Språkteknologi för skrivande i skolan

En förstudie inom VINNOVA-projektet *Roligt, lärorikt och meningsfullt skrivande i skolan: Utveckling av digital pedagogik och skrivverktyg*

Augusti-december 2014

Ola Knutsson
knutsson@dsv.su.se
Institutionen för data- och systemvetenskap, Stockholms universitet

Innehållsförteckning

1 Inledning

Den här rapporten handlar om språkteknologi för skrivande. Med språkteknologi avses programvara som något sätt är “aktiv” med språket, dvs. en programvara som antingen analyserar mänskligt språk eller genererar det (t.ex. skapar rättningsförslag i en stavningskontroll). Detta till skillnad mot digitala lexikala resurser (t.ex. SAOL) och korpusar (stora mängder språkliga data, text och tal) som innehåller språklig information men som inte är “aktiv” med språket.

Syftet med denna rapport är att ge överblick över språkteknologi för skolan. Målsättningen har varit att presentera tre till fem mogna språktekniker som matchar tydliga behov enligt målen för projektet ”Alla kan skriva” samt identifiera en till två forskningsområden som på lite längre sikt bidrar till att uppnå projektmålen, dvs. att radikalt förbättra digitala skrivverktyg, skrivpedagogik och skrivkunskaper i skolan.

Rapporten kommer att inledas med slutsatser av förstudien i form av rekommendationer för att sedan ge en fördjupning inom de olika delområdena i snittet mellan språkteknologi och skrivande.

2 Rekommendationer

I det följande kommer jag med rekommendationer när det gäller det språkteknologiska (praktiska-tekniska) och det mer forskningsorienterade. Jag kommer också att ge rekommendationer för lämplig programvara för att börja implementera de tekniska rekommendationerna. Rekommendationerna som följer har en inbördes prioriteringsordning. De är listade enligt den ordning som bestämdes på den andra prioriteringsworkshop som hölls med projektets behovsgranskningsgrupp den 11 november 2014.

2.1 Tekniska rekommendationer

För en förklaring och för exempel på lämplig och tillgänglig öppen programvara se avsnitt 7.

2.1.1 Nyckelfrasidentifikation för att kunna ge specifik istället för generell feedback

Ett verktyg för att identifiera nyckelfraser skulle vara mycket användbart. Genom att känna igen genrebundna nyckelfraser kan ett program känna igen olika delar av en text, även vilken typ av text det är. Nyckelfraserna finns ofta i specifika delar av texten. Om programvaran har god kontroll över texten kan en mer pedagogiskt lämplig feedback ges, till exempel genom att ge specifika kommentarer på hur t.ex. en inledning lämpligen skrivs eller vad som är lämpligt att ha med i slutsats- eller diskussionsavsnitt, om vi tänker oss att det är en essä eller rapport som eleven skriver.

2.1.2 En semantisk modell att ställa innehållsorienterade frågor till

En semantisk modell, en ordrumsmodell som kan användas för att förbättra språkkontroll, för att göra bättre jämförelser av texter samt flera andra tillämpningar där en viss "förståelse" av ordens betydelse är nödvändig. Den första huvudsakliga nyttan med att använda en semantisk modell är att innehållet i texten kan börja betyda något för de bedömningar som språkkontrollen gör. Den andra huvudsakliga nyttan handlar om att när texter jämförs kan inte bara de exakta ordformerna jämföras utan även betydelsenära ord som inte finns med i den ena av de texter som kan tas med i textjämförelsen. Ett exempel skulle kunna vara att ordet "hund" finns med i den ena texten men inte i den andra där hunden betecknas med ordet "labrador" och "blindhund". Ordrumsmodellerna utgår ofta ifrån hypotesen att ord som förekommer i likartade sammanhang har likartade betydelser.

2.1.3 En utvecklad språkkontroll av typen ensemblegranskare

Det finns en omfattande forskning om språkkontroll för svenska. Flera olika regelbaserade system finns men även sådana som bygger på statistik och maskininlärning. Genom att kombinera några av dessa kan en mer omfattande bild av elevens språkriktighet ges. En ensemblegranskare som utnyttjar information från flera källor kan peka på specifika avsnitt i texten som är problematiska än andra.

2.1.4 Verktyg för textjämförelser

En vektorrumsmodell för att kunna göra bra textjämförelser, gärna kombinerad med standardverktyg för statistisk textanalys för att kunna ge mer begriplig feedback till eleven eller läraren. Kort sagt kan man säga att vektorsrumsmodellen är bättre på att göra själva textjämförelsen medan den statistiska textanalys kan påtala mer konkret vilka skillnaderna är mellan texterna är. Detta verktyg kan användas för att t.ex. påvisa textprogression eller hur nära en elevens text ligger en modell- eller måltext.

2.2 Programvarutekniska rekommendationer

Här följer ett antal programvarutekniska rekommendationer. Utgångpunkten har varit att identifiera program som bygger på öppen källkod, men programvarorna har lite olika licensmodeller som bör beaktas var för sig. Denna del har inte varit med under projektets prioriteringsworkshop i november.

2.2.1 Maskininlärning för språkkontroll

Ett sätt att utvidga täckningen (öka antalet fel som upptäcks) i en grammatikkontroll är att träna upp ett system på texter med uppmärkta fel i. Ut från ett sådant system kommer regler som känner igen vad som de feltyper som fanns med i textmaterialet. Studier har visat att sådana regelsamlingar delvis upptäcker andra fel än handskrivna regler (Sjöbergh & Knutsson, 2005).

2.2.1.1 Transformationsbaserad maskininlärning med fnTBL

I samma artikel av Sjöbergh och Knutsson (op.cit.) användes fnTBL-implementation av transformationsbaserad maskininlärning (TBL) för att generera felregler: http://www.cs.jhu.edu/~rflorian/fntbl/download.noform.html

Det finns flera andra sätt att göra detta på men detta är ett välprövat sätt. Ett bibliotek som Weka ger fler möjligheter.

2.2.2 Ordrumsmodeller för att börja komma åt texters innehåll

Det finns flera olika sätt att bygga upp ordrumsmodeller som det går att "ställa" semantiska frågor till. En del av dessa modeller kräver mycket datorkraft, vilket bör beaktas vid användning. Här följer två olika implementationer som är lämpliga att arbeta vidare med.

2.2.2.1 Semantic vectors

(http://code.google.com/p/semanticvectors/), välanvänd programvara som dels kan bygga ordrum från olika textmängder, dels kan göra textjämförelser. Bygger vidare på den välkända och öppna textsökningsmotorn Apache Lucene (http://lucene.apache.org/). Apache Lucene kan också användas för att textjämförelser a' la traditionell sökmotorsteknologi utan språkteknologiska algoritmer. Semantic vectors bör för detta ändamål vara bättre än Lucene.

2.2.2.2 Java SDM – A Java Package for Random Indexing

Martin Hassels implementation i Java (http://www.csc.kth.se/tcs/humanlang/tools.html) som bygger på Random Indexing (http://www.sics.se/~mange/random_indexing.html). Random Indexing (RI) är ett erkänt och effektivt sätt att bygga och tillämpa ordrumsmodeller.

Fördelen med JAVA SDM är att detta är en svensk "produkt" och kunskapen finns i Stockholmsområdet.

2.2.3 Statistisk textanalys

Med statistisk textanalys avser jag här programvara som gör en mer rå statistisk analys än traditionell automatisk textanalys som bygger på en lingvistik analys i någon form i grunden. Behovet av denna typ av programvara är att kunna säga något mer än att få ett numeriskt värde på ord eller texter likhet, dvs. något som en elev eller lärare kan ha nytta av. Ett enkelt exempel på detta är att verktygen kan ta fram ordfrekvenstabeller över olika texttyper, dvs. en tabell över vilka ord som är vanligast i en speciell texttyp. Ett mer avancerat verktyg skulle kunna identifiera att "begå brott" är en mer sannolik ordkombination än "göra brott" i svenskt skriftspråk, s.k. kollokationsanalys. Verktyget StringNet (se nedan) som ligger som ett lingvistiskt sökgränssnitt mot British National Corpus (BNC, se nedan) får väl ses som en kombination av dessa. Dock är StringNet just ett sökgränssnitt mot BNC och inte en programvara som kan laddas ner och köras mot valfri korpus. Det finns flera andra sätt att visa på likheter skillnader mellan texter och hur ord används, se t.ex. verktygen från Voyant Tools nedan.

