Rapporter från
Språkdata
GÖTEBORGS UNIVERSITET
Institutionen för språklig databehandling
Department of Computational Linguistics

3

NORDISKA DATALINGVISTIKDAGAR 1977

Föredrag från en konferens i Göteborg
10-11 oktober 1977
utgivna av *Martin Gellerstam*

Postadress
Språkdata
Norra Allégatan 6
S-413 01 GÖTEBORG

Telefon
031 - 11 13 60
11 13 70
11 13 80
11 23 80

ISSN 0347 - 9218

INNEHÅLL

FÖRORD

Institutionen för språklig databehandling vid Göteborgs universitet anordnade under oktober månad 1977 en nordisk konferens kring ämnet datalingvistik. Initiativet till konferensen – som gick under namnet Nordiska datalingvistikdagar – kom från Nordiska samarbetsgruppen för språklig databehandling (Sture Allén, Kolbjörn Heggstad, Baldur Jönsson, Viljo Kohonen, Bente Maegaard). Konferensen planlades av en arbetsgrupp bestående av Sture Allén, Mats Eeg-Olofsson, Rolf Gavare och Martin Gellerstam med Kolbjörn Heggstad, gästforskare vid institutionen, som adjungerad. Sekreterare för konferensen var Martin Gellerstam.

Vid konferensen bestämdes dels att föredragen skulle publiceras i någon enkel form, dels att liknande konferenser i fortsättningen skulle anordnas i olika nordiska länder. Bente Maegaard ställde i utsikt att nästa konferens skulle kunna hållas i Köpenhamn 1979.

Denna volym, som är andra numret i en ny rapportserie utgiven av Språkdata, upptar samtliga föredrag vid konferensen. Förutom föredragen förekom också en avslutande diskussion, inledd av Kolbjörn Heggstad, om datalingvistikens tillstånd och behov.

Göteborg i december 1977

Sture Allén Martin Gellerstam

Gulbrand Alhaug

NOEN PROBLEMER VED OPPBYGGING AV ET DATAMASKINELT MORFEM-LEKSIKON.

Oppbyggingen av ei morfemordbok kan kreve en betydelig arbeidsinnsats. Arbeidsmengden vil i stor grad avhenge av hvor stort materialet for ordboka er, og hvor mye informasjon en vil legge inn i ordboka, f.eks. av statistisk art.

Hittil er det utgitt forholdsvis få morfemordbøker. Ei av de første som baserte seg på datateknikk, var "Russian Derivational Dictionary", utgitt i New York 1970. I forordet til denne ordboka forteller de tre utgiverne (Worth, Cozak og Johnson) at arbeidet med ordboka har tatt 7 år. Det står ikke noe om hvor mange som har arbeidet med dette prosjektet, men det er uten videre klart at det må ha gått med mange årsverk.

Kunne så et tilsvarende norsk prosjekt regne med bevilgninger til f.eks. 20 årsverk (4 personer i 5 år)? Dette er neppe realistisk.

Den russiske morfemordboka bygger på ordbokmateriale (ca. 110 000 oppslagsord) og har således ingen informasjoner om frekvenser i løpende tekst. Når det gjelder leksikalsk tekst, er det ikke gitt opplysninger om statistiske forhold. En leser som er interessert i frekvensopplysninger på leksikalsk nivå, må altså sjøl gjøre telle-arbeidet. Morfemordboka har bare én sorteringsversjon - forlengs alfabetisk sortering i KWIC-format.

Et atskillig mer ambisiøst prosjekt er den svenske morfem-ordboka. Arbeidet med denne ordboka har foregått i flere år under ledelse av Sture Allén i Språkdata og er nå så å si avsluttet. Rapportene som er utgitt om denne ordboka, viser at det er lagt ned et betydelig arbeid for å gi mest mulig informasjon om morfologiske forhold i svensk (jfr. f.eks. "Morf-o-log-i och poly-sem-i" av Sture Allén).

Det kan diskuteres hvor mye arbeid en skal legge ned i slike ordbøker. Jo større arbeidsinnsats, desto større informasjon for leseren. Et mål for den norske morfemordboka bør nok være at den skal gi minst like mye informasjon som den russiske morfemordboka. Men det er neppe realistisk å gå i gang med et så omfattende prosjekt som den svenske morfemordboka.

Ved denne konferansen deltar det flere av de som har arbeidet med den svenske morfemordboka. De vil sikkert kunne gi opplysende svar på en rekke av de problemene som knytter seg til oppbyggingen av ei morfemordbok. I resten av dette foredraget vil jeg begrense meg til et spesielt problem-område - nemlig den morfologiske behandlingen av innlånte ord.

Først noen opplysninger om materialet:

1. Avistekster - ca. 700 000 løpende ord og ca. 45 000 lemma
2. Bokmålsdelen av "Norsk ordregistrant" - ca. 60 000 oppsl.ord
3. Nynorskdelen " " " " 60 000 "

En stor del av de innlånte orda er av romansk eller gresk opphav. Med kjennskap til morfemstrukturen i det långivende språket kan en lett skille ut de enkelte morfemene i et ord. Eksempelvis har verbet assistere i latin morfemstrukturen as-sist-ere. Men skal dette fremmedordet også få tilsvarende deling i norsk? I så fall ville det være vanskelig å angi betydningen for de utskilte elementene. Hvilken betydning skulle f.eks. ligge i as og sist assistere?

For noen brukere av ordboka vil det sannsynligvis være nyttig at de forskjellige morfemer i det långivende språk er skilt ut. For andre brukere kan dette føre til at morfemordboka blir unødig vanskelig å finne fram i. Spørsmålet er så om en kan tilgodese begge brukergrupper - med sorteringsversjoner tilpasset hver gruppe?

Dersom morfemene i det långivende språket ble behandlet på linje med regulære morfemer i norsk, ville en få sorteringer som denne i KWIC-format (forlengs alfabetisering av "nøkkel-morfem" med sorteringsprioritering av venstre kolonne før høyre kolonne):

Sorteringsversjon A

film pro duc er
turtel due miljø
dug nad s time
av duk ing
intro duk sjon
lin duk
pro duk t
pro duk t iv
re duk sjon
styrke re duk sjon
dukk ert
edder dun s dyn e
fly dur
intro dus er e
re dus er ing
pro dus ent
sal at pro dus ent

Denne sorteringsversjonen har interesse for dem som vil ha en oversikt over innlånte ord med et bestemt element, f.eks. -dus-.

De som har kjennskap til latin, vil her finne rota av verbet ducere (= "føre") i forskjellige kontekster. P.g.a. ortografiske regler i norsk kommer imidlertid ikke disse beleggene direkte etter hverandre, men det har trolig mindre for seg å gruppere sammen rotvarianter som -duk-, -dus- og duc-.

De som er interessert i en slik sorteringsversjon, er sannsynligvis så godt inne i morfemsystemet i andre språk at de sjøl finner fram til slike rotvarianter. Dessuten vil det i en del tilfelle by på problemer om enheter med samme etymologiske rot skal sammenføres eller ikke. Skal f.eks. loj- i loial (lånt fra fransk) og leg- i legal (fra latin) betraktes som rotvarianter og i så fall presenteres under hverandre? Begge disse rotvariantene skriver seg fra lat. lex. Sannsynligvis vil det være vanskelig å formulere klare regler for når slike rotvarianter skal sammenføres eller ikke.

Dersom en innfører spesialtegn som markører av morfemgrenser i det långivende språket, vil en ha mulighet for sorteringsversjoner som er bedre egnet for de fleste brukere av morfemordboka. Nedafor har vi presentert det samme ordmaterialet som på forrige side, men markert grensa mellom prefiks og rot i det långivende språket med tegnet *. Vi kommer senere tilbake til andre grensemarkører, f.eks. mellom rot og suffiks. Vi får da denne sorteringsversjonen:

Sorteringsversjon_B

```
turtel due miljø
       dug nad s time
    av duk ing
   lin duk
       dukk ert
 edder dun s dyn e
   fly dur
       ---

       intro*duk sjon
       intro*dus er e

       ---

  film pro*duc er
       pro*duk t
       pro*duk t iv
       pro*dus ent
sal at pro*dus ent

       ---

       re*duk sjon
styrke re*duk sjon
       re*dus er ing
```

Her kommer altså ordstammer med felles betydningsinnhold samlet. Dette gjelder f.eks. ordstammen produk (og varianter av denne) som er belagt i substantiv (f.eks. produksjon) og i verb (produsere). Prefiksene intro- , pro- og re- er ikke skilt ut som sjølstendige enheter, men er sammenkoblet med resten av stammen med tegnet *. Dersom dette tegnet kan tenkes å virke forstyrrende på leseren, kan det om ønskelig sløyfes i utskriftsversjonen. Men da vil det i en del tilfelle komme andre ordtyper innimellom ord med samme prefiks, f.eks.:

Uten prefiks-markering	Med prefiks-markering
rebell	re*bell
redd hare	re*dus er e
redus er e	redd hare

I enkelte tilfelle kan prefikser som pro- og re- betraktes som regulære morfemer i og med at de kan skilles ut ved null-kommutasjon, f.eks. prorektor. Disse blir som andre regulare morfemer markert med "blanktegn" og vil dermed komme umiddelbart foran de ukommutable beleggene med samme prefiks. f.eks.

```
re engasj er e
re vurd er e
re*form
re*prise
re*volt
```

Leseren har dermed grei oversikt over kommutabiliteten til de enkelte innlånte prefikser og kan raskt sjekke om han/hun er enig i kommutasjonene.

Da det i en del tilfelle vil være av interesse å sondre mellom prefiks, rot og suffiks i det långivende språket, kunne vi tenke oss å bruke spesialtegn for dette formålet:

*	det foregående segmentet er et prefiks	pro*test
+	det etterfølgende segmentet er et suffiks	leg+al
.	segmentet foran og etter er ei rot	akva.duk+t

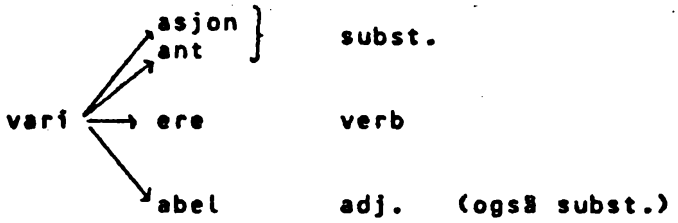
Dette innebærer at ei rot i det långivende språket kan være markert på flere måter. (Tegnet u brukes her til å markere "blank").

1.	* u	f.eks. *testu	i	pro*testu
2.	u +	uleg+	i	uleg+al
3.	u .	uakva.	i u	akva.duk+t
4.	. +	.duk+	i	"

De innlånte suffiksene spiller på flere måter en viktigere rolle enn de innlånte prefiksene i det norske ordlagingssystemet. Dette viser seg bl.a. ved at suffiksene har:

1. Ordklasseavledende funksjon
2. Større kommutabilitet

Suffiksene har en ordklasseavledende funksjon i den forstand at de tilknyttet samme rot kan danne forskjellige ordklasser, f.eks.



Når det gjelder kommutabilitet, foreligger det tilsynelatende innbyrdes kommutasjon mellom f.eks. de-, in-, ob-, pro- og re- i dedusere, indusere, obdusere, produsere og redusere. Men da verken prefiksene eller det gjenværende segmentet -dusere tilfredsstiller det vanlige kravet ved kommutasjon, nemlig at de utskilte segmentene skal ha betydning, blir ikke prefiksene skilt ut som morfemer i disse orda. De får altså markering med *, f.eks. pro*dusere - i motsetning til de få forekomstene hvor prefikset er kommutabelt, f.eks. provarabisk.

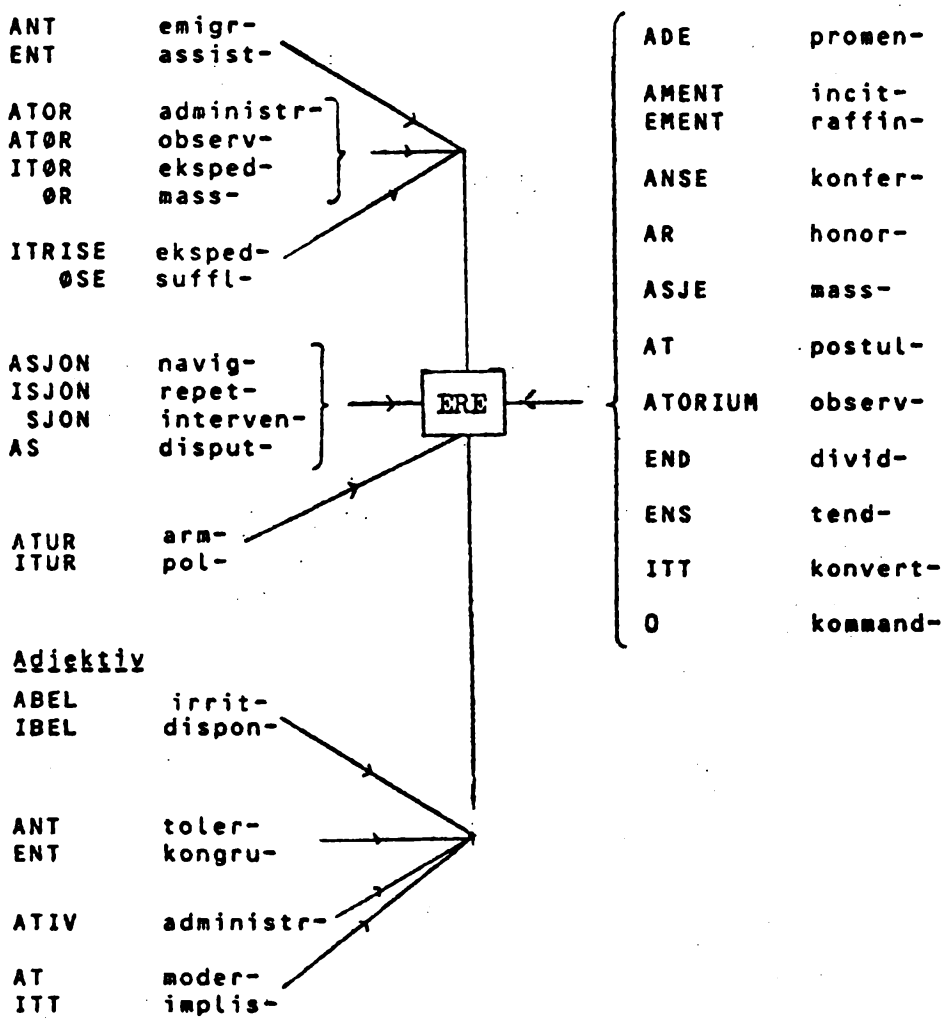
Suffiksene derimot kan skilles ut som morfemer i en rekke innlånte ord. Dette kan skje ved to former for kommutasjon:

1.	Nullkommutasjon	lakk|ere lakk|Ø
2.	Innbyrdes kommutasjon	interven|sjon interven|ere

En rekke innlånte ord av romansk og gresk opphav har innbyrdes kommutasjon, men ikke nullkommutasjon. I en del tilfelle foreligger det ingen kommutasjonsmulighet, f.eks. i ingeniør hvor -ør ikke kan kommuteres mot -ere, som i f.eks. gravør.

Vi vil først ta for oss de romanske fremmedorda, hvor det kan stilles opp et system med innbyrdes kommutasjon av suffikser. Sentralt i dette kommutasjonssystemet står det verbale -ere. Dette kan illustreres med denne skissen:

Noen subst.suffikser
(gruppert etter type)



Andre subst.suffikser
(i alfabetisk rekkefølge)

Adjektiv

I tillegg til kommutasjon med -ere har en rekke suffikser også kommutasjon med andre suffikser, f.eks.:

-asjon	emigr-asjon	-ent	kongru-ent
-ant	emigr-ant	-ens	kongru-ens

I noen tilfelle forekommer det at -ere-kommutasjon ikke er mulig, mens andre suffikser kan kommuteres, f.eks.:

arrog-ant
arrog-anse

Blant ord av overveiende gresk opphav har suffikset -isk en sentral plass som ordklasseavledende suffiks:

-i	iron-i	-ikk	polit-ikk
-isk	iron-isk	-isk	polit-isk

Det kunne komme på tale å markere ord med innbyrdes kommutasjon med et spesialtegn, f.eks. #. I så fall ville en i noen tilfelle få en tre-delt oppføring av ord med samme suffiks:

ambassad ør im*port ør sjarm ør	1. null-kommutasjon
fris#ør grav#ør mont#ør	2. innbyrdes kommutasjon (med -ere)
fav+ør in*geni+ør vig+ør	3. ingen kommutasjon

Ved å definere # og + som "skilletegn" under sorteringen (på linje med ␣ i sjarmu.ør) oppnår en at beleggene med samme suffiks - men med forskjellig kommutabilitet - kommer gruppert umiddelbart etter hverandre. Leseren kan da lett sjekke om han/hun er enig i kommutasjonene.

For leseren ville det for oversiktens skyld være en fordel at også ør-ord uten suffiksstatus kom på samme sted. Dette gjelder f.eks. malør (fra fransk malheur). Dermed ville en få en fir-delt oppføring av ord med samme element -ør. Dette kunne oppnås ved baklengssortering (fra høyre mot venstre fra og med "nøkkelmorfem"). Et ord som malør ville da komme umiddelbart etter ør-ord med spesialtegn, dvs. med suffiksstatus.

I sorteringen ovafor ser en at skillet mellom kommutable og ukommutable suffikser gir et semantisk biresultat: Substantiv med kommutable suffikser er personbetegnelser, mens de ukommutable suffiksene står i en mer uensartet gruppe (inkluderer også abstrakter som favør).

Det viktigste suffikset i de innlånte orda er nok det verbale -ere. Med utgangspunkt i slike verb kan det på norsk grunn bli dannet nye substantiv og adjektiv med romanske suffikser. Et eksempel på dette er sonder (avledet av sondere) som har fått betydningen "person som på vegne av sitt parti undersøker mulighetene for samarbeid med andre partier".

Det dynamiske elementet i ordlagingen innebærer at markeringene i ordboka kan bli foreldet. Hvis det f.eks. skulle oppstå et ord som spaser (person som spaserer), måtte spasere få markeringen for innbyrdes kommutasjon: spas#ere.

Det markeringssystemet som vi har antydet her, kan kanskje bli vel komplisert. En kan også være i tvil om et segment skal skilles ut eller ikke. Semantikken kommer inn som et problem ved en rekke ord, f.eks. president - presidere. Er betydnings-

fellesskapet av en slik art at -ent og -ere skal skilles ut som morfemer? Eller skal -al skilles ut i substantivet kolonial (= "forretning som fortrinnsvis fører matvarer"). En eldre språkbruker ser nok forbindelsen til koloni, hvor en stor del av matvarene kom fra. Men nå er koloniveldet avviklet, og en yngre språkbruker ser ikke nødvendigvis denne forbindelsen i kolonial.

Etter det jeg har forstått, behandles fremmedorda i den svenske morfemordboka på en enklere og mer "mekanisk" måte. En skiller således mellom to typer suffikser:

1. Svake suffikser (2 til 4 kommutasjoner)
2. Sterke suffikser (minst 5 kommutasjoner)

Når et suffiks har minst 5 kommutasjoner, ansees suffikset som "så väletablerat at det kan urskiljas också när restled uppkommer" (Allén 1977,6). Et eksempel på et slikt sterkt suffiks er -in, som bl.a. kan kommuteres i balsamin og blondin. Fordi suffikset klassifiseres som sterkt, blir -in også skilt ut i f.eks. giljotin (som skriver seg fra personnavnet Guillotin).

Det er mulig at det er språkpsykologiske betraktninger som ligger bak sondringen mellom svake og sterke suffikser. Ordet "väletablerat" kan tyde på dette, dvs. veletablert i språkbevisstheten. Men det er vanskelig å uttale seg om språkbrukeren ser samme suffiks i såvidt forskjellige ord som blondin, margarin, appelsin, kanin osv.

Et vesentlig poeng med den svenske morfemordboka er at "ordleddene" behandles på flere nivåer. Således kan et element som er skilt ut på et nivå, f.eks. -in i giljot-in og kre- i kre-atur, sammenføres med tilstøtende ordledd på et annet nivå. Dette innebærer at f.eks. kre- i kreatur ikke opprettholdes som ordledd på det semantiske nivået, mens derimot kre- blir stående på dette nivået i kre-ativ - med betydningskjernen "skape" i kre-.

Litteratur:

Allén, Sture: Morf-o-log-i och poly-sem-i (Göteborg 1977)

Worth, D.S. et al.: Russian Derivational Dictionary (N.Y. 1970).

COMPUTERGENERERING AF LYDSKREVET DANSK UD FRA EN QUASIORTOGRAFISK NOTATION VED HJÆLP AF GENERATIVE FONOLOGISKE REGLER

Hans Basbøll og Kjeld Kristensen

1. Indledning

Arbejdet med computergenerering af lydskrevet dansk ved hjælp af en generativ fonologi for dansk rigsmål, foregår inden for rammerne af et projekt (DANFON) som vi satte i gang ved det 3. nordiske forskerkursus i datamatisk lingvistik i København 1974. Formålet med projektet var oprindelig at anvende computeren til at teste og herved evt. forbedre en i forvejen (af HB) udarbejdet generativ fonologi for dansk. Men det datamatiske system vi har opbygget, finder anvendelse i endnu bredere sammenhæng: 1) fonologien kan modificeres så at en datamatisk parsing-analyses output der indeholder ortografiske former adskilt og opdelt af forskellige slags grammatiske grænser (boundaries), kan være input til DANFON-systemet; dettes fonetiske output kan gøres så specifikt at det kan bruges som input ved regelsyntese af dansk tale (ved hjælp af en "talemaskine"); vores projekt beskæftiger sig altså med en strækning af vejen fra skrift til lyd (om punkt 1, se afsnit 3.1-3); 2) da vores system kan manipulere fakultative regler, kan vi undersøge disse reglers indbyrdes sammenhæng i en hierarkisk struktur og herigennem berøre spørgsmålet om fakultative reglers reelle status inden for en adækvat generativ fonologisk grammatik: fakultativ/variabel (om punkt 2, se afsnit 3.4).

2. Datamatiske aspekter (specielt DANFON-systemets opbygning)

Fig. 1 viser systemets opbygning. Det centrale led i strukturen er naturligvis PROGRAM hvortil de fonologiske former, evt. i (quasi)ortografisk notation, er input, sammen med de tre sæt baggrundsdata: UNITMATRIX, RULEINDEX og RULEMATRIX. Inputformen af et givet ord underkastes forsøg på applikation af regel nr. 1 og ændres måske af denne regel; resultatet af applikationsforsøget (hvad enten der er sket ændringer eller ej) er inputform til regel nr. 2 osv. Output fra databehandlingen er de færdiggenererede overfladeformer med angivelse af derivationsvej og intermediære former (se fig. 2). Det kan nævnes at der i programmet findes procedurer til omsætning af de fonologiske former fra almindelig strengrepræsentation (som i fig. 2) til talre-

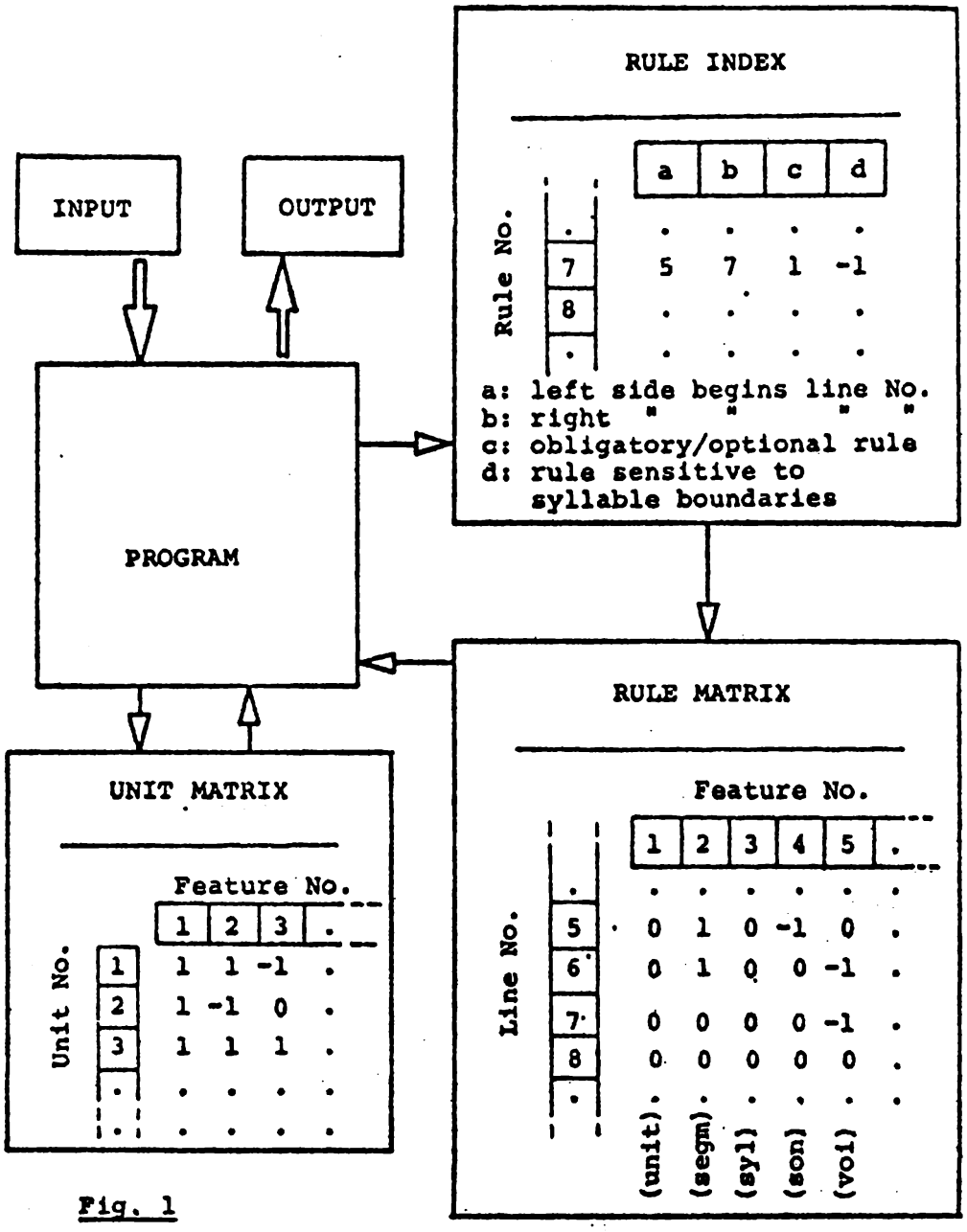

INPUT
OUTPUT
PROGRAM
RULE INDEX
a
b
c
d
Rule No.
7
5 7 1 -1
8
a: left side begins line No.
b: right " " " "
c: obligatory/optional rule
d: rule sensitive to
syllable boundaries
RULE MATRIX
Feature No.
1 2 3 4 5
Line No.
5
0 1 0 -1 0
6
0 1 0 0 -1
7
0 0 0 0 -1
8
0 0 0 0 0
(unit) (segm) (syl) (son) (voi)
UNIT MATRIX
Feature No.
1 2 3
Unit No.
1
1 1 -1
2
1 -1 0
3
1 1 1

Fig. 1

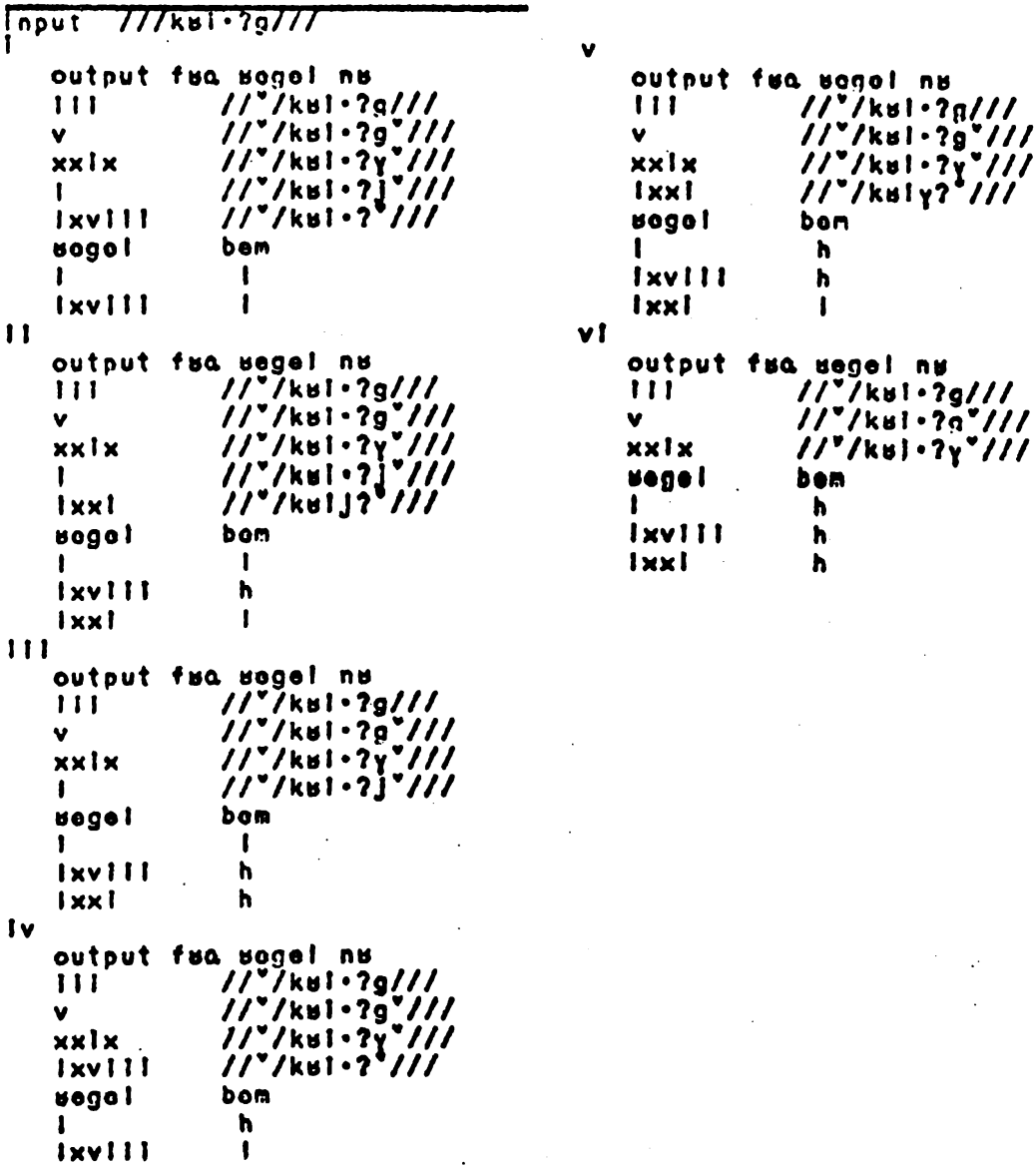


Figure 2

Derivations of the word krig. Both optional and obligatory rules are applied. The word has six versions.

præsentation, og omvendt. Det er talrepræsentationer som grammatikkens regler bearbejder under genereringen. Databehandlingens output (i strengrepræsentation) hulles ud på papertape der monteres i en papertape puncher forsynet med et særligt IPA-lydskrift-kuglehoved. Herved fremkommer det lydskrevne outprint.

