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.

Nyord-registrering i database
Kolbjørn Heggstad, Harald Solevåg
Proceedings of NODALIDA 1977, pages 40-49

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 eksempl. 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:

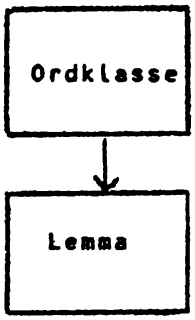
Post-type betegner samlingen av dataelementer av en gitt type, og blir anskueliggjort med et rektangel.

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:

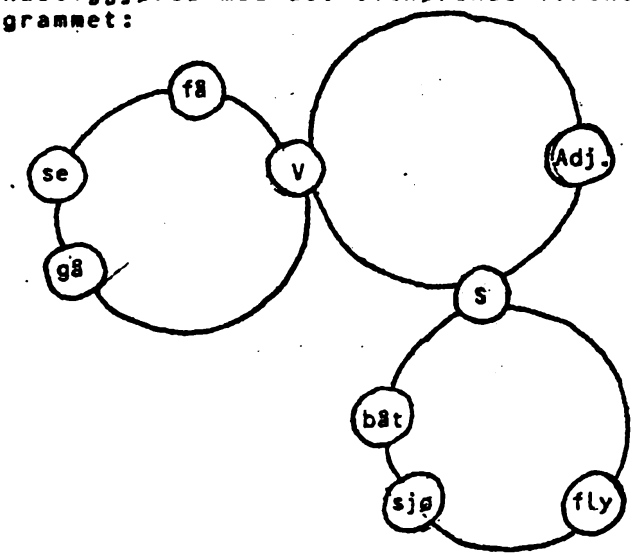
Typediagrammet:



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.

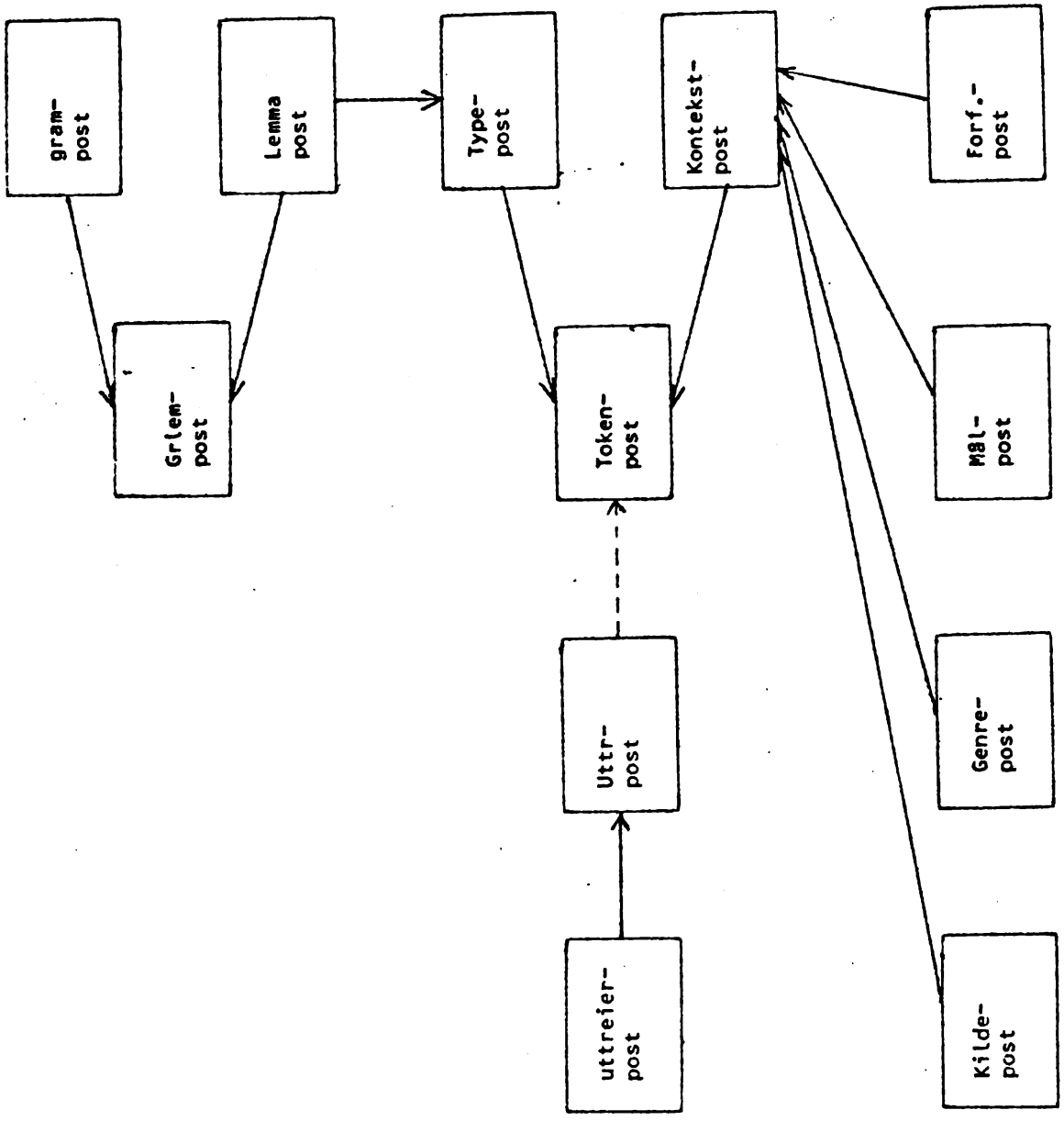
MÅL: Bokmål eller nynorsk.

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.

TYPEDIAGRAM



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.
```