2.2.3.1 Natural Language Toolkit (NLTK)

http://www.nltk.org/
Det går att göra mycket av den statistiska textanalys som beskrivs ovan med språkteknologiska programvarupaktet NLTK, se vidare här:
http://www.nltk.org/book/ch01.html.

2.2.3.2 Statistisk textanalys baserat på R

Mer avancerad processning än ovan kan göras med R-systemet för statistisk analys (http://www.r-project.org och http://www.jstatsoft.org/v25/i05/paper) som tillsammans med det omfattande kodbiblioteket WEKA för textorienterad maskininlärning ger många textanalytiska möjligheter. Ytterligare alternativ ges av tm-paketet (http://cran.r-project.org/web/packages/tm/vignettes/tm.pdf) som också bygger på R. Samtliga dessa är öppen källkod.

2.2.4 Förbättrad ordklass- och sammansättningsanalys

Den grundläggande textanalysen i många språkteknologiska system görs av en ordklasstaggare, ofta hanterar den även så kallad lemmatisering (ordens grundform matas ut). En stor utmaning för svensk textanalys är nyskapandet av ord genom sammansättningar samt låneord från engelskan. Även små förbättringar av denna textanalys spelar stor roll för slutresultatet, det finns därför anledning att alltid ha den bästa tekniken på detta område.

2.2.4.1 Ordklasstaggaren Stagger

Även om ordklasstaggaren Stagger är till synes marginellt bättre än Granska Tagger (Östling 2013) så är det rimligt att anta att Stagger skulle ge ett bättre resultat för program som bygger på ordklasstaggning som t.ex. grammatikkontroll. Stagger fungerar även bra för mer rörig text som t.ex. bloggar vilket kan vara bra i skolsammanhang där elevernas texter i många fall inte ser ut som korrekturlästa nyhetstexter. Det är också sannolikt att Stagger är bättre på att analysera sammansatta ord än vad Granska Tagger är.

http://www.ling.su.se/english/nlp/tools/stagger/stagger-the-stockholm-tagger-1.98986

2.2.5 Tekniker för nyckelfrasidentifikation

Det finns olika tekniker för att identifiera nyckelfraser i en text: det går att använda maskininlärning eller regelbaserade metoder för detta. Eftersom vi vet vilka de centrala

nyckelfraserna är så kan en god början vara att skriva regler för att känna igen dessa, men att maskininlärning sannolikt behövs i en mer långsiktig ansats. Det finns anledning att tro att elevernas nyckelfraser inte nödvändigtvis ser ut som de av reglerna eftersökta, och då blir maskininlärning ett nödvändigt medel för att hantera denna problematik.

2.2.5.1 Nyckelfrasidentifikation med hjälp av Granskas regelspråk

Granska är framförallt ett system för grammatikkontroll men regelspråket kan användas för generell textanalys. Att vidarutveckla denna del skulle kunna ge ganska stor vinster mot ganska lite arbete. Det som jag framförallt rekommenderar här är att skriva regler för att känna igen nyckelfraser, och att utveckla utdatadelens format så att det enkelt kan användas av andra applikationer genom validerad XML.

http://www.csc.kth.se/tcs/projects/granska/rapporter/rulelang20010308.pdf

2.3 Forskningsorienterade rekommendationer

I det följande kommer tre olika förslag på vägar för forskningen att ta. De olika förslagen hänger samman, men det tredje förslaget om diskurscentrerad learning analytics får ses som mest grundforskningsorienterat. Förslagen är rangordnade enlighet med de prioriteringsbeslut som togs under prioriteringsworkshoppen med projektets behovsgranskningsgrupp den 11 november 2014.

2.3.1 Hjälp eleven att lära genom andras texter

En viktig del i genrepedagogiken är att kunna visa på alternativ genom den funktionella grammatikens möjligheter att visa betydelsepotentialer. Om det går att koppla elevens text till andras texter för att visa på möjligheter skulle nog skrivandet lossna för många. Vi kan se det som en mycket avancerad ordprediktion som utifrån mikrogenrer förslår vägar vidare: – "nu har du kommit så här långt i texten – nu finns det följande möjligheter att skapa mening" eller "så här gör andra skribenter". Bör kunna kopplas ihop med plagiatkontroll på ett pedagogiskt sätt.

2.3.2 Automatisk identifikation av basgenrer-orienterade strukturer

Att automatiskt identifiera strukturer/mönster i en text och koppla till bas(mikrogenre)- och texttyper (makrogenrer) skulle göra att en större del av genrepedagogiken skulle kunna digitaliseras. Denna programvara kan t.ex. återkoppling till skribenten av vad som finns eller inte finns för att det skall bli en basgenre (se Holmlund 2009; Knapp & Watkins). En basgenre kan vara en berättelse, återberättelse, argumentation, förklaring, instruktion m.fl. Basgenrer bygger upp makrogenrer som essäer, tekniska rapporter, nyhetstexter osv. Nyckelord- och fraser är ganska straightforward, mycket fine-tuning dock för att få det bra. Mer avancerad Funktionell grammatisk analys betydligt mer avancerat att åstadkomma, men t.ex. en del grammatiska metaforer kan identifieras (nominaliseringar, substantiveringar alt. behov av grammatiska metaforer).

2.3.3 Diskurscentrerad learning analytics

Diskurscentrerad learning analytics handlar om att analysera elevens språk, texter och LMS-aktiviteter: koppla ihop en språklig analys av elevens texter med "learning analytics" (analys av alla andra aktiviteter i ett VLE/LMS, och annat som kollaborativt lärande – hur mycket deltar studenten i online-diskussioner osv). En textuell analys behövs också – vilken typ av texter skriver egentligen eleven (jämförelse mot modelltexter), för self-assessment, och för lärares återkoppling och bedömning. Det senare kan kanske få upp ögonen hos en del lärare att en del elever egentligen skriver bra men deras många språkfel står i vägen för att läraren skall kunna se detta. Denna fråga måste utveckla en del av tekniken ovan men har ett mer allmänvetenskapligt fokus.

2.4 Kopplingar mellan rekommendationerna

Det finns en rad kopplingar mellan de olika rekommendationerna. Följande tabell visar de viktigaste kopplingarna som finns.

Tabell 1. Kopplingar mellan rekommendationer. X=stark koppling, x=koppling

REKOMMENDATION	fntbl	Semantic vectors	JavaSDM	NLTK	R	Stagger	Granska
Nyckelfrasidentifikation		x	x			X	X
Semantisk modell		X	X				
Utvecklad språkkontroll	X	x	x			X	X
Textjämförelser		X	X	X	X	x	x
Lära genom andras texter		X	X	X	X	X	
Identifikation av basgenrer		X	X	X	X	X	X
Diskurscentrerad learning analytics		X	X	X	X	X	X

3 Nyckelforskare

Att bedöma vilka forskare som är nyckelforskare är svårt men en del skriver t.ex. mer än andra, skapar intressanta och innovativa system osv. Min bedömning utgår ifrån forskare som varit med länge och i en rad olika projekt. Här följer en lista på några sådana nyckelforskare och ett av deras styrkeområden:

- **Jill Burstein, essay scoring**: Director of the NLP Group in Research & Development at Educational Testing Service in Princeton, New Jersey, USA.
- **Detmar Meurers, WERTi – Working with English Real Texts:** Full Professor of Computational Linguistics and head of the Theoretical Computational Linguistics group at the Department of Linguistics of the University of Tübingen.
- **Claudia Leacock, Microsoft ESL Assistant**, Research Scientist III, CTB McGraw-Hill, USA.
- **Martin Chodrow**, **grammar checking**, Professor of Psychology and Linguistics, Hunter College and the Graduate Center of CUNY, USA.
- **Trude Heift**, **intelligent language tutoring**: Professor at the Department of Linguistics, Simon Fraser University, Canada.

4 Nyckeltidskrifter och nyckelkonferenser

Ämnets tvärvetenskapliga karaktär gör att relevanta artiklar publiceras i en rad olika tidskrifter. Även en mängd olika konferenser spelar stor roll för publicering av forskningsrön inom ämnet.