UNITMATRIX (se fig. 1) indeholder en analyse i 18 distinktive træk af alle de 96 units (89 fonetiske segmenter (se afsn. 3.2) + 6 grænsesymboler (se afsn. 3.1) + "blank") som systemet opererer med. Af de 89 segmenter er de 4 x 16 fuldvokaler, idet hvert af de 4 sæt á 16 fuldvokaler er karakteriseret ved én af de 4 mulige kombinationer af værdierne af de to binære træk stød og længde, se afsn. 3.3. RULEINDEX bestemmer i hvilken rækkefølge reglerne skal forsøges appliceret. For hver regel findes der i RULEINDEX bestemmelser som "fakultativ/obligatorisk" og "reglens rang" (se afsn. 3.1) samt, ved stavelsesgrænseindsættende regler, oplysning om på hvilket sted i strengen stavelsesgrænsen skal indsættes. Foruden disse bestemmelser der angår de mere almene betingelser for reglens applikation, findes der henvisninger til hvor reglens venstre- og højreside (SD hhv. SC), som er defineret i værdier af de distinktive træk, er lagret i RULEMATRIX (se fig. 1). Programmet undersøger ved hjælp af UNITMATRIX om en delstreng af den talrepræsenterede inputform til en given regel, er kompatibel med reglens SD i RULEMATRIX (hvortil der altså henvises fra RULEINDEX); hvis det er tilfældet, foretages ud fra reglens SC i RULEMATRIX (igen med henvisning fra RULEINDEX) de nødvendige rettelser af værdierne af de distinktive træk, hvorefter outputtet gives talrepræsentation ved opslag i UNITMATRIX.

Det er nævnt at systemet tillader at reglerne er fakultative. På den måde har vi sørget for at vi i kraft af fakultativitetsprincippet kan få frembragt former fra forskellige stilistiske delkoder hørende til det sprog som den generative grammatik beskriver. Vores hypotese er at ikketom applikation af en fakultativ fonologisk regel giver en outputform der tilhører en "lavere" (mindre formel el. lign.) stil, end den form gør som ikke forudsætter applikation af reglen. Programmet er udformet således at en given inputform gennemløber alle mulige derivationsveje, for så er vi sikre på at alle i sprogsamfundet forekommende former (+ evt. nogle ikke-forekommende former) genereres. Samtlige genererede former kan bruges i en undersøgelse af de fakultative reglers plads i et hierarki, jf. punkt 2 ovenfor og afsnit 3.4. I udskriften anføres så de resulterende outputformers forskellige derivationsveje (se fig. 2):

"1" (i kolonnen mrk. "bem") betyder at den fakultative regel hvis nummer står i kolonnen mrk. "Regel", er appliceret ikke-tomt (dvs. at den pågældende regels outputform er forskellig fra inputformen), og "h" at den pågældende fakultative regel er oversprunget, men ville have kunnet appliceres ikke-tomt; hvilke regler (fakultative så vel som obligatoriske) der har været appliceret, kan man se af regelnumrene lige oven over "Regel".

3. Lingvistiske aspekter

3.1. Reglers rang. Systemet råder over 6 grænser der er ordnet lineært fra stavelsesgrænsen (den svageste) til ytringsgrænsen (den stærkeste). De 4 øvrige grænser er: / (svag ordgrænse) som sættes efter præfikser og prætonale ord og før visse "uafhængige" suffikser; // (svag sammensætningsgrænse) og /// (stærk sammensætningsgrænse), f.eks. for//bunds///dom//stol; //// er grænsen mellem trykgrupper. Stavelsesgrænserne indsættes ved regler; de øvrige grænser må vi i øjeblikket selv indføre i inputtet (se afsnit 3.5). Enhver regel har (i en given stil) én af disse grænser som sin rang, dvs. at alle svagere grænser og kun de ignoreres, når det skal afgøres om reglen kan appliceres. Det skal fremhæves at vi regner sammensætningsgrænserne for stærkere end grænsen efter prætonale ord (f. eks. ytringen /////han/tænker////på/en/for//bunds///dom//stol/////), hvorved reglerne for finalt ordtryk og finalt frasetryk (med sammensætningstrykreglen "imellem") kan reduceres til én og samme proces.

3.2. Distinktive træk. Vi søger at gøre vor trækanalyse så fonetisk (og i øvrigt også fonologisk) rimelig at vort output kan være input til talesyntese. Systemet opererer med 41 kvalitativt forskellige segmenter der er analyseret ved hjælp af 12 "kvalitative" distinktive træk (heri ikke indregnet de prosodiske træk, se afsnit 3.3 nedenfor).

Nogle nydannelser: constriction tredeler segmenter i mundlukkelyd, frikativer og approximanter (negativt defineret) - herved kan h få en rimelig definition som en ustemt approximant; consonantal er defineret som en'cover feature' ved hjælp af ækvivalensen: (-conson)$=_{def}$ (1 cnstr,+son,-lat); glottal constriction udskiller (som -) p,t,k,f,s (herved redegøres der for såvel afstemning i dækket forlyd som for de særlige stedforhold ved gammelt ustemt r); back og distant inddeler 'vowel space' så at dimensionerne fra en ekstrem faryngalvokal til i og u er henh. (-back) og (+back), og hver af disse dimensioner har 5 værdier af (dist) regnet fra faryngalvokalen (denne analyse gør på en klarere måde end normalt rede for de fysiologiske, akustiske og perceptuelle vokaldimensioner, og for principperne for r-påvirkning).

3.3. Prosodi. Syllabisk betragtes som et prosodisk træk i den forstand at ved schwa-sletning bevares stavelsesstrukturen intakt (uden at enkelte segmenter ændrer syllabicitet). Stød er et træk ved stavelsestoppen (ligesom længde), men tilskrives i outprint henh. langvokal eller sonorant efter kortvokal, eller slettes (jvf. mulig "underliggende stødforskel" i 'flæsket' etc., der kan manifesteres som en intonationsforskel i vestdansk, selv om de pågældende stavelser mangler "stødbasis"). Tryk noteres som et tal tilordnet stavelsestoppen, og tryktallene manipuleres ved regler af en særlig form: fx "1 1" til "3 1" (finalt tryk, rang //) og "1 1" til "1 2" (simplificeret sammensætningstrykregel, rang /// og ////).

3.4. Fakultative regler. Ét af vore formål med projektet er at undersøge sammenhængen mellem forskellige fakultative regler (se afsnit 1 og 2), dvs. at indkredse det hierarki (eller anden struktur) som disse regler indgår i. Det kan ske ved at forsøgspersoner får forelagt en lang række udtaleformer der eksemplificerer forskellige fakultative regler (disse udtaleformer kan efter at være genereret af vort program være indtalt på bånd af et menneske, eller - i en senere fase - direkte være output fra vort program via en syntesemaskine). Hypotesen om et hierarki af fakultative regler bygger på sociolingvistiske erfaringer, herunder en undersøgelse af 'dialektudtynding' ved én af forfatterne (KK).

3.5. Overvejelser over input. Vi anvender i øjeblikket inputformer der er quasiortografiske, og vi søger at ændre systemet henimod rent ortografiske startformer. F. eks. har vi nu regler der omdanner skriftens e'er til schwa i bestemte omgivelser og herudfra forudsiger ordenes trykfordeling i de almindeligste tilfælde. Vort system vil komme langt nærmere sit endelige mål, at omsætte skrift til lyd (på dansk), hvis det kan udnytte resultaterne af en tidligere morfologisk-syntaktisk analyse, især i to henseender: for at forudsige de grammatiske grænser kræves dels en syntaktisk analyse i konstituenter og derpå i trykgrupper (det er et meget kompliceret problem), og dels en morfologisk analyse (i præfikser, suffikser etc.); og til at forudsige stødet behøves en morfologisk analyse i ordklasser, rødder osv. Det kan endelig anføres at viden om segmenter, længde, stød, tryk og grænser vil være nødvendig for at man kan tilordne den fonologiske streng en nogenlunde akceptabel "neutral" intonation.

Litteratur:

Basbøll, Hans og Kjeld Kristensen: "Preliminary work on computer testing of a generative phonology of Danish", Annual Report of the Institute of Phonetics, University of Copenhagen (= ARIPUC) 8, pp. 216-226. 1974.

Basbøll, Hans og Kjeld Kristensen: "Further work on computer testing of a generative phonology of Danish", ARIPUC 9, pp. 265-291. 1975.

Ben 76-12-20

BETA-SYSTEMET: En sammanfattning.

Benny Brodda, Stockholm.

BETA-systemet är ett programmeringsspråk uppbyggt helt och hållet kring substitutionsgrammatikens principer. Substitutionsregler har länge använts inom lingvistiken för att beskriva vissa fenomen (både syntaktiska och fonologiska). Chomsky är väl den som gjort metoden mest känd, men även andra har förespråkat substitutionsreglernas viktiga roll i lingvistiken oberoende av Chomsky, t.ex. Hocket och Harris. I logiken och matematiken har substitutionsmetoden varit länge känd; den "uppfanns" av norrmannen Thue omkr 1910, och logikerna E. Post och A.M. Turing undersökte dess teoretiska aspekter under 30- och 40-talen.

Det aktuella systemet har regler som innebär en mild "generalisering" av regler av Turing-typ utformade som "productions" av Post-typ; generaliseringen är gjord med syfte att åstadkomma regler som är "lagom" bekväma för morfologisk analys. (Som bekant är Turing-reglerna - trots sin enkelhet - kraftfulla nog att åstadkomma allt som överhuvud taget är möjligt att åstadkomma i den här branschen. Ev. generaliseringar adderar i sak ingenting annat än möjligen bekvämlighet.)

1: SUBSTITUTIONSGRAMMATIK:

En substitution innebär att man i en sträng W (mer om detta begrepp strax) ersätter en delsträng med X med en annan delsträng Y. Man erhåller så en ny sträng på vilken man sen kan utföra ytterligare subtitutioner etc. De substitutioner man får utföra bestäms av vilka substitutionsregler man har; dessa sammantagna utgör en substitutionsgrammatik.

En grammatik i den här meningen skall nu användas på följande sätt: Man tar en sträng av ett visst slag som input till grammatiken. Denna knådar så om strängen så länge det finns regler som passar, och när inga regler längre är tillämpliga definieras den nu kanske rätt kraftigt omknådade strängen som grammatikens output.

Vad som får vara input och vad som blir output är nu inte en gång för alla givet; det beror ju på vilken sorts grammatik man skrivit och för vilket syfte. Om t.ex. input är vanliga meningar och output syntaktiska analyser av demsamma kallar man grammatiken en analysgrammatik. Om input utgöres av en enda abstrakt meningssymbol S och output av en faktisk mening har vi en syntesgrammatik, vanligen kallad en generativ grammatik. Många andra slag finnes och det generella begreppet utgör transduktor (transducer). BETA-systemet är ett sätt att göra varje dator till en transduktor.

För att datorn skall förstå hur den skall tillämpa reglerna måste det vara ordning och reda på sättet att skriva reglerna. Reglerna

bör ha något så när fixt format (utseende), och det måste vara väldefinierat vad som skall hända när man tillämpar regeln. Har man väl fixerat regel-formatet och dess tolkning ger det sig nästan själv hur man skall få datorn att uppföra sig på det förväntade sättet; de programmeringstekniska detaljerna lämnar vi helt därhån för ögonblicket - i någon mening är det ju ointressant h u r datorn bär sig åt att göra det man ber den och kanske mer intressant att veta v a d man kan be den göra. Här följer nu en kort sammanfattning av det senare.

2: TECKENREPRESENTATION OCH STRÄNGAR:

Med en sträng menas en sammanhängande sekvens av tecken (karaktärer), dvs bokstäver, siffror, typografiska tecken o dyl. Kort sagt allt det man kan skriva ned på en skrivmaskin (inkl. sådana tecken som normalt inte "syns", och därför brukar kallas "vita" tecken: vagnretur, radframmatning, mellanslag, tabbar o dyl). I det aktuella BETA-systemet antages den interna representationen vara ASCII-alfabetet, d v s den representation man erhåller om materialet stansas på en sk Teletype-maskin. I och för sig är BETA-systemet helt generellt och behöver inte alls förutsättas vara ASCII-orienterat, men för att förstå de exempel som ges är det bra att veta vad det innebär. För att kunna benämna tecknen användes då och då tecknets decimala kodnummer. Se separat bilaga. (Varav framgår att t.ex. mellanslag har kodnr 32, "!" har kod 33, "0" har kod 48, "a" har kod 65 etc.

Med en delsträng menas helt enkelt en lämplig sammanhängande sträng som ingår som del av den större; cd är alltså en delsträng av bcda, men ca är inte en delsträng. Om vi har en regel som tillåter utbyte av cd mot ers kan vi alltså få strängen berse.

3: INPUT:

Om man vill databehandla en hel text, kan det kanske vara opraktiskt (ja, omöjligt) att i maskinen handskas med hela texten på en gång. Om det är syntaktiska egenskaper på meningsnivå man vill syssla med, vill man som input ge en mening i taget. Om det är morfologisk analys på ordnivå är det ju ordet som är det meningsfulla objektet. Man måste alltså kunna tala om för datorn hur stora sjok som skall in åt gången. Varje tecken skall i BETA-systemet åsättas ett typvärde som användes för att styra input. Allt som typats som typ 1 (under rubriken DEFTYP i exemplen) betraktas som postavskiljare, eller om man så vill, allt mellan två 1:or blir det datorn får som input. För att också styra output (så att man få läsbart format) behandlas allt som är typat som en 2:a som en potentiell radavslutare. Om man vill jobba på meningsnivå bör man alltså typa punkt och frågetecken som 1:or, mellanslag och komma som 2:or och allt annat som högre. Om bokstäver typas som 4:a får man vid inläsningen sammanföring av ord avbuggna vid radslut med bindestreck. Typningen måste obligatoriskt vara med i en BETA-regeluppsättning (anges under rubriken DEFTYP). Konventionen är den, att de tecken som räknas upp till höger åsättes det typvärde som står längst till vänster; vid denna uppräkning kan man använda antingen tecknet självt eller dess ASCII-kod. Siffror, plus och minus samt vita tecken måste anges med sin kod.

Följande typning rekommenderas som standard (post=rad)

DEFTYP
1: 0
2: 32-34 36-47 58-64 (=Skiljetecken)
3: 48-57 (=Siffror)
4: 65-127 (=Bokstäver)

Anm.: "0" Användes som "stand in" för radslut. Om "0" typas som 2 men "," och "?" blir post=mening. Om mellanslag (32) dessutom typas som 1 blir post=ord.

4: REGELFORMAT:

BETA-systemet arbetar på följande sätt: När en post kommer in (vilket sker automatiskt: det finns ingen särskild läsinstruktion, ej heller någon tryck-d:o, när en post är färdigbehandlad åker den ut och nästa kommer in; när man trycker på "START" kommer första in) kan man tänka sig att en liten tomte, dot kallad, ställer sig längst till vänster i strängen. Sen kutar han fram och tillbaka, tjänstevillig som bara den, och substituerar och stär i. Vid varje ögonblick när han inte håller på och substituerar "är" han någonstans i strängen och det han där gör är att kolla om någon regel är tillämplig just d ä r. Om så i c k e är fallet tar han ett steg till höger och upprepar processen. När han ramlat utanför strängen till höger är "jobbet" klart och man erhåller resultatet som output. Det andra fallet, dvs han upptäcker e t t en regel är tillämplig är förstås det intressantaste:

En regel innehåller tre huvuddelar
A) vilken substitution som skall utföras
B) villkoren för att substitutionen skall få utföras
C) vad göra sen. ("Action")

Regelformatet är mer specifikt följande

X	Y	LC	RC	SC	SR	MV	MD
A			B			C	

Där symbolerna har följande betydelse:

X är den delsträng som ev. skall substitueras: dot bryr sig enbart om sådana delsträngar som är omedelbart till höger om honom.

Y är den delsträng som skall in i stället för X (dvs om substitutionen utföres).

LC = Left Context Condition, villkor på tecknet omedelbart till vänster om X (se nedan)

RC = Right Context Condition, villkor på 1:a tecknet t h om X

SC = State Condition: vid varje ögonblick anteges ett "tillstånd" råda och SC uttrycker ett villkor på det rådande tillståndet

SR = Resulting State: (det tillstånd som inträder om regeln tilläm-

pas)

MV = MOVE, order om vart dot skall ställa sig härnäst

MD = MODE, uppgift om ev. alternativa substitutioner skall utföras (för att ta hand om ambiguösa strängar).

5: VILLKOREN:

För att regeln i fråga över huvud taget skall tillämpas skall tre villkor vara uppfyllda, nämligen på vänsterkontext, på högerkontext och på rådande "tillstånd". Om något av dessa villkor icke är uppfyllt tillämpas regeln icke (och dot prövar en annan regel eller - om ingen sådan finns - knallar ett steg åt höger). Dessa tre villkor utvärderas helt enligt samma princip. Men innan jag beskriver den principen kanske vi skulle tala något om begreppet "rådande tillstånd".

Vid varje ögonblick antages systemet befinnas vara i ett slags tillstånd S, som dot måste kolla av för att se om han överhuvudtaget får tillämpa regeln. Man kan tänka sig det hela som att det hänger kulörta lyktor lite runt omkring, och i varje ögonblick lyser en av dessa (vid starten lyser den "neutrala" vita). I en viss regel kan det ingå instruktioner att ändra detta rådande tillstånd, dvs släcka den aktuella lyktan och tända en ny (detta är innebörden av SR, resulterande tillstånd). Detta är ett sätt att minnas vad som hänt tidigare. I systemet anges tillstånden med tal liggande i intervallet 1-127 (SJälva talvärdet betyder ingenting i sig, det är bara ett namn).

Parametern SC (som också är ett tal i samma intervall men kan också vara negerat) innebär nu ett v i l l k o r på det rådande tillståndet, och detta är viktigt att komma ihåg. Om det rådande tillståndet är '12' och villkoret i regeln säger '17' kan det mycket väl inträffa at '12' uppfyller villkoret '17'. Alltså: parametern SC är ett villkor på rådande tillstånd och inte ett namn på det tillstånd som skall råda. Det är lätt att tänka fel här, men just denna subtilitet gör systemet mycket flexibelt.

Hur vet man nu att rådande tillstånd S uppfyller villkoret SC? Ja, ta exemplet ovan: '12' uppfyller '17' om '12' finns uppräknad bland de tillstånd som är angivna till höger om rubriken 17: under DEFSET. DEFSET har samma konvention som DEFTYP men tillåter "cross-classification". Ex:

```
DEFSET
1:  1 2 3
2:  3
17: 3 12 15 17 32 35 , . ?
```

Tillstånd '1' uppfyller enbart villkor '1', likaså '2' (som alltså ej uppfyller '2'), '3' uppfyller både '1', '2' och '17' och som sagt vad, '12' uppfyller '17'. Villkoret '-17' vilket innebär villkoret '17' negerat, dvs rådande tillstånd får i n t e ingå i '17' uppfylles bara av '1' och '2'.

När det gäller vänster- och högerkontext tittar dot bara på t e c k n e n närmast till vänster och till höger om den sträng X (som

ev. skall substitueras). Vill man jobba med längre kontexter får man använda sig av tillstånd som klättrar upp och ned; BETA-systemet är närmast tänkt för fonologiskt/motologiska tillämpningar och där gäller i förbluffande hög grad att man endast behöver närmaste grannkontext). Parametrarna LC och RC är nu v i l l k o r på dessa närgrannar, och villkoren är av samma art som tillståndsvillkoren. Anger man villkoret '17' som LC betyder det att tecknet omedelbart till vänster om X skall finnas uppräknat till höger om 17: under DEFSET. I exemplet ovan är tecknen mellanslag (32), punkt, komma, frågetecken uppräknade vid '17', dvs typiska ordavskiljare; LC = 17 skulle alltså innebära att regeln endast får tillämpas om tecknet till vänster är en ordavskiljare, eller m a o enbart i ordbörjan.

LC, RC eller SC = 0 innebär "intet villkor" (noll defaultvärdet)

6: ACTION:

Om dot nu konstaterat att regeln f å r tillämpas, vad gör då dot? Ja, först och främst utföres substitutionen, men vad mer? Först och främst skall rådande tillstånd ändras till SR resulterande tillstånd, men endast under förutsättning att SR är = 0, i annat fall b i b eh å l l e s rådande tillstånd.

Nästa parameter MV (MOVE) säger var dot skall gå härnäst, och man har i stort sett 6 standardpositioner att gå till, och dessa är inte fixa utan relaterade till den nyss utförda substitutionen. För att förklara dessa framställer vi det hela schematiskt: Fig 1, är hur det ser ut just innan substitutionen utförts. Varje ruta symboliserar ett tecken, en avlång låda en sträng. Den sträng som skall bytas ut är "mittlådan". De tre lådorna tillsammans utgör den aktuella strängen.

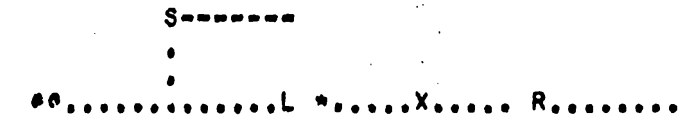


Fig 1.

(Dot, dvs "*", finns inte med i själve strängen utan kutar s å s ovanpå)

Vi antager nu att dot kollat att R ingår i RC, L ingår i LC och S ingår i SC. Dot plockar då bort X och plugger in Y i stället. Just vid själva substitutionsögonblicket kan man tänka sig att hela strängen man arbetar med har tre delar:
Delsträngen till vänster om Y, Y självt och delsträngen till höger om Y, detta symboliserat med de tre lådorna nedan i fig 2. Varje sådan delsträng har (kan ha) ett första tecken och ett sista tecken, vi har alltså 6 väldefinierade punkter i strängen. Vi numrerar dessa från vänster till höger 1 - 6, och får då de standardpositioner till vilka man kan dirigera dot. Skriver man 4 som MV parameter ställer sig dot alldeles t v om det sista tecknet i Y (Om Y är tom, d v s X deleterats blir förstås positionerna 3, 4 och 5 desamma). Utöver dessa sex standardpositioner kan man dirigera dot till några andra positioner utanför den aktuella strängen.

Dessa är:

0: Hela strängen deleteras
7: Strängen betraktas som färdigbehandlad och ges som output (ut 1)
8: Strängen ges som output på separat fil (ut 2)
9: Strängen dirigeras till radskrivaren (Lpr)

Genom möjligheterna 8 och 9 kan man använda BETA som ett mycket avancerat excerperingsprogram. 0-Ning kan tillämpas om flera alternativ bearbetas samtidigt och något av dem visade sig vara dödfött; jfr MD-parametern nedan)

```
0   1        2 3    4 5        6 7    8   9
•   •........L YYYYYY R........• •    •   •
```

Positionerna 0, 1, 6 och 7 finns faktiskt som en del av strängen som BETA arbetar med och innehåller alla tecknet " " ("brädgård"). Strängen i utrymmet mellan 1 och 6 är den egna strängen man jobbar med. Startposition är 1, "halkar" man ut till höger är man färdig, "halkar" man ut till vänster dör strängen.

Default för MV är 5.

7: ALTERNATIV

Om man håller på med syntaktisk analys är det bekant att man ibland kan erhålla strängar (konstruktioner) som är analyserbara på mer än ett sätt. Problemet är att kunna ange samtliga möjliga analyser. Detta är ganska lätt att åstadkomma i BETA, ÅTMINSTONE OM MAN KAN SKRIVA REGLERNA SÅ ATT MAN SER ATT EN VALSITUATION ÄR FÖR HANDEN. Detta anges i BETA-reglerna med den sk MODE-(MD) parametern. Denna parameter kan ha 3 värden 1,2 och 3 (samt också "ekvivalenta" värden 5, 6 och 7; Om dessa senare användes "frågar" dot om regeln i fråga får användas. Denna facilitet förutsätter att man kör BETA interaktivt och sitter vid terminalen och besvarar dessa frågor).

Reglerna i BETA tolkas normalt disjunktivt. Det betyder att den första regel uppifrån som är tillämpbar tillämpas och om MD är = 1, vilket är det normala (Jfr dock avsnittet om regelordning), så är det därmed inget mer tal om den saken. Om MD är satt till 2, läggas det nyss erhållna resultatet en stund åt sidan, och dot söker efter ytterligare en regel, och om en sådan återfinnes utföres även denna substitution, är även denna regel märkt som en 2:a slaskas även det nya resultatet undan och nästa tillämpbara regel tillämpas etc.

De uppstående alternativen läggas sedan i en kö, och denna kö avbetas först innan någon ny post inhämtas utifrån. I denna kö lagras det "halvfabrikat" som hittintills uppstått inklusive det aktuella tillståndet. När detta halvfabrikat återhämtas från kön fortsätter man alltså behandlingen precis där man avbröt den (för att fortsätta med andra alternativ).

MD = 3 är ett specialfall av 2. Substitutionen utföres och resultatet av densamma lagras i kön precis som i fallet MD = 2. Någon ny regel uppsökes inte och dot fortsätter att jobba med "originalet" som ingenting hade hänt,: man kan alltså både ta och släppa en regel.

Default för MD är 1.

8: GENERELLA STYRPAR:

Överst på varje regellista skall upp till 3 generella parametrar skrivas. Dessa kallar vi REGDEL, HL och PROGVAR. REGDEL (Regeldelimitern) är ASCII-koden för det tecken som användes för att definiera vad som menas med att vänsterledet och högerledet i en substitution är "slut". Default är 32 (mellanslag), men om man måste skriva en regel omfattande mellanslaget måste man välja ett annat tecken (som dock aldrig får förekomma i en regel).

