

STEFFEN LEO HANSEN

På vej mod en fagsproglig tekstfortolker?

Abstract

The FAGFLADE Project

“FAGFLADE” is short for Danish “fagsproglig grænseflade” (“special purpose language interface”). The aim of the FAGFLADE project is to develop and test theories and methods for automatic interpretation of texts written in special purpose language. We use the expression “text interpreter” to designate a program which transfers the information contained in a natural language text into knowledge representations in a knowledge base. Thus a text interpreter is meant to perform part of the task of a knowledge acquisition system in an expert system.

The FAGFLADE project takes as its basis the text of the Danish Companies Act (“Lov om aktieselskaber”). It is not our ambition to build an expert system, nor do we aim at the construction of a complete text interpretation system, fit for use, and capable of translating statutes into legal knowledge bases. Rather, we take the development of a specific interpreter to be an ideal goal which defines an overall project capable of giving rise to a handful of interesting subprojects for the investigation of general theories and principles concerning interpreters, e.g. in the domain of syntactic and semantic analysis, parsing strategies, knowledge representation, dictionary databases, and terminological analysis.

The choice of the Danish Companies Act as a text basis is motivated partly by our desire to work with an LSP central to industry and commerce, partly by the evident practical perspectives which will open up if text interpreters for legal directives become a realistic possibility.

This paper deals with the initial phase of the project. In this phase we have constructed a framework for the syntactic and semantic analysis of simple sentence structures, and we have developed a program, written in Quintus Prolog, which carries out syntactic and semantic parsing of these structures.

1 Introduktion

På Institut for Datalingvistik på Handelshøjskolen i København har vi siden instituttet blev oprettet i 1985 udviklet en tradition for et forskningssamarbejde,

der i starten formede sig som en række datalingvistiske seminarer over udvalgte emner, men som siden 1987 har ført til arbejdet på et egentligt forskningsprojekt med deltagelse af alle forskningsmedarbejdere, nemlig projektet FAGFLADE, hvilket står for 'fagsproglig grænseflade'.

Målet for dette projekt er at afprøve og udvikle teorier og metoder bag konstruktionen af en tekstfortolker, dvs. et system som analyserer en tekst og overfører viden herfra til vidensrepræsentation i en vidensbase. En sådan tekstfortolker, forestiller vi os, kan indgå som komponent i et ekspertsystem som en del af et vidensindlæringsmodul.

Den tekst vi har valgt som grundlag for projektet er den danske „Lov om aktieselskaber“, og det sprog som er repræsenteret i loven er dermed det sublanguage som fortolkeren skal kunne analysere og forstå.

Projektet sigter ikke mod at udvikle et færdigt ekspertsystem, men er udgangspunkt for en række mindre projekter der fokuserer på problemer og områder knyttet til udviklingen af en tekstfortolker, det vil først og fremmest sige grundlaget for den syntaktiske og semantiske analyse, vidensrepræsentation, udvikling og afprøvning af parsing strategier, den maskinlæsbare ordbogs struktur og omfang samt terminologiske problemer så som identifikation og repræsentation af flerledelede termer, synonymy, definitioner og begrebsrelationer.

I dag er projektet nået dertil at der er udviklet en parser i Quintus Prolog som kan klare simple sætningsstrukturer, dvs. sætninger der kun indeholder valensbundne led som optræder på den forventede position i sætningen. Vi kan altså ikke på nuværende tidspunkt klare spørgebisætninger eller topikaliseringer, heller ikke adverbialer, ledsætninger og adskillige andre ting.

Det output vi får fra parseren er dels sætningens syntaktiske struktur i form af et konstituentstrukturtræ, dels en semantisk struktur. Den syntaktiske struktur anvendes ikke i øvrigt af tekstfortolkeren, hvorimod den semantiske struktur skal danne udgangspunkt for den vidensrepræsentation som skal overføres til vidensbasen.

Et input som

Denne aftale påfører selskabet en forpligtelse

vil derfor give flg. output:

- (1) $s(np(det(denne), n(aftale)), vp(v(påfører), np(n(selskabet)), np(det(en), n(forpligtelse))))$.
- (2) $påføre(agent(aftale), theme(forpligtelse), locus(selskab))$

Grundlaget for den syntaktiske analyse og den semantiske repræsentation er dels en konstituentstruktur grammatik, dels et leksikon med en leksikalsk beskrivelse af verberne ud fra en valensbaseret kombination af grammatiske funktioner og roller. Jeg skal i det følgende komme nærmere ind på de overvejelser der ligger til grund for denne leksikalske beskrivelse og på dens implementering i selve parserprogrammet.