4.1 Viktiga tidskrifter

1. ReCALL: http://www.eurocall-languages.org/publications/recall

2. Computers and education, An International Journal:
http://www.journals.elsevier.com/computers-and-education/

3. Natural language engineering:
http://journals.cambridge.org/action/displayJournal?jid=NLE
4. Journal of Writing research: http://www.jowr.org
5. Computer Assisted Language Learning:
http://www.tandfonline.com/action/journalInformation?journalCode=ncal20#.VJfoJJ3oA

4.2 Viktiga konferenser

1. **Workshop on Innovative Use of NLP for Building Educational Applications**:
http://www.cs.rochester.edu/~tetreaul/naacl-bea10.html
2. **EuroCALL**: http://www.eurocall-languages.org/conferences
3. **Annual Meetings of the Association for Computational Linguistics (ACL)** (inkl. North American chapter of ACL, European Chapter of ACL): http://www.aclweb.org/website/acl
4. **CoNLL**, maskininlärningskonferens som alltid har en s.k. shared task, flera gånger under senare år har det handlat om grammatical error detection: http://ifarm.nl/signll/conll/

5 Sju tillämpningsområden och hur de valts ut

Delrapporten av "Språkteknologi för skrivande i skolan" (Knutsson, 12 aug 2014) skickades ut på remiss till projektets behovsgranskningsgrupp. Efter en prioriteringsworkshop med behovsgranskningsgruppen den 13 augusti 2014, som utgick från de sju tillämpningsområden som presenterades under workshopen prioriterades ett antal tillämpningsområden ut enligt följande:

Område 1: Automatisk språklig och textuell formativ bedömning
Detta område handlar om program som gör en språklig, innehållsmässig eller argumentationsorienterad bedömning av en text. Det kan handla både om att avgöra vad som anses tillhöra språket eller ej (språkkontroll) till att ge omdömen om argumentation och hela texters uppbyggnad. Någon individuell hänsyn till varje skribent tas normalt inte utan alla texter behandlas på samma sätt, däremot finns det programvara specifikt utvecklad för olika målgrupper t.ex. för skribenter med engelska som andraspråk. Området har utvecklats under lång tid (se t.ex De Smedt, 2005)

Området är prioriterat som en delmängd av Område 2.

Område 2: Intelligent language tutoring
Grundidén med intelligent language tutoring är att modellera elevens språknivå och domänkunskaper samtidigt som eleven arbetar med olika språkuppgifter som datorn förser eleven med. Med "modellera" avses att samla data om eleven och att analysera dessa. Utifrån modellen av eleven ger programmet eleven relevant feedback till stor del baserat på teknologier från Område 1 ovan. Den stora skillnaden mot Område 1 är att elevens språkkunskaper modelleras, och en individanpassning därmed kan ske. Historiskt har detta område varit kritiserat på grund av sin tro att avancerad artificiell intelligens finns bakom hörnet.

Området är prioriterat.

Område 3: Verktyg för elever med speciella behov
Det finns en rad olika språkteknologiska tillämpningar som stödjer elever med speciella behov såsom talsyntes, skräddarsydda språkgranskningsprogram för dyslektiker m.fl. Det finns också programvara som försöker känna språkliga drag i elevtexter som indikerar att det sannolikt finns speciella behov hos skribenten.

Den del av detta område som handlar om att med automatiska medel hjälpa till med att identifiera elever med speciella behov utifrån deras skrivande är prioriterat. Denna kommer att placeras under Område 2, och mer specifikt under användarmodellering (User modelling).

Område 4: Korpusar för lärande (Use of corpora in educational tools)
Att använda autentiska språkexempel i stor mängd är ett sätt att dra nytta av korpuslingvistiken. Det verkar dock svårt att bedöma vilken nivå i utbildningssystemet som korpusar kan användas. Eller handlar det om pedagogisk metod? Korpusar är en rik källa för att snabbt och storskaligt få tillgång till hur språket verkligen användas, och därmed bjuda in till ett spännande utforskande av olika texttyper.

Området är ej prioriterat.

Område 5: Verktyg för lärare och/eller testutvecklare
När det gäller verktyg för lärare för att skapa övningar för eleverna finns det många olika sätt, från lucktexter till att skapa instuderingsfrågor från vilken text som helst, men här handlar det ofta mer om att utveckla språkfärdigheter än skrivande.

Området är ej prioriterat.

Område 6: Skrivteknologi (ej nödvändigtvis med stöd av språkteknologi)
Det finns en lång tradition av att stödja skrivande med olika datorhjälpmedel. Det handlar mest om avancerade ordbehandlare på senare år inte minst för kollaborativt skrivande. Flower och Hayes klassiska studier som delade upp skrivprocessen i delarna planering, textproduktion, och granskning som samverkade genom en övergripande funktion monitoring har forskare och utvecklare fokuserat på. Stödverktyg för dessa delprocesser finns utvecklade (t.e.x det enkla dispositionsläget i MS Word).

Detta är Invigos fokusområde, och vi har därför valt att inte prioritera detta område i denna studie.

Område 7: Teknikstött genrepedagogisk angreppssätt
Genrepedagogiken vilar på flera olika grundläggande teoretiska ramverk, framförallt Halliday, Bernstein och Vygotskij. Man kan väl också ganska förenkla säga att dessa ramverk stödjer genrepedagogikens tre ben på lite olika sätt: den pedagogiska metoden att explciti försöka öppna upp samhällets texter för alla medborgare (Bernstein), att visa och förklara hur texter är uppbyggda genom Hallidays språkvetenskapliga ramverk som ges av Systemic Functional Linguistics (SFL) samt att ge stöd (scaffolding-stöttning) för elevens arbete med dessa texter (Bruner; Vygotskij). Genrepedagogiken (åtminstone Sydneyskolan), som har utvecklats av flera olika forskare men Martin och Rose kan nämnas som extra viktiga, tar hjälp av systemisk funktionell grammatik (SFG) för att visa och lära eleverna vilka olika språkliga möjligheter de har i olika skrivsituationer. Notera att det finns konkurrerande genrepedagogiska angreppssätt som The New Rhetoric och ESP (English for Special Purposes), men vi håller oss till den så kallade Sydneyskolan här.

För lärare som behärskar den systemiska funktionella grammatiken fungerar genrepedagogiken i klassrummet men många lärare upplever svårigheter med att lära sig dess ganska avancerade begreppsapparat och praktik. Dagens språkteknologi kan komma en bit på väg som hjälpmedel för

att identifera tänkbara valmöjligheter i elevernas texter för olika genrer, t.ex. mellan att använda verb eller substantiv (grammatiska metaforer), för att röra sig från en mer vardaglig text till en mer akademisk. Eller ge stöd för det textuella, nämligen hur informationen i satserna kan bindas ihop på olika sätt osv. Ytterligare någon analysnivå "ovan" ordklasserna borde vara möjligt att identifiera automatiskt. I ett längre perspektiv borde det även vara möjligt stödja samtliga SFLs tre metafunktioner (textuella, ideationella, interpersonella), se vidare Holmberg 2009 för en bra introduktion till genrepedagogik. När det gäller vilka delar som språkteknologin kan stödja är det mest uppenbara att ge lärare och elev stöd med att identifiera språkliga drag som kan mappas mot det metaspråk som SFG utgör, men även teaching-learning cycle (genrehjulet, cirkelmodellen) kan stödjas med programvara genom att ge struktur för dess delar samt ge stöttning under hela processen när läraren inte finns tillgänglig.

Området är prioriterat.

6 Område 1: Automatisk språklig och textuell formativ bedömning

Två typer av programvaror är mycket tydliga inom detta: språkkontroll och automatiserad betygssättning av uppsatser.

6.1 Språkkontroll (stavnings- och grammatikkontroll)

Det är lätt att förledas att tro att stavningskontroll med avseende på detektion, korrektion och orsak till felet är ett löst problem efter mycket användbara stavningskontroller

Ett större lexikon medför att även esoteriska ord kommer med och därmed blir felstavade ord (utifrån kontexten), jämför:

Det som var utmärkande var den parantes vackra kostym.

I programmeringsspråket LISP saknas det ofta en parantes.

