Tekstanalyse per computer

Renkema, J.; Kempff, H.; van Opstal, T.

Published in:
Gramma

Publication date:
1987

TEKSTANALYSE PER COMPUTER; HET LEXICON

H. Kempff, T. van Opstal & J. Renkema

0. Inleiding

In de taal- en tekstwetenschap kost het beantwoorden van vragen waarvoor grote tekstbestanden moeten worden geanalyseerd, veel tijd en geld omdat het onderzoekmateriaal met de hand bewerkt moet worden. Wanneer men een bepaalde tekstsoort wil onderzoeken, bijvoorbeeld opstellen uit verschillende leerjaren wil vergelijken, dan dient men te beschikken over gegevens die gebaseerd zijn op tekstbestanden van een behoorlijke omvang. Hetzelfde geldt uiteraard voor vragen op terreinen als 'begrijpelijkheidsonderzoek' en 'taalverwerving van etnische minderheden', en voor het onderzoek naar de frequentie van bepaalde grammaticale constructies.

Om het onderzoek naar deze en soortgelijke vragen te vergemakkelijken is door de afdeling toegepaste taalkunde in Eindhoven en de letterenfaculteit in Tilburg een project gestart onder de titel 'Halfautomatische Tekstanalyse'. Doel van dit project is het ontwikkelen van programmatuur die aan woorden in teksten de juiste woordsoort toekent. Om optimale bereikbaarheid voor toekomstige gebruikers te garanderen is de programmatuur behalve op (VAX) mainframe ook beschikbaar op een personal computer<1>.

Dit project richt zich op woordbenoeming. Op het gebied van automatische tekstanalyse wordt ook veel aandacht besteed aan morfologische, syntactische en semantische verschijnselen. Waarom dan toch een project dat alleen beoogt de woordbenoeming te geven? De belangrijkste reden is dat andere vormen van tekstanalyse in ernstige mate worden bemoeilijkt, wanneer de lexicale status van een woord niet bekend is. Wanneer men bijvoorbeeld onderzoek wil doen naar werkwoordsvormen in een tekst, dient eerst vast te staan welke tekstwoorden tot de categorie werkwoord behoren. Een dergelijk onderzoek zou over de informatie moeten kunnen beschikken welke voorkomens van een woordvorm als *werken* verbaal (*ze werken*) zijn, en welke nominaal (*de werken*). Bij pogingen tot automatische zinsontleding blijkt dat het aantal mogelijke zinsanalyses sterk toeneemt wanneer er geen sluitende informatie aanwezig is over de woordsoorten.

GRAMMA 11 (1987), 3, p.169-190

Automatische woordbenoeming bestaat uit twee stappen. In de eerste plaats moeten woorden worden opgezocht in een lexicon. Inspectie van het lexicon moet resulteren in de toekenning van de volledige woordsoortinformatie. Dit houdt in dat in veel gevallen meer dan één woordsoortcode toegekend moet worden. Een woord als *verbonden* moet behalve als zelfstandig naamwoord (meervoud) ook herkend worden als voltooid deelwoord en onvoltooid verleden tijd (meervoud) van het werkwoord *verbinden*. In tweede instantie moet beslist worden welke van de in het lexicon genoemde mogelijke woordsoorten in een bepaalde context de juiste is. In dit artikel wordt verslag gedaan van de resultaten van de eerste helft van dit onderzoeksproject: de opbouw van een lexicon en de ontwikkeling van programmatuur die op basis van dit lexicon alle relevante woordsoorten toekent aan woorden.

De opzet van het project is in grote lijnen bepaald door de resultaten van een voor de start van het eigenlijke project uitgevoerd literatuur-onderzoek (Renkema e.a., 1984). In (1) zal ingegaan worden op de relatie van dit project met een aantal soortgelijke projecten. Aan de hand daarvan zal de opzet van het project worden besproken. In (2) wordt informatie gegeven over het gekozen systeem voor woordbenoeming. In (3) - de kern van dit artikel - wordt uiteengezet hoe het lexicon is opgebouwd, met een illustratie aan de hand van de analyse van een voorbeeldzin. Ook wordt ingegaan op de oplossing van enige probleemgevallen. Een evaluatie van de tot nu toe bereikte resultaten en een bespreking van de toekomstige werkzaamheden volgen in (4).