HL förklaras i avsnittet om regelordning nedan. PROGVAR kan ha värden 1 eller 3 och att man har lite olika programvarianter. 1 är defaultvärdet och innebär precis det system som vi förutsett här, PROGVAR = 3 innebär att man har inga statusparametrar i reglerna. Regelformatet blir då:

X Y LC RC MV MD

9: REGELORDNING:

Reglerna utvärderas i princip i den ordning de lästs in. Den första regeln uppifrån som är tillämpbar tillämpas. En sådan regelordning kallas för disjunktiv. Att reglerna utvärderas disjunktivt är dock en sanning med modifikation, det är i stort sett sant, men inte alldeles. Hur utvärderingen i detalj utföres bestämmes av parameter nr 2, HL, i de generella styrparametrarna. Denna parameter lägger upp reglerna i s k hash-klasser enligt vänsterleden i substitutionsreglerna. Om HL t.ex. är =2 (defaultvärdet) betraktas alla regler med minst två identiska tecken i vänsterledet ("X") i substitutionsreglerna som hörande till samma hashklass, och dessa utvärderas före de som innehåller endast ett tecken i vänsterledet. Om HL sättes = 3 utvärderas de regler som innehåller minst tre tecken före de som innehåller två, vilka i sin tur utvärderas före de som innehåller endast ett tecken. Inom varje hash-klass utvärderas reglerna strikt disjunktivt (detta är viktigt att komma ihåg, så att dot inte "nyper" fel regel tidigare). Huvudprincipen för regelskrivandet skall alltså vara att regler med starkare villkor (d v s längre eller havande "härdare" kontext-villkor) måste skrivas tidigare i listan. Något krav på att reglerna skall skrivas i bokstavsordning eller dyl föreligger inte. Tvärtom, det rekommenderas att regler som "logiskt" hör ihop sammanföres. Sådana logiskt hopförda regler kommer då att fungera som subrutiner i vanliga programmeringsspråk. Konventionen med hash-klassernas utvärderingsordning underlättar att man kan skriva reglerna på detta sätt.

GÖTEBORGS UNIVERSITET
SPRÅKDATA
Algoritmisk textanalys
Mats Eeg-Olofsson
Sep -77

Algoritmisk textanalys - en presentation

Inom projektet Algoritmisk textanalys håller vi på med att utarbeta formaliserade metoder för grammatisk analys av autentisk svensk text. Ett av projektets syften är praktiskt - vi vill konstruera ett fungerande programsystem som på ett ekonomiskt sätt kan analysera stora textmassor. Existensen av ett sådant system är väsentlig för verksamheten inom Logoteket, det nationella serviceorgan som tillhandahåller maskinläsbara texter och textbearbetningar. Arbetet ger givetvis också teoretiskt relevanta resultat. Man kan t.ex. peka på Staffan Hellbergs inom projektet utarbetade formella beskrivning av svenskans morfologi. Andra lingvistiskt intressanta regelsystem som ligger till grund för analysen är exempelvis Jerker Järborgs ytstruktursyntax. Analyssystemets utformning har också teoretiskt intresse som uttryck för en perceptionsstrategi.

Konkreta delmål för projektarbetet just nu är dels att disambiguera all homografi i den inmatade texten, dels att förse den med en enkel syntaktisk strukturbeskrivning. En praktiskt betydelsefull biprodukt av systemets verksamhet är alltså en lemmatisering av texten. Den syntaktiska ytstrukturbeskrivningen har givetvis ett värde i sig, men kan också tjäna som utgångspunkt för en djupare syntaktisk-semantisk analys.

I dagens läge är det ju knappast möjligt att uppnå dessa delmål på helt automatisk väg. Orsaken tycks vara att det är svårt att med maskinen efterlikna människans förmåga att använda extralingvistiska data även vid en rent "formell" grammatisk analys. Vi förutser därför att en av komponenterna i systemet blir en mänsklig informant, en lingvist som kan granska maskinens lösningsförslag och eventuellt komma med korrektioner. Vi vill emellertid gärna se hur bra den helt automatiska analysen kan bli innan vi bestämmer hur interaktionen lingvist-maskin skall utformas.

Medan andra projekt, framför allt sådana som syftar till någon form av textförståelse, i stor utsträckning har övervunnit svårigheterna vid den grammatiska analysen genom en långtgående integrering av syntaktisk och semantisk bearbetning av texten, vill vi i Algoritmisk textanalys se hur långt man kan komma med rent formella hjälpmedel. Statistiska data om frekvenser för ord och ordförbindelser, hämtade från projektet Nusvensk Frekvensordbok (NFO), kommer att användas på flera sätt. Syftet är att bygga upp ett slags sannolikhetsmodell, som för varje alternativ analys av en mening ger ett mått på analysens rimlighet.

Strategin för textanalysen blir i korthet följande:

Steg 1: junkturanalys och förberedande morfologisk analys.

Detta steg är helt automatiskt. Det utförs delvis för att spara tid och minnesutrymme för bearbetningarna i de följande stegen. Indata är en opreparerad text. Utdata är samma text uppdelad i meningar och försedd med ordklass- och böjningsangivelser för vissa ord.

Svårigheten vid skiljeteckensanalys är ju framför allt att skilja mellan förkortningspunkt och meningspunkt. För att göra denna analys säkrare används en lista från NFO3 över vanliga meningsinledande fraser. De ord som i detta steg förses med grammatiska uppgifter är (vanliga) heterografer och kvasiheterografer (formellt homografa ord, t.ex. "är", där en av homografkomponenterna är helt dominerande). Även ord som ingår i vissa frekventa "konstruktioner" (grammatiskt välformade rekurrenta ordförbindelser) disambigueras därigenom (nästan) säkert och blir därför goda utgångspunkter för homografsepareringen i de följande stegen.

Steg 2: morfologisk analys och syntaktisk ytstrukturanalys.

I detta steg kommer flera olika processer att samverka på ett sätt som vi ännu inte har utformat mera detaljerat. Utdata från steg 1 underkastas först en morfologisk analys, där de ord som inte redan har en grammatisk märkning slås upp i ett lexikon över stammar. Återstoden av ordkropparna undersöks sedan av en morfologisk grammatik som anger möjliga böjningsändelser och fogar för de ord som

tillhör stammens paradigm. Sammansatta ord uppdelas alltså i sina beståndsdelar. Ofta nog kan en stam ha flera alternativa paradigmnummer. Dessa undersöks (och presenteras) då i en ordning som är baserad på de textuella frekvenserna för motsvarande lemman i NFO2. Flertydighet uppkommer också genom att det är morfologiskt möjligt att segmentera längre sammansatta ord på många olika sätt.

Utdata från den morfologiska analysen är en riktad graf, där de olika bågarna representerar alternativa homografkomponenter i de analyserade orden (jämför Martin Kays "chart"). Syntaxkomponenten väljer nu ut strängar av homografkomponenter och märker de ingående orden med avseende på de konstituenttyper de kan ingå i. Strängen kan sedan (i allmänhet på flera sätt) uppdelas i en följd av kontinuerliga "ytstrukturkonstituenter". Varje sådan ytkonstituentsträng värderas därefter genom att de enheter som finns med i ett lexikon över konstituentfraser tilldelas ett högt värderingstal. Detta är en pseudosemantisk kontroll, eftersom de fraser som är frekventa nog att vara belagda i lexikonet är semantiskt selekterade. Ytkonstituenterna underkastas sedan en intern kontroll och värdering; härvid används bl.a. kongruensfenomen. Efter den interna evalueringen följer en extern. Värderingsgrunder härvidlag är regler för en menings totala sammansättning, för ordningen mellan ytkonstituenter och för antalet konstituenter i en viss position. Dessutom utnyttjas information i ett särskilt subkategoriseringslexikon, som t.ex. för verb skall ange hur många och vilka syntaktiska argument som verbet brukar ta. Slutligen sammanvägs de olika evalueringarna av varje ytkonstituentsträng till ett totalvärde, som uttrycker analysens rimlighet.

Steg 3: relationell syntaktisk analys.

I detta steg, som ännu bara befinner sig på planeringsstadiet, skulle man kunna söka efter relationella kategorier som subjekt, objekt osv. bland ytkonstituenterna.

Det återstår att närmare utforma evalueringsreglerna och fastställa värden på de ingående parametrarna. Detta kommer att kräva mycket experimenterande. Betydelsefull för systemets effektivitet blir också återkopplingen mellan de olika analysprocesserna. När skall man

t.ex. avbryta den syntaktiska analysen av en homografkomponentsträng som verkar dålig och välja en annan homografkomponentsträng i stället?

En alternativ systemlösning som något diskuterats inom projektet utgår från texter lagrade i en form som föreslagits för Logotekets ändamål. Texterna finns då i databaser, där man via länkar har direkt tillgång till såväl hela texten som samtliga belägg på varje grafordstyp, initial- och finalalfabetisk sortering av ordtyperna osv. Denna lagringsform är säkert fördelaktig för en användare som vill studera texten intensivt. Den borde också kunna ge vissa fördelar för den algoritmiska textanalysen. Varje grafordstyp behöver bara analyseras en gång. Man har också möjlighet att arbeta med större kontexter än meningar. Vid bestämning av ordklass och böjning för ord som saknas i lexikon kan det vara värdefullt att förfoga över alla belägg i texten på de okända orden. Emellertid torde en sådan metod vara ganska resurskrävande i tid och minnesutrymme. För närvarande verkar det mera angeläget att utveckla den andra systemlösningen, som använder sekventiell bearbetning mening för mening.

Programmeringen sker i SIMULA för IBM-anläggningen vid Göteborgs Datacentral. För vissa interaktiva bearbetningar av lexika och regelsystem används BASIC vid Språkdatas NOVA-anläggning.

Referenser:

Allén, S. et al.: Nusvensk Frekvensordbok 1-3 (NFO1-3).
Kay, M.: The MIND System (i Rustin (ed.): Natural Language Processing, New York 1973).

Ivar Fonnes:

EDB I ORDBOKSPRODUKSJON:

TILRETTELEGGING AV NORSK LANDBRUKSORDBOK FOR TRYKKING.

---

1. Innledning.

Norsk Landbruksordbok er en definisjonsordbok bestående av ca. 17000 oppslag som innleder definerte ordartikler, og ca. 8000 oppslag med referanse til definerte ord, altså totalt ca. 25000 oppslag. Hele materialet er registrert i maskinleselig form og lagt opp som en databank. På dette grunnlag produseres det så magnetbånd for fotosetting og trykking av ordboken.

I de fleste ordartikler er det angitt synonymer på en del andre språk. Disse synonymene trekkes ut i egne lister (én for hvert språk) med det norske oppslagsord som referanse, alfabetiseres og legges opp på magnetbånd for fotosetting.

Ordboken vil bli trykket i to bind, hvorav bind 1 (hoveddelen) inneholder de norske oppslag med definisjoner og referanser, og bind 2 (registerbindet) inneholder alfabetiserte synonymlister på 7 språk (samisk, svensk, dansk, finsk, islandsk, engelsk og tysk) med henvisning til de norske oppslagsord i hoveddelen.

Databehandling har spilt en sentral rolle i produksjonsarbeidet. Det har vært et verdifullt hjelpemiddel i det avsluttende redigeringsarbeid, og har muliggjort fullstendig automatisert produksjon av registerbindet. Dessuten har vi kunnet produsere hele materialet på fullt ferdig såkalt "drivetape" som kan gå direkte til fotosetting.

I Norge har det ikke tidligere vært produsert definisjonsordbøker ved hjelp av slike metoder, og vi har ikke hatt kjennskap til lignende prosjekter som kunne danne forbilde for vårt opplegg. Materialet i Norsk Landbruksordbok er forøvrig nokså spesielt og i vår sammenheng komplisert. Det forekommer en rekke trykkvarianter i hyppig veksling, stort tegnrepertoar, formler osv. (jfr. vedlegg).

Materialet er også av betydelig størrelse, anslagsvis omkring 8 mill. tegn. Det nærmer seg altså størrelsen på f.eks. Brown Corpus. I tillegg kommer så synonymlistene som bygges opp maskinelt. Disse vil i trykk utgjøre nesten like mange sider som hovedmaterialet.

Denne framstilling tar sikte på å redegjøre for hvordan EDB kan være til nytte i produksjonen av en slik ordbok, og trekke fram noen av de mest sentrale problemer som knytter seg til EDB-arbeidet i en slik sammenheng.

2. Materialet.

Hver ordartikkel (se eksempel i vedlegg) består av et oppslagsord med eventuelle bøyningsformer og varianter for nynorsk og bokmål, eventuelt etymologi, definisjon av ordet m.v., angivelse av fag-

område, signatur for den faglig ansvarlige, og synonymer på norsk og inntil 6 andre språk. I tillegg kan det forekomme spesifikasjoner av oppslagsordet (f.eks. med adjektiv) og påhengte oppslagsord med egne definisjoner og synonymer.

Alle disse opplysningene følger tett etter hverandre, men ulike trykkvarianter gjør det likevel forholdsvis lett for leserne å holde dem fra hverandre: Oppslagsord i halvfet, bøyningsformer i kursiv, fagmerking med versaler, signaturer med kursiverte versaler, utenlandske synonymer i hakeparenteser, spesifikasjoner i sperrede kapitéler, nynorsk- og bokmålsvarianter med hevet henholdsvis n og b osv.

Materialet er delt inn i ca. 40 fagområder, og før trykkeprosjektet startet forelå det meste i stensilerte hefter (maskinskrevet) med ett hefte for hvert fagområde. Oppslagsordene var ordnet alfabetisk innen hvert hefte.

Ved trykking av ordboken skulle hele materialet ordnes i ett alfabet. Dette måtte nødvendigvis bli et nokså omfattende sorteringsarbeid. Dessuten måtte det foretas en del redaksjonelle endringer som følge av den nye sorteringen. Eksempelvis måtte oppslagsord som fantes i flere hefter, bygges sammen i én og samme ordartikkel.

Redaksjonen ønsket også å bygge inn en del opplysninger som var kommet til etter at heftene ble skrevet, og foreta en del justeringer. F.eks. tok man sikte på å endre genusmarkeringen for alle substantiver.

Endelig skulle de utenlandske synonymer trekkes ut med referanser og alfabetiseres innen hvert språk.

3. Formål og fordeler ved bruk av EDB.

Blant flere mulige opplegg for trykking av ordboken valgte man fotosetting via EDB. Fotosats gir samme kvalitet som blysats og praktisk talt fritt valg med hensyn til tegn- og type-repertoar.

Bruk av EDB i produksjonen ble valgt av flere grunner:

a) Lavere kostnader enn ved manuelt opplegg. Dette var delvis basert på billig maskintid ved Universitetet i Oslo. Men til tross for at prosjektet har måttet kjøpe en del maskintid utenfor universitetet, har vurderingen vist seg å være riktig.

b) Biprodukt i form av en databank, tilgjengelig for forskningsformål og en fordel ved eventuelle senere utgaver av boken.

En slik databank vil også kunne få betydning utover det som var planlagt. Ved EF-kommisjonen i Brussel arbeides det med en databank for landbruksterminologi, og det er interesse for en kopi av vårt materiale. Fra vår side er det ønskelig å bygge inn synonymer på flere språk, f.eks. fransk. Utveksling vil trolig finne sted etter at trykkeprosjektet er avsluttet.

c) Reduksjon av redaksjonens avsluttende arbeid i betydelig grad, og muligheter for systematisk kontroll av materialet.

Alfabetisering av materialet samt utplukking og alfabetisering av synonymer er meget betydelige arbeidsoppgaver som nå er blitt utført automatisk.

Finalalfabetisk liste over oppslagsord muliggjør systematisk kontroll av morfemmarkeringer, betoningsaksenter og skrivemåte. Lister over alle ulike fagmerkinger og signaturer muliggjør kontroll av disse osv.

Totalt sett vurderte man det slik at EDB ville gi betydelige gevinster i form av lavere kostnader, systematiske kontrollmuligheter og en databank. Dette har vist seg å være en korrekt vurdering. I tillegg bør nevnes den erfaring prosjektet har gitt i arbeidet med å produsere trykkeklare data på et slikt materiale, ikke minst fordi materialet både er stort i omfang og komplisert.

4. Sentrale problemområder i EDB-opplegget.

4.1. Utgangspunktet.

Materialet til Norsk Landbruksordbok var samlet inn gjennom mange år og lagt til rette for trykking på tradisjonell måte. Spørsmålet om EDB hadde aldri vært inne i bildet, og opplegget var naturligvis da heller ikke på noen måte tilpasset slike metoder. På den annen side var det en selvsagt forutsetning at bruk av EDB ikke skulle påvirke ordbokens utseende - fotosetting via EDB skulle gi samme resultat som manuell blysats.

Dette noe ugunstige utgangspunkt, sett fra EDB-siden, representerte imidlertid ikke noe særlig betydelig problem. Markeringer av forskjellige typer av opplysninger i ordartiklene var gjort med henblikk på at det skulle være lett å finne fram for leserne. Det var lagt opp til å anvende ulike skrifttyper som halvfet og kursiv i tillegg til vanlig skrift, og dessuten kapitéler og versaler samt parenteser, hakeparenteser, skråstreker osv. Dette opplegget passet egentlig veldig bra for databehandling. Ved å legge inn de markeringer som var nødvendige for å skille mellom ulike trykkvarianter, fikk vi samtidig inn de opplysninger maskinen trengte for å produsere listeprodukter for redaksjonen, og også hoveddelen av de markeringer som ble ansett nødvendige for å utnytte materialet maskinelt i en databank. Noen få tilleggsmarkeringer ble lagt inn med henblikk på maskinell utnyttelse av materialet, og disse ble naturligvis fjernet under opplegget av et trykkeklart magnetbånd.

Et større problem var egentlig det forhold at man ikke hadde full oversikt over hva som kunne forekomme av tegn og kombinasjoner i materialet. For eksempel: Vi visste at det forekom både greske bokstaver og matematiske formler, men kunne en gresk bokstav forekomme som eksponent i en formel (altså som hevet tegn i trykk)? I tillegg kom at det foregikk redaksjonsarbeid parallelt med registreringen slik at det kunne oppstå nye varianter underveis.

Det var altså ikke til å unngå at det dukket opp nye problemer underveis i prosjektet, og løsning av disse måtte da innpasses i opplegget på best mulig måte. For programmeringen er dette alltid en ugunstig situasjon. Likevel er det klart at det ikke dukket opp ting som representerte noe alvorlig problem i forhold til det opplegg som var valgt.

4.2. Datarepresentasjon.

Vi skulle altså representere et materiale med et tegnrepertoar som langt overstiger det man vanligvis har tilgjengelig i en datamaskin og dessuten markeringer for ulike skrifttyper og andre trykkvarianter. (Ser man bort fra forskjellen mellom vanlig skrift, kursiv og halvfet er repertoaret på ca. 220 tegn. Tar man med kursiv- og halvfet-variantene stiger tallet til ca. 420). Dette måtte markeres med en eller annen form for funksjonskoder. Vi valgte å bruke en funksjonsmarkering (tegnet %) etterfulgt av et tall eller en bokstav som angir hvilken funksjon det dreier seg om, og la funksjonen gjelde inntil den blir opphevet (med tegnet $). Flere funksjoner kan forekomme inne i hverandre, og et $-tegn opphever alltid sist angitte funksjon.

På denne måte kan vi representere nær sagt alle tenkelige varianter av tegn og tegnkombinasjoner. Vi bruker slike funksjoner for å markere kursiv og halvfet, hevede og senkede tegn, gammelnorsk, samisk og gresk alfabet, aksenter, brøkstrek, kvadratrot osv. På den annen side koster det noe ekstra arbeid å skrive slike funksjoner under dataregistreringen, og det vil også kreve ekstra påpasselighet å holde rede på hvor man til enhver tid befinner seg, særlig hvis man har flere funksjoner inne i hverandre. For å lette registreringsarbeidet ble det innført forenklede varianter av enkelte av de mest høyfrekvente funksjoner, f.eks. at * (asterisk) medfører at neste tegn skal være hevet (meget hyppig ved n og b for nynorsk og bokmål), og at & markerer at neste tegn er en aksent som skal over foregående bokstav.

Til tross for denne forenkling er det klart at registreringsarbeidet var forholdsvis krevende, og det var utvilsomt en betydelig fordel at registratoren kjente materialet og oppbyggingen av ordartiklene gjennom arbeid i redaksjonen.

4.3. Presentasjon av data for korrektur.

Med alle de funksjonskoder som måtte brukes, til dels meget hyppig, er det klart at materialet ikke var særlig lett å lese. Det var derfor en viktig oppgave å finne fram til en bedre presentasjonsform med henblikk på korrekturarbeidet.

Her hadde vi et velegnet grunnlag i de stensilerte hefter hvor materialet forelå maskinskrevet. I heftene var trykkvariantene markert med ulike typer understrekinger, understreking med + - tegn punktum, likhetstegn, asterisk, vanlig understreking osv. Dette systemet kunne vi anvende ved å skrive ut selve materialet på annenhver linje, og la maskinen konvertere en del av funksjonskodene til understrekinger på de mellomliggende linjer.

Dette systemet viste seg meget vellykket. Materialet kunne nå skrives ut i en etter forholdene oversiktlig form, og dessuten i en form som redaksjonen var vant til fra før gjennom de stensilerte heftene.

4.4. Alfabetisering av materialet etter oppslagsord.

Alfabetisk sortering av materialet kunne ikke gjennomføres direkte på oppslagsordene slik de forelå i materialet. For det første inneholder ordene en del informasjon som måtte bort (punkturering for morfemgrenser, betoningsaksenter og annen ikke-alfabetisk informasjon som f.eks. vanlige aksenter og tall).

For det andre måtte en del informasjon konverteres. Bokstaver som ä, ö, ü (svensk eller tysk) og andre ikke-norske bokstaver, måtte konverteres til norske ekvivalenter som ga dem riktig plass i alfabetet. Romertall (betydningsnummer) foran oppslagsord måtte flyttes bak ordet, det samme gjaldt bindestrek som stod foran ordet. Oppslag på mer enn ett ord måtte markeres spesielt osv.

Det måtte altså etableres egne sorteringsfelter for de enkelte oppslagsord. Regelverket for etablering av sorteringsfelt ble relativt komplisert. Dette skyldes at det forekommer diverse ikke-alfabetiske tegn i oppslagsordene som ofte representerer signifikant informasjon for sorteringsrekkefølgen. En ekstra komplikasjon var at noen få ord måtte behandles særskilt på tvers av regelverket.

4.5. Klargjøring av materialet for fotosetting.

Klargjøringen for fotosetting omfattet i hovedsak følgende arbeidsoppgaver:

- Fjerning av informasjon som var lagt inn med henblikk på databehandling.
- Ny linjeinndeling.
- Bearbeidelse av materialet etter trykkeriets spesifikasjoner og opplegg av korrekte koder på magnetbånd.

4.5.1. Linjeinndeling.

I og med at innholdet i hver ordartikkel skrives fortløpende, må nødvendigvis linjeinndelingen i den trykte versjon bli forskjellig fra den man brukte ved dataregistreringen. Dette ville ikke representere noe betydelig problem i et vanlig tekstmateriale, men med alle de spesielle markeringer som forekommer i denne ordboken, viste det seg forholdsvis komplisert å utarbeide kriterier og programmer for linjedeling. Hvilke tegn skulle følge ordet foran, hvilke skulle følge neste ord (dvs. ned på neste linje) hvilke skulle behandles som selvstendige enheter, hvilke tegn kunne ikke innlede en linje, hvilke kunne ikke avslutte en linje osv. Så kommer spørsmålet om deling av formler, at fagmerking og signaturer ikke kan deles, og dessuten vanlig orddeling som kompliseres ved at en rekke ord inneholder ikke-alfabetiske tegn, deler av ordet kan stå i parentes osv.

A) Oppbygging av en linje.

Rent teknisk har man følgende utgangspunkt: Tegnene har varierende bredde som angis i såkalte relative enheter (RE). F.eks. har bokstaven i en bredde på 4RE og bokstaven m hele 13RE i vanlig skrift. I kursiv er de tilsvarende tall 4,5 RE og 12,5RE (tallene gjelder for skriften Times). Totalbredden på spalten (= linjelengden) er også angitt i relative enheter, i vårt tilfelle 486RE. Programmet løper altså gjennom materialet tegn for tegn,

summerer opp breddeverdier og ser hvor mye det blir plass til innenfor de 486RE.

Man kan imidlertid ikke bare avslutte linjen etter siste hele ord før grensen er nådd. Alle linjer unntatt den siste i hver ordartikkel skal ha rett høyremarg, og jokeren i arbeidet med å få dette til er ordmellomrommene (space). Disse kan variere betydelig i bredde (i vårt tilfelle fra 4 til 14RE), og dette gir forholdsvis gode muligheter til å få delt linjen på et naturlig sted og likevel oppnå rett høyremarg. Programmet må altså normalt finne fram til ett eller to steder hvor det er naturlig å dele linjen (mellom ord el.lign.) og undersøke om breddeintervallet på ordmellomrom tillater deling der. Hvis ikke må det foretas orddeling.

B) Orddeling.

Orddeling er et velkjent problem innen automatisert tekstbehandling. I vårt tilfelle ble forholdet ytterligere komplisert ved at materialet inneholder ord på 6 andre språk som f.eks. tillater en del konsonantkombinasjoner som ikke finnes i norsk. Likeledes måtte vi ta hensyn til ikke-alfabetiske tegn samt parenteser inne i ord.

I vår sammenheng syntes det lite hensiktsmessig å legge mye arbeid i å komme fram til et perfekt orddelingsprogram som kunne gi korrekt orddeling i alle tilfeller. Vi vurderte det som mindre arbeidskrevende å bruke et enkelt program som ga riktig deling i de fleste tilfeller, og så rette opp manuelt de delinger som var feil. Det program vi har brukt er utviklet for norsk tekst og bygger på enkle prinsipper.

Vårt opplegg går da i korthet ut på at ord som må deles blir avkledt all ikke-alfabetisk informasjon, parenteser holdes utenfor, og så overlates ordet til orddelingsprogrammet. Alle orddelinger som foretas skrives ut i en egen oversiktlig liste. Redaksjonen leser korrektur på denne listen og anmerker feil. Feilene kan rettes av oss som vanlig korrektur eller av trykkeriet under settingen. Vi har kommet til at det siste vil være det enkleste.

Resultatet av kjøring på ca. 20% av materialet antyder følgende statistikk for orddelinger:

> Totalmaterialet er på ca. 100.000 trykte linjer.
> Orddeling foretas i ca. 5.000 linjer, dvs. ca. 5% av linjene.
> Feil orddeling forekommer i ca. 800 linjer, dvs. ca. 16% av orddelingene er feil, eller: Feil orddeling forekommer i ca. 0,8% av linjene.

Vi anser dette for å være fullt tilfredsstillende - det vil ikke koste mye arbeid å gjennomføre slik kontroll og korrektur.

De typer av feil som forekommer, er først og fremst plasseringen av s i sammensatte ord, f.eks.

årsvekst blir til år-svekst

og feil av typen

endring blir til en-dring.

Noen feil skyldes også spesielle konsonantkombinasjoner i andre språk, f.eks. tysk.

4.5.2. Tilrettelegging av magnetbånd for fotosetting ("drivetape").

Etter at linjeinndelingen var klar skulle så materialet organiseres etter trykkeriets spesifikasjoner og legges opp på magnetbånd.

Tegnene som den aktuelle fotosetteren har til rådighet er organisert på såkalte fonter, med inntil 112 tegn på hver. De vanligste tegn er lagt på 3 parallelle fonter, en for vanlig skrift, en for kursiv og en for halvfet. Dessuten har vi til rådighet 2 fonter med mer spesielle tegn, hvorav den ene er lagt opp med spesielt henblikk på dette prosjektet.

Angivelsen av hvilken font et tegn skal hentes fra skjer ved funksjonskoder, noe i likhet med det prinsipp vi har brukt i databanken. Når riktig font er angitt, følger så koden for det eller de tegn som skal hentes fra denne fonten, deretter ny fontangivelse osv. Systemet har så pass mange felles trekk med vårt funksjonskodeopplegg at det gikk forholdsvis greitt å tilrettelegge materialet på denne måten.

Magnetbåndet ("drivetapen") skrives i såkalt TTS-kode som er en 6-bits kode og dermed bare har plass til 64 ulike tegn. Dette gjør det nødvendig med nokså mye skift (f.eks. for store og små bokstaver) og funksjonskoder. Et problem i denne forbindelse har vært å kontrollere innholdet av dette magnetbåndet i forbindelse med uttesting av programmet. Kontroll på grunnlag av "dump" fra båndet er både spesielt tidkrevende og forholdsvis utsatt for feil. Den virkelige kontroll har derfor måtte vente til det aktuelle prøvemateriale var kjørt gjennom trykkeriets fotosetter (i Stockholm).