En annat exempel som visar att det finns många svårrättade fall för en stavningskontroll, jämför: *cykolåggiska* med *psykologiska*

Vad är orsaken till felen? Förslag till orsaksmodellering finns i Deorowicz och Ciura (2005), och det är ett knepigt område. Frågan vi bör ställa oss är vad det pedagogiska värdet är av en stavningskontroll som kan förklara vilken eller vilka stavningsregler som skribenten brutit mot? Det kan t.ex. ställas mot att lägga resurser på att ranka ersättningsförslagen på ett avancerat och mestadels sätt (jmf. Kann om Stava)

De första grammatikontrollerna var regelbaserade, och mig veterligen är de flesta kommersiella grammatikkontrollerna också regelbaserade. Senare års forskning om språkkontroll har dock framförallt handlat om angreppssätt baserade på maskininlärning. I många fall är det specifika feltyper som angrips var för sig som t.ex. felaktig preposition eller felvald artikel. För att nå riktigt bra resultat används i flera fall semantisk information av olika slag. Kochmar och Briscoe (2014) uppnår t.ex. mycket intressanta resultat, deras system hittar fel av typen:

big*/large quantity,
big*/great importance.

Här handlar det alltså om felaktiga ordkombinationer av adjektiv och substantiv. Kochmar och Briscoe angriper delvis samma problem som t.ex. Östling och Knutsson (2009), men angreppssättet är inte direkt korpusbaserat utan bygger framförallt på en semantisk modell över adjektiv- och substantivkombinationer utvecklad baserad på korpusdata. Detta gör att Kochmar och Briscoes ansats blir mer generell, och gör att ett fenomen som inte påträffats tidigare i en korpus trots allt kan identifieras som problematiskt. Det är ju faktiskt så att många språkliga fenomen sett som lexikogrammatiska instanser inte har "setts" förut. När detta att utvärdering av grammatikkontroll så är en tumregel att hög precision är att föredra eftersom falska alarm förbryllar och stör användaren. En tydlig brytpunkt blir om precisionen går under 0.5 (50 %) då är det alltså fler falska alarm än korrekta alarm från felklassificeraren (grammatikkontrollen). Min egen åsikt är att denna gräns måste vara högre för osäkra språkbrukare. Kochmar och Briscoe kommer över denna gräns, med en lägsta täckning (hur många fel som hittas) på 58 % och en lägsta precision ("goda alarm"/"alla alarm") på 62 %. Detta måste dock ställas mot värdet av att kunna upptäcka dessa fel. För en språkbrukare som kan bedöma alarmen korrekt är detta mycket värdefull information som dock inte går att att lita blint på. Kanske skall klassificeringen av de olika feltyperna förses med olika grader av sannolikheter för att påvisa säkerhet vs. osäkerhet i detektionen?

6.1.1 Är en svensk version möjlig?

Är det möjligt att med små medel bygga en svensk version av Kochmar och Briscoes system? Eftersom det handlar om maskininlärning kan vi lätt tro att endast små medel måste till, men är det verkligen så? Följande tekniker och resurser används för att bygga felklassificeraren:

Stor ordklasstaggad kontrollerad korpus: British National Corpus (http://www.natcorp.ox.ac.uk) består av 100 miljoner ordklasstaggade ord i olika texttyper (skriftspråk och talspråk), korpusen är balanserad. Någon sådan kvalitetsprodukt för svenska finns inte, men Parole (ca 19 miljoner ord, http://spraakbanken.gu.se/parole/ och http://spraakbanken.gu.se/parole/parole_material_kommentar.phtml) och SUC 3.0 (ca. 1 miljon ord, http://spraakbanken.gu.se/eng/resources/suc) är en bit på väg.

Ordrumsmodell: Mer eller mindre språkoberoende programvara finns här, och det finns inga uppenbara problem att gå över till svenska om korpusen finns. Olika "mätningar" i ordrummet (den semantiska modellen) görs för att särskilja korrekta och inkorrekta ordkombinationer, baserat på en antal semantisk ledtrådar (s.k. features) om vad som "bör" skilja dessa åt.

Felklassificeringen: Felklassificeringen sker med hjälp av Decision trees-verktyg från NLTK (open source, språkteknologimoduler för Python, http://www.nltk.org), dessa är mer eller mindre språkoberoende. Verktyget använder utdata från den semantiska modellen som byggs upp av ordrumsmodellen som tränas på BNC (orden i BNC har lemmatiserats, dvs. alla ord är nerförda på sin grundform, exempelvis *bil, bils, bilen, bilens, bilar, bilars, bilarna, bilarnas* --> ***bil*** som ett svenskt exempel). Själva felklassificeringen möjlig på svenska men resultatet sannolikt sämre eftersom en svensk BNC inte finns.

6.1.2 Öppna frågor

Det finns en rad öppna frågor inom detta område, och här följer några av de viktigaste:

- Hur mycket bättre blir egentligen en ensemblegranskare som kombinerar olika språkkontroller till en? Vilka problem finns det med ett sådant angreppssätt? En praktisk förstudie från 2009 ger en del ledtrådar (Norelius, 2009).

- Vilka feltyper (särskrivningar, kongruens etc.) är viktigaste att komma åt för svenska idag? Empiriska studier krävs. En rangordnad lista skulle vara önskvärt, och utifrån denna kan beslut om teknologi tas.
- Vilka nivåer för täckning och precision är rimliga för respektive feltyp? Här finns det en del forskning från tidigare projekt (Knutsson, 2005)
- Hur omfattande är arbetet att utveckla en stavningskontroll som modellerar orsaken till ett stavfel, och som kan "redogöra" för detta?

Räcker de svenska fria språkresurserna till för att bygga en lika bra feldetektor som Kochmar och Briscoes (2014) gjort?

6.2 Programvara för språkkontroll

6.2.1 Stavningskontroll

- Oribi (http://www.oribi.se)
- Stava (KTH)

6.2.2 Grammatikkontroll

- Microsoft Words stavnings- och grammatikkontroll för flera språk
- Grammatifix (lingsoft.fi)
- Granska (KTH)
- ProbGranska (KTH)
- SnålGranska (KTH)
- Lightside Revision Assistant (http://lightsidelabs.com)
- Microsoft Research ESL http://research.microsoft.com/en-us/projects/msreslassistant/Konkordans-views för hur felet ser ut och hur det korrigerade textsekvensen ser ut i kontext från webbexempel. Av någon anledning finns inte ESL längre som webbtjänst.
- Languagetool (https://languagetool.org/): Den fria programvaran languagetool.org ger möjlighet att skriva error patterns

6.3 Automated Essay Scoring (AES)

Program som automatiskt kan sätta betyg på en uppsats är på stark frammarsch, och det finns flera kommersiella aktörer på detta område, eftersom AES som verktyg kan spara pengar, och passar dessutom bra ihop med den standardiseringsiver som växer fram inom skolväsendet.

Ett företag som sysslat länge med detta är Educational Testing Service (ETS), och de har utvecklat Criterion där en viktig del är AES. Vad gör egentligen ett verktyg som Criterion? Hur långa kan texterna vara? Kan läraren påverka bedömningen? Hur ser ett betyg/score ut? Kan vi lita på bedömningen? Vad finns det för kritik mot dessa verktyg? Det finns många frågor, och olika svar beroende på vem som svarar. En allmän kritik finns mot den här typen av verktyg är att de bygger på en modell av skrivande, och den blir normerande. Studenterna skriver för att bli testade, och andra syften blir underordnade.