2 Konstituenterne

Den parser vi har bygget er en 'left-corner, bottom-up' parser. Den starter således med det første ord i en input-sekvens og forsøger at opbygge en konstituentstruktur for et S som har dette første ord som sit venstre hjørne. Grammatikken der rummer fig. genskrivningsregler, hvoraf 2-9 repræsenterer de simple sætningsstrukturer som parserne kan klare netop nu:

(1)	S	-->	NP	VP	
(2)	VP	-->	V		
(3)	VP	-->	V	NP	
(4)	VP	-->	V	PP	
(5)	VP	-->	V	AP	
(6)	VP	-->	V	NP	NP
(7)	VP	-->	V	NP	PP
(8)	VP	-->	V	NP	AP
(9)	VP	-->	V	PP	PP
(10)	PP	-->	P	NP	
(11)	PP	-->	P	AP	
(12)	NP	-->	N		
(13)	NP	-->	DET	N	
(14)	AP	-->	A		
(15)	AP	-->	A	PP	

I leksikon er hvert enkelt ordform forsynet med en oplysning om konstituenttype. Subkategoriseringen af VP'erne genfindes i leksikon som en subkategoriseringsramme, der repræsenterer de konstituenttyper som verbet kan forbindes med som valensled. Denne konstituentramme skal unificeres med en identisk ramme i en Prolog regel og tjener derfor det formål at effektivisere selve parsningen. Hvis unificationen lykkes, vil parseren vælge den pågældende regel.

Vi kan således begynde at se på opbygningen af en post i leksikon der har form af en Prolog klausul med prædikatet `d/6` hvis argumenter alle er sammensatte termer:

```
d(word(_), lexeme(_), lexcat(_), gram_form(_),
  synt_spec(_,_), sem_spec(_,_)).
```

For verbet 'påføre' som optræder i eksempelsætningen ovenfor ser udfyldningen af de første argumenter således ud:

```
d
  word:           påfører
  lexeme:         påføre
  gram_form:      [pres]
  lexcat:         v
  synt_spec
    const_frame:  v_np_np
    adj:          [nil]
```

Argumentet *lexcat* indeholder oplysning om konstituenttype, *synt_spec* har to sammensatte termer som argumenter: dels *const_frame*, der angiver konstituentrammen, dels *adj* (adjekt), der som argument har en liste over tilladte præpositioner hvis der kan optræde et valensbundet PP. Notationen [nil] angiver at det ikke kan forekomme ved dette verbum.

3 Funktioner

De funktioner der opereres med i FAGFLADE er hentet fra den valensteori som Herslund og Sørensen (1985, 1988a, 1988b) har udviklet i deres arbejde med franske verber. Der kan her kun blive plads til at skitsere hovedtrækkene i denne teori, der bygger på en distinktion mellem sætningsled der hører til hele sætningen, frie tilføjelser, og sætningsled som er dikteret af det konkrete verbum som optræder i sætningen. Et valensled defineres som det syntaktiske udtryk for de relationer der består mellem et verbum som semantisk prædikat og dets argumenter og mellem argumenter indbyrdes, og det er derfor kun verbalafhængige sætningsled med argumentstatus der kan opfattes som valensled. Endvidere antages det at valensled alene kan beskrives i form af funktioner, og at der til enhver leksikalsk beskrivelse af et verbum hører et valensskema med angivelse af hvilke funktioner, resp. valensled, det pågældende verbum forbindes med. Der opereres med et inventar på tre grammatiske funktioner: subjekt, objekt og adjekt. Hertil har vi i FAGFLADE for egen regning føjet en fjerde funktion, agential, for også at kunne håndtere passiv.

Sammenlignet med antallet af valensled i traditionelle valensgrammatikker er der sket en drastisk reduktion i antallet af funktioner. Funktionen adjekt repræsenterer på den ene side en valensgrammatisk nytænkning, og er på den anden side den funktion som rummer nøglen til rationaliseringen i antallet af funktioner.