Det er utdrag fra en slik trykkprøve som er gjengitt i vedlegget.

jo'rd.plante -a/-en BOT. plante der berre underjordiske organ overlever ei ulagleg årstid (vinteren). J.ene blir rekna som ein eigen *livsformtype* //*geofytt*. Jf. *kryptofytt* EAU [sv. *geofyt*; da. *j.*, *geofyt*; eng. *geophyte*; ty. *Erdpflanze* f, *Geophyt* m; isl. *jarðplanta* f]

jo'rd.potte -a DYRK.TEKN. (til å plante i:) blokk av samanpressa jord el. jordblandingar (med gjødsel) AAA [sv. *jordkruka*; da. *j.*, *gødningspotte*; eng. *soil block*, *soil pot*; ty. *Erdtopf* m, *Presstopf*; isl. *moldarpottur* m]

jo'rd.pro.fil -et/ -en JORD. 1) loddrett snitt av jordlag, vanl. frå yta og ned mot undergrunnen 2) skjematisk teikning av snitt som nemnt under 1 ///OEI [sv. *markprofil*, *j.*, *jordmånsprofil*; da. *j.*, *jordbundsprofil*; eng. *soil profile*; ty. *Bodenprofil* n; isl. *jarðvegssnið* n]

jo'rd.prøve -a/ -en DYRKJORD. prøve av jord som skal nyttast til kulturar, utteken til analyse og gransking AAA [sv. *jordprov*; da. *j.*; eng. *soil sample*; ty. *Bodenprobe* f; isl. *jarðvegssýni* n]

jo'rd.prøve -a/ -en JORD. jord som er teken ut or ei større mengd, og som skal nyttast til analysar og granskingar. Jf. *jordanalyse* IAA [sv. *jordprov*; da. *j.*; eng. *soil sample*; ty. *Bodenprobe* f; isl. *jarðvegssýni* n]

kalori' -en [lat. *calor*/varme] FYS.KJEM. (internasjonal fork. *cal*, tidlegare nytta i no. *kal*) 1) måleining for varmemengd a) (cal[illegible], def. av *International Conference on Properties of steam*, 1956:) varmemengd som er lik 4,1868 J b) (def. av *National Bureau of Standards*, USA:) varmemengd som er lik 4,184 J //*termokjemisk kalori* c) (cal_{15}) den varmen som trengst for å auke temperaturen til 1 gram luftfritt vatn frå 14,5 til 15,5 °C ved eit konstant trykk på 1 atm., dvs. 4,1850-4,1860 J ///VHA 2 a) mål for energi som kjem fram når næringsemne blir omlaga i ein organisme; varmemengd som trengst til å varme opp 1 g vatn frå 14,5 til 15,5 °C b) næringsmengd som kan produsere 1 cal som nemnd under 2 a [sv. *k.*; da. *k.*, *kalorie*; eng. *calorie*, *calory*; ty. *Kalorie* f; isl. *kaloria* f, *hitaeining* f]

kalori.me'ter -(e)ret; -ter/ -tre, -tra/ -trene FYS.KJEM. 1) apparat til å måle varmemengd (t.d. smeltevarme, reaksjonsvarme (forbrenningsvarme)) OHO.VHA 2) apparat til å måle *spesifikk* *varmekapasitet* /// *varmemåler* (-målar, -mælar) OHO.VHA [sv. da. *k.*; eng. *calorimeter*; ty. *K.* n; isl. *varmamælir* m]

kalori.metri' -en FYS.KJEM. lære om måling av varmemengder //*varmemåling* (-mæling) OHO [sv. da. *k.*; eng. *calorimetry*; ty. *Kalorimetrie* f; isl. *varmamæling* f] [illegible]isk /k. *brennverdi* [sv. da. *k.*; eng. *calorimetric(al)*; ty. *kalorimetrisch*]

kalsedo'n -et [gr. *khalkedon*, etter eit gammalt gr. namn på ein by ved Bosporos] JORD. kryptokrystallinsk mineral som høyrer til gruppa kvarts (SiO_2) og vanl. er gjennomskinleg og gråvore, men som òg kan ha varietetar med vakre fargar OHØ [sv. da. *kalcedon*; eng. *chalcedony*; ty. *Chalzedon* m]

kalsi- *smsl.* [lat. *calx*/kalk(stein)]

kalsi.fer.o'l -en [gr. *pherein*/bere] BIOL.KJEM. ($C_{28}H_{43}OH$) feittløyseleg, umetta krystallinsk alkohol som m.a. finst i mjølk og tran, og som ein får når ultrafiolette strålar påverkar ergosterol //*ergokalsiferol*, D_2*vitamin* [sv. da. *ergokalciferol*; eng. *calciferol*, *ergocalciferol*; ty. *Ergokalziferol* n; isl. D_2*vitamin* n, D_2*fjörvi* n]

karbo.hydra't -et KJEM. kvar av ei rekkje sambindingar som er samansette av grunnstoffa karbon (C), hydrogen (H) og oksygen (O), og som kan vere sukkerarter, stoff som er bygde opp av sukkerarter el. stoff som står sukkerartene kjemisk nær. Den allmenne formelen er $C_n(H_2O)_m$ //*kolhydrat* (*kull-*) OHO.OME [sv. *kolhydrat*; da. *kulhydrat*; eng. *carbohydrate*; ty. *Kohlenhydrat* n; isl. *kolvetni* n]

karbo.hydra't.fôr -et HUSD.DYRK. proteinfattig kraftfôr av vegetabilsk opphav, i hovudsaka av stive (korn, manioka m.m.) el. sukker (melasse m.m.) //(for:) *kolhydratfôr* (*kull-*) FSU [sv. *kolhydratfoder*; da. *kulhydratfoder*; eng. *carbohydrate concentrate*; ty. *Kohlenhydratfutter* n, *Karbohydratfutter*; isl. *kolvetnafóður* n] **forklistra** (-et) **k.** HUSD. karbohydratfôr som er kokt og torka, her i landet i hovudsaka kveite og havre FSU [sv. *kokt torkad cerealiefoder*; da. *forklistret kulhydratfoder*; eng. *precooked cereals*; ty. *gekochtes und getrocknetes Zerealienfutter*]

karb.oksi'n -en/ -et [fork. for det kjemiske namnet, s. ned.] VERN.KJEM. *systemisk soppmiddel med verknad mot sotsoppar //*2,3-dihydro-6-metyl-5-fenyl-karbamoyl-1,4-oksatiin* LSU [sv. da. *karboxin*; eng. *carboxin*; ty. *Carboxin* n]

karb.oks.y'l -et; -/ -er, -a/ -ene KJEM. (-COOH) einverdig radikal som finst i alle organiske syrer OHO [sv. da. *karboxyl*; eng. *carboxyl*; ty. *Karboxyl* n]

karb.oks.yl.a'se -en KJEM. enzym som aktiviserer utviklinga av *karbondioksyd frå *karboksyl i ketonsyrer. Slike reaksjonar er ein viktig del av alkoholgjæringa og omlaginga av karbohydrat i kroppen: pyrodruesyre ($CH_3 \cdot CO \cdot COOH$) blir til karbondioksyd og *acetaldehyd OHØ [sv. *karboxylas*; da. *karboxylase*; eng. *carboxylase*; ty. *Karboxylase* f]

karb.oli'n.eum -et/ -en [lat. *carbo*/kol] KJEM.BYGN. produkt av steinkoltjøre, nytta som impregneringsmiddel for trevyrke OHJ [sv. da. *k.*; eng. *carbolineum*; ty. *K.* n; isl. *karbólín* n]

mano.me'ter -(e)ret; -ter/ -tre, -tra/ -trene [gr. *manos*/tynn] FYS. apparat til å måle gasstrykk i stengde rom (til skilnad frå barometer, som syner trykket i open atmosfære) // *trykkmåler* (-målar, -mælar) VHA [sv. *m.*, *tryckmätare*; da. *m.*; eng. *m.*, *pressure gauge*; ty. *M.* n, *Druckmesser* m; isl. *þrýstingsmælir* m]

mano.me'trisk -; -e FYS. som gjeld trykk i gass(ar) og væske(r) /*m. lyftehøgd* dss. *pumpehøgd* /*m.* *sugehøgd* (tyd. 2) /*m.* *trykkhøgd* (tyd. 2) ///SCH [sv. da. *m.*; eng. *manometric*; ty. *manometrisch*]

mansa'rd.tak -et [etter den franske arkitekten *François Mansarde* (Mansart) (1598-1666)] BYGN. tak der takflatene er brotne slik at nedre del har større fall enn øvre OHJ [sv. *m.*; da. *mansardtag*; eng. *gambrel roof*, (valma m.:) *mansard roof*; ty. *Mansardendach* n; isl. *Mansardþak* n]

maqui's -en [fr. 'snar', 'kjerr'; utt.: *maki:*'] dss. *maki*.

ma'r.bakke -en [gno. *marr*/sjø, hav] LAND. bratt skråning utanfor strandområde i sjø el. vatn //*molbakke*, *mekbakke*). Jf. *1 bakke 1 c* KJE [sam. *bæg'ku*; sv. *bråddjup*, (*brant sluttning längs strandbrädden*); da. (*skråning fra land ud til havdybet*); eng. *shelf*, (am.:) *drop-off*; ty. *Meeresgesenk* n; isl. *marbakki* m]

ma'r.buske -a [trul. sv.] SKOG. undertrykt plante, svært lita i høve til alderen, ofte med vid krone OBO [sv. *m.*; da. *undertrykt plante*; eng. *severely suppressed tree*; ty. *unterdrückte Pflanze*]

ma're.halm -en [gno. *marr*/hav; jf. ty. *meer* & lat. *mare*] BOT. 1) (*Ammo'phila* Host.) slekt i grasfamilien: (fl.:) planter av denne slekta 2) (*Ammo'phila arena'ria* Link) art av denne slekta: plante med krypande jordstengel, høgt stivt strå, og slette blad, mykje brukt til å binde flygesand // *ranlig* (-leg) *marehalm*, *hjelmegras* ///*sandrør* (*røyr*) ORF [sam. *rid'dohdvrar*; sv. 2) *sandrör*, *marhalm*; da. *hjelme*; eng. *marram grass*; ty. 1) *Helmgras* n, 2) *Strandhafer* m; isl. *sandhálmur* m]

marg -en dss. *merg*.

marg -en dss. *merg*.

marg- *smsl.* *merg-*.

margari'n -en [fr. *margarine*, av *acide margarique*, av gr. *margaron*/perle (pga. fargen), opph. eit orientalsk ord] MEIER. næringsmiddel som ein lagar med å blande reine, raffinerte feittstoff, mjølk, salt m.m. til ein emulsjon som etter temperering og brå nedkjøling blir elta til ein smørliknande masse //(stundom:) *smør* [sam. *mår'går*; sv. *m.*; da. *margarine*; eng. *margarine*, *m.*; ty. *Margarine* f]

margi'n -en [lat. *marginem*, til *margo*/kant, rand, grense] ØKON. prisskilnad; depositum; garantisum; MEIER. (koll.:) normale kostnader per liter innsett *mengd av mjølk el. per kilogram produkt, rekna ut på grunnlag av eit *normalbudsjett RBR [sv. (fl.:) *normalkostnader*; da. *margen*, *m.*, (fl.:) *normaldriftsomkostninger*; eng. *working margin*, *claculated working expenses for dairies with normal output*; ty. *Normalkosten* fl.]

ma'sse.ein.ing -a *-enhet* -en FYS.KJEM. eining som blir nytta når ein skal måle massen av noko /*atomær m.* (symbol: u) (etter 1960:) internasjonal eining for atommasse. 1 u = 1/12 av massen av karbonisotopen ^{12}C (dvs. $1{,}66032 \cdot 10^{-27}$ kg). Tidlegare nytta ein den fysiske atomvektsskalaen, der måleininga var 1/16 av massen av oksygenisotopen ^{16}O, el. den kjemiske atomvektsskalaen, der måleininga var 1/16 av den blandinga av oksygenisotopar som finst i naturen ///VHA [sv. *massenhet*, *atomär massenhet*; da. *masseenhed*, *atommasseenhed*; eng. *mass unit*, *atomic mass unit*; ty. *Masseneinheit* f, *atomare Masseneinheit*; isl. *massaeining* f]

ma'sse.en.het -en dss. *masseeining*.

ma'sse.fabr.ikk -en SKOG.TEKN. fabrikk som produserer fibermasse av trevyrke ØME [sv. *massafabrik*, *massabruk*; da. *fibermassefabrik*; eng. *pulp mill*; ty. *Zellstoff-Fabrik* f]

ma'sse.fag.verk -et *fagverk 2*.

ma'sse.flis -a dss. *1 flis 2*.

ma'sse.flis -a dss. *celluloseflis*.

mi'dd.el.feil -en *medel-* MAT. (i statistikk:) mål for presisjon etter formelen m = grenseverdien for [illegible] $\Sigma\, \varepsilon^2/n$ når n går mot uendeleg, der m 32800 er m., ε er sanne feil og *n* er talet på observasjonar. Sannsynet for å gjere feil med talverdi ε mindre el. lik talverdien av m. er 0,683 dersom feila er fordelte i samsvar med den gaussiske feillova //*standardavvik* IRE [sv. *medelfel*, *standardavvikelse*; da. *middelfejl*, *standardafvigelse*; eng. *standard error*, *mean square error*, *standard deviation*; ty. *mittlerer Fehler*, *Standardabweichung* f, *mittlere quadratische Abweichung*, *mittlerer zu befürchtender Fehler*; isl. *meðalskekkja* f]

mi'dd.el.høgd -a *medel-*, *-høyde* -en SKOG. gjennomsnittshøgd for tre (t.d. for trea i eit bestand). Med m. meiner ein oftast *grunnflateveegen m.* (òg kalla *Loreys m.*) der ein bruker grunnflatene til trea som vekter i utrekninga. *Aritmetisk m.* blir sjeldan brukt AFI [sv. *medelhöjd*; da. *middelhøjde*; eng. *mean height*, *average height*; ty. *Mittelhöhe* f; isl. *meðalhæð* f]

mi'dd.el.korn.stor.leik -en *medel-*, *-størrelse* JORD. (fork.: *md.*) kornstorleik som er slik at halvparten av materialet er finare og halvparten grovare OEI [da. *middelkornstørrelse*; eng. *median particle size*; ty. *mittlere Korngrösse*; isl. *meðalkornastærð* f]

mi'dd.el.pro.duk.sjon -en *medel-* dss. *middeltilvekst*.

mi'dd.els -; - *medels* som har gjennomsnittsstorleik (-verdi); som er (ligg) om lag midt mellom to ytterpunkt /*m. moldhaldig* *mineraljord* /*m. stiv leire* dss. *leire 1 a* /*m. tett skjerm* s. *skjerm 2 b* [sv. *genomsnittlig*, *medel-*; da. *middel*, *middel-*; eng. *average*, *medium*; ty. *durchschnittlich*, *mittel-*; isl. *meðal-*]

mi'dd.el.stamme -a/ -en *medel-*, *-stomn* -en SKOG. (tre)stamme som svarar til gjennomsnittstreet i ein skogteig (t.d. i eit bestand) når det gjeld volum, diameter, grunnflate, høgd el. l. AFI [sv. *medelträd*, *medelstam*; da. *middelstamme*; eng. *mean tree*, *average tree*; ty. *Mittelstamm* m; isl. *meðaltré* n]

ohm (-en); - [etter den tyske fysikaren *G. S. Ohm* (1789-1854)] FYS. (symbol: Ω) eining for elektrisk motstand (resistans). 1 ohm (= 1 volt/ampere) er motstanden i ein leiar der ein volt produserer ein straum på ein ampere. Ohm blir også nytta som eining for *impedans og *reaktans VHA [sv. da. eng. *o.*; ty. *O.* n]

ohm.me'ter -(e)ret; -ter/ -tre, -tra/ -trene FYS. instrument som måler elektrisk motstand i ohm [sv. da. eng. *o.*; ty. *O.* n, *Ohmmesser* m]

ohm.sk mo't.stand dss. *resistans*.

-oid *smsl.* [gr. *-o-*, bindevokal & *-eides*/som liknar på, av *eidos*/form] t.d. i *diploid*, *haploid*.

o.i'd.ium -iet; -ium/ -ier, -ia/ -iene [gr. *oon*/egg] BOT. tynnveggja spore, laga når soppbyfar deler seg. Jf. *artrospore* [sv. da. eng. *o.*; ty.

EXPERIMENTS WITH ODD LANGUAGES

Eric Grinstead

Asian languages may seem exotic, with the special difficulties of strange cultural vocabulary, script, and the few opportunities of communicating in them. However, Chinese, Arabic, Indonesian, Hindi, and Persian are not only languages with long cultural histories, but also very influential linguae francae in their political and cultural spheres today.

The problems in Chinese seem to be the reverse of those studied by Europeans. One begins with basic roots, neatly separated, and a "machine translation" must show how they are put together. The Chinese written language, being composed of unit areas, has always been much better organised grammatically, phonetically, and bibliographically.

The line-plotter has been used for Chinese since 1968, at least, and now we have the data screen, the matrix printer, jet plotters, and, in Japan, the line-printer. There is teaching in the University of Illinois. Coding in four-digit numbers is well established, though the usual goal, to input all available texts, would be too much for foreign students to attempt, but could be easily achieved by Chinese students.

I am interested in: (1) Design of special vocabularies to be put into a small computer, in compact Chinese and English.

(2) A database for finding quotations, built on the existing concordances and indexes. Two rather rare characters could well establish a unique original source. This is an attempt to bypass the dictionary, and go to the original text, for which there is often a standard English or French translation.

(3) Index to the second character of a two-character compound. This is to speed up technical translation by reducing look-up time.

Alphabetic scripts present few problems. Golfballs are available for most languages already. Here I am interested in inputting the basic dictionary, and finding some algorithm that will enable the text to be compacted as much as possible for use in small desk computers.

NYORD-REGISTRERING I DATABASE.

Kolbjørn Heggstad Harald Solevåg

1. Innledning

Norsk språkråd og Norsk Leksikografisk Institutt, Universitetet i Oslo, arbeider begge med innsamling av materiale for å belyse nyordstilfanget i norsk. Materialet er primært et leksikalsk utvalg: ord og fraseologi med betydninger og bruksmåter som før ikke er registrert i det hele tatt, eller som ikke ansees registrert i tilstrekkelig bredde og dybde. De nevnte institusjonene har knyttet til seg et nett av kontaktmenn rundt omkring i landet, som har til oppgave å ekserpere i aviser, tidsskrifter, lærebøker o.l. og sende dem inn til Språkrådet eller NLI, som så legger til rette dataene, lemmatiserer oppslagsordene og klassifiserer dem grammatisk. Deretter blir materialet gjort maskintilgjengelig og databehandlet ved Nordisk institutt, PDS, Universitetet i Bergen.

Tidligere blei nyord-ekserptene skrevet ut på kartotekkort og ordnet alfabetisk i et tradisjonelt arkiv. Denne arbeidsgangen hadde vesentlige ulemper, både med hensyn til arbeidsmengden og tilgangen til data.

I 1969 blei det etablert et samarbeid med Nordisk institutt, Universitetet i Bergen, som tok på seg å utarbeide et datamaskinelt opplegg for nyord-registrering.

2. Prosjektbeskrivelse

Prosjektet baserer seg på en "manuell" utplukking av leksikalske enheter som deretter blir registrert og databehandlet. Selv om en i dag gjør store tekstsamlinger maskintilgjengelig for databehandling, ville en automatisk registrering fra de samlede datamengder en arbeider med i dette prosjektet være urealistisk, både på grunn av datastørrelsen og de kompliserte søkerutiner.

2.1 Ekserpering

Medarbeiderne som ekserperer data markerer direkte i kildene

1) det aktuelle ordet/uttrykket som skal tjene som oppslag,
2) og hvor stor kontekst som skal være med i hvert enkelt tilfelle.

I samme kontekst kan en markere flere ord/uttrykk, slik at samme kontekst skal kunne brukes til å belyse flere oppslag.

2.2 Klassifisering

Alle oppslagsord får satt på en eller flere koder etter et klassifiseringssystem som er utarbeidet for å muliggjøre maskinell søking etter ordklasse, ordsammensetningstype og en lang rekke andre grammatiske, ortografiske og stilistiske kriterier. Systemet kan stadig utvides, og inneholder for tida ca. 100 ulike koder. (Se eksempel i Bilag.)

2.3 Registrering

Før en bestemte seg for et datamaskinelt opplegg av nyord-arkivet, blei alle eksempler skrevet ut på kartotekkort. Alle oppslagsord måtte ha eget kort påført språklig klassifisering, kildenavn, kildehenvisning, (dato, år, side, spalte) eventuelt forfatternavn, stofftype, målform (bokmål, nynorsk) og kontekst.

I det nåværende punche-opplegg skrives konteksten inn med en markering direkte satt til kontekstformen av oppslagsordet som fører til at riktig oppslagsform blir generert. Den språklige klassifisering av oppslagsordet blir også satt til kontekstformen.

Dersom det er flere eksempler fra samme kilde, oppgis bare kontekstopplysningene (kilde, forfatter osv.) ved første eksempel. Seinere oppgis bare forandringer, f.eks. ny sideangivelse osv.

2.4 Databehandling

Siden prosjektet startet i 1969, har flere ulike programsystem vært i bruk, både program spesielt utviklet ved PDS, og et generelt informasjonssystem (IBM's dokumentsøkesystem STAIRS).

Av krav en må stille til et leksikalsk datasarkiv av denne type, er at det må være lett å oppdatere, og at det finnes effektive søkemetoder. Videre må det være mulig å få data presentert i et formålstjenlig format. Med i et fullstendig opplegg hører også rutiner for innlesing og korrigering av data.

Når det gjelder interaktiv søking i databasen som nå er under utvikling, har følgende blitt prioritert:

a) søking etter et bestemt lemma for å finne belegg eller andre opplysninger under dette.

b) søking etter alle lemma med en bestemt klassifisering (f.eks. ordklasse, sammensetningstype) eller en viss kombinasjon av klassifiseringer (jf. 2.2).

Fullstendige utlistinger i forskjellige sorteringer eller andre større systematiske utvalg fra databasen vil bli mulig ved hjelp av et sett med brukerprogrammer. Data kan tas ut på papir eller på mikrokort.

3. Database

Hva er hensikten med å bruke et database-system? For å besvare dette skal vi først kort resymere den tradisjonelle måten å programmere på, som kalles den programorienterte metoden.

Når programmereren får seg forelagt en oppgave, har han en tendens til først å begynne med å utarbeide programlogikken, for deretter å tilpasse dataene til programmet. Dette vil utvilsomt resultere i effektive program, men datastrukturen vil bli nøye knyttet til programmet, slik at når behovet for å utvide systemet med flere program oppstår - og det skjer alltid -, så kan det bli problematisk å få dataene til å passe til det nye programmet. Løsningen er som oftest å forandre data-strukturen, men det medfører at dataene må lagres flere ganger og til dels uøkonomisk. F.eks. vil ordlister og konkordanser kreve mangedobbel lagerplass i forhold til teksten de er basert på. Endelig har vi problemet med korreksjoner og oppdateringer: når dataene er lagret flere ganger må de selvsagt korrigeres på samtlige steder. Og korreksjonen av en konkordans er vel for de fleste av oss en lite fristende oppgave.

En mulig løsning på slike problem er å bygge på database-metoder. Her legger man hovedvekten på data- og lagrings-strukturen.

Man forsøker å strukturere dataene på en slik måte at man

a) Unngår dobbellagring.
   Dette innebærer ikke bare økonomisering med lagerplass, men forenkler også data-administrasjonen.

b) Forenkler oppdateringsmulighetene.

c) Forenkler programforandringer/-utvidelser.

Litt forenklet kan man si at med den programorienterte metode er programmene målet, mens dataene er midlet. Med database-metoden er det omvendt, her er dataene målet, mens programmene er midlet.

3.1 Hvorfor DMS 1100?

Vi skal ikke gå noe særlig inn på hvorfor vi valgte DMS 1100, men én grunn er jo innlysende: DMS 1100 er implementert på vårt dataanlegg. I stedet vil vi kort skissere alternativene:

a) "Hjemmelaget" system.

Med "hjemmelaget" menes et system som kun er basert på et kjent programmeringsspråk. Dette har vi forsøkt med betinget suksess.

Ulempene er helt åpenbare:

1) Stor arbeidsmengde.

2) Lite generelle program. Man vil uvilkårlig ha et spesielt prosjekt og en spesiell arbeidsrutine i tankene under konstruksjonen. Følgelig vil systemet ta farge av dette, og det blir vanskelig å tilpasse systemet nye rutiner og eventuelle beslektede prosjekt.

b) Informasjonssystemer (dokumentsøkesystemer).

Fordelene med disse systemene er åpenbare: de er forholdsvis enkle å implementere, søkeprosedyrene er ferdige så man slipper programmering av disse.

Disse systemene er utviklet for søking i dokumenter, brev etc. Man bruker nøkkelord for å finne riktig dokument, nøkkelordene er derfor bare et hjelpemiddel i søkeprosessen. Når det gjelder leksikalske ordarkiv, er ordene det essensielle, mens dokumentene - kontekstene - bare er tilleggsopplysninger for å belyse ordene. Eventuelle opplysninger som kan gis i et informasjonssystem, er opplysninger om selve dokumentet. I vårt tilfelle skal også mange opplysninger gis om nøkkelordet. Informasjonssystemene må derfor nærmest "misbrukes" for å passe til våre formål.

Et annet moment er størrelsen. NYORD-prosjektet omfatter for tida ca. 25 mill. tegn, derfor er det viktig at man lagrer dette så komprimert som mulig. Et dokument, i vårt tilfelle en kontekst, kan inneholde flere ekserperte ord. Dette tilsier at p.gr.a. opplysninger som skal med om hvert enkelt nøkkelord, må et dokument gjentas like mange ganger som vi har ekserperte ord fra det aktuelle dokument.

3.2 DMS 1100

DMS 1100 er Univac's CODASYL-basesystem, og består av 3 språk:

Data_Definition_Language (DDL) som brukes for å definere databasen.

Data_Manipulation_Language (DML) som brukes til lasting, søking og oppdatering.

Utility_Language som brukes til forskjellige hjelpefunksjoner: pack, dump etc.

Vi skal ikke gå noe inn på den datatekniske siden av DMS 1100, men heller konsentrere oss om konstruksjonen av databasen. Det følgende har derfor først og fremst sammenheng med DDL.

Før vi ser nærmere på vårt opplegg, skal vi kort forklare noen begreper:

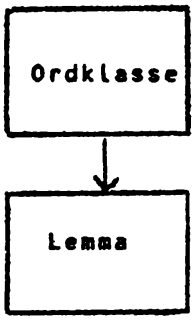
Post-type betegner samlingen av dataelementer av en gitt type, og blir anskueliggjort med et rektangel.

Post-forekomst betegner selve dataelementet av en gitt type, og anskueliggjøres med en sirkel. Dvs. post-type er betegnelsen på en samling post-forekomster.

Logiske sammenhenger mellom post-typer representeres grafisk med pil.

La oss belyse dette med et eksempel:

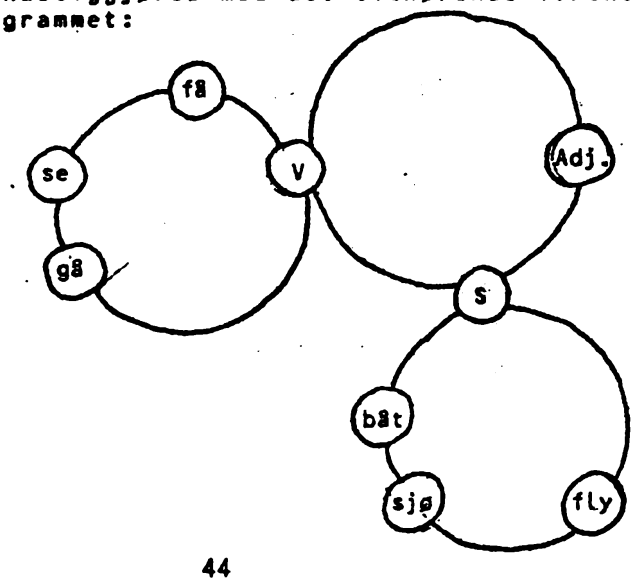
Typediagrammet:



skal forståes slik:

Post-typen Ordklasse ersamlingen av ordklasser, de enkelte ordklasser (f.eks. verb, substantiv etc.) blir post-forekomster. På samme måte med post-typen Lemma, som er et samlenavn på en rekke ord (lemma) og de enkelte lemmaene blir post-forekomster av post-typen Lemma.

Pila - som indikerer den logiske sammenhengen mellom post-typene - betyr at lemmaene grupperes under sine respektive ordklasser. Dette kan best anskueliggjøres med det tilhørende forekomst-diagrammet:



Vi skal nå punktvis se på de forskjellige faser i konstruksjonen av databasen:

1. Definere informasjonen som skal lagres. (Jfr. prosjektbeskrivelsen).

2. Samle informasjonen i poster (records).

3. Definere logiske sammenhenger mellom post-typene.

4. Bestemme lagringsmetodene til postene.

5. Bestemme lagringsstrukturen.

Pkt. 4 og 5 er først og fremst datateknisk interessant og dessuten maskinavhengig, slik at det ikke skal tas opp her. La oss heller se litt på pkt. 2 og 3 anvendt på vår database (se vedlagte typediagram).

Postene er:

GRAM: Denne post-typen inneholder de grammatikalske 2-bokstavers kodene. Post-typen inneholder derfor 29 x 29 post-forekomster.