6.3.1 Criterion

Programmet Criterion (http://www.ets.org/criterion/) som är utvecklat av amerikanske Educational Testing Services (ETS) innehåller flera olika intressanta skrivverktyg för eleven/studenten såsom assessment, formativ bedömning, skrivinstruktionsmaterial samt kommunikationsverktyg för att underlätta samarbete mellan elev och lärare. Det som kanske väcker mest uppmärksamhet är att en mängd olika typer av essäer/uppsatser kan betygsättas automatiskt med ett betyg mellan 1 och 6 (högst), se Figur X och Y för exempel på vad dessa betyg

motsvarar i kvalitativa termer[1]. Det verktyg som gör detta heter e-rater och innehåller programvara för att undersöka tio olika aspekter av en text (Burstein, 2013)[2]:

Följande parametrar undersöks och viktas sedan ihop till ett betyg mellan 1 och 6:

a. grammatical errors (number of errors/essay length)

b. word usage errors, t.ex. there/their (number of errors/essay length)

c. mechanics errors, t.ex. spelling (number of errors/essay length)

d. presence of essay-based discourse elements: att tes, stödjande idéer, argument, slutsats och likartade element finns med i texten. (number of required discourse elements)

e. development of essay-based discourse elements: hur mycket studenen utvecklar elementen i d, och detta mäts i hur långa dessa avsnitt är. (average length of discourse elements/essay length)

f. style weaknesses (number of style diagnostics/length)

g. content vector analysis, comparing the essay to the set of essays: enkelt förklarat så jämförs essäns vokabulär mot manuellt rättade essäer i databasen för att se var likheten är störst. (score assigned to essays with similar vocabulary)

h. content vector analysis, comparing the essay to the set of essays with a score of 6:
enkelt förklarat så jämförs essäns vokabulär mot manuellt rättade essäer med betyget 6 i databasen för att se var hur stort/litet överlappet i vokabulär är. (similarity of vocabulary to essays with score 6)

i. average word length

j. a word-frequency based feature: denna baseras

Programmet e-rater kan bygga upp så kallade topic-specific models där även innehållet granskas genom parametrarna g och h ovan. Topic-specific models bygger på att det finns en stor mängd rättade uppsatser om ett speciellt ämne. Om det inte gör det kan så kallade generic models skapas som inte tar hänsyn till vad texterna handlar om utan får förlita sig på strukturella parametrar.

[1] En intressant användning: Some institutions use the Criterion service scores for exit testing — combining a Criterion service score with the score from a reader in the same way they combine scores from two different readers. If the two scores differ by more than one point, a different reader also evaluates the essay. Some institutions use the Criterion service for benchmark testing, assigning the Criterion service-scored essays at specified points during an academic term.

[2] I en artikel av Burstein, Chodorow och Leacock (2004) finns även #word types/#word tokens och det totala antalet ord med som parametrar/vikter.

Figur 1: De tre verktygen som ingår i Criterion är Instructional, Assessment och Communication.

▪ Clearly states the author's position, and effectively persuades the reader of the validity of the author's argument. ▪ Well organized, with strong transitions helping to link words and ideas. ▪ Develops its arguments with specific, well-elaborated support. ▪ Varies sentence structures and makes good word choices; very few errors in spelling, grammar, or punctuation.

Figur 2. Beskrivning av en uppsats med högsta betyget (6)..

▪ Little effort is made to persuade, either because there is no position taken or because no support is given. ▪ Lacks organization, and is confused and difficult to follow; may be too brief to assess organization. ▪ Lacks support. ▪ Little or no control over sentences, and incorrect word choices may cause confusion; many errors in spelling, grammar, and punctuation severely hinder reader understanding.

Figur 3. Beskrivning av en uppsats med lägsta betyget (1).

6.3.2 Semantiskt avancerad konkurrent till Criterion: Intelligent Essay Assessor

Intelligent Essay Assessor (IEA) från Pearson bygger framförallt på Latent Semantic Analysis som är en ordrumsmodell som bygger på att modellera en essäs vokabulär. Systemet har också en modul som inte bygger på LSA, och det är den som kan ge återkoppling på stavning och grammatik.

6.3.3 Andra konkurrerande Essay scoring system

Project Essay Grader (PEG): det första AES-systemet
MY Access! (Vantage Learning) med IntelliMetric.
Robert Öslings et als essay scorer är mig veterligen den enda för svenska[3].
LightBox (http://lightsidelabs.com)
Writing Roadmap (CTB McGraw-Hill)

7 Område 2: Intelligent language tutoring

Ett intelligent tutoring system består normalt av tre kärnmoduler (Gamper & Knapp, 2002):

1. en expert-modul som hanterar domänkunskapen
2. en inlärarmodul som hanterar elevens domänkunskap
3. en tutor-modul som hanterar tutor-strategier och lärandemål.

Med hjälp av dessa moduler kan ett intelligent tutorsystem ge specifik återkoppling till en elev. Men eftersom det krävs domänkunskap så blir sammanhangen begränsade. Många av de lite äldre systemen bygger på en pipelinearkitektur med olika språkteknologiska moduler som ligger i följd, och där feedback utlöses genom att studenten producerar något som stämmer överens med något som kan förutsägas av systemet, eller att en språkkontroll hittar ett fel. Förutsägelsen bygger på att det är begränsade aktiviteter som studenten ägnar sig åt, och studenten förses med ett ordförråd som skall användas (domänkunskapen). Modernare system försöker komma bort ifrån detta, och bygger mer på att söka igenom en mer fri språkproduktion, och utifrån identifierade mönster ge studenten återkoppling (Amaral, Meurers & Ziai, 2011). Dessa fungerar ungefär som ett informationsextraktionssystem som letar efter specifika fakta och händelser i en textmassa.

Ett system som verkar intressant och som gör en enklare användarmodellering baserat på en demografisk enkät är SAT-systemet, Andersen et al (2013). Det är specialgjort för skribenter med B2-nivå enligt Common European Framework of Reference for Languages (http://www.coe.int/t/dg4/linguistic/source/framework_en.pdf). Användarmodelleringen är då i stort gjord på förhand, vilket verkar vettigt, och den bygger dessutom på en välutecklad modell. Återkopplingen i SAT ges på tre olika nivåer:

1. ett övergripande omdöme
2. poäng för varje mening, vilket möjliggör markering (highlightning) av bra stycken resp. de stycken som kräver mer arbete
3. specifika kommentarer på lokala problem inkluderande stavning och ordval.

SAT skall enligt författarna vara unika i världen med system genom att bedöma en relativ kvalitet hos olika meningar. Det skall enligt Andersen et al (2013) vara ett mer pedagogiskt sätt att ge

[3] http://www.ling.su.se/english/nlp/tools/automated-essay-scoring/automated-essay-scoring-1.130608

feedback än det traditionella sättet att markera felen i texten. Visualiseringen skall också vara viktig enligt författarna. Vad är egentligen tutoring i detta system? Den relativa feedbacken som ges är väl kanske mer i linje med en vad en lärare skulle göra speciellt när det utgår ifrån en specifik språklig kunskapsnivå med beskrivningar av vad språkbrukaren då kan läsa, skriva, lyssna på för typ av texter.

7.1.1 User modelling, learning analytics och språkanalys

Learning analytics är ett stort område i sig och kan väl delvis ses som en modern form av user modelling, även om det ofta är koppla till stora grupper, och att predicera framtida studieresultat. Förenklat kan man säga att området handlar om att analysera alla aktiviteter som sker i de virtuella lärmiljöerna (Sakai, Moodle, Fronter osv.) och finna olika samband för att påvisa vad studenterna verkar lära sig av. Ett nytt spännande fält som försöker koppla en språklig analys till Learning analytics kan kallas Discourse-centric Learning Analtyics, och det finns en årlig workshop för detta DCLA ((http://solaresearch.org/conferences/lak/lak13/dcla13/ och , och de har följande "mission statement":

Devise and validate analytics that look beyond surface measures in order to quantify linguistic proxies for deeper learning.

8 Område 7: Teknikstött genrepedagogiskt angreppssätt

Det finns inte många språkteknologiska forskare eller praktiker som tar avstamp i någon genreorienterad syn på text och skrivande, men i en artikel av Moreale och Vargas-Vera (2004) beskrivs ett system som identifierar nyckelfraser som indikerar argumentation utifrån genreorienterade synsätt på texters uppbyggnad. Detta system presenteras nedan som The student essay viewer (SEV).

8.1 The student essay viewer

Med utgångspunkt i Swales (CARS-modellen), Teufel och Hyland (genrepedagogik för andraspråksinlärning) skapar de en taxonomi för vad som bygger upp argumentationen i en studentessä, och hur olika kategorier kan knytas till specifika konkreta nyckelfraser, exempel på dessa finns i kolumn 3 i Tabell 1 nedan. Taxonomins kategorier kan användas för att “visa” hur texter är uppbyggda samt ge eleverna återkoppling vilka delar av en text som de har med samt vilka de bör arbeta mer med. Om ett datorprogram kan identifiera dessa kategorier i en text så läraren väsentlig avlastas och eleven kan få betydligt mer stöd under själva skrivandet. För läraren kan ett sådant program ge en snabb bild av vad eleven behöver träna mer på, och vilka delar som behöver ytterligare instruktion.