Traditionelt er man vant til at arbejde med valensled som DO (dativobjekt), IO (indirekte objekt), A eller LO (adverbial/lokativobjekt), SP og OP (hhv. subjekts- og objektsprædikat). Ved anvendelsen af et givet verbum kan man således komme ud for at det er muligt at vælge mellem en konstruktion hvor verbet optræder med DO eller IO: man kan fx. 'give *nogen* noget' eller 'give noget *til nogen*', eller det kan være et valg mellem DO og LO: 'jeg har efterladt ham mine bøger' overfor 'jeg har efterladt mine bøger hos ham'. Hvad man derimod ikke kan komme ud for, er en kombination af begge funktioner samtidig, dvs. at de ovennævnte, traditionelle valensled gensidigt udelukker hinanden. Derfor antages det, at de alle repræsenterer en og samme ledfunktion som imidlertid kan realiseres på forskellig måde. Vi kan illustrere det med fig. eksempler:

- (1) Han efterlod børnene en betragtelig formue
 DO O
- (2) Han efterlod en betragtelig formue til børnene
 O IO

- (3) Han efterlod $\underbrace{\text{sin bog}}_O$ $\underbrace{\text{hos sin broder}}_{LO}$
- (4) Han efterlod $\underbrace{\text{sin bil}}_O$ $\underbrace{\text{på banegården}}_{LO}$
- (5) Han efterlod $\underbrace{\text{hende}}_O$ $\underbrace{\text{dypt frustreret}}_{OP}$

Argumentationen for at sammenfatte i dette tilfælde DO, IO, LO og OP i een og samme funktion, *adjekt*, er, at uanset hvilken betegnelse man vælger at anvende så gælder det i alle konstruktioner at funktionen som sådan etablerer en ny relation i sætningen, i ovennævnte tilfælde en relation mellem *objekt* + *adjekt*. Denne relation opfattes som en sekundær prædikation der kan parafraseres med enten VÆRE eller HAVE og som supplerer verbets betydning ved at placere referenten for O i forhold til referenten for adjektet.

Placeringen kan være enten meget konkret som i eksemplerne (3): bogen er hos hans broder, og (4): bilen er på banegården, eller mere abstrakt som i (1): børnene har en betragtelig formue. Der kan imidlertid også være tale om en placering i forhold til en egenskab eller en klasse, nemlig i de tilfælde hvor adjektet svarer til en prædikativ konstruktion med enten SP eller, som i (5), med OP.

Begrundelsen for at indføre adjekt er derfor på den ene side at det er afhængigt af sætningens verbal som argument i logisk forstand, og på den anden side at det som argument gør det muligt at relatere et af de to fundamentale valensled, S og O, til et andet argument i sætningen og at denne relation mellem valensled indbyrdes er udtryk for en sekundær prædikation. Udfyldningen af adjektet kan være forskellig, men den sekundære prædikation er konstant.

Som tidligere nævnt omfatter den leksikalske beskrivelse af et verbum et valensskema som angiver hvilke funktioner der kan forekomme sammen med et givet verbum. Et sådant valensskema findes også i det leksikon som vores parser benytter, men inden vi skal se nærmere på dette vil det være nødvendigt at behandle de tematiske roller, idet de grammatiske funktioner i valensskemaet associeres med en semantisk rolle.

4 Roller

Med udgangspunkt i opfattelsen af verbet som et prædikat med tilhørende argumenter kan de *semantiske relationer* mellem argumenterne beskrives vha. semantiske roller. Den semantiske analyse støtter sig ligeledes på Herslund og Sørensen (1985) samt endvidere på Korzen et alii (1983) og deres arbejde med den såkaldte PK-grammatik, idet den ovenfor omtalte valensteori og de syntaktiske funktioner er udgangspunkt for at associere funktioner med roller.

Antallet af roller synes at være en temmelig ubestemmelig størrelse, men vi har foretrukket at reducere antallet i henhold til de nævnte arbejder og i vores projekt indskrænket det til tre: agent, tema og locus.

agent svarer til den traditionelle opfattelse af agent og kan kun associeres med funktionerne *subjekt* og *agential*.

locus optræder kun ved prædikater med mindst to argumenter. Denne rolle relaterer referenten for et andet argument til den location som locus-argumentet udtrykker. *Locus* kan associeres med funktionerne *subjekt*, *adjekt* og *agential*.

tema er semantisk set den meste generelle og den mindst præcise af de tre roller. Referenten for tema er den 'entity' som enten påvirkes af handlingen eller placeres mht. en location. *Tema* kan associeres med funktionerne *subjekt*, *object* og *adjekt*.