LEMMA: Inneholder lemma-formen av de ekserperte ord med frekvenser, samt de grammatikalske kodene tilhørende dette lemma.

TYPE: Post-typen har som forekomster kontekstformen av de ekserperte ord med frekvens.

KONTEKST: Under denne post-typen lagres kontekstene.

KILDE: Denne post-typen inneholder kilde-henvisninger til tekstene (navn, nummer/dato, side/spalte).

GENRE: Stofftype.

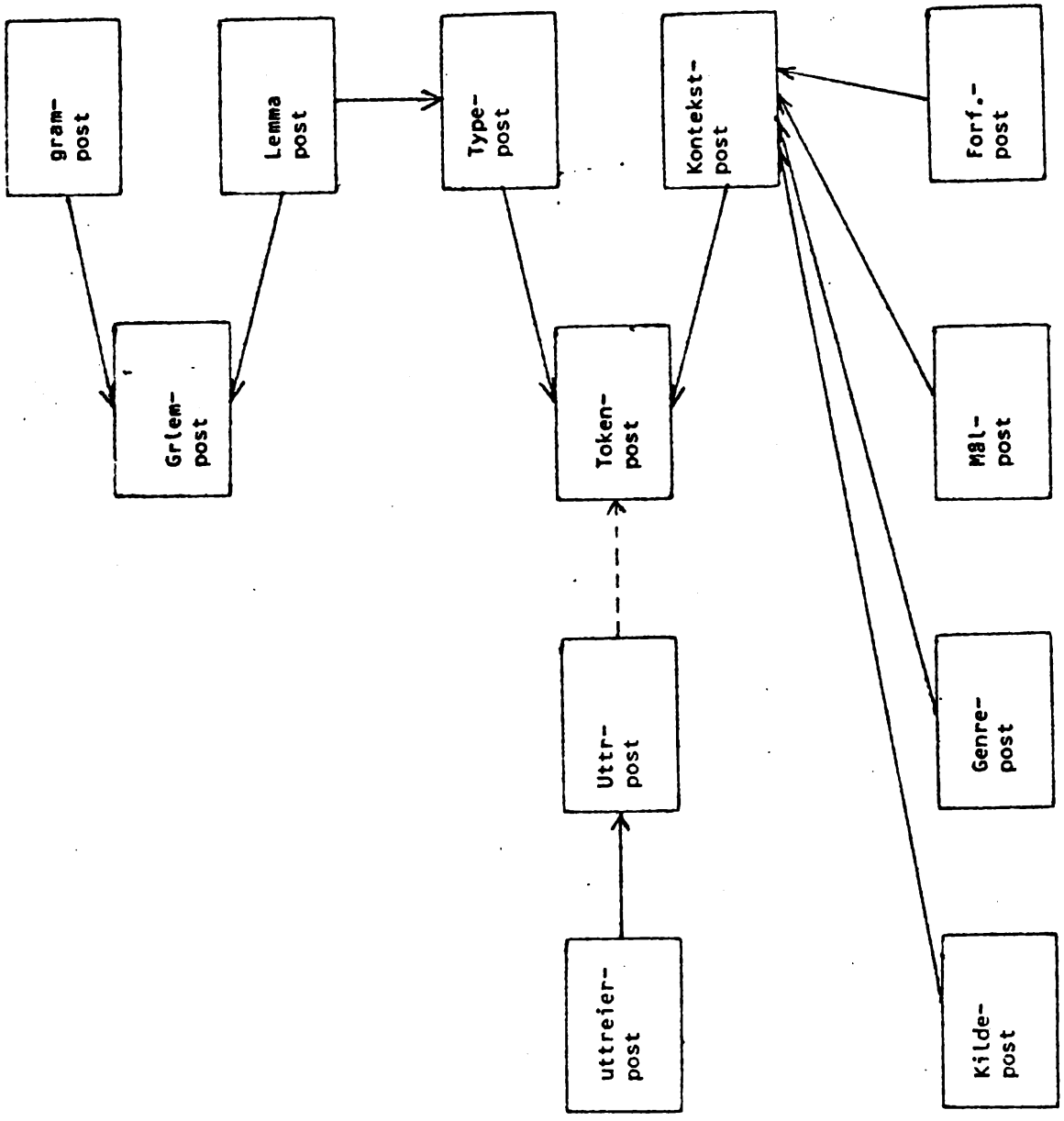
MÅL: Bokmål eller nynorsk.

FORFATTER: Forfatternavnet eller blank hvis forfatter ikke er oppgitt.

De øvrige post-typene inneholder ingen direkte språklig informasjon, men brukes til strukturering av dataene.

Av type-diagrammet ser vi at kontekst-formene er gruppert under sine respektive lemma. Forekomstene av Token-post-typen er gruppert både under Type-post og Kontekst-post. Postene inneholder ingen bruker-informasjon, men virker som en kopling for en "mange-til-mange"-forbindelse: flere ord kan være ekserpert i samme kontekst, et ord kan være ekserpert i flere kontekster. Det samme forholdet har vi mellom Lemma-post og Gram-post.

T Y P E D I A G R A M



B I L A G.

Det følgende er et eksempel på noen enkle spørsmål som blei stilt over terminal til en prøveversjon av databasen.

Databasen inneholder i dette tilfelle bare ca. 600 ord med kontekst.

Det er to typer spørsmål som er demonstrert:

1. Spørsmål etter ord som tilhører en bestemt kategori (jf. 2.2).

2. Spørsmål etter bestemte ord for å få et fullstendig ekserpt.

Ad. 1: Spørsmål blir innledet med '>g' og en må gi opp klassifiseringskode.
Flere betingelser kan knyttes sammen med &.
En får da ord som oppfyller begge betingelser.
(Eks. '>g tt&g og').

Følgende klassifikasjoner er brukt i eksemplene:

ag	Anglisismer
an	Det ekserperte ordet står i anførsel (angis alltid).
cx	Felleskjønn, usammensatt.
pv	Passiv, ikke påfallende (angis alltid).
ss	Både forledd og etterledd sammensatt.
tt	Teknisk terminologi og andre ord som har særlig tilknytning til yrke.
vx	Verb, usammensatt.

Ad 2. Spørsmålet blir innledet med '>l'. (Eks. '>l høring')

(Vi ønsker en fortegnelse av samtlige tekniske termer:)

```
>g tt

permeabel
seismikk
wildcat
aluminatsement
smeltesement
avionikk
spin-off-produkt
cushiongummi
datalogger
temperbehandling
duktilitet
overelde
hydrologi
nekton
benthos
nekton
feromon
benthos
planktonisk
blow_out
paleooseanografi
litosfære
_syndrom
etologi
etologisk
mikrofossil
geokjemisk
shelf-is
nefridium
metanefridium
protonefridium
protostom
protostomium
deuterostomium
koordinatpunkt
megapopulasjon
antall-lemma:36
```

(Vi ønsker en liste over de termene som er klassifisert som anglisismer:)

```
>g tt&g ag

wildcat
spin-off-produkt
cushiongummi
datalogger
blow_out
shelf-is
antall-lemma:6
```

(Vi ønsker en liste over de termene som er brukt i teksten i anførselstegn:)

```
>g tt&g an

spin-off-produkt
blow_out
antall-lemma:2
```

(Vi ønsker en liste over alle anglisismer:)

```
>g ag

wildcat
turnover
drive
gamble
spin-off-produkt
cushiongummi
wild_session
høring
datalogger
monitoring
blow_out
non_profit_service
shelf-is
antall-lemma:13
```

(Vi ønsker å se det fullstendige eksempt av "høring":)

```
>l høring

høring,b,cxagan


Denne diskusjonen var praktisk
talt fri for politiske problem-
stillinger.  Disse kom imidler-
tid i noen grad frem i en
"høring" om datapolitikk.
,TU,1975/31,7/2,Art.
```

(Vi ønsker en liste med sammensatte ord der
både forledd og etterledd er sammensatt:)

```
>g ss

undervannsfarkost
klarværutstyr
grunthavsområde
overrislingsanlegg
antall-lemma:4
```

(Vi ønsker en liste over de ord som inngår
i en bestemt passivkonstruksjon:)

```
>g pv

droppe
initiere
hevde
overelde
antall-lemma:4
```

```
>l initiere

initiere,b,vxpv
To perspektivanalyser for utviklingen av Bergens-
regionen er nettopp fullført. Den ene analysen er
initiert av Generalplanutvalget og tar for seg
utviklingen av sysselsetting og næringsliv i Bergen.
,TU,1975/25,22/1,Art.
```

NAVF'S EDB-SENTER FOR HUMANISTISK FORSKNING
NORGES ALMENVITENSKAPELIGE FORSKNINGSRÅD
Villavei 10, 5000 Bergen — Telefon 21 00 40
Postadresse: Postboks 53, 5014 Bergen - Universitetet

Knut Hofland: Implementering av en metode for syntaktisk analyse av norsk.

Foredrag til nordiska datalingvistikdagar i Göteborg 10. - 11. oktober 1977.

---

1. INNLEDNING

Denne artikkelen gir en oversikt over et samarbeidsprosjekt mellom stipendiat Svein Lie, Nordisk institutt, Universitetet i Oslo og NAVF's EDB-senter.

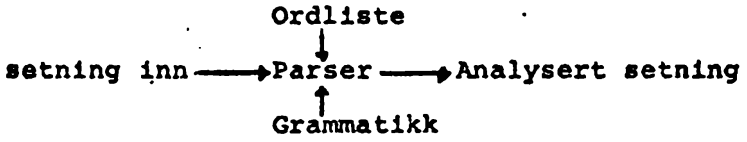
Formålet med prosjektet har vært å lage et programsystem som leser inn en vanlig norsk setning og som foretar en syntaktisk analyse av denne. Prosjektet har to hovedmål:

1. En ønsker å vinne økt innsikt i norsk syntaks ved å prøve ut grammatikken på et stort antall setninger og undersøke de ukorrekte analyser for å modifisere reglene og prøve på nytt.

2. Bygge opp et generelt programsystem for syntaktisk analyse som også kan nyttes av andre.

Prosjektet bygger på et notat av Martin Kay:"Morphological and syntactic analysis" utdelt på nordisk sommerskole i språklig databehandling sommeren 1974, samt et uferdig program av samme forfatter. Etter en første kontakt høst 75/vår 76 ble arbeidet ved NAVF's EDB-senter påbegynt i mai 1976. Sommeren 1976 var en første versjon av programmet klart. Dette opererte med en liten engelsk grammatikk beskrevet i Martin Kays notat og analyserte som kontrollsetning en setning fra et større eksempel i dette notatet. En norsk grammatikk ble så lagt inn og denne var i stadig utvikling høsten 1976/våren 1977. Våren 1977 ble programmet overført til Oslo slik at det nå kan kjøres både i Bergen og Oslo. NAVF's EDB-senter har bidradd i prosjektet med 2-3 månedsverk.

Oversiktsfigur:



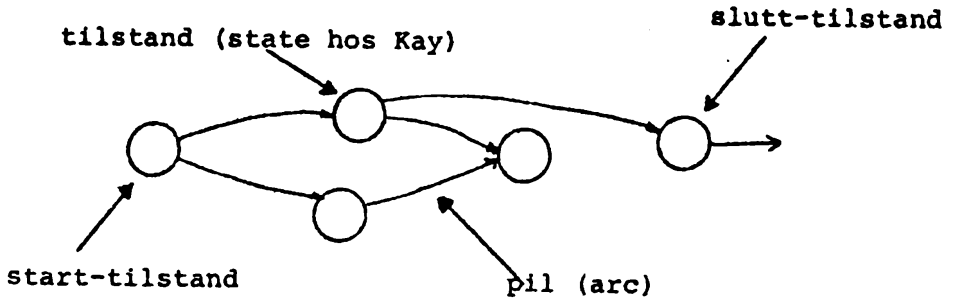
Siden det er det syntaktiske aspektet som i denne forbindelse er det mest interessante, har en holdt utenfor en morfologisk analyse. Ordlisten består derfor av ordformer med opplysninger om ordklasse og et sett med egenskaper som f.eks. kjønn, tall, person, transitiv o.l. For ord som ikke står i ordlisten må brukeren slå inn disse opplysningene og ordet vil da bli satt inn i ordlisten.

Eksempel på analyse.

```
GI SETNING :

>DA VI KOM FRAM BAD VI DIREKTØREN PÅ HOTELLET
>VERE SNILL OG GI OSS ET VERELSE MED BAD.

FØLGENDE ORD FINNES IKKE I ORDLISTE. ANGI ORDKLASSE NN.

VERE

>VERB INFINITIV HJVB COP
VERELSE

>SUBST NEUTR SG UBEST

S /
   •ADVL2
   ADVL
      •S
      S / ADV-LS
         •UKONJ
         UKONJ / ADVKNJ                          DA
         •SUBJ
         NP / NOMINATIV / BEST
            •OVERLEDD
            PRON / NOMINATIV                     VI
         •VERBFIN
         VF / PRET / INTRANS                     KOM
         •ADVL3
         ADVL
            •ADVERB
            ADV                                  FRAM
   •VERBFIN
   VF / PRET / SAMSEVB                           BAD
   •SUBJ
   NP / NOMINATIV / BEST
      •OVERLEDD
      PRON / NOMINATIV                           VI
   •OBJ
   NP / BEST / SG / MASK
      •OVERLEDD
      SUBST / MASK / SG / BEST                   DIREKTØREN
      •ADVL
      ADVL / PREPLEDD
         •PREP
         PREP                                    PÅ
         •STYRING
         NP / BEST / SG / NEUTR
            •OVERLEDD
            SUBST / NEUTR / SG / BEST            HOTELLET
   •OBJINF
   NP / NP-S/INF
      •INF-FRASE
      INF-FRASE
         •INFINITIV/OVERLEDD
         VERB / INFINITIV / HJVB / COP           VERE
         •PREDIKATIV
         ADJP
            •OVERLEDD
            ADJ                                  SNILL
      •SKONJ
      SKONJ                                      OG
      •INF-FRASE
      INF-FRASE
         •INFINITIV/OVERLEDD
         VERB / INFINITIV                        GI
         •INDIROBJ
         NP / BEST
            •OVERLEDD
            PRON / AKK                           OSS
         •OBJ
         NP / UBEST / SG / NEUTR
            •BEST
            ART / NEUTR / SG / UBEST             ET
            •OVERLEDD
            SUBST / NEUTR / SG / UBEST           VERELSE
            •ADVL
            ADVL / PREPLEDD
               •PREP
               PREP                              MED
               •STYRING
               NP / UBEST / SG / NEUTR
                  •OVERLEDD
                  SUBST / NEUTR / SG / UBEST     BAD
```

2. OVERSIKT OVER METODEN

Det gis her en oversikt over Martin Kays metode med de forandringer som er gjort av oss.
Grammatikken beskrives som et nettverk.



Et nettverk er et sett med tilstander (states) som er forbundet med (overgangs)piler (arcs). Pilene ut fra en tilstand gir de mulige etterfølgende tilstander. En tilstand som ikke har noen innløpende piler vil være en starttilstand. Tilsvarende vil en tilstand som ikke har noen neste tilstand, være en slutt-tilstand. Til enhver pil i nettverket vil det være et sett med betingelser som må være oppfylt dersom en skal ta denne veien i nettverket (dette settet kan være tomt og det finnes ubetingete overganger i nettverket). Til en pil kan det også høre et sett av aksjoner som skal utføres dersom denne overgangen i nettverket blir foretatt. En betingelse kan f.eks. være knyttet til ordklasse og egenskaper til et aktuelt ord i en setning mens aksjonen kan være å sette navn på et ord eller et setningsledd for senere å kunne bruke dette navnet for å sjekke egenskaper ved ordet. Denne fysiske sammenknytningen av et navn med en del av setningen kalles et register, og det vil bli opprettet nye registre underveis i analysen. Disse er metodens hukommelse og er en av de viktigste delene i hele prosessen.

Deler av nettverket som brukes flere ganger f.eks. for å analysere nominale uttrykk, skilles ut som egne undernettverk og det knyttes forbindelse fra hovednettverket til undernettverkene.

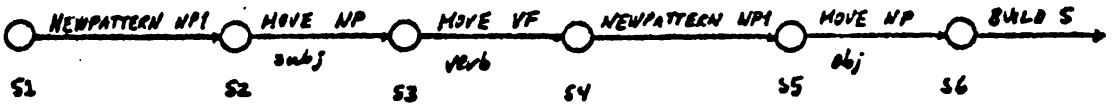
Det er 4 hovedtyper av overgangspiler i nettverket:

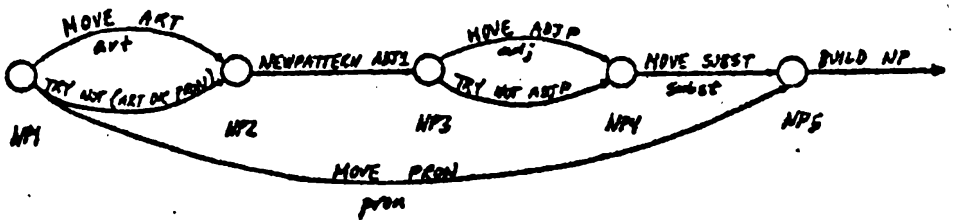
- overganger som godkjenner ord eller setningsledd på grunnlag av ordklasseopplysninger (MOVE, GO TO hos Kay) og går videre i setning
- overganger som starter søking etter nye konstituenter i et subnettverk (NEWPATTERN)
- overganger som gjør det mulig å gå ubetinget fram i nettverket (TRY)
- overganger som knytter sammen ord eller setningsledd til større enheter (BUILD)

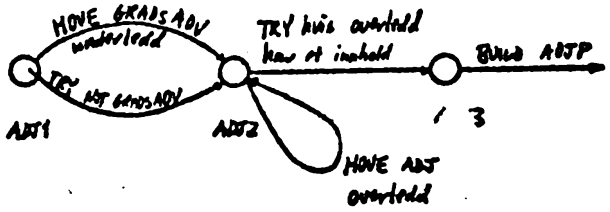
Vi har valgt å la NEWPATTERN bare være en pil som starter søking i et subnettverk. Undersøkelsene om en fant det en lette etter i subnettverket, gjøres i tilstanden etter NEWPATTERN pilen.

Eksempel på en grammatikk:

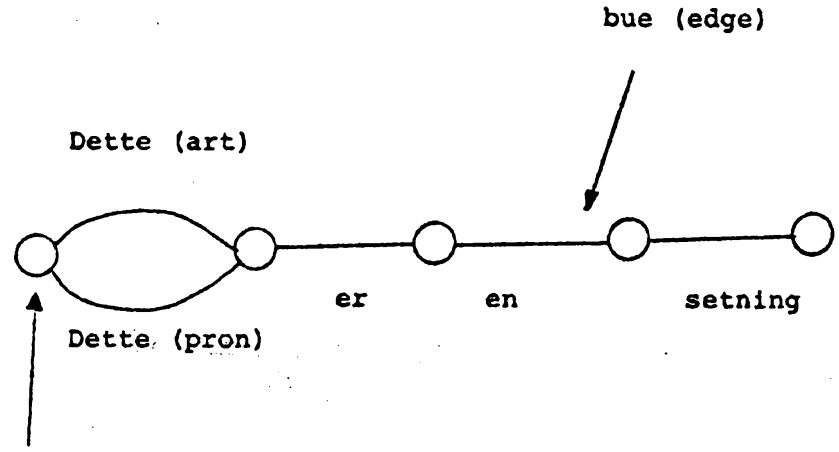
Symbolene etter MOVE eller TRY er en betingelse som er knyttet til ordklassen til den aktuelle bue. Navnene under en pil er et registernavn som den aktuelle bue blir satt i dersom en velger denne pilen.







Setningen som skal analyseres (og senere resultatene som fremkommer fra analysen), lagres i et nettverk som har fått navnet kart (chart hos Kay).



Et kart består av et sett med knuter, en for hvert ordmellomrom. Mellom to knuter går det en bue (edge). En bue inneholder et ord med tilhørende opplysninger som ordklasse o.l. som kan være fremkommet fra en morfologisk analyse eller tatt fra en ordliste. Dersom et ord har flere betydninger, vil det gå flere buer mellom de samme knutene.

Analysen foregår på den måten at alle mulige veier i kartet prøves mot alle mulige veier i grammatikken. Dersom slutt-tilstanden i grammatikken nås og en har kommet gjennom hele setningen, er setningen grammatisk. Registrene vil da inneholde opplysninger om hva som er subjekt, objekt o.l.

Når en finner konstituenter i setningen, så innføres det nye buer i kartet som spenner over de enkelte ordene som hører sammen.

Prøvingen av alle veier i kartet mot alle veier i grammatikken gjøres ved hjelp av en jobbliste. Et element i denne listen består av en bue (et ord eller en konstituent), en tilstand og registerlisten slik den var idet en forlot forrige tilstand.

På grunnlag av opplysningene om buen (edge) og innholdet i registrene, kan en velge ingen, en eller flere av pilene ut fra tilstanden, og dette gjøres ved at en produserer nye elementer (jobber) og henger disse på jobblisten (ett element for hver pil).

For også å kunne prøve nye buer som blir laget underveis mot grammatikken, har en innført begrepet venteliste (wait list). Dette er en liste med tilstander som er knyttet til en knute. For hver tilstand inneholder ventelisten også registerlisten slik den var da en kom til knuten. Dersom en tilstand inneholder en pil som starter en søking etter et nytt ledd (NEWPATTERN) setter en neste tilstand på ventelisten til den venstre knuten til den aktuelle bue.

Siden en jobb inneholder alle opplysninger som skal til for å sammenligne en bue mot et sett med piler fra en tilstand, er det likegyldig i hvilken rekkefølge en velger jobbene. Det mest vanlige er å velge den jobben som sist ble satt på jobblisten.

DEN FULLSTENDIGE ALGORITMEN

I Ved start.

For hver bue b_i som går ut fra knute nr. 1 sett R til registeret start = b_i, lag en ny jobb

b_i, S1, R

og sett denne på jobblisten.

S1 er første tilstand i grammatikken.

II Idet en setter nye jobber på jobblisten, undersøkes først om jobben finnes der fra før, eller har blitt utført tidligere.

III Så lenge det er jobber igjen på jobblisten, tar en ut og utfører denne.

Aktuell jobb har bue B, tilstand T og registerliste R.
For hver pil P_i fra tilstand T (som går til tilstand t_i) utføres følgende:

1. fortsett dersom betingelsene er oppfylt

2. utfør aksjonene som hører til den aktuelle pil, R' er nå den eventuelt ajourførte registerlisten

3. hvis P_i er MOVE.
   Lag en ny jobb for hver etterfølgende bue b_j til B (dersom t_i har en pil ut som er en BUILD ordre, lages jobb bare for en bue) og med oppdatert registerliste.
   ny jobb: b_j, t_i, R'

4. hvis P_i er TRY,
lag en ny jobb med samme bue og oppdatert registerliste.
Ny jobb: B, t_i, R'

5. hvis P_i er NEWPATTERN t,
t er første tilstand i et subnettverk.

a) Sett tilstand t_i og registerliste R på venteliste til B's venstre knute (dersom disse ikke finnes der fra før).

b) Sett R' til start = B
Lag to nye jobber:
B, t_i, R (denne jobben er for å komme videre i hovednettverk dersom vi ikke finner det ønskede ledd).
B, t_i, R'.

6. hvis P_i er BUILD K

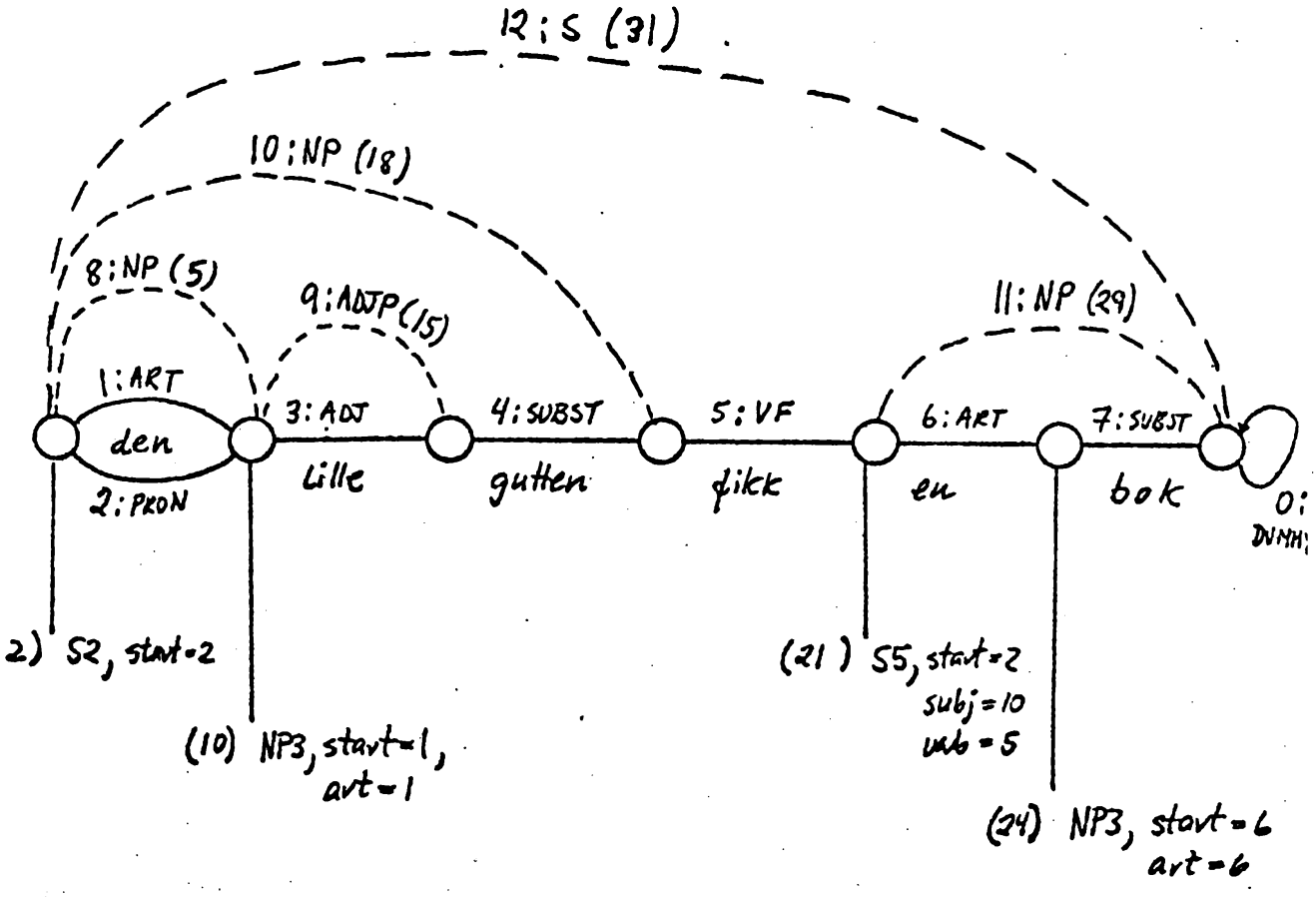
Lag en ny bue b_1 med ordklasseopplysning K og registerlisten R som innhold, fra knute K_1 (venstre knute til bue i registeret start) og til knute K_2 (venstre knute for bue B).

For hver tilstand t_j og registerliste r_j på ventelisten i K_1 lag en ny jobb:

b_1, t_j, r_j

Et eksempel.

Buene i kartet nummereres fortløpende. De heltrukne buene er kartet slik det var ved starten. I parentes står jobb-nummeret der buen ble laget eller der elementet ble satt på venteliste. Ventelisten henger under en knute. Den jobben som sist ble satt på jobblisten blir valgt som neste jobb.



jobb-nr.	produsert av jobb-nr.	utførelses-rekkefølge	bue	tilstand	registerliste
1 *)	start	7	1	S1	start=1
2	start	1	2	S1	start=2
3	2	6	2	S2	start=2
4	2	2	2	NP1	start=2
5	4	3	3	NP5	start=2, pron=2
6	5	4	8	S2	start=2
7	6	5	3	S3	start=2, subj=8
8	1	33	1	S2	start=1
9	1	8	1	NP1	start=1
10	9	9	3	NP2	start=1, art=1
11	10	31	3	NP3	start=1, art=1
12	10	10	3	ADJ1	start=3
13	12	11	3	ADJ2	start=3
14	13	12	4	ADJ2	start=3, overledd=3
15	14	13	4	ADJ3	start=3, overledd=3
16	15	14	9	NP3	start=1, art=1
17	16	15	4	NP4	start=1, art=1, adj=9
18	17	16	5	NP5	start=1, art=1, adj=9 subst=4
19	18	17	10	S2	start=2
20	19	18	5	S3	start=2, subj=10
21	20	19	6	S4	start=2, subj=10, verb=5
22	21	30	6	S5	start=2, subj=10, verb=5
23	21	20	6	NP1	start=6
24	23	21	7	NP2	start=6, art=6
25	21	24	7	NP3	start=6, art=6
26	21	22	7	ADJ1	start=7
27	26	23	7	ADJ2	start=7
28	25	25	7	NP4	start=6, art=6
29	28	26	0	NP5	start=6, art=6, subst=7
30	29	27	11	S5	start=2, subj=10, verb=5
31	30	28	0	S6	start=2, subj=10, verb=5 obj=11
32	31	29	12	S2	start=2
33	11	32	3	NP4	start=1, art=1

*) Her settes ikke nytt element på ventelisten til første knute fordi tilstanden er den samme og fordi registrene "start" inneholder buer som har samme venstre knute.