Tabell 1. A Taxonomy for Argumentation in Student Essays (Moreale & Vargas-Vera, 2004)

Category	Description	Cue phrases (examples)
DEFINITION	Items relating to the definition of a term. Often towards the beginning. IS_ABOUT, COMPARISONS	is about, concerns, refers to, definition; is the same; is similar /analogous to;
REPORTING	Sentences describing other research in neutral way	"X discusses", "Y suggests", "Z warns"
POSITIONING	Sentences critiquing other research VIEWPOINTS	"I accept", "I am unhappy with", "personally";
STRATEGY	Explicit statements about the method or the textual section structure of the essay	"I will attempt to", "in section 2"
PROBLEM	Sentences indicating a gap or inconsistency, question-raising, counter-claiming	"There are difficulties", "is problematic", "impossible task", "limitations"
LINK	Statements indicating how categories of concepts relate to others TAXONOMIC, EVIDENCE, CAUSAL	"subclass of", "example of", "would seem to confirm", "has caused"
CONTENT/ EXPECTED	Any concept that the tutor expects students to mention in their essay. Tutor-editable	Essay-dependent
CONNECTORS	Links between propositions may serve different purposes (topic introduction, support, inference, additive, parallel, summative, contrast, reformulation)	"With regard to", "As to", "Therefore", "In fact", "In addition", "Overall", "However", "In short"
GENERAL	Generic association links	"is related to"

Vilka språktekniker använder de då? Ja, initialt prövades ett ganska avancerat informationsextraheringssystem (lingvistisk analys och maskininlärning) men resultaten var inte tillräckligt bra, så ett mer pragmatiskt angrepssätt användes. Det bygger på reguljära uttryck och mönstermatchning av de nyckelfraser (cue phrases) som finns i tabellen ovan. När analysen är gjord kan den visas i ett grafiskt gränssnitt (Student Essay Viewer, SEV). Tanken är att texter med många "highlights" av nyckelfraser innehåller mer argumentation och innehåll, och därmed bör ha ett högre betyg. Förutom färgkodade markeringar (highlights) i texten så räknas förekomsterna av de olika kategorierna samman och presenteras i en tabell nedan elevens texten. SEV kan förutom att visa Moreales och Vargas-Veras kategorier även visa Swales och Hylands kategorier. De går också att visa en eller flera kategorier i taget. Morealse och Vargas har utvärderat sitt verktyg på tolv betygsatta essäer skrivna av doktorander, och de fann en korrelation mellan betyg och totala antalet annoteringar som SEV gjorde. Den kategori som var viktigast för korrelationen mellan

betyg och kategori, var "positioning" som visar hur studenten förhåller sig till andra källor. Med andra ord det är viktigt att kunna förstå, kritisera och skriva om vad andra har gjort för att få ett högt betyg, åtminstone enligt de lärarna har satt betyg i denna undersökning. Sammanfattningsvis är det viktiga att SEV kan dels ge eleven återkoppling på hur väl genren "täcks" in, vilka delar som saknas och borde vara med. För läraren ger ett verktyg som SEV möjlighet att snabbt få en överblick över en text, och också ge en påminnelse om bedömningens olika delar. Nu har inte Morelae och Vargas-Vera utgått från SFG men något på liknade nivå bör vara möjligt att åstadkomma baserat på funktionell grammatik, texters särdrag och deras koppling till olika mikrogenrer (berättelse, återberättelse, beskrivning, ställningstagande, m.fl.).

8.1.1 Är en svensk version av SEV möjlig?

Ja, i alla högst grad, att identifera nyckelfraser av den typ som görs i SEV kan ganska enkelt göras med regler i Granska. Det som krävs är en efterbehandling av utdata från Granska, räkna kategorier, "highlighta" konstruktionerna i texten osv. (Går Willes arbete att återanvända för detta?).

8.2 Andra system i en svensk kontext

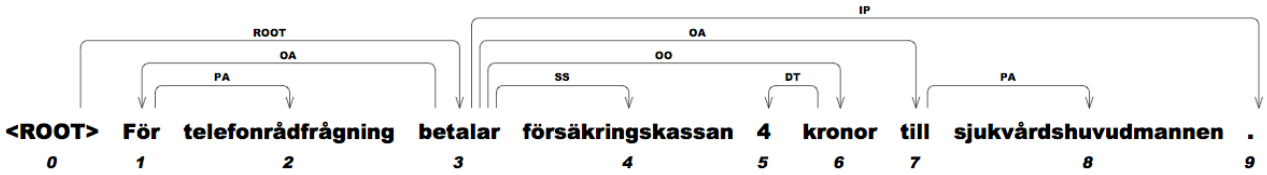
Ett annat exempel på hur språkteknologi kan stödja ett genrepedagogiskt arbetssätt i klassrummet görs i en studie av Karlström och Lundin (2013) där de använder en sk. ordklasstaggare som märker upp t.ex. alla verb i studenternas texter för att arbeta med grammatiska metaforer (nominaliseringar) i språkmiljön Grim (Knutsson et al, 2007). Ordklasstaggning är ett språkteknologiskt område som är mycket moget och det går att lita på. De fel som görs är sammanblandning av specifika "smala" ordklasser, och dessa kan då helt enkelt undantas "skrivövningen" (Nerbonne, 2007). En bra utgångspunkt för olika genrepedagogiska övningar är Knapp och Watkins (2005).

8.3 Öppna frågor och forskning att gräva djupare i

Jag hittar inte någon som egentligen har byggt vidare på Moreale & Vargas-Vera (2003; 2004) pragmatiska angreppssätt för att bygga programvara utifrån någon form av genrepedagogiskt angreppsätt. Det finns inte heller någon forskningsmässig fortsättning på Karlström och Lundin (2012) förutom Kalborg, Knutsson och Blåsjö (2014, submitted) som dock är mer på den koncepteulla nivå om hur de olika stegen i genrehjulet/cirkelmodellen skulle kunna se ut i en digital miljö.

Det finns dock en del andra mer ambitiösa spår, som gruppen runt Biber (2007) som är mer rent korpuslingvistiska men vars resultat borde kunna vara användbara i en applikation lik SEV (se ovan). Honnibal med medforskare (Honnibal & Curran, 2007) har gjort en del arbete med att utveckla SFG-annoterat material (SFG= Systemic Functional Grammar; SFL= Systemic Functional Linguistics) på redan existerande välkontrollerade resurser som Penn Treebank. En trädbank är en samling analyserade meningar vars syntaktiska struktur har representerats som träd (se Figur 1). Mer nutid forskning försöker bygga vidare på att stora framsteg har gjorts med automatisk analys av dependensgrammatik, och hur dessa syntaktiska strukturer kan omvandlas till SFG (Yan 2014; Costechi, 2013). Vad blir då nytta med denna analys för genrepedagogiken? Vilka frågor kan ställas till en sådan analysen av en mening, en text?

För svenska: En automatisk analys av texter utifrån en dependensgrammatik skulle för svenska kunna motsvaras av MALT-parsern (Nivre et al 2007, , och det finns en svensk trädbank (Swedish TreeBank, http://stp.lingfil.uu.se/~nivre/swedish_treebank/). 1,4 miljoner tokens (ord+skiljetecken).