Distribution og kombination af roller associeret med funktioner kan således se ud som vist i flg. skemaet:

VERB	SUBJ	OBJ	ADJ	AGL
anse	agent	tema	locus	none
anses	tema	none	locus	agent
bestemme	agent	tema	none	none
bortfalde	tema	none	none	none
eje	locus	tema	none	none
ejes	tema	none	none	locus
finde	agent	tema	locus	none
påføre	agent	tema	locus	none
være	tema	none	locus	none

Af skemaet fremgår at fx. verbet *anse* optræder med en funktionsramme 'subjekt + objekt + adjekt' og at der dertil svarer rollerammen 'agent + tema + locus'. Udtrykket 'none' betyder at den pågældende funktion ikke kan forekomme ved det anførte verbum.

Jeg vil gerne knytte en enkelt kommentar til rollen locus med udgangspunkt i disse sætninger:

- | | | |
|--|------------|--------------------------|
| (1) spørgsmålet <i>bortfalder</i> | sub(theme) | |
| (2) ministeren <i>bestemmer reglen</i> | sub(agent) | obj(theme) |
| (3) stifterne <i>ejer dette selskab</i> | sub(locus) | obj(theme) |
| (4) disse aktier <i>er fondsaktier</i> | sub(theme) | adj(locus) |
| (5) bestyrelsen <i>finder disse undersøgelser nødvendige</i> | sub(agent) | obj(theme)
adj(locus) |
| (6) denne aftale <i>påfører selskabet en forpligtelse</i> | sub(agent) | obj(theme)
adj(locus) |
| (7) disse undersøgelser <i>anses for nødvendige af bestyrelsen</i> | sub(theme) | adj(locus)
agt(agent) |

I sætning (4) og (5) er locus associeret med funktionen adjekt, som her svarer til hhv. det traditionelle subjektsprædikativ (4) og objektsprædikativ (5), og i (6) er locus associeret med et adjekt der svarer til det traditionelle dativobjekt.

Jeg anførte tidligere at rollen locus kun optræder sammen med et andet valensled og at den denoterer en location for referenten denoteret af det andet

valensled. Ser vi på de tre sætninger ovenfor så er det subjektet, *disse aktier*, der i (4) localiseres ekstensionalt i forhold til en klasse, klassen af *fondsaktier*, adjektet, og i (5) er det objektet, *disse undersøgelser*, der localiseres intensionalt med hensyn til egenskaben at være nødvendig, udtrykt i adjektet. Ligesom for funktionernes vedkommende kan locationen også være mere abstrakt sådan som det er tilfældet i (6) hvor objektet, *en forpligtelse*, localiseres i forhold til *selskabet*, adjektet, forstået på den måde at det er selskabet som har en forpligtelse.

Vi kan hermed afslutte udfyldningen af argumenter i leksikonposten og indsætte valensskemaet i form af de semantiske specifikationer:

```

d
  word:           påfører
  lexeme:         påføre
  gram_form:      [pres]
  lexcat:         v
  synt_spec
    const_frame:  v_np_np
    adj:          [nil]
  sem_spec
    func_to_role
      sub:         agent
      obj:         theme
      adj:         locus
      agl:         none
      aspect:     nil

```

Jeg vil ikke komme yderligere ind på rollerne her. På nuværende tidspunkt i projektet fungerer denne opfattelse af roller kombineret med valensteorien udmærket når det gælder om at producere en semantisk repræsentation af de simple sætninger vi arbejder med. Når vi skal i gang med at etablere selve vidensbasen kan det imidlertid meget vel vise sig at denne semantiske repræsentation er utilstrækkelig, og at fx. antallet af roller skal suppleres.