3. IMPLEMENTASJONEN I SIMULA

SIMULA er et programmeringsspråk med innebyggede funksjoner for listemanipulering. Det uferdige programmet fra Martin Kay's side var også skrevet i SIMULA. Dette er grunnene til at SIMULA ble valgt som programmeringsspråk.

Grammatikken beskrives direkte i SIMULA og oversettes sammen med resten av programmet. En tilstand i grammatikken er definert som en klasse i SIMULA. Dette er et sett med (data og) instruksjoner som gis et navn. En kan definere en forekomst av en klasse med navn og utføre denne ved å aktivisere forekomsten En forekomst kan videre aktiviseres og passiviseres. Først i alle tilstander er en ordre for passivisering av tilstanden (siden dette er felles for alle klassene gjøres dette i en prototyp som alle tilstandene defineres som underklasse av). Før analysen starter blir alle tilstandene aktivisert (for så å passiviseres med en gang).

Når en tar ut en jobb fra jobblisten, blir den aktuelle tilstand aktivisert. Instruksjonene for denne tilstanden lager nye jobber og når instruksjonene er utført, passiviseres tilstanden. Tilstanden i neste jobb som tas fra jobblisten blir så aktivisert.

Det er skrevet en del prosedyrer for å beskrive grammatikken i SIMULA. Her skal nevnes noen:

CE ("ordklasse")	boolsk funksjon som returnerer true dersom den aktuelle bue har samme ordklasse som angitt mellom anførselstegnene.
NEWREG("regnavn")	setter registeret "regnavn" til å peke på aktuell bue.
REG("regnavn")	gir oss bue som registeret "regnavn" peker på, er NONE dersom det ikke finnes et register med dette navn.
TEST("regnavn", "egenskap")	boolsk funksjon som er true dersom buen i registeret "regnavn" har den aktuelle egenskapen. "regnavn" kan være "CUR" og buen blir da den aktuelle bue .
MOVE(t)	godkjenner den aktuelle bue og lar neste tilstand være t . Går videre i kartet.
TRY(t)	blir stående på den aktuelle bue og lar neste tilstand være t.
NEWPATTERN(t_1,t_2)	lar neste tilstand være t_1 (start på et subnettverk). t_2 blir første tilstand etter en er ferdig med subnettverket .
BUILD("ordklasse")	lager en ny bue i kartet .
PLACE("regnavn")	lar buen som registeret "regnavn" inneholder, henge på den buen som skal bygges

SREG("regnavn","verdi")	lager et register som ikke inneholder en bue, men en tekstkonstant.
GREG("regnavn")	henter ut tekstkonstanten fra registeret.
NREG("regnavn1", "regnavn2")	lar et register med navn "regnavn1" inneholde samme bue som "regnavn2".

Grammatikken i eksempelet på side 4 vil nå se slik ut:

Alle tilstander har C foran navnet i definisjonen.
Tilstandene som utføres (forekomstene) har navn uten C og det er disse som bruke i MOVE, TRY, og NEWPATTERN.

```
tilstand class CS1;
newpattern (NP1,S2);

tilstand class CS2;
If CE("NP") then
begin
  newreg ("subj");
  move (S3);
end;

tilstand class CS3;
if CE ("VF") then
begin
  newreg ("verb");
end;

tilstand class CS4;
newpattern (NP1,S5);

tilstand class CS5;
if CE ("NP") then
begin
  newreg ("obj")
  move (S6);
end;

tilstand class CS6;
begin
  place ("subj");
  place ("verb");
  place ("obj");
  build ("S") ;
end;
```

```
tilstand class CNP1;
if CE("ART") then
begin
  newreg ("art");
  move (NP2);
end
else
if CE("PRON") then
begin
  newreg ("pron")
  move (NP5);
end
else
  try (NP2);

tilstand class CNP2;
newpattern (ADJ1,NP3);

tilstand class CNP2;
if CE ("ADJP") then
begin
  newreg ("adj");
  move (NP4);
end
else
  try (NP4);

tilstand class CNP4;
if CE ("SUBST") then
begin
  newreg ("subst");
  move (NP5);
end;
```

MOVE art pil

MOVE art pil

TRY pil

```
tilstand class CNP5;
begin
  place ("art");
  place ("adj");
  place ("subst");
  place ("pron");
  build ("NP");
end;
```

```
tilstand class CADJ1;
if CE ("GRADSADV") then
begin
  newreg ("Underledd");
  place ("underledd");
  move (ADJ2);
end
else
  try (ADJ2);
```

```
tilstand class CADJ2;
if CE ("ADJ") then
begin
  newreg ("overledd");
  place ("overledd");
  move (ADJ2);
end
else
if reg ("overledd") =/=NONE then
  try (ADJ3);
```

```
tilstand class CADJ3;
build ("ADJP");
```

```
ref (CS1) S1;
      "
      "          } Definisjon av forekomst,
      "            en for hver tilstand
      "
ref (CADJ3) ADJ3;
```

```
S1: - new (C1("S1",false));
          "
          "      } Aktivisering, en for
          "        hver tilstand
          "
ADJ3:- newreg(CADJ3("ADJ3",true));
```

Andre parameter er true dersom tilstanden inneholder en BUILD ordre (pil).

Før en slår inn en setning kan en angi hvor mye en vil ha skrevet ut mens analysen pågår. En kan her velge mellom ingen utskrift (bare den analyserte setningen) moderat utskrift og full utskrift. En utskrift som vist under der en får skrevet ut tilstandene og buene i den rekkefølge de sammenlignes, er til god hjelp ved analyse av feilanalyser.

```
S6A      30    NP-1     30    S7       9    ANCHOR   9
ADV1      9    ADV2      9    ADV4     9    ANCHOR   9
S10       9    ADV4A     9    S7A      9    S8       9
S9        9    S10       9    ANCHOR   9    ADJ1     9
ADJ2      9    S10A      9    S11      9    ANCHOR   9
NP1       9    NP2       9    ANCHOR   9    ADJ1     9
```

4. GRAMMATIKKEN

Programmet behandler jobblisten som en stack, dvs. den velger ut som neste jobb den siste som ble satt på jobblisten.

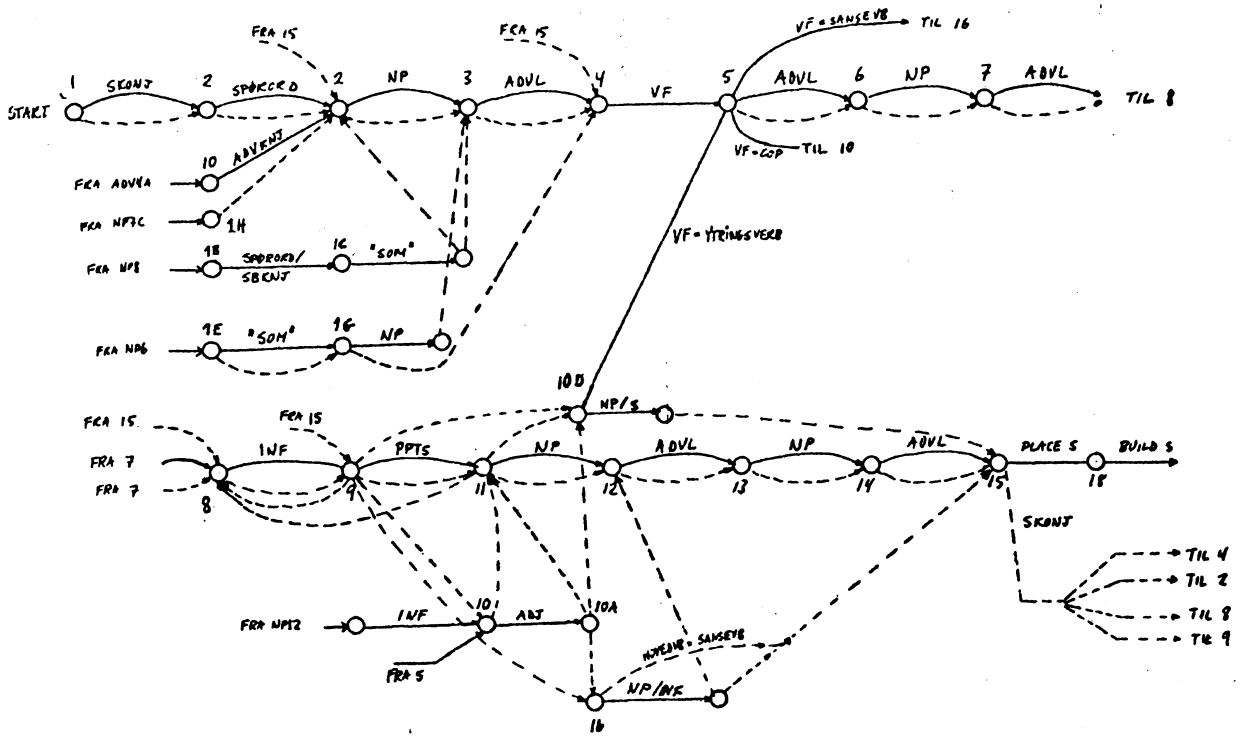
Dette forholdet har en utnyttet idet en har beskrevet grammatikken på en slik måte at en får først ut den mest vanlige tolkningen av en setning.

Grammatikken består av et hovednettverk for setning og 3 subnettverk, et for nominale ledd, et for adjektiviske ledd og et for adverbialer.
Fra NP nettverket eller adverbialnettverket går en igjen tilbake til setningsnettverket for å se etter substantiviske, adjektiviske leddsetninger og adverbiale leddsetninger.
Grammatikken kan behandle relativsetninger eller at-setninger uten som eller at. Det siste som er lagt inn er sideordning på alle nivåer i grammatikken.

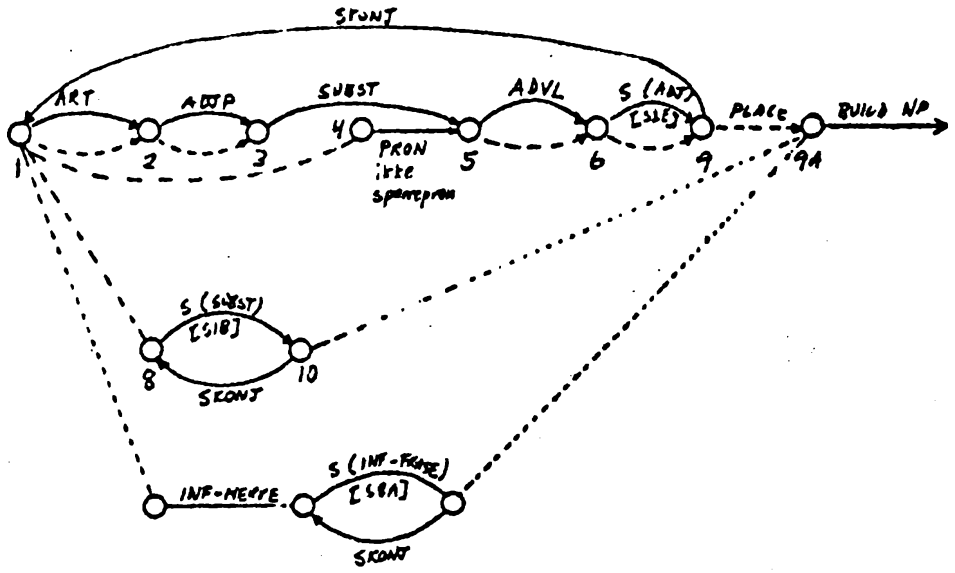
I setningsgrammatikken setter en først inn NP'ene i registre som har navn etter hvilket nummer NP'en har i setningen NP-1, NP-2 ..... Først når en skal bygge setningen blir NP'ene satt inn i registre for subjekt, objekt, predikat o.l.

På de neste sidene er de grammatiske nettverk litt forenklet. De prikkede linjene står for TRY piler. Piler som har NP, ADVL eller ADJP er en kombinasjon av en NEWPATTERN pil og en MOVE pil.

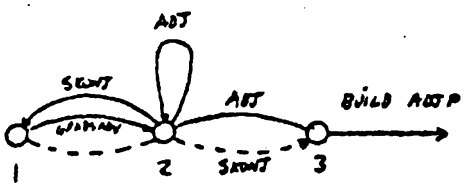
SETNINGSGRAMMATIKKEN (forenklet)



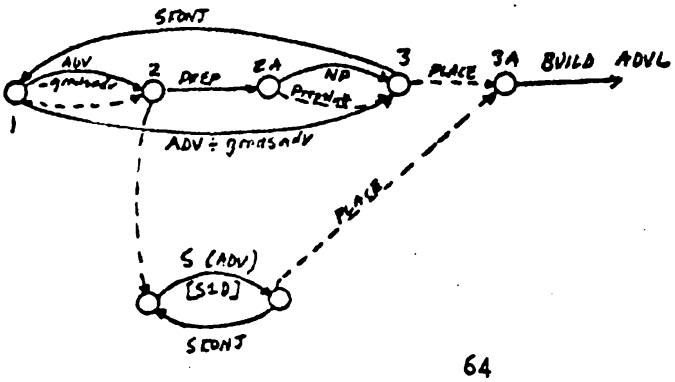
NP


SKONJ
ART
ADJP
SUBST
ADVL
S (ADJ)
[S1E]
PLACE
BUILD NP
1
2
3
4
PRON
irte
spzrzprzn
5
6
9
9A
S (SUBST)
[S1B]
8
10
SKONJ
INF-MEYFE
S (INF-FRAZE)
[S8A]
SKONJ

ADJP


ADJ
SKONJ
ADJ
BUILD ADJP
1
2
SKONJ
3

ADVL


SKONJ
ADV
2
PREP
EA
NP
3
PLACE
3A
BUILD ADVL
1
ADV ÷ gmasadv
PLACE
S (ADV)
[S1D]
SKONJ

DANwORD

Hyppighedsundersøgelser i moderne dansk

Bente Maegaard og Hanne Ruus

Formålet med projektet DANwORD er at undersøge et repræsentativt udsnit (på 1.25 million løbende ord) af moderne dansk med automatiske metoder, især med henblik på frekvens af ord. Udover de beregnede frekvenser har projektet interesse ved udviklingen af de automatiske metoder til bl.a. morfologisk analyse og entydiggørelse af homografer og ved anvendelse af databaseteknik til lagring af de fundne resultater.

Vi har arbejdet med projektet i ca. 1 1/2 år, og har i den forløbne periode indsamlet og registreret 250.000 løbende ord, dvs. en femtedel af vort materiale. Før vi kunne påbegynde denne indsamling, måtte vi bestemme, hvilke tekster der skulle indgå i vort materiale eller corpus. Vi skal her først gennemgå de overvejelser, man må gøre sig, når man skal sammensætte et corpus og de konklusioner vi er nået til, og derefter omtale, hvorledes vi vil behandle teksterne og hvilke oplysninger om teksterne, vi ønsker at finde og lagre.

I. Tekstgrundlag.

Vi har ved udvælgelsen af tekstmateriale anlagt et forbrugskriterium, således at vi har forsøgt at finde frem til det mest brugte sprog. Det mest brugte sprog består af de mest læste tekster og de mest hørte tekster; men af praktiske grunde har vi, ligesom alle andre, der har lavet større frekvensundersøgelser, måttet begrænse os til de mest læste tekster, altså til skriftsprog, selv om forbrugskriteriet peger både på ikke offentliggjorte skrevne tekster (breve, huskesedler, mv.) og på det talte sprog (der også findes i både offentliggjort (radio og fjernsyn) og ikke offentliggjort form).

Inden for det offentliggjorte skriftsprog, som altså er vores grundlagsmateriale, har vi som nævnt forsøgt at finde frem til det i nutiden mest anvendte, dvs. mest trykte og dermed mest læste, sprog. Ved nutiden har vi forstået 5-årsperioden 1970-1974, hvad vi senere skal vende tilbage til. Dette udvælgelsesprincip gør, at vore resultater med rette vil kunne opfattes som gældende for moderne dansk skriftsprog, idet vi dog igen skal understrege, at det sprog, vi undersøger, er det sprog, der læstes i begyndelsen af 1970'erne, ikke det sprog, der blev skrevet.

Det er væsentligt at være opmærksom på, at materialevalget har stor betydning for resultaternes generaliseringsværdi; det gælder nemlig om mange frekvensordbøger, at deres resultater opfattes som mere generelle end de egentlig er, idet kriterierne for udvælgelse af materiale ikke altid fremgår klart af titelbladet (en gennemgang af eksisterende frekvensordbøger fremkommer i Danske Studier 1978).

2. Valg af corpus.

Ved frekvensundersøgelser af andre sprog har man anvendt to hovedtyper af corpora: den ene type består udelukkende af avisprog, mens den anden er søgt sammensat således, at corpus afspejler det trykte sprogs sammensætning.

I begge tilfælde ønsker man, at resultaterne af undersøgelser i corpus skal gælde for det mest "almindelige" sprog inden for det undersøgte nationalsprog; men det er faktisk et spørgsmål, om der findes sådan noget som "det mest almindelige sprog", dvs. om man ud fra en undersøgelse af en blanding af tekster kan konkludere noget om sproget som helhed.

Ordhyppigheder i en bestemt tekst er nemlig for de fleste forskellige ords vedkommende specifikke for denne tekst og dens indhold, idet de afhænger af, hvad teksten handler om (jvf. Henning Spang-Hanssen, 1960). Hvis man tager en anden tekst, vil det være andre ord, der er hyppige. De eneste ord, hvis hyppighed er tekstuafhængig, er nogle få (et par hundrede) meget hyppige ord.

Nu kan man ved ret begrænsede tællinger bestemme disse tekstuafhængige ord; men hvilke af sprogets øvrige ord man ellers får med, afhænger af valget af tekster og af, hvor stort et materiale man har. Hvis materialet er stort nok, og hvis man vælger tekstprøverne små nok og rimeligt varieret, er det muligt inden for en nærmere afgrænset, homogen, tekstart også at få gyldige resultater for andre end de meget hyppige ord. De enkelte tekstprøver bør være korte, fordi man derved mindsker risikoen for, at enkelte tekstspecifikke ord får en urimelig overvægt: tekstproverne kommer til at handle om forskellige ting.

Problemet bliver herefter at afgrænse en tekstart således, at den bliver homogen, dvs. således at den består af tekster af nogenlunde ensartet karakter. Det er ikke muligt på forhånd at angive, hvad der er en rimelig afgrænsning af en tekstart; man må skønsmæssigt opstille kriterier

for afgrænsning af hver enkelt tekstart, og kan først, når den faktiske sammensætning af tekstarten foreligger, undersøge, om den er homogen.

En homogenitetsprøve vil bestå i tilfældigt at udtage delmængder af hele tekstarten og undersøge, om tællinger i disse stemmer overens med tællinger i hele tekstarten.

3. Størrelse af tekstarter og tekstprøver.

Vi er ved fastlæggelsen af størrelsen af de enkelte tekstarter gået ud fra den alment bekræftede antagelse (Zipf's lov, omtales f. eks. i Charles Muller, 1968) at en tekstmasse på 1/4 million løbende ord indeholder ca. 5000 ord med en hyppighed på 5 eller mere. Da denne størrelse af en tekstart synes rimelig også med henblik på en sammenligning med andre undersøgelser, har vi besluttet at lade hver tekstart omfatte 250.000 løbende ord. For de enkelte prøver, der som allerede nævnt bør være forholdsvis korte, har vi valgt størrelsen 250 ord, dvs. ca. 1 side. Hver tekstart består altså af 1000 prøver. Prøverne er udvalgt i teksterne ved anvendelse af tilfældige tal. De tilfældige tal benyttes til at beregne den side og linie, hvor prøven starter. En prøve begynder altså altid ved det første hele ord på en linie, men af hensyn til entydiggørelse af homografer er ved indkodning af prøven også medtaget "forkonteksten", der strækker sig indtil den nærmeste periodegrænse før prøvens start, og "bagkonteksten", der strækker sig fra prøvens slutning til nærmeste periodegrænse.

En tekstprøve. Forkonteksten slutter ved //.

```
<< Anders Bodelsen
Tænk på et tal
Gyldendal 1970, s.47 l.26 >>
<<L18>>
                                                  Midt i en
//mørk skov standsede han og stod ud af vognen.
  Det var bidende koldt. Han slukkede lygterne, øjnene vænnede sig til mørket og han hørte ikke nogen anden lyd end den kolde susen i grantræerne. Han gik et par skridt ind i skoven og borede med skosnuden i jorden, der allerede var fast af frost.
  En bil nærmede sig. Det lød som om den ville standse,
<< s.48 >>
men den satte atter farten op og lysvifften forsvandt bagude mellem træerne. Jorden var for hård, stedet for befærdet. Desuden havde han fået et nyt indfald.
  Han gik tilbage til bilen igen og kørte hjemad. Ideen han netop havde fået, forbandt han med de dybe, næsten berusende indåndinger af frostluft. Den var en inspiration. Han parkerede noget fra butikstorvet, gik roligt det sidste stykke med mappen i hånden, låste sig fra trappegangen ind i banklokalet og fandt lommelygten i bunden af skabet med de elektriske propper.
  Nøglerne til de ubenyttede boxe lå i Miriams skuffer. Han skrev en box-kvittering ud til F. Hjulmand, fandt frem til den tomme box via nummeret på nøglen, et hundrede og niogtyve, og låste ved hjælp af bankens universalnøgle til alle boxene og specialnøglen til et hundrede og niogtyve den blå madkasse ind i det snævre rum. Kartoteket med boxkvitteringerne var låst, denne del af manøvren måtte han undsætte. Han lagde lommelygten på plads i bunden af elektricitetsskabet, låste sig ud og sad lidt efter i sin vogn. Ingen var fulgt efter ham, ingen fulgte ham, da han startede vognen. Men han mødte vagtselskabets mand på hans cykel, idet han passerede butikstorvets udmunding.
```

4. Valg af tekstarter.

Når man, som vi, udgår fra et forbrugskriterium, vil man være interesseret i, at udvalget af tekstarter tilsammen skal dække størstedelen af enkelttekster på det undersøgte sprog, her dansk. Flere forskellige undersøgelser (se f.eks. Socialforskningsinstituttets fritidsundersøgelser og Hans Hertel 1972) af voksne danskeres læsevaner viser, at 90 pct. dagligt læser avis, 75 pct. læser ugeblade og 50 pct. læser fiktionsprosa; på bibliotekerne låner børn 2-3 gange så meget som voksne og er således store forbrugere.

Hvis man kun vil undersøge én tekstart, er det altså klart, at den må bestå af avistekster; men da der også er andre tekstarter, der bruges af en ikke ubetydelig del af befolkningen, og da det vil være interessant at sammenligne forskellige tekstarter, har vi ikke ment at kunne nøjes med avisteksterne,og har valgt følgende 5 tekstarter: fiktionsprosa for voksne, fiktionsprosa for børn, aviser, ugeblade, faglitteratur.

Den første tekstart, vi behandler, er fiktionsprosa for voksne, og for denne er vi færdige med udtagelse og indkodning af prøverne, idet der dog mangler sidste korrekturlæsning på en del.

5. Udvælgelse af tekstprøver.

Også ved udvælgelse af de enkelte tekstprøver i en tekst anlægger vi et forbrugssynspunkt, dvs. at vi for fiktionsprosaen har forsøgt at finde frem til de mest læste danske forfattere. Det er imidlertid ikke så nemt, som man skulle tro, idet 1) oplagstal ikke kan bruges, fordi visse forlag ikke vil opgive det, 2) folkebibliotekerne ikke fører udlånsstatistik på titler, og 3) der ikke findes nyere undersøgelser af læsevaner, der når ned til enkelttitler. Vi har derfor valgt at bygge på udgivelsesfrekvens og har fundet frem til de mest udgivne forfattere i perioden 1970-1974 i Dansk Bogfortegnelse; vi har medtaget alle forfattere, der i denne periode har fået udgivet mindst 5 bøger, med én prøve fra hver af udgivelserne. Dette gav 976 tekstprøver, og de manglende 24 er derefter fundet ved forholdsmæssig fordeling på de allerede repræsenterede forfattere (en nærmere redegørelse for vores udvælgelsesprocedure fremkommer i Danske Studier 1978).

Grunden til, at den periode, vi undersøger, er forholdsvis lang og f.eks. længere end ét år, er, at udgivelsestallene for de enkelte forfattere viser ret store udsving fra år til år, men stabiliserer sig, når man betragter en længere periode.

Nedenfor angives de 20 mest udgivne forfattere i perioden 1970-1974, og antallet af tekstprøver for hver af dem.

nr.		antal
1	Cavling, Ib Henrik	58
2	Koch, Morten	43
3	Nielsen, Lars	38
4	Forsberg, Bodil	33
5	Panduro, Leif	28
6	Rifbjerg, Klaus	28
7	Hazel, Sven	27
8	Bodelsen, Anders	26
9	Nohr, Else-Marie	23
10	Edson, Ed	22
11	Andersen, H.C.	20
12	Søeborg, Finn	20
13	Kampmann, Christian	19
14	Poulsen, Erling	19
15	Ørum, Poul	19
16	Blicher, Steen Steensen	17
17	Hansen, Martin A.	16
18	Bang, Herman	15
19	Johansen, Orla	15
20	Scherfig, Hans	15

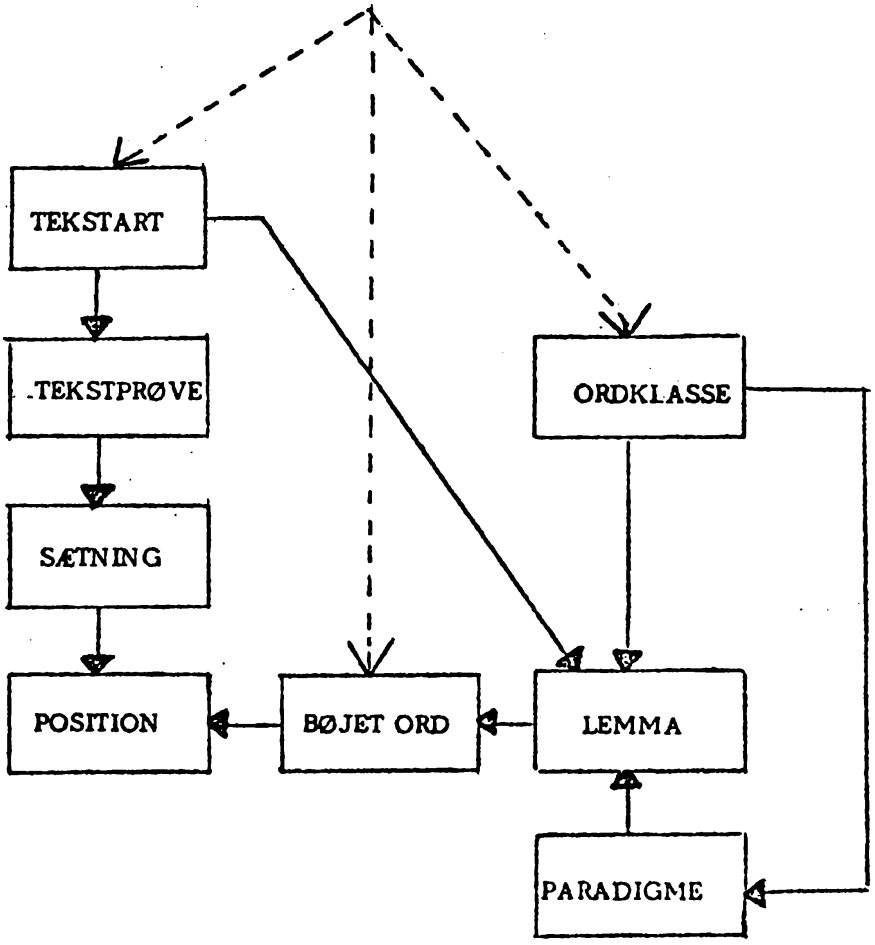
II. Den datamatiske behandling af teksterne.

Hovedformålet med projektet DANwORD er at lave hyppighedsundersøgelser. Derfor vil de primære resultater af vores bearbejdning af materialet være opgørelser over frekvenser af grafemer, lemmaer, ordklasser, paradigmer osv. Vor undersøgelsesmetode og den videre behandling af de analyserede tekstprøver vil imidlertid også bringe metodisk interessante resultater på andre områder: På det lingvistiske område giver arbejdet med at automatisere den sproglige analyse ny indsigt i form af mere strukturerede og udførlige grammatiske beskrivelser. På det datamatiske område vil lagringen af analyseresultaterne i en database bringe viden om samspillet mellem lagring af en større datamængde med en kompleks struktur og den effektive udnyttelse af de informationer, strukturen indeholder.