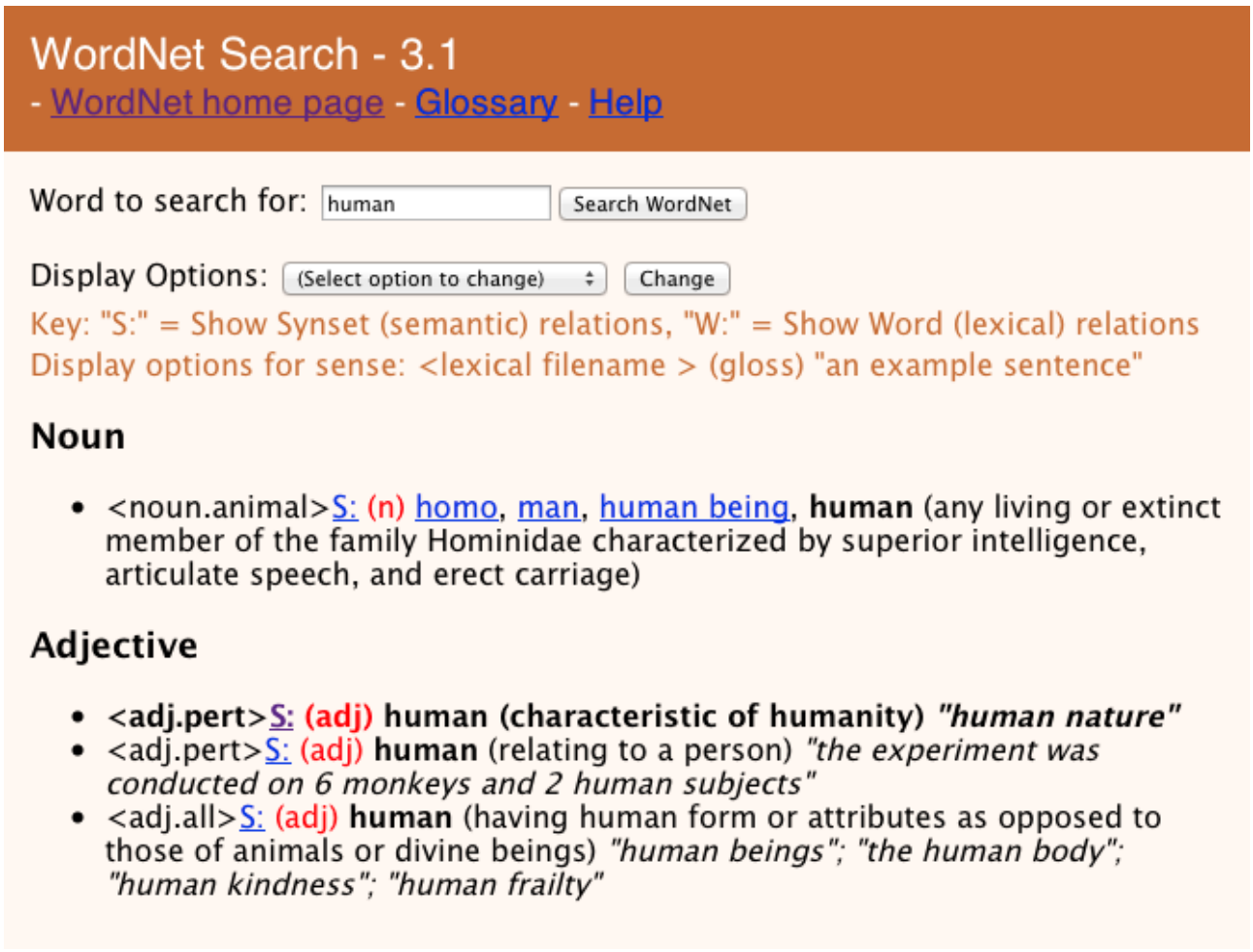
Figur 1: Ett depedensanalyserat träd från The Swedish Treebank. Root-elementet (ROOT) pekar på satsens viktigaste ord, nämligen verbet "betalar" som resten av orden i satsen är beroende av.

8.4 Framtida forskningsfrågor

Hallidays komplexa och semiotiska språksyn gör att annoteringen av ett språkmaterial är en mycket utmanande uppgift – detta innebär i praktiken att endast små mängder text har annoterats i respektive forskningsprojekt. Storskaliga manuella ansatser saknas mig veterligen. Att bygga upp ett stort SFG-annoterat material är ett enormt projekt, och en mer framkomlig väg skulle vara det som Honnibal och Curran (2007) samt Yan (2014) och Costechi (2013) försöker att göra, nämligen att utnyttja resurser och verktyg som redan existerar och sedan försöka konvertera annoterade data till SFG-annoterade data. Det verkar också vara så att mindre steg på vägen mot en mer fullständig SFG-analys skulle kunna vara användbart som ett stöd för genrepedagogik (se Karlström och Lundin, ovan). Vad skulle en ytparser (se nedan) kunna göra för nytta om den grupperar orden efter frastillhörighet som ju är en del i en SFG-analys? Vad kan en sådan analys "berätta" om en text? Eller en dependensparser som annoterar ordens syntaktiska funktioner, och sedan mappar dessa mot SFG-kategorier? Ett sådant system skulle i så fall vara det första i sitt slag för svenska. En sådan SFG-orienterad analys skulle vara långt ifrån fullständig men ändock sannolikt användbar.

9 Språkteknologiska resurser och programvara

I det följande kommer ett antal resurser och programvara som har vettiga licenser och kan anses vara state-of-art (ung. bäst just nu) lyftas fram för svenska. Vad är då syftet med att lyfta fram en state-of-the-art lista över generella tekniker och verktyg som är open source oftast och fungerar för svenska? En majoritet av de språkteknologiska tillämpningarna som utvecklas kräver omfattande "generella" resurser som t.ex. WordNet (ett semantisk lexikon, http://wordnet.princeton.edu, se Figur 2). WordNet kan användas för att upptäcka ordrelationer genom så kallade synsets; delar av språkets hierakiska struktur kan spåras med hjälp av ett sådant lexikon. Penn TreeBank är ett annat exempel på en generell resurs (en trädbank se ovan, och http://www.cis.upenn.edu/~treebank/), vilket gör att det blir mer eller mindre omöjligt utan mycket omfattande ekonomiska resurser att "porta" tillämpningar som bygger på liknande resurser till svenska. Wordnet och Penn Treebank är också språkresurser som kräver mycket resurser för att utveckla för svenska. Detta medför att vissa mycket lovande tillämpningar inte passar i det här projektet för att grundförutsättningarna saknas.



Figur 2. En exempelsökning i WordNet för ordet 'human'.

9.1 Morfologisk analys och generering

Morfologisk analys handlar om att analysera orden och deras delar. En morfologisk analysator kan t.ex. berätta att ordet "för" kan tolkas som verb, substantiv, konjunktion, subjunktion, preposition, verbpartikel. Analysatorn kan också ange ordens lemmaformer (grundformer) och också eventuellt markera var i orden ord- och sammansättningsgränser finns samt hantera böjningsformer och avledningar. Morfologisk generering i sin tur "skapar" ordformer utifrån givna specifikationer som t.ex. ordformen "bilar" från specifikationen lemma="bil" ordklass=substantiv, genus=utrum, numerus=pluralis, species=indefinit, kasus=nominativ.

Morfologisk analys och generering är en grundläggande resurs för i stort sett alla språkteknologiska tillämpningar för alla språk.

9.1.1 Stava extended

Med hjälp av stavningskontrollen Stava från kan en avancerad sammansättningsanalys göras.
Licens: Viggo Kann

9.1.2 Swedish Python Routines (SPyRo)

http://www.ling.su.se/english/nlp/tools/spyro/swedish-python-routines-spyro-1.106863
Innehåler moduler för SALDO-analys samt sammansättningsanalys
Känd praktisk användning:
Licens: GPL 3

9.2 Morfosyntaktisk analys

Morfosyntaktisk analys bygger på att en morfologisk analys görs först, och sedan väljer en ordklasstaggare ut vilken tolkning som gäller i den specifika meningen som analyseras. Ordklasstaggarna innehåller oftast en morfologisk analysator, men åtkomsten till denna är oftas begränsad. En annan funktion inom morfosyntax är lemmatisering som innebär att ordet analyseras i sin kontext, och ordens grundform (lemmat) bestäms utifrån denna. Det krävs alltså en ordklasstaggning för att lemmatiseringen skall fungera.

9.2.1 Stagger

Stagger skall vara den bästa taggaren som finns för svenska just nu, korrekthet på 96,4 % på ordnivå.
Licens: Robert Östling

9.2.2 Granskas Tagger

Granska Tagger är den ordklasstaggare som används i grammatikkontrollen Granska.
Känd praktisk användning: Granska
Licens: GNU-licens

9.3 Syntaktisk analys

9.3.1 MaltParser – a data-driven dependency parser

http://www.maltparser.org
En svensk verison finns att ladda ner (Pre-trained model for Swedish is available). Utdata kan transformeras till frasstrukturformat.
Utvärderingsresultat: 80-90 % korrekthet beroende på språk. Svenska: Unlabeled attachment score (ord-ord-relationer, vilka ord som domineras av andra ord): 89, 5%, och Label accuracy: 87.39. Label accuracy avser att peka ut syntaktisk funktion hos varje ord. MaltParsern fick bäst resultat för svenska i CoNLL-2006.
Känd praktisk användning:
Licens: open source license, http://www.maltparser.org/license.html
Distribution: Java jar

9.3.2 Granska Text Analyzer

Ytsyntaktisk analys med hjälp av frasstrukturregler i Granskas regelspråk (Knutsson, Bigert & Kann, 2003).

