert tokenisering. Hvis først et ord er tokeniseret forkert ofte som konsekvens af en fejl i tekstkonverteringen, fx

<ana token="RÅ"… korrekt: <ana token="RÅD"…
<ana token="Dnaturen"… korrekt: < ana token="naturen"…

får det naturligvis konsekvens for både pos-tagging og lemmatisering.

Den største fejlkilde opstår hvor der ikke er tale om løbende tekst. Teksterne i korpus består jo både af løbende tekst, overskrifter, indholdsfortegnelser, kolofoner etc. Så længe der blot er tale om en oplistning af navne klarer taggeren det ok, men når der fx kommer lister af navne efterfulgt af folks titler, går taggingen skævt.

<ana token="Mogens" pos="EGEN"
<ana token="Rishøj" pos="ADJ"
<ana token="Forlagsredaktion" pos="N_INDEF_SING"

Hvis vi til valideringen havde nøjedes med at udvælge materiale med løbende tekst, ville vi have opnået en meget højere korrekthed.

Underlige tegn, som særlige punkttegn fx, har taggingen naturligt nok også svært ved at fortolke.

Fejl i lemmatisering

Fejl I lemmatisering taltes ikke med I følgende tilfælde:

- I starten af et afsnit lemmatiseres det første ord konsekvent med stort bogstav. Dette er ikke korrekt, men da det er en option der kan ændres i lemmatiserens opsætning, og altså en indstilling der er valgt inden lemmatiseringen, har vi valgt ikke at tælle fejlen med her.

Ordformer der har fået tildelt forkert pos-tag, vil ofte som konsekvens have en forkert lemmatisering, fx, lim – fejlagtigt tagget V-IMP med lemma: lime. Da fejlen ligger i taggingen, kan man diskutere om den bør tælle med som lemmatiseringsfejl. Da vi her er interesserede i antal fejl i teksterne generelt, har vi talt det samlede antal fejl uanset fejlens årsag og altså fundet 26 fejl svarende til en fejlrate på 2%. Hvis man i stedet ville måle lemmatiserens performans, skulle de fejl som skyldes forkert pos-tagging, ikke tælles med. Resultatet ville så være 14 fejl i alt svarende til en fejlrate på godt 1 %. I begge tilfælde er det et godt resultat.

6.3.2 Evaluering af term-annoteringer

Evalueringen af term-annoteringerne er som sagt ikke nogen helt enkel sag. Primært fordi en formel evaluering af recall (genkaldelsesrate) ville kræve at samtlige termer i analysedokumentet blev manuelt identificeret af flere uafhængige domæneeksperter. Evaluering af precision (træfrate) ville kræve at samtlige termer i en givet delmængde af output blev manuelt evalueret på samme facon. Ud over denne resursemæssige problemstilling så er der det problem at selv domæneeksperter kan være uenige om en kandidats termstatus (jf. eksperimentet med de fire farmaceuter i Halskov (2007: 115). Endelig er der den udfordring at termer kan være flerordsudtryk, og disse er svære at identificere automatisk, hvorfor nærværende termtagger alene tager unigrammer i betragtning.

Ulempen ved denne begrænsning er naturligvis at termfragmenter kan blive tilskrevet en høj termhood. Et eksempel er kandidaten ”blev” i 1. kolonne på bilag 1. Denne tekststreng indgår med stor sandsynlighed i en række passivkonstruktioner med forskellige verber, idet der er tale om en faglig rapport, og denne teksttype indeholder erfaringsmæssigt mange passiver.

I den uformelle evaluering af termtaggerens precision (se bilag 1) har arbejdspakkens fire medarbejdere ageret som domæneeksperter. 14 tekster (to fra hvert af de syv domæner) er blevet udvalgt tilfældigt, og arbejdspakkemedarbejderne har evalueret termstatus af de 20 ordformer som er blevet tilskrevet den højeste termhood af systemet i hver enkel tekst.

Precision varierer mellem 70 % og 100 % (hvis man ser bort fra en enkelt tekst med en meget lav precision på 25 %). Den gennemsnitlige precision er 79 %. Da opgaven er vanskeligere end eksempelvis pos-tagging, så er denne precision i den bedre ende af skalaen, men en mere formel og omfattende evaluering (også af systemets recall) vil naturligvis være ønskværdig.

Den mest iøjnefaldende ”støj”, altså ikke-termerne i bilag 1, er:

1. Personlige pronominer (din, dig …)
2. Hjælpeverbet ”blev”
3. Engelske ord (acute, surgery)
4. ”Hvis”

Forklaringen på at personlige pronominer bliver tilskrevet høj termhood i en it-tekst og en økonomitekst er sandsynligvis skrivestilen i den instruerende teksttype (fx ”Åbn din browser og…”, ”Udfyld dit skattekort …”). Desuden forekommer personlige pronominer kun sjældent i referencekorpusset, da dette primært består af nyhedsartikler fra store dagblade. De engelske ord stammer sandsynligvis fra engelsksprogede abstracts i faglige rapporter. I visse domæner findes der naturligvis slet ikke danske ækvivalenter til engelske fagudtryk, så det vil faktisk kræve en nærmere analyse af konteksten at afgøre om den engelskklingende kandidat er en term eller ej.

7 Referencer:

Bourigault, Didier (1992). ”Surface grammatical analysis for the extraction of terminological noun phrases”. *Proceedings of COLING-92*.

Halskov, Jakob (2007). *The semiautomatic expansion of existing terminological ontologies using knowledge patterns discovered on the WWW – an implementation and evaluation*. Department of International Language Studies and Computational Linguistics, Copenhagen Business School. PhD Series 28.2007.

Madsen, Bodil Nistrup (1991). ”In terms of concepts ”. *Copenhagen Studies in Language*, 14: 67-91. Copenhagen Business School.