Første del af en automatisk sproglig analyse til brug for frekvensundersøgelser er lemmatiseringen. Denne kræver, at man opstiller en fuldstændig fleksionsgrammatik for dansk. Til brug for den morfologiske analyse disponerer vi over en udtømmende og sammenhængende grammatik over den regelmæssige bøjning i moderne dansk (se Hanne Ruus, 1977). Udover den morfologiske analyse omfatter den automatiske lemmatisering entydiggørelsesprocedurer for homografemerne. Ved entydiggørelsen af homografer fra lemmaer af forskellig ordklasse vil man kunne komme ganske langt automatisk, idet man kan bygge på ordklassernes forskellige syntaktiske funktion, ligesom homografer inden for samme lemma i stor

udstrækning vil kunne entydiggøres ved at søge efter kongruerende former i den nærmeste kontekst og ved også her at bygge på formernes forskellige syntaktiske funktion.

Da vi vil opgøre frekvenser på forskellige lingvistiske niveauer, har vi brug for at have de analyserede tekstord med tilhørende oplysninger om forekomststeder, lemma osv. kædet sammen på forskellige måder. For at opnå dette vil vi lagre alle de fundne oplysninger om tekstordene i en database, således at det er muligt at uddrage ord eller sætninger med bestemte egneskaber. En database er en lagringsmetode, hvor man ved lagringen angiver, hvilke oplysninger der skal være kædet sammen, og derved strukturerer sine data. Vi vil f.eks. sørge for, at tekstordene er kædet sammen med deres lemma og deres ordklasse, herved bliver det nemt at opgøre frekvenser f.eks. for alle former, der hører til samme lemma og for alle ord fra samme ordklasse. I en database er det lettest at uddrage de oplysninger, der allerede er kædet sammen, men man kan selvfølgelig også finde frem til de delmængder af materialet, der ikke er nedlagt direkte i strukturen. Hvis man ønsker at kunne udtage mange forskelligartede delmængder, vil databasen imidlertid let få en ret indviklet struktur, så både førstegangslagring, lagerforbrug og søgninger vil blive tidskrævende og dermed dyre. Det drejer sig derfor om at begrænse databasen til den struktur, der er nødvendig for de hyppigste opslag. Vi forestiller os følgende databasestruktur:



De stiplede linjer angiver indgange til databasen, altså de kasser, man har direkte adgang til. Alle de andre kasser kan man finde frem til ved at benytte strukturen, dvs. følge pilene.

Kasserne i figuren kan opfattes som de steder, hvor man lagrer forekomsten af lemmaer, tekstord (bøjet ord) m.v. Lagringen er foretaget således, at et lemma f.eks. pen som angivet ved pilen fra LEMMA til BØJET ORD er forbundet med de forskellige, faktisk forekommende bøjningsformer af pen, altså f.eks. pen, penne, pennen, pennenes. Hver af de disse bøjningsformer peger via POSITION på de steder i teksterne, hvor ordformen er fundet. Et bøjet ords kontekst finder man i SÆTNING, hvor teksternes sætninger er lagret hver for sig. Pilen fra TEKSTPRØVE til SÆTNING angiver, at den enkelte tekstprøve peger på de sætninger, den indeholder.

2. Analyse og lagring.

Inden tekstprøverne kan lagres i databasen, skal de analyseres: bibliografiske oplysninger og for- og bagkontekst skal skilles fra, og selve prøven deles op i ord.

Når ordene fra en tekstprøve skal lagres i databasen foretages for hvert ord følgende:

I. Ordet slås op i basen. Findes ordet i basen, lagres en forekomst af det. Hvis det er en homograf, *) entydiggøres det først.

II. Hvis ordet ikke findes i basen endnu, skal det lemmatiseres. De allerhyppigste småord, hvoraf mange er ubøjelige, klares ved opslag i en liste på et par hundrede ord. I en forkortelsesliste eftersøges ord før punktum. De øvrige ord analyseres morfologisk, hvorved potentielle fleksiver fjernes og der opstilles et antal mulige lemmaer. De foreslåede lemmaer slås automatisk op i en let bearbejdet udgave af Nudansk Ordbog. Hvis flere af de foreslåede lemmaer findes i ordbogen, er det analyserede ord en homograf og sendes til entydiggørelse. Til sidst lagres tekstordet og analyseresultaterne i basen. De ord, der ikke kan findes i ordbogen, og de homografer, der ikke kan entydiggøres automatisk, skrives ud til håndbehandling.

Efterhånden som vi får lagret tekstprøverne i basen, vil stadig flere ord blive klaret under I, således at lemmatisering og ordbogsopslag begrænses mest muligt. En anden fordel ved den beskrevne fremgangsmåde er, at vi kan udtage frekvensoplysninger på forskellige niveauer på dele af materialet, inden vi har indsamlet prøver fra alle tekstarterne. Dele af den analyserede datamængde (f.eks. tekster af en forfatter) vil også kunne anvendes ved andre former for sprogvidenskabelige undersøgelser.

3. Fremtidig anvendelse af materialet.

Når vi har afsluttet frekvensundersøgelserne på de 1.25 million ord, vil vi bevare databasen som hjælpemiddel for sprog- og stilforskere. Ved at bruge 5000 sider gennemanalyseret tekst som basis for arbejdet med sprogbeskrivelse og tekstkarakteristik vil det være muligt kvantitativt at skelne mellem mere og mindre væsentlige regler i grammatikken, ligesom et gennemsnit af materialet for stilforskeren vil kunne være en tilnærmelse til den norm, der danner grundlaget for enhver tekstkarakteristik.

---

*) Man siger, at to ord er homografer, hvis de har samme grafiske form - staves ens -, men enten tilhører forskellige lemmaer (<u>helt</u> substantiv, <u>helt</u> adjektiv) eller er forskellige bøjningsformer af samme lemma (<u>kunne</u> infinitiv, <u>kunne</u> præteritum). At entydiggøre en homograf betyder at bestemme hvilket lemma, den tilhører, og hvilken bøjningsform, den er, i den aktuelle kontekst.

Nedenfor anføres de 75 hyppigste graford i vore skønlitterære prøver, og til sammenligning de 75 hyppigste graford fra Noesgaards undersøgelser af skønlitteratur (Aksel Noesgaard: Hyppigheds Undersøgelser II, 1937).

DAN-ORD				NOESGAARD			
1	OG	43	OVER	1	OG	43	MAN
2	DET	44	NOGET	2	I	44	SIN
3	I	45	HVAD	3	DET	45	IND
4	AT	46	KUNNE	4	HAN	46	HANS
5	HAN	47	MAN	5	AT	47	HENDE
6	VAR	48	SIN	6	VAR	48	NAAR
7	JEG	49	EFTER	7	EN	49	HVOR
8	EN	50	IND	8	DEN	50	MIG
9	PÅ	51	BLEV	9	TIL	51	SKULDE
10	IKKE	52	SKAL	10	PAA	52	DEM
11	TIL	53	VIL	11	SAA	53	EFTER
12	ER	54	KOM	12	DER	54	KAN
13	DE	55	VÆRE	13	MED	55	NOGET
14	MED	56	JO	14	DE	56	NED
15	HUN	57	GIK	15	AF	57	STOD
16	DEN	58	SELV	16	IKKE	58	JO
17	DER	59	HVOR	17	FOR	59	HVAD
18	SÅ	60	JA	18	SOM	60	OGSAA
19	AF	61	VILLE	19	HUN	61	VILDE
20	FOR	62	DIG	20	SIG	62	ALLE
21	SIG	63	HENDES	21	JEG	63	VÆRE
22	MEN	64	OGSÅ	22	HAVDE	64	LILLE
23	HAVDE	65	ELLER	23	ER	65	SELV
24	SOM	66	NED	24	MEN	66	SKAL
25	ET	67	HER	25	ET	67	HER
26	OM	68	MEGET	26	OM	68	HEN
27	DU	69	NÅR	27	HAM	69	JA
28	HAR	70	MIN	28	VED	70	HENDES
29	HAM	71	HAVE	29	OVER	71	HAVE
30	SAGDE	72	MÅ	30	DA	72	GAMLE
31	MIG	73	IGEN	31	FRA	73	FIK
32	NU	74	SKULLE	32	NU	74	MOD
33	VED	75	LIDT	33	SAGDE	75	ELLER
34	VI			34	UD		
35	UD			35	HAR		
36	KAN			36	KUNDE		
37	DA			37	OP		
38	OP			38	DU		
39	FRA			39	VI		
40	HANS			40	BLEV		
41	DEM			41	KOM		
42	HENDE			42	GIK		

Liste over værker, der er henvist til i foredraget:

Hans Hertel: Det litterære system i Danmark (i Robert Escarpit: Bogen og læseren, Reitzel 1972).

Bente Maegaard og Hanne Ruus: DANwORD, Baggrund og materiale (i Danske Studier 1978 (i Trykken)).

Charles Muller: Initiation à la statistique linguistique, Dunod 1968.

Hanne Ruus: Ordmekanik (i SAML III, 1977, s. R1 - R28).

Socialforskningsinstituttets fritidsundersøgelser:
P.-H. Kühl, Inger Koch-Nielsen, Kaj Westergaard: Fritidsvaner i Danmark, 1966.
P.-H. Kühl og Inger Koch-Nielsen: Fritid 1975, 1976.

Henning Spang-Hanssen: Aksel Noesgaards ordstatistiske pionerarbejde (i Danske Studier 1960, s. 81-90).

Marina Mundt
Bergen

FUNCTION WORDS IN HÁKONAR SAGA

Med Mario Peis noe mer utførlige definisjon av begrepet function words i bakhode kommer jeg i første omgang til å konsentrere meg om de fire av Daniel Steible anførte kategorier: Preposisjoner, konjunksjoner, korrelativer, negasjoner. Man får da likevel mer enn nok å gjøre. Det materiale man finner i en norrøn tekst, lar seg nemlig ikke behendig putte inn i de fire esker vi blir tilbudt. En systematisk behandling av dette utvalget vanskeliggjøres ved at det foreligger formal kongruens av noen høyfrekvente preposisjoner med konjunksjoner, til dels også med andre ordklasser. Går man igjennom en KWIC-index for å kunne skille homograf-komponentene fra hverandre, finner man f.eks. at man ved til får med ikke mindre enn fem komponenter å gjøre.

Ved samtlige preposisjoner og en del konjunksjoner må vi regne med tvillingsformer som er adverbier. Tar vi med alle negasjonene, ser vi at det i tre av våre fire esker finnes noe som henviser oss til gruppen adverb, rettere sagt, til gruppen "rene" adverb. Jeg anser det som rimelig at dette faktum får konsekvenser for det arbeid jeg holder på med.

Hákonar saga Hákonarsonar hører inn under genren kongesaga. Den ble til 1263-65. En kritisk utgave av eldre datum ble lagt til grunn, da jeg fikk punchet teksten, sammen med en del andre sagatekster, for vel ti år siden. Jeg hadde i mellomtiden rikelig anledning til å komme tilbake til Hákonar saga og dens særpreg, under arbeidet med en diplomatarisk utgave av en bestemt versjon av samme saga. Etter at jeg således hadde vært opptatt av Hákonar sagas språkdrakt i lengre tid og under vekslende synsvinkler, kom min interesse, så langt som den angår ordforrådet, til å bli konsentrert om gruppen function words.

Function words - det er stort sett de samme som i norsk grammatikk går under navnet "lukkede ordklasser". Det gjelder også for function words som i første omgang faller inn under de fire åpne ordklasser (verb, subst., adj., adv.). Dette

kommer tydelig fram i Olav Næs' fremstilling, hvor han anfører de modale hjelpeverb som "lukket" underklasse til verbene eller de nektende adverbier som lukket underklasse til adverbiene. For ikke å vanskeliggjøre kommunikasjonen med diskusjonsdeltakere fra nabolandene unødig, tror jeg likevel det er lurest i det følgende å unngå betegnelsen "lukkede ordklasser".

Mine hittil tentative overveielser i tilknytning til gruppen function words i norrønt har sitt utspring i interessen for to problemstillinger. Den ene vedrører attribusjonsspørsmålet. I den utstrekning det arbeides med ordfrekvenser på dette området, har utviklingen i den senere tid gått i retning av å bruke stadig flere function words som diskriminatorer enn meningsbærende gloser/ordkombinasjoner.[1] Den andre problemstillingen jeg er opptatt av, er målbarheten av stilistiske meritter eller unoter. Jeg unngår her bevisst "evaluering", for jeg har liten tro på at spørsmålet god/dårlig stil kan besvares ved hjelp av frekvensundersøkelser, bortsett fra at de helt elendige tekster kanskje ville skille seg ut: Til det er det for stort spillerom mellom reseptorene m.h.t. hva de anser som "godt".[2] Dessuten er det mange måter å være god på. Men det vil som oftest være mulig å oppnå enighet om en teksts karakterisering som tørr eller spennende, stiv eller tilnærmet hverdagstale, om det er

---

1) Sml. Tore Johannisson, Ett språkligt signalement, 1973, s.150: Som särskiljande kriterier lämpar sig framför allt de grammatiska, lexikalska och grafiska kategorier, inom vilka två eller flera möjligheter står till förfogande för att uttrycka samma betydelseinnehåll. Hit hör framför allt många av formorden.

2) Det illustreres ved et forsøk som Hardi Fischer har beskrevet i "Entwicklung und Beurteilung des Stils" (Mathematik und Dichtung v/ Kreuzer og Gunzenhäuser, 1965, s.171ff.). Ti stiler skrevet av trettenåringer skulle bedømmes av syv "sakkyndige". Stilene skulle gis en rekkefølge tilsvarende den stilistiske kvaliteten. Overensstemmelsen mellom de sakkyndige var størst ved de mellomste plasser i skalaen. Ved ytterpunktene var det verre: En av stilene kom på 1.plass hos to sakkyndige, mens den kom på siste (10.) plass hos to andre.

flyt i fremstillingen eller om den virker avhakket. Da alle disse egenskaper kan tilskrives tekster med hvilket som helst innhold, hvor de meningsbærende ord må være forskjellige p.g.a. vekslende innhold, er det fristende å undersøke om gruppen function words her kanskje spiller en nokså bestemt, om enn foreløpig udefinert rolle, i hvilken grad det kanskje er nettopp deres anvendelse resp. fordeling som fremkaller de forskjellige inntrykk. Meg bekjent er det lite som er gjort på dette område hittil.

Selv om jeg ville sette pris på å få kjennskap til eventuelle forsök med formordene i de moderne skandinaviske språk, må det sies med en gang, at man ikke kunne vente seg å treffe liknende forhold i en norrøn tekst. Det skyldes hovedsakelig to grunner. Preposisjonene har etter hvert fått et langt større virkeområde enn før, fordi de i dag brukes til å indisere en rekke forhold som tidligere ble uttrykt ved oblikve kasus alene. En annen grunn er at man finner en annen gruppering av de adverbiale konjunksjoner i en norrøn tekst enn hva som er tilfelle i nåtidens skandinaviske språk.

Det ligger i sakens natur at function words er ord som ikke gjør mye av seg. Slike "fargeløse" ord, spesielt de unnværlige, er dessverre mindre konsistent enn andre,hvor man har med håndskrifter fra middelalderen å gjøre. For Hákonar saga - som for de fleste norrøne tekster - betyr det: Det antall belegg vi finner i en (annenhånds-)avskrift eller i en kritisk utgave som er laget på grunnlag av to/tre slike avskrifter, kan ikke tas for god fisk bestandig. Så her gjelder i høyeste grad: No variable is entirely safe. Investigate. - Det kanskje beste eksempel jeg kan anføre i den forbindelse, er "ok". ok topper ranglisten ved alle islendingesagaer som jeg har tall for. Den relative frekvens ligger noe høyere i Hákonar saga enn i Knytlinga, men den stemmer godt overens med det en finner i Heimskringla.Så den er det lite å gjøre med. Med anvendelsen av "ok" i teksten, d.v.s. hvor og hvordan det brukes, vil man heller ikke kunne stille opp meget, dette mest p.g.a. overleveringssituasjonen. Skrivere i middelalderen kunne nemlig etter forgodtbefinnende slenge på et par "ok", likeledes kunne de utelate et par

forekomster som de vurderte som overflødige. Som illustrasjon av skrivernes behandling av ok tok jeg vare på et eks. fra Sth.perg.no 8, bl.42v20: ....ut. ok Vóru þeir þegar drepnir. Mistanken om at ok ikke var med i den opprinnelige teksten, blir bekreftet, når man ser på andre avskrifter av samme saga.

For ikke bare å oppholde meg ved negative resultater: Når man omsider kommer fram til negasjonene, vil man der bl.a. finne noe, hvor Hákonar saga tydelig skiller seg ut blant (konge-)sagaer som det er naturlig å sammenlikne med. Vi kommer da i siste avdeling til aldri, aldrigi. Heimskringla har 109 belegg på 228 000 ord, Knytlinga har 29 på 48 500 ord. I forhold til det er belegg-antallet i Hákonar saga påfallende lite:25 på 99 000 ord Hvorfor er det slik? Går forfatteren av veien for å bruke sterke ord? En slik tanke har nok i første omgang ikke mer for seg enn den gjengse oppfatning at utstrakt bruk av polysyndese gjør en tekst kjedelig. Men spørsmålet må jo bli, hvor mange andre avvik kan vi konstatere som fører til samme indikasjon, og - minst like viktig - om vi greier å finne noe som peker i motsatt retning.

GÖTEBORGS UNIVERSITET
SPRÅKDATA

Lexikalisk databas

September 1977

PROJEKTET LEXIKALISK DATABAS

Vid Språkdata pågår förarbete för ett projekt kallat Lexikalisk databas. Projektet har som syfte att etablera en databas med svenskt språkligt material av huvudsakligen lexikalisk karaktär. En enspråkig svensk ordbok är tänkt som den primära avkastningen.

Det är naturligt att databasen av praktiska skäl till en början starkt präglas av ordbokens behov. Uppbyggnaden av databasen får emellertid betraktas som en egen uppgift, principiellt skild från framtagningen av ordboken. Det senare momentet blir främst en manifestation av databasens tillämpningsmöjligheter.

Den exakta lingvistiska informationen i databasen är ännu inte slutgiltigt fastställd. Bland de diskuterade uppgiftstyperna återfinns – utan strikt inbördes viktning – uttal, ortografi, avstavning, morfemindelning, flexion, ordklasstillhörighet, syntaktiskt konstruktionssätt, fraseologi, idiomatik, definition, specialbetydelse(r), synonymer, antonymer, stilnivå, användningsområde, frekvens/bruklighet, implicita värderingar samt etymologi. Som tunga informationstyper betraktas i synnerhet innehållslig definition och syntaktiskt konstruktionssätt.

Arbetet är fast datoranknutet. Orienteringen mot datamaskinen tar sig emellertid något olika uttryck under olika faser av projektet.

(1) Vid materialurvalet utnyttjas Logoteket. Det stoff Logoteket tillhandahåller har delvis genomgått maskinell bearbetning.

(2) Det kompletterande material som får tillföras databasen inkodas via textskärm. Likaså utförs i princip all analys och bearbetning interaktivt via textskärmen.

Materialet presenteras i fast format på skärmen. Vi diskuterar i vilken utsträckning inkodningsprogrammen också skall kunna ge förslag till analys på olika punkter. Det kan nämnas att tidningsspråksprojektet har rymt sådana moment (med olika grader av förfining) som partiellt automatisk lemmatisering och tentativ automatisk uppdelning av ord i ordsegment, och liknande operationer har vidareutvecklats inom andra projekt vid institutionen (främst Algoritmisk textanalys).

(3) Databasen lagras som ett länkat nätverk. Detta kan ge praktiska fördelar, men framför allt är det modellbygget som lockar. Nätverksstrukturen förefaller oss ha något slags psykologisk realitet när det gäller lexikalisk lagring.

En nätverksstruktur kan naturligtvis få olika grader av komplexitet. Ambitionsnivån får till en början inte bli alltför hög, utan det får endast bli vissa typer av information som lagras på detta sätt.

(4) Under planeringsstadiet har en del experimentellt arbete utförts. Härvid har de maskinella faciliteterna utnyttjats.

Det teoretiska förarbetet har främst gällt stickordens konstruktionssätt och innehållsliga definitioner. Som allmän teoretisk ram har en form av kasusgrammatik valts. En viktig del av förundersökningarna har tagit sikte på just verbens kasusramar.

I definitionerna skall en minimalordlista (definitionsvokabulär) användas för undvikande av cirkularitet. Alla ord som används i definitionerna måste vara nedbrytbara till definitionsvokabulärens innehåll inom ramen för databasens eget material. Denna infallsvinkel förutsätter att orden ordnas i ett konvertibilitetssystem. Det krävs att alla ord väljs noggrant, inte bara de som skall ingå i definitionsvokabulären utan även de icke elementära men nedbrytbara orden i definitionerna. Förundersökningar har också ägnats åt detta problem.

(5) Maskinella kontroller blir efter hand aktuella. Till dels är dessa lingvistiskt triviala, men viktiga undantag finns. Exempelvis framstår kontrollen av definitionsvokabulärens användning i definitionerna som ett språkligt intressant företag.

(6) En uteslutande praktisk – men betydelsefull – poäng med den tänkta uppläggningen utgör slutligen möjligheten att utnyttja datorstyrd fotosättning vid produktionen av ordboken. Ett magnetband kan framställas på ett enkelt sätt eftersom databasen är välstrukturerad. Kontinuerlig uppdatering av databasen, och principiellt sett även ordboken, är genomförbar med enkla medel.

- - -

Denna rapport speglar arbete som utförts av flera personer gemensamt, främst vid Språkdata. Rapporten har formulerats av Bo Ralph.

Jussi Salmela
Viljo Kohonen

CHITAB - a "poor man's" shortcut to computer processing of linguistic data

1. Background. The primary purpose of the computer programme CHITAB is modest: we hope it to be of use mainly to individual linguists or small projects with limited resources, who want to cope with sizable corpuses involving delicate classifications. In such a case processing the data manually will soon become laborious. To be more adaptable to linguistic data processing, the present programme introduces some improvements over similar programmes already existing in various programme libraries (e.g., in HYLPS, in the Univac 1108 system of the University of Helsinki, and in the SPSS, for Dec 10). These improvements include the possibility for alpha-numeric coding, the use of up to ten "filter variables" to extract from the corpus precisely the desired variables for cross-tabulation, and the possibility to pick, out of the classifications of any variable, only the frequencies of any individual classificatory principles (e.g., codes 1,3,8,A,C, out of the total range from 1-F). These improvements mean a more economic use of the card space, and a more versatile use of the computer.

The basic idea in the system is that, instead of processing both the text and the coded symbols, only the symbols are fed into the computer. This solution naturally excludes certain kinds of research, such as vocabulary frequency studies, but it is adequate for frequency counts and cross-tabulations of, e.g., various semantic, syntactic and textual features. It is thus adaptable for a variety of research purposes. An important advantage of the system is that it is remarkably cheap: the punching of the cards is fairly quick and straightforward, one card can accommodate a large number of classifications, and the computer processing is also quick.

For the benefit of the individual researcher, who is frequently unsophisticated in computer technology, we have also attempted to make the actual use of the computer as simple as possible. Thus, in order to have the computer carry out the desired cross-tabulations, the user only needs

to punch one or two cards: one card for the specification of the variables to be cross-tabulated (and giving the title of the table if desired), and another card to give the "filter" variables (if these are needed; cf. below).

2. Coding of the data. When analysing the primary material (text), the researcher transforms the features chosen for the study directly into a series of coding symbols. To do this he must have an idea of what he is looking for from the text, in the form of hypotheses to be tested or questions to which he wants to get quantitative answers. The codings are entered manually into primary matrices, according to a specific coding plan. The development of such a plan frequently involves pilot studies with experimental data. For computer processing, the data are punched onto cards.

In the present programme, one variable can be given a field of max. 5 columns, and a record can comprise a maximum of 5 cards (i.e., 400 one-column variables); with the alpha-numeric coding, one column can accommodate some 40 different symbols. In most cases, however, some 15-20 subclassifications will be enough, and one column will thus suffice. The first few columns will be two-or-more column variables, for an exact identification of the data (e.g., text/page/line, or consecutive numbering). We have not adopted the mnemonic coding symbols used, e.g., in MAMBA,[1] because the one-digit alpha-numeric symbols are more economical. In practice, we have noticed that the researcher can (and does) memorize quite a detailed coding plan with "decontextualized" symbols within the first few days (or weeks) of intensive coding work. This relieves him from constant checking of the symbols and thus speeds up the coding process. In the presentation of the results one naturally has to refer to the original text for relevant examples.

3. Checking and correcting of the data. In addition to the verifying punching, the programme provides four possibilities for the checking of the data. (1) The checking of the min. and max. limits of codings in each variable (e.g., from 1 to F; anything above F must be an error) reveals errors that are typically due to punching (these can be eliminated by verifying punching, though this is laborious in a large corpus). (2) The programme also gives the totals of coded and lacking symbols in each variable.

---

1 Ulf Teleman, Manual för grammatisk beskrivning av talad och skriven svenska, Lund 1974.

These are useful in cases in which the researcher knows that certain variables should have the same total frequencies (cf. check 3), and in extreme cases in which the researcher is reasonably sure that given variables should contain coding in all or very few records. (3) With interrelated variables, the programme can be used to check simultaneous presence/absence of coding in any combination of variables. For example, if the coding plan has four columns for different properties of the subjects of sentences (such as length, structure, givenness and position), all of these must have some coding or be blank. The computer prints out the records showing lack of agreement. (4) Further, "impossible" cross-tabulations can also be used to spot errors. Thus, for example, if passive sentences are entered in column X with codings 3,4,6, and their subjects are analysed in a subsequent column Y,1-8, a tabulation of the identification variables with X=3,4,6 and Y=blank/0 (i.e., no coding) as the filter variables will give a list of records in which the subject properties had not been entered into the matrix. Errors in (3) and (4) are thus frequently due to the researcher forgetting to enter the relevant information in all the places; such errors cannot be revealed by verifying punching. With (4) one can even get inside the total frequencies of the variables, by specifying parts of them to be checked for agreement. Finally, the actual data cross-tabulations will, of course, serve as further checks in cases of impossible or meaningless contingencies. The programme can be used to print out the dubious records for checking and eventual correction. The erroneous records will be replaced on the magnetic tape by the corrected ones.

4. The cross-tabulation programme can be utilized for the following purposes: (1) extraction of data according to the desired specifications or their combinations ("filters"), (2) cross-tabulation of any two variables against each other, (3) calculation of the Chi square test and the contingency coefficient, and (4) calculation of the relative frequencies by the totals of the rows and columns, and the sum total.

(1) The maximum of the filter variables is 10. These are given on a separate card (or more than one card), after the card specifying the variables to be cross-tabulated. An example of a table request thus looks as follows:

```
16, 28 = 1  First Table
    5 = 1-4,5,9,A   6 = 3,4   25 = 1-8,F
```

This means that variables 16 and 28 will be cross-tabulated against each other, and the data is extracted from that part of the corpus which is specified by the above values of variables 5, 6 and 25. If only portions of

the variables are desired for the tabulation, these must be given in the filter variables (e.g., giving 16 = 1,3,5,A on the second card in the above example). To save computer time, the programme has a "peeping device", i.e., if precisely the same filters are used in the following table, an auxiliary file is formed of the relevant records. This file is kept until a new combination of the filter variables is read. To make use of this possibility to save time, the researcher should group his table requests in such a way that the tables to be run from precisely the same sub-part of the corpus are in a consecutive order. This is not necessary, however, if it is more desirable to group the runs in some other way. A consequence of this limitation is that the tables to be run without any filter variables (i.e., from the whole corpus) must be run first.

(2) For the cross-tabulations, the printout format provides 25 columns and 155 rows, if needed. These limitations should be taken into account when grouping the row and column specifications. The variable given first in the table request is always interpreted as the row variable, and the second as the column variable. The frequency table is automatically supplemented by the percentages of the row and column totals out of the sum total, for quick reviews. By way of speeding up the programme, the testing of the filter variables and the up-dating of the frequency tables are both done as a uniform binary search. The programming language is FORTRAN 4.

(3) The calculation of the Chi square test is optional. It is done according to the formulas given in Siegel (1956).[1] Before applying the formulas, the programme checks that the conditions for the calculation of the Chi are satisfied by the contingency table. In the negative case, the programme prints out the reason for the inapplicability of the test. As this module also accepts contingency table data from the cards, it can be used as an independent unit for calculating the test, after possible re-groupings of the tabulated frequency data. When the Chi test is applicable, the programme also calculates the value of the contingency coefficient (C), as a measure of the degree of association between the two variables. These are given immediately below the frequency tables, with the df value.