5 Opbygningen af den semantiske repræsentation

De oplysninger der ligger i leksikon som vist ovenfor indtastes manuelt ved hjælp af en særlig editor, DICTED, udviklet af Henrik Kersting (under udgivelse) i forbindelse med FAGFLADE projektet. De danner udgangspunkt for en procedure som genererer et nyt leksikon i form af en prolog database, hvor lexemnavnet associeres med en semantisk repræsentation i form af et lambdaudtryk beregnet på baggrund af argumenterne i den sammensatte term `func(tion)_to_role`. Det skal her indskydes, at kun verber og adjektiver, herunder også participier, på nuværende tidspunkt forekommer med en specifik semantisk repræsentation. Alle andre ordklasser, dvs. determinativer, præpositioner og substantiver, har det pågældende leksemnavn som semantisk repræsentation. Sammensatte NP'er har

kerneleddets leksemnavn som semantisk repræsentation og PP'er har styrelsens kerne som semantisk repræsentation. Udtrykkene *aktie*, *aktierne*, *disse aktier* og *af disse aktier* har således alle en og samme repræsentation, nemlig *aktie*.

Argumenterne i termen `func_to_role` er de fire syntaktiske funktioner i ordnet rækkefølge: `subj`, `obj`, `adj`, `agl`, afbildet på de semantiske roller `agent`, `tema` og `locus`. Alle fire funktioner optræder, også i de tilfælde hvor de rent faktisk ikke kan forekomme og derfor som værdi har 'none' som vi så det ovenfor. De anvendes ved konstruktionen af lambdaudtrykket, hvis argumenter ligeledes forekommer i kanonisk rækkefølge: `agent(_)`, `theme(_)`, `locus(_)`.

Verbet 'påføre' har således flg. `func_to_role` repræsentation:

```
func_to_role( sub(agent), obj(theme), adj(locus), agl(none))
```

som konverteres til flg. lambdaudtryk:

```
X^Y^Z^påføre(agent(Z), theme(Y), locus(X) )
```

Dette udtryk beregnes af flg. regel:

```
% Verb + adjunct + object (V NP NP).
% Verb + object + adjunct (V NP PP & V NP AP).

build_LambdaExpr(Lexeme,
  func_to_role(sub(RoleS),obj(RoleO),adj(RoleAd),agl(none)),
  X^Y^Z^Expr) :-
    RoleS \== none,
    RoleO \== none,
    RoleAd \== none,
    RoleSTerm =.. [RoleS,Z],
    RoleOTerm =.. [RoleO,Y],
    RoleAdTerm =.. [RoleAd,X],
    sort_RoleTerms([RoleSTerm,RoleOTerm,
                    RoleAdTerm], SortedRT),
    Expr =.. [Lexeme|SortedRT].
```

Når parseren skal bygge en sætnings semantiske struktur har den da for verbernes vedkommende den type informationer til sin rådighed som vist på flg. side.

Kolonnen med funktioner og roller kan læses på to måder: dels som udtryk for en afbildning af syntaktiske funktioner på roller, dels kan man for hvert verbum aflæse en funktionsramme og en rolleramme ved alene at læse funktioner og roller. Rammerne udtrykker tilladte kontekster for et verbum i form af konstituenten, funktioner eller roller. Strukturerne, den syntaktiske og semantiske, bygges af parseren ved at relatere ordene i den konkrete sætning til konstituenttyper, roller eller funktioner.

	Konstituentramme	Func-to-role mapping	Prolog lambda udtryk
VERBER			
anser	v_np_pp	sub(agent) obj(theme) adj(locus)	$X^{\sim}Y^{\sim}Z^{\sim}\text{anse}(\text{agent}(Z), \text{theme}(Y), \text{locus}(X))$
anses	v_pp_pp	sub(theme) adj(locus) agl(agent)	$X^{\sim}Y^{\sim}Z^{\sim}\text{anse}(\text{agent}(Y), \text{theme}(Z), \text{locus}(X))$
bestemmer	v_np	sub(agent) obj(theme)	$Y^{\sim}Z^{\sim}\text{bestemme}(\text{agent}(Z), \text{theme}(Y))$
bortfalder	v	sub(theme)	$Z^{\sim}\text{bortfalde}(\text{theme}(Z))$
ejer	v_np	sub(locus) obj(theme)	$Y^{\sim}Z^{\sim}\text{eje}(\text{theme}(Y), \text{locus}(Z))$
ejes	v_pp	sub(theme) agl(locus)	$Y^{\sim}Z^{\sim}\text{eje}(\text{theme}(Z), \text{locus}(Y))$
er	v_ap	sub(theme) adj(locus)	$Y^{\sim}Z^{\sim}\text{v\ae}re(\text{theme}(Z), \text{locus}(Y))$
er	v_np	sub(theme) adj(locus)	$Y^{\sim}Z^{\sim}\text{v\ae}re(\text{theme}(Z), \text{locus}(Y))$
finder	v_np_ap	sub(agent) obj(theme) adj(locus)	$X^{\sim}Y^{\sim}Z^{\sim}\text{finde}(\text{agent}(Z), \text{theme}(Y), \text{locus}(X))$
følger	v_pp	sub(theme) adj(locus)	$Y^{\sim}Z^{\sim}\text{f\o}lge(\text{theme}(Z), \text{locus}(Y))$
påfører	v_np_np	sub(agent) obj(theme) adj(locus)	$X^{\sim}Y^{\sim}Z^{\sim}\text{p\aa}f\o}re(\text{agent}(Z), \text{theme}(Y), \text{locus}(X))$
undtager	v_np_pp	sub(agent) obj(theme) adj(locus)	$X^{\sim}Y^{\sim}Z^{\sim}\text{undtage}(\text{agent}(Z), \text{theme}(Y), \text{locus}(X))$