9.4 Lexikogrammatiska resurser

SweCcn -- ett svenskt konstruktikon (ung. konstruktionsgrammatiskt lexikon)
http://spraakbanken.gu.se/swe/sweccn

Svenskt frasnät (ung. lexikosemantiskt lexikon)
http://spraakbanken.gu.se/swe/resurs/swefn

9.5 Semantiska modeller av ordens betydelser i kontext

Ordrumsmodeller finns det flera olika.

9.5.1 Random indexing

JAVASDM, Martins Hassels Random Indexing-paket:
http://www.csc.kth.se/tcs/humanlang/tools.html

9.5.2 Semantic Vectors

SemanticVectors creates semantic WordSpace models from free natural language text. Such models are designed to represent words and documents in terms of underlying concepts. They can be used for many semantic (concept-aware) matching tasks such as automatic thesaurus generation, knowledge representation, and concept matching.

9.6 Statistisk textanalys

9.6.1 Voyant tools

Voyant tools kan visa en rad olika samband mellan ord och texter på ett spännande sätt.

Exempelvis Mandala allows the importing of "textual" files to perform analysis on the frequency and linkage of terms. For example, importing a play would allow the user to find the linkage and frequency between a term and its speaker.
http://docs.voyant-tools.org/tools/

9.6.2 Kollokationer

Kollokationer handlar i princip om tvåordskombinationer som förekommer mer tillsammans än vad slumpne säger att de gör för att hårddra det. På engelska säger man "strong tea" och inte "powerful tea". Några olika intressanta verktyg för statistisk textanalys är följande: WordSmith Tools 4 (WST 4), Collocate, Xaira and the Ngram Statistics Package (NSP). The first two are commercial solutions; Xaira and the NSP are open source and freeware.

9.6.3 Key Phrase Extractor från SemaText

Extrakt från SemaTexts produktsida (http://www.sematext.com/products/key-phrase-extractor/): Key Phrase Extractor is a toolkit for extracting key terms (key words) and phrases from text. aka. Keyword Extractor, Key Word Extractor, Concept Extractor, Collocation Extractor, SIP Extractor. It is designed to be used in two main modes: Mode 1: Extractor of common (frequently occurring) phrases. These phrases are known as Collocations. Mode 2: Extractor of phrases based on the comparison of two sets of documents (also known as background and foreground corpora). These phrases are known as Statistically Improbable Phrases or SIPs.
In this mode the Key Phrase Extractor finds the most differentiating phrases between two document sets.

9.6.4 Ngram Statistics Package (NSP)

http://www.d.umn.edu/~tpederse/nsp.html

9.6.5 Weka

Programvarubibilotek för maskininlärning: http://www.cs.waikato.ac.nz/ml/weka/

9.6.6 NLTK

Natural Language ToolKit, http://www.nltk.org/. Python-moduler för språkteknologi.

9.6.7 ClearTK

ClearTK is a toolkit for developing statistical natural language processing components in Java and is based on the Apache UIMA framework for text analysis.
https://code.google.com/p/cleartk/

9.7 Lexikon

9.7.1 Saldo

http://spraakbanken.gu.se/resurs/saldo

Alla delar är tillgängliga med en Creative Commons Attribute-Share Alike-licens. Följande information finns i lexikonet:
Känd praktisk användning: Stagger. Spyro

9.8 Annoteringsverktyg (för SFG-annotering)

9.8.1 Knowtator

http://knowtator.sourceforge.net

9.9 En svensk automated essay scorer

Automated essay scoring, dvs. automatisk betygsbedömning är ett stort område för engelska men för svenska finns det mycket litet med undantaget av Robert Östling m.fl. system som bygger på nationella prov i svenska B. Det bör kunna vara en rimlig utgångspunkt för något liknade på svenska: http://www.ling.su.se/english/nlp/tools/automated-essay-scoring

10 Referenser

Amaral, L., Meurers, D., & Ziai, R. (2011). Analyzing learner language: towards a flexible natural language processing architecture for intelligent language tutors. Computer Assisted Language Learning, 24(1), 1-16.

Andersen, Ø. E., Yannakoudakis, H., Barker, F., & Parish, T. (2013, June). Developing and testing a self-assessment and tutoring system. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, BEA (pp. 32-41).

Biber, D., U. Connor & T. A. Upton (eds.) (2007). Discourse on the move: Using corpus analysis to describe discourse structure. Amsterdam: John Benjamins.

Burstein, J. (2003). The E-rater® scoring engine: Automated essay scoring with natural language processing.

Costetchi, E. (2013). A method to generate simplified Systemic Functional Parses from Dependency Parses. DepLing 2013, 68.

Deorowicz, S., & Ciura, M. G. (2005). Correcting spelling errors by modelling their causes. International journal of applied mathematics and computer science, 15, 275-285.

De Smedt, K. (2009). NLP for writing: What has changed?. NEALT PROCEEDINGS SERIES VOL. 3, 1.

Holmberg, P. (2009). Text, språk och lärande – Introduktion till genrepedagogik. [Text, language and learning – Introduction to genre pedagogy]. Symposium 2009.

Honnibal, M., & Curran, J. R. (2007, June). Creating a systemic functional grammar corpus from the Penn treebank. In Proceedings of the Workshop on Deep Linguistic Processing (pp. 89-96). Association for Computational Linguistics.

Karlström, P., & Lundin, E. (2013). CALL in the zone of proximal development: novelty effects and teacher guidance. Computer Assisted Language Learning, 26(5), 412-429.

Knapp, P., & Watkins, M. (2005). Genre, text, grammar: Technologies for teaching and assessing writing. UNSW Press.

Knutsson, O., Pargman, T. C., Eklundh, K. S., & Westlund, S. (2007). Designing and developing a language environment for second language writers. Computers & Education, 49(4), 1122-1146.

Knutsson, O. (2005). Developing and evaluating language tools for writers and learners of Swedish. Ph.D. thesis, Royal Institute of Technology, Stockholm, Sweden.

Knutsson, O., Bigert, J., & Kann, V. (2003). A robust shallow parser for Swedish. In Proceedings of Nodalida (Vol. 2003).

Kochmar, E., & Briscoe, T. (2014). Detecting learner errors in the choice of content words using compositional distributional semantics. In Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014). Association for Computational Linguistics.

Moreale, E., & Vargas-Vera, M. (2004). Semantic Services in e-Learning: an Argumentation Case Study. Educational Technology & Society, 7(4), 112-128.

Moreale, E., & Vargas-Vera, M. (2003). Genre analysis and the automated extraction of arguments from student essays. In The Seventh International Computer Assisted Assessment Conference (CAA-2003), Loughborough University.

Nerbonne, J. (2002). Computer-assisted language learning and natural language processing. In The Oxford Handbook of computational linguistics.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., ... & Marsi, E. (2007). MaltParser: A language-independent system for data-driven dependency parsing. Natural Language Engineering, 13(02), 95-135.

Norelius, L. (2009) Majoritetsgranskaren: ett sätt att förbättra grammatikgranskare genom att kombinera dem. Masteruppsats vid Skolan för datavetenskap och kommunikation, Kungliga Tekniska högskolan.

Sjöbergh, J., & Knutsson, O. (2005). Faking errors to avoid making errors: Very weakly supervised learning for error detection in writing. In Proc. RANLP (Vol. 2005).

Yan, H. (2014). Automatic labelling of transitivity functional roles. Journal of World Languages, 1(2), 157-170.

Östling, R. (2013). Stagger: an Open-Source Part of Speech Tagger for Swedish. Northern European Journal of Language Technology (NEJLT), 3, 1-18.

Östling, R., & Knutsson, O. (2009). A corpus-based tool for helping writers with Swedish collocations. In Proceedings of the Workshop on Extracting and Using Constructions in NLP.