(4) The calculation of the percentages by the totals of the rows, columns and the sum total is also optional, and it is possible to choose any of these.

---

1 Sidney Siegel, Nonparametric Statistics, New York 1956.

5. Concluding remarks. The members of the Text Linguistics Research Group are working on syntactic data collected from Finnish, Swedish and English, to be processed on the CHITAB. Kohonen has collected a corpus of some 4,000 clauses from Old and Early Middle English (between ca. 1000 and 1200) for a study of the development of OE word order. The results will be published in the research reports of the group in 1978. A documentation of the programme and the coding plans developed for Finnish and Swedish, with some preliminary findings, will appear in Erik Andersson (ed.), Working Papers on Computer Processing of Syntactic Data, Abo 1978.

Our future plans for the development of the programme include a module that can be used for matching a dependent clause to its matrix clause and forming a separate file of the matrix clauses, for analyses of desired features in them. This module could also be used in contrastive studies, when comparing translations with the originals. Another improvement could be added to the tables, where the decontextualized symbols could be automatically replaced, if desired, by mnemonic 4-character titles, to be fed into the computer separately. This would still preserve the ease and economy of the input, while making the interpretation of the tables more convenient. A further improvement could also be added to the "peeping device": instead of taking the full records into the auxiliary file (when precisely the same part of the data is going to be used for several consecutive tables), only the relevant variables, i.e., those actually needed for the tables requested, would be extracted from the data. This would again speed up the programme.

The programme is now available in the Dec 10 system of the University of Turku. We are planning to get it into the Univac 1108 system of the University of Helsinki, with terminals all over Finland.[1]

1 The designing of the operations carried out by the programme has been done jointly by the two authors, while all the technical programming work has been done by Jussi Salmela. If somebody is interested in getting a copy of the programme he should contact Jussi Salmela.

Anna-Lena Sågvall Hein

CHARTANALYS OCH MORFOLOGI

Med automatisk grammatisk analys menas vanligen en automatiserad process, varigenom en grammatisk interpretation tillordnas till en språklig enhet. Beroende på om den språkliga enheten utgörs av en isolerad ordform eller ett längre uttryck brukar man traditionellt skilja mellan automatisk morfologisk resp syntaktisk analys.

Skiljelinjen mellan morfologi och syntax tenderar inom området datalingvistik att få en extra, artificiell markering av det faktum att existerande dataprogram som regel är inriktade antingen på morfologi eller syntax. Det rör sig om programsystem med helt skild uppbyggnad och komplexitet.

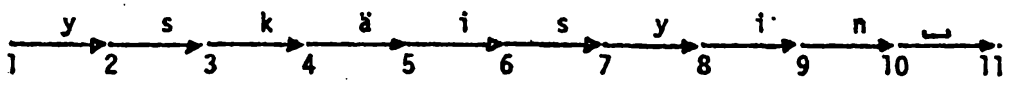
Existerande program för morfologisk analys är vanligen specifika till sin uppbyggnad i två avseenden; dels avgränsar man sig mot syntaxen genom vissa allmänna restriktioner som t ex att man utgår från att analysenheten utgörs av en teckensträng, avgränsad i löpande text genom mellanslag eller skiljetecken, dels anpassar man sig vanligen till de språkspecifika dragen i det språk man vill analysera. En sådan anpassning är givetvis eftersträvansvärd; problemen ligger i att dessa begränsningar vanligen byggs in i själva programlogiken. Man kan därför inte utveckla ett sådant system utöver de i förväg uppställda ramarna, t ex till att handskas med ett språk med större morfologisk variation eller till att klara av syntaktiska problem. Sålunda kan man t ex inte i ett traditionellt system för morfologisk analys känna igen analytiskt resp syntetiskt bildad komparativform som varianter.

Inom förefintliga system för syntaktisk analys, vilka vanligen skrivits för analys av engelska, har å andra sidan morfologin en rudimentär behandling. Man vill koncentrera sig på de 'intressanta' problemen och strävar efter att komma förbi snarare än igenom morfologin. Den morfologiska komponenten i sådana system är därför sällan utvecklingsbar.

Möjligheterna att inom ramen för ett och samma system kunna utföra såväl morfologisk som syntaktisk analys är en av grundtankarna bakom M Kay's 'Chartanalys' (1). Den kommer till uttryck bl a däri att analysenheten antingen den utgörs av en eller flera ordformer representeras med hjälp av samma struktur, en chart.

Fig 1 visar en grafisk representation av charten för den finska ordformen 'yskäisyin' (='hostning' i instr sg el pl) sådan den ter sig innan själva bearbetningen påbörjats.

fig 1

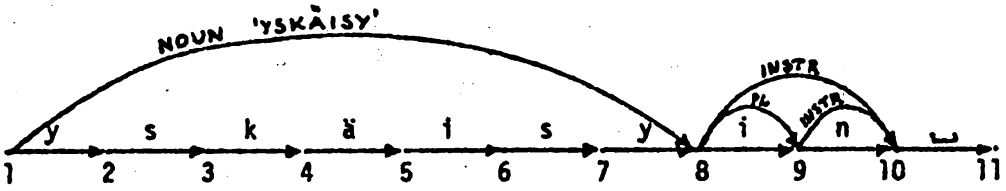


Anm. '␣' markerar mellanslag

Charten består av numrerade vertices (1 till 11), sammanbundna av riktade, etiketterade edgar (1 - 2 'y', 2 - 3 's', etc). Den intar en central plats i det av Kay föreslagna systemet för grammatisk analys. Den används inte bara för att representera analysenheten sådan den ser ut innan analysen påbörjats utan även för att lagra såväl delresultat under pågående analys som slutresultat efter avslutad bearbetning. Väsentlig blir härvid möjligheten att inom charten representera alternativa analyser.

Fig 2 visar hur charten för 'yskäisyn' skulle kunna se ut efter avslutad morfologisk analys.

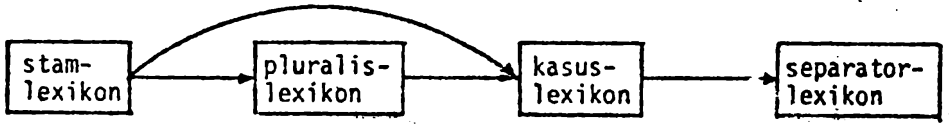
fig 2



Om vi jämför med fig 1, så finner vi att 4 nya edgar har införts, nämligen från 1 till 8, från 8 till 10, från 8 till 9 och från 9 till 10. De svarar mot två alternativa läsningar, d v s två alternativa segmenteringar, nämligen
1. 'yskäisy' - 'in' (subst 'yskäisy' i instruktiv) och
2. 'yskäisy' - 'i' - 'n' (subst 'yskäisy' i instruktiv pl).[1)]
Dessa analyser förutsätter att systemet konsulterat ett stamlexikon (huvudlexikon enl Kay's terminologi), där stammen 'yskäisy' med informationen 'substantiv' återfinns, ett suffixlexikon, som upptar suffixet 'i' med informationen 'pluralis', ett suffixlexikon, som upptar suffixen 'n' och 'in', båda med informationen 'instruktiv' samt slutligen ett 'separatorlexikon', som upptar ' '.[2)]

Det finns i det generella systemet som sådant inga begränsningar på hur många morfemsegment som kan följa på varandra i en ordform eller någon uppgift om vilka de är eller om den inbördes ordningen mellan dem. Som språkspecifik information måste man därför, förutom ett huvudlexikon, även tillföra systemet ett antal lexikon (Kay's terminologi) som upptar de faktiska realisationerna av de grammatiska morfemen, jämte de morfotaktiska reglerna. Morfotaxen avspeglas i en hänvändelse vid de olika segmenten i huvudlexikon resp suffixlexikon till vilket lexikon skall konsulteras för igenkänning av följande segment. Kopplingen mellan de olika lexikonen, som ligger bakom analysen i fig 2, dvs den bakomliggande morfotaxen illustreras i fig 3.

fig 3



1) Tolkningen av morfemsegmenten ligger som synes i etiketterna för motsvarande edgar.
2) Konsultationen i separatorlexikonet har till uppgift att kontrollera att ordformen i fråga är slut, d v s i det anförda exemplet att analysenheten verkligen är uttömd i och med återfinnandet av kasussegmentet.

Införandet av ett särskilt separatorlexikon bidrar till systemets generalitet och konsistens. Det innebär, att kontroll av huruvida en ordform är slut eller inte kan ske helt i analogi med hur en jämförelse mellan en bokstav i analysenheten - en edge i charten - och en bokstav i något av lexikonen, går till. Det allmänna begreppet för en sådan aktion är 'task' (uppgift). En uppgift kan t ex också svara mot tillämpningen av en grammatisk regel på en eller flera edgar. Under bearbetningens gång lagras uppgifterna i en (eller flera) agenda (agendor), varifrån de aktiveras. Hela analysprocessen kan beskrivas som en följd av uppgifter. [1]

Att låta de morfotaktiska reglerna komma till uttryck enbart via kopplingen mellan de olika lexikonen är emellertid inte tillräckligt för att garantera korrekta analysresultat. Vi kan illustrera problematiken med ett exempel. Antag att våra lexikon (för den finska morfologin) innehåller bl a följande uppslagsord, fig 4.

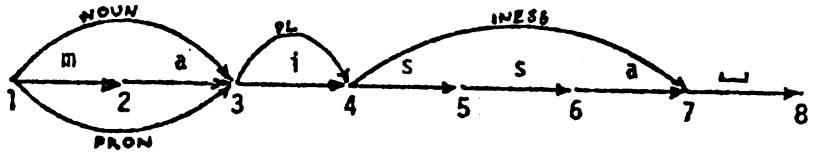
fig 4

```
*MAIN:

        MA      1. MORPHCAT=WORD, LEXEM=MA, SYNCAT=PRON
                   continue in dictionary SEP

                2. MORPHCAT=STEM, LEXEM=MAA, SYNCAT=NOUN
                   continue in dictionary NUM

+ + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +

NUM:

        I       1. NUM=PL
                   continue in dictionary CASE

+ + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +

CASE:

        SSA     1. CASE=INESS
                   continue in dictionary SEP

+ + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +

SEP:

        ␣␣      1. continue in dictionary *MAIN
```

Då skulle den morfologiska analysen av ordformen 'maissa' (inessiv plur av 'land') ge följande chartstruktur, fig 5.

1) Se vidare om systemets generella uppbyggnad i (1) och (2).

fig 5



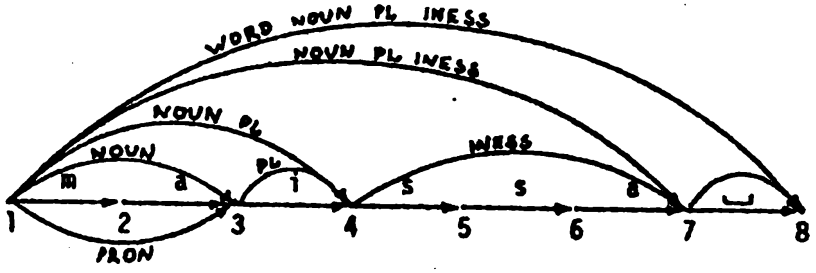
Denna chartstruktur medger två läsningar, nämligen 1. substantivet MAA i inessiv plur och 2. pronominet MA i iness plur, varav endast den första är korrekt. Orsaken till att vi i fall av homografi, t ex stamhomografi som ovan, får en felaktig alternativ analys är tvåfaldig; dels beror det givetvis på själva chartstrukturen, dels på det faktum att lexikonsökningen och segmenteringen sker helt kontextoberoende, d v s att en ny edge läggs in i charten så snart man finner överensstämmelse mellan ett segment (uppslagsord) i något av lexikonen och en följd av edgar i charten. Med bibehållande av chartstrukturen kan vi nalkas problematiken på två sätt, antingen genom att göra segmenteringsprocessen kontextsensitiv eller genom att formulera fristående kompatibilitetsregler, som appliceras på den kontextfritt genererade chartstrukturen, och vars tillämpning leder till att övergripande edgar införs för att markera kompatibilitet mellan segment.

I min egen implementering av chartanalysen har jag valt att göra segmenteringsprocessen kontextkänslig. Här arbetar jag på så sätt, att ingen ny edge introduceras i charten förrän systemet verifierat, att det segment, som representeras av edgen, är kompatibelt med angränsande segment. När man arbetar med en ordform som analysenhet innebär det i praktiken, att införandet av nya (segment-)edgar måste undertryckas, till dess man undersökt hela ordformen, inkl separatorn. I exemplet ovan (fig 5) skulle alternativet 'pron' aldrig ha införts i charten, enligt min implementering, då systemet ej återfunnit den förväntade separatorn. Fördelen med den här metoden är dels den, att man får ett väsentligt mindre belastat chart, eftersom endast de korrekta alternativen återspeglas i charten, dels den att analysen kommer att ske i färre etapper, då man i samma bearbetningssteg både segmenterar och verifierar kompatibilitet.[1] Detta tillvägagångssätt har å andra sidan den nackdelen, att det är svårt att vara lika generell som vid en kontextfri segmenteringsprocess. Språkspecifik information smyger sig lätt in i programmen, varför programsystemet måste ha en modulär uppbyggnad, som gör de språkspecifika programmen lätt utbytbara.

Låt oss diskutera det andra alternativet, där man bibehåller en kontextoberoende segmenteringsprocess och låter kompatibiliteten mellan segmenten kontrolleras i påföljande bearbetningssteg. Fig 5 illustrerar då endast en etapp i analysen av 'maissa'. En möjlig slutgiltig chartstruktur visas i fig 6.

1) För en illustration av denna metod, se (3).

fig 6



Vid läsningen av chartstrukturen tillämpar man den konventionen, att det slutgiltiga analysresultatet finns i etiketten (etiketterna) till den (eller de) edge (edgar) som går från första till sista vertex. Om den resulterande charten inte uppvisar någon sådan edge har analysen inte lyckats. Felaktiga delanalyser, som i exemplet tolkningen av 'ma' som ett pronomen, stör därigenom inte slutresultatet. Edgen från vertex 1 till vertex 7 representerar enbart en delanalys, då man ännu inte har något belägg för att ordet de facto slutar efter kasussegmentet. Återfinnandet av separatorn tillför här igen ny information utan har endast kontrollfunktion.

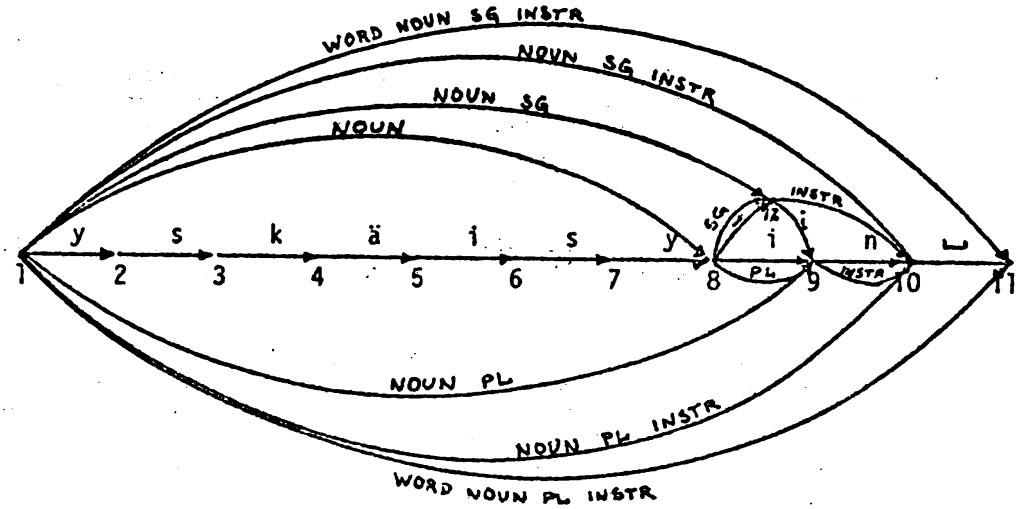
Vilka är då problemen inom chartanalysens ram, då det gäller att komma fram till en sådan analys som i fig 6? Successivt måste man verifiera kompatibiliteten mellan segmenten, t ex 'noun - pl', 'pron - pl'. Om det råder kompatibilitet, skall en övergripande edge införas, som i sin etikett bär den relevanta informationen från de båda edgarna. Detta skall ske genom att en regel appliceras på de båda edgarna. Tillämpningen av en regel på en eller flera edgar formuleras som alla andra aktioner inom chartanalysteorin som en uppgift. När skall nu den uppgiften genereras och varifrån skall den hämta information om vilket test, som skall utföras, samt om vad, som skall ingå i etiketten till den övergripande edgen? Jag har valt följande strategi:

1. Kompatibilitetstestet jämte informationen om etikettens innehåll och uppbyggnad sammanförs i en regel.
2. Reglerna skall formuleras enligt samma notation, i vilken de syntaktiska reglerna är uttryckta samt kunna utnyttja samma grammatiska operatorer, se (4).
3. Reglerna skall sammanföras i en morfologisk grammatik, helt i analogi med den syntaktiska grammatiken.
4. Namn på aktuell regel skall ingå i lexikoninformationen till resp suffix.
5. Reglerna skall initieras, d v s leda till generering av aktuell uppgift, i samband med återfinnandet av suffix-segmenten.
6. Reglerna skall verka från höger till vänster, d v s på aktuellt suffix jämte föregående segment.

Strategin fungerar, och jag har formulerat 3 morfologiska regler, som garanterar kompatibilitet mellan stamsegment, numerussegment, kasussegment och separator hos finska nomina. Kopplingen till de syntaktiska faciliteterna, d v s möjlighet att utnyttja de väldefinierade grammatiska operatorerna, har också gjort det möjligt att utan ytterligare utveckling av Kay-systemet kunna utföra kompatibilitetskontroll, som motiveras av förekomst av såväl stam- som suffixallomorfer.

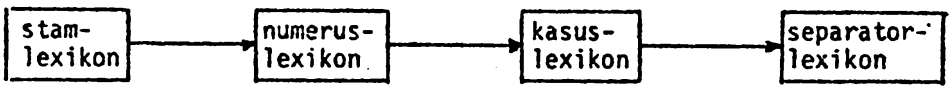
Låt oss gå tillbaka till fig 2. Här betraktas distinktionen mellan singularis och pluralis som en distinktion markerad/omarkerad, där pluralis är den markerade parten, d v s om inget uttryck för pluralis återfinns, så tolkas formen i fråga som singulär. Alternativt kan man vilja betrakta uttryck för numerus som obligatoriskt, där det singulära segmentet realiseras som ett noll-segment. Systemet erbjuder denna möjlighet. Då får man som alternativ till segmenteringen i fig 2 den segmentering som presenteras i fig 7.

fig 7



1. 'yskäisy' - '␣' - 'in'
2. 'yskäisy' - 'i' - 'n'

Som synes sker en omskrivning i ursprungscharten på så sätt att ett noll-segment explicit läggs in i charten som en edge (8 - 12 '␣'). Denna omskrivning sker selektivt, nämligen vid morfemgräns vars andra morfem kan realiseras som ett noll-segment. Motsvarande suffixlexikon upptar noll-segmentet med tillhörande tolkning. Med denna strategi kommer också kopplingen mellan de olika lexikonen att se något annorlunda ut, jfr fig 8 och fig 3.



Behovet av att kunna göra omskrivning ('rewriting') av ursprungscharten aktualiseras inte bara i samband med noll-segment utan även vad det gäller hela det problemkomplex som rör hantering av morfofonematiska växlingar, såvida man ej väljer att i sina lexikon explicit uttrycka alla varianter. Det senare alternativet tvingar oss att avstå från att

fånga upp intressanta generaliseringar i det språkliga materialet och därigenom skapa ett mindre insiktsfullt system, förutom att det leder till en väsentlig belastning på de olika lexikonen. För finskans del gäller det såväl fonologiskt som grammatiskt betingad morfofonematisk växling, nämligen volkalharmoni, vokalstrykning, kvantitativ resp kvalitativ stadieväxling, vokalförändring före suffix på -i och vokalassimilation. Hela detta problemkomplex kan spaltas upp i ett antal mindre frågor. Hur skall omskrivningsreglerna formuleras? Vilken alternant skall väljas som lexikonrepresentant? Hur skall reglerna integreras i den övriga bearbetningen? Kan och bör skillnaderna mellan de fonologiskt och de grammatiskt betingade morfofenomatiska växlingarna reflekteras genom principiellt olika behandling under analysen?

Av de språkliga fenomenen ovan har jag hittills endast bearbetat vokalharmonin. Den metod jag utarbetat ger möjlighet att behandla den finska vokalharmonin i icke sammansatta ord. Som arkisymboler för harmonivokalerna har jag valt /a,o,u/. Sålunda representeras t ex inessivmorfemet, med allomorferna 'ssa', 'ssä', av segmentet 'ssa' i kasuslexikonet.
Omskrivningen av 'ä' till 'a', 'ö' till 'o' samt 'u' till 'y' sker helt kontextfritt i samband med uppbyggandet av ursprungscharten, varvid ett speciellt 'bokstavslexikon' konsulteras. Erforderlig kompatibilitetskontroll sker med hjälp av den ovan skisserade metodiken. Stammar, som enbart innehåller de neutrala vokalerna 'e' och 'i', måste markeras som främre i lexikon. Övriga stammar innehåller ingen lexikoninformation om huruvida de är främre eller bakre, utan den informationen härleds under analysens gång.

Anm. För närmare information om status på projektet 'Chartanalys och finsk morfologi', vilket jag bedriver i samarbete med Erling Wande från Finsk-ugriska institutionen i Uppsala hänvisas till (5).

Referenser

1. Kay, M, Morfological and syntactic analysis, Lecture notes from the 3rd International Summer School of Computational and Mathematical Linguistics, Pisa 1974

2. Kay M, Syntactic Processing and Functional Sentence Perspective, Handbook from 1977 Nordic Summer School in Computational Linguistics

3. Sågvall Hein, A-L, An Approach to the Construction of a Text Comprehension System for X-ray Reports, Proc of the IFIP Working Conference on Computational Linguistics in Medicine, eds W Schneider, A-L Sågvall Hein, North-Holland Publ Comp 1977, pp. 91-99

4. Kay, M, Reversible Grammar, A Summary of the Formalism, Handbook from 1977 Nordic Summer School in Computational Linguistics

5. Analysresultat, lexikon, grammatiska regler och funktionsdefinitioner från projektet 'Chartanalys och finsk morfologi', Datalistor, UDAC, Uppsala

NORDISKA DATALINGVISTIKDAGAR

10-11 oktober 1977

Deltagarförteckning

Gulbrand Alhaug	Inst. for språk, Pb 1090	N-9001 Tromsø
Sture Allén	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Hans Basbøll	Nordisk inst., Odense univ. Nils Bohrs Allé	DK-5260 Odense
Sture Berg	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Inger Bierschenk	Ljungsätersvägen 10, Ljunghusen	S-230 12 Höllviksnäs
Leiv Egil Breivik	Inst. for språk og litteratur, Universitetet i Tromsø, Pb 1090	N-9001 Tromsø
Benny Brodda	Inst. för lingvistik, Stockholms universitet	S-106 91 Stockholm
Thorkil Damsgaard Olsen	Gærdet 15	DK-3460 Birkerød
Carin Davidsson	St. Olofsgatan 1 A	S-752 35 Uppsala
Mats Eeg-Olofsson	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Knut Fintoft	Lingvistisk inst. Universitetet i Trondheim	N-7000 Trondheim
Ivar Fonnes	Oslo univ., Pb 1102, Blindern	N-Oslo 3
Rolf Gavare	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Martin Gellerstam	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Eric Grinstead	Nordisk Asieninst., Kejsergade 2	DK-1155 Köpenhamn
Suzanne Hanon	Rypebakken 48	DK-5210 Odense NV
Kolbjörn Heggstad	Nord. inst. PDS, Harald Hårfagresgate 29II, Bergens universitet	N-5014 Bergen
Jostein Helland Hauge	NAVF:s EDB-senter for humanistisk forskning, Villavei 10, Box 53	N-5014 Bergen
Knut Hofland	NAVF:s EDB-senter for humanistisk forskning, Villavei 10, Box 53	N-5014 Bergen
Henrik Holmboe	Inst. for lingvistik, Otto Rudsg. 67-69	DK-8200 Århus N
Hans Holmgren	Ämnesgruppen för datalogi, Mat. inst. Linköpings universitet	S-581 83 Linköping
Jan Hultgren	Stockholms datamaskincentral, Arrheniuslab., Stockholms univ.	S-106 91 Stockholm
Elisabeth Ingram	Anv. Språkvitenskap, Univ. i Trondheim, Lade	N-7000 Trondheim
Jens Juhl Jensen	Inst. for lingvistik, Njalsgade 96	DK-2300 Köpenhamn
Stig Johansson	Britisk inst., Univ. i Oslo, Pb 1003, Blindern	N-/slo 3
Baldur Jónsson	Hraunbæ 194	I-110 Reykjavík
Ingela Josefson	Inst. för nordiska språk, Göteborgs univ., Viktor Rydbergsgatan 24	S412 53 Göteborg
Paul Jørgensen	Datalogisk avd., Matematisk inst. Aarhus univ., Ny Munkgade	DK-8000 Aarhus
Gregers Koch	Datalogisk inst., Sigurdsgade 41	DK-2200 Köpenhamn

Viljo Kohonen	Turun yliopisto, Juslenia, English Dept.	F-20500 Turku 50
Kjeld Kristensen	Cedervaenget 7 st. tv.	DK-2830 Virum
Svein Lie	Nordisk inst., Univ. i Oslo	N-Oslo 3
Eirik Lien	Univ. i Trondheim, NLHT	N-7000 Trondheim
Mogens Baumann Larsen	Jagtvej 8	DK-9000 Ålborg
Lise Lorentzen	Avd. for fil. fag: fransk, Univ. i Trondheim, NLHT	N-7000 Trondheim
Jonas Löfström	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Bente Maegaard	IAMLL Njalsgade 96	DK-2300 Köpenhamn
Ulla-Britt Mattsson	Biskopsgatan 10 A 30	F-20500 Åbo 50
Brian Mayoh	Datalogisk avd., Aarhus universitet	DK-3000 Århus
Zuzanna Michalska	Avd. för fonetik, Umeå universitet	S-901 87 Umeå
David Mighetto	Nyponvägen 39	S-440 03 Floda
Marina Mundt	Univ. i Bergen, Nord. inst. avd. B, Pb 23	N-5014 Bergen
Johan Nedregård	Germanistisk inst., Avd. A, Blindern	N-Oslo 3
K. Hvidtfelt Nielsen	Inst. for germ. filologi, Århus univ., Bygning 326, Ndr. Ringgade	DK-8000 Århus
Kerstin Nordin	Inst. för nordiska språk, Uppsala univ., Thunbergsvägen 3	S-752 41 Uppsala
Antti J. Pitkänen	Koulukatu 5 B 36	F-33200 Tammerfors 20
Per-Bjørn Pedersen	Rogaland distriktshøgskole, Studiesenteret Ullandhaug	N-4001 Stavanger
Harald Pors	Inst. for germ. filologi, Århus univ.	DK-8000 Århus C
Bo Ralph	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Anna Karin Ro	Engelsk inst., Univ. i Trondheim	N-7000 Trondheim
Hanne Ruus	Inst. for nord. filologi, Köpenhamns univ., Amager, Njalsg. 80	K-2300 Köpenhamn
Jussi Salmela	Turun yliopisto, Fennicum	F-20500 Turku 50
Christin Sjögreen	Språkdata, N. Allégatan 6	S-413 01 Göteborg
Harald Solevåg	Nord. inst., PDS, Harald Hårfagres-gate 29II, Bergens univ.	N-5014 Bergen
Ebbe Spang-Hanssen	Romansk inst., Rigensgade 13	DK-1316 Köpenhamn
Jan Svartvik	Eng. inst., Lunds univ., Helgonabacken 14	S-223-62
Anna-Lena Sågvall Hein	UDAC, Box 2103	S-750 02 Uppsala
Georg Søndergaard	Nordisk inst., Odense univ.	DK-5230 Odense M
Cecilia Thavenius	Eng. inst., Lunds univ., Helgonabacken 14	S-223 62 Lund
Mats Thelander	FUMS, Thunbergsv. 7I (Gula villan)	S-752 38 Uppsala
Jan Gunnar Tingsell	GUME/Prov, Lärarhögsk. i Mölndal, Fack	S-431 20 Mölndal
Alan Tucker	Inst. for språk og litteratur, Universitetet i Tromsø, Pb 1090	N-9001 Tromsø
Carl Wilhelm Welin	Inst. för lingvistik, Stockholms univ.	S-106 91 Stockholm
Staffan Wåhlin	Språkdata, N. Allégatan 6	S-413 01 Göteborg