Under parsningen af sætningen „Denne aftale påfører selskabet en forpligtelse“ vil parseren—når den skal bygge den semantiske struktur for sætningen—starte med at identificere *denne aftale* som et NP der opbevares i en variabel. I leksikon under *påfører* vil den få oplysninger om den konstituentramme der gælder for verbet samt det lambdaudtryk der svarer til verbet. Ud fra en matching af konstituentrammen finder den frem til den regel som skal anvendes og identificerer herefter en konstituentstruktur med to NP'er efter verbet. Det første NP, *selskabet*, associeres med rollen locus som er tildelt adjektet, det andet NP, *en forpligtelse*, med rollen tema som er tildelt objektet. Til sidst vil den udfylde rollen agent med subjektet, det første NP den læste, og aflevere det færdige resultat:

$\text{p\aa}f\o}re(\text{agent}(\text{aftale}), \text{theme}(\text{forpligtelse}), \text{locus}(\text{selskab}))$

Den semantiske repræsentation vi kan få frem nu er baseret på rollestrukturen fordi vi anser den for et centralt element når den information sætningen indeholder skal fastholdes som vidensrepræsentation i vidensbasen. Vi er godt klar over at der er flere forskellige semantiske fænomener som ikke er repræsenteret i den nuværende fase, tidsrelationer, modalitet eller fx. adverbialsemantik. Vi har også en liste over punkter som skal løses og andre punkter som forestår. Det

er også derfor jeg har kaldt indlægget „På vej mod en fagsproglig tekstfortolker“, og forhåbentlig kan vi når vi mødes igen om to år præsentere en yderligere udbygning og forbedring af vores tekstfortolker.

Hvis der er nogen der er interesseret i en mere udførlig rapport om projektet samt en detaljeret gennemgang af programmet, så kan man orientere sig nærmere i LAMBDA NR. 11.

Litteratur

- Hansen, Steffen Leo (forthcoming), FAGFLADE: Text, Types and Tokens, *Lambda*.
- Hansen, Steffen Leo and Carl Vikner. 1989. FAGFLADE: The Initial Phase of a Project in Natural Language Interpretation. *Lambda*, 11. Institut for Datalingvistik, HHk.
- Herslund, Michael. 1988a. On Valence and Grammatical Relations. *Copenhagen Studies on Language*, 11:3–35, Copenhagen.
- Herslund, Michael. 1988b. *Le datif en français*. Louvain-Paris, Editions Peeters.
- Herslund, Michael and Finn Sørensen. 1985. *De franske verber. En valensgrammatisk fremstilling. I. Verbernes syntaks*. Romansk Institut, Københavns Universitet.
- Kersting, Henrik. 1989. DICTED: An Editor for Dictionaries Stored as Prolog Databases. *Lambda*, 12.
- Korzen, Hanne, Henning Nølke, Henrik Prebensen and Finn Sørensen. 1983. PC-Grammar: An Alternative?, *Acta Linguistica Hafniensia*, 18:5–53, Copenhagen.
- Sørensen, Finn. 1988. Om rollen Locus, Notat. Institut for Datalingvistik, Handelshøjskolen i København.

Institut for Datalingvistik
Dalgas Have 15
DK-2000 København F.
Danmark