1. Plaats van het project

In het vervolg van de tekst worden de termen *woordsoorttoekenning* en *disambiguering* gebruikt. Onder *woordsoorttoekenning* wordt verstaan het toekennen aan een woord van alle in isolatie van toepassing zijnde woordsoorten. Deze woordsoorten worden gevonden door het woord op te zoeken in het lexicon. Onder *disambiguering* wordt verstaan het selecteren van de in de context juiste woordsoort.
Uit het vooronderzoek bleken, mede gezien de keuze voor een micro-computer toepassing, twee projecten voor Hata van belang te zijn, namelijk een in Nijmegen ontwikkeld programma 'Lexical 1' (zie van Bakel e.a., 1977) en het zogenaamde LOB-project (zie Leech & Garside, 1982). De opzet van het Hata-lexicon en de nadruk op morfologische woordeindeanalyse is geïnspireerd door 'Lexical 1', terwijl de organisatie van het Hata-programma in twee fasen (woordsoorttoekenning gevolgd door een disambigueringsfase) sterke overeenkomst vertoont met het LOB-project. Het Nijmeegse programma, vanwege het voorlopige karakter 'Lexical 1' genoemd, kent op grond van morfologische beslissingen woordsoorten toe aan woorden in een tekst, en trekt zo mogelijk conclusies uit de morfologie van die woorden. Deze conclusies worden opgeslagen in een lexicon, dat tijdens de analyse wordt opgebouwd. Ter verduidelijking een voorbeeld. Bij de analyse van het woord *eetbaarheid* wordt op grond van het suffix '-heid' besloten tot de woordbenoeming zelfstandig naamwoord, en het suffix '-baar' leidt tot de conclusie dat *eet* een

werkwoordstam is. Deze laatste informatie (de conclusie) wordt opgeslagen in een dynamisch, veranderbaar lexicon, zodat bijvoorbeeld een later voorkomen van de vorm *eten* herkenbaar is als werkwoord.
Zoals de auteurs zelf al opmerken, is het programma 'nog voor duchtige verbeteringen vatbaar': van de 1384 aangeboden testwoorden werd in de eerste testrun 72.8% gekarakteriseerd, en in de tweede, met gebruikmaking van de conclusies uit de eerste testrun, 73.8% waarvan 62.6% helemaal goed. Met 'helemaal goed' wordt bedoeld dat karakterisering van het woord volledig is: karakterisering van een woord als *fiets* als 'zelfstandig naamwoord' wordt als onvolledig gekenmerkt. Pas als ook de karakterisering 'werkwoordstam' aanwezig is, is de benoeming helemaal goed.
Bij het ontwerpen van Lexical 1 heeft men de volgende twee uitgangspunten gehanteerd: 1. een oppervlakkige morfologische analyse is voldoende voor de toekenning van de juiste woordsoort; 2. een lexicon kan gaandeweg de tekstverwerking worden opgebouwd op grond van de regelmatigheden in de morfologie. Het eerste uitgangspunt bleek niet juist. Er is meer dan een oppervlakkige morfologische analyse vereist voor woordbenoeming. Hier slechts enkele voorbeelden. Als het suffix '-ste' leidt tot de benoeming adjectief, dan behoort op grond van die regel ook *vereiste* tot deze woordsoort. Als het suffix '-baar' aangeeft dat het woord een werkwoordstam bevat, zoals in *eetbaar*, dan bevat ook *onmiskenbaar* de werkwoordstam *onmisken*. Uiteraard kan men dan een regel opstellen waardoor pas na afkapping van 'on-' tot de benoeming werkwoordstam wordt besloten, maar een woord als *misbaar* wordt daarmee nog niet als adjectief èn nomen gekarakteriseerd.
Het uitgangspunt dat woordbenoeming zou kunnen plaatsvinden aan de hand van morfologische regelmatigheden is ook de oorzaak van veel andere problemen. Zo zal zonder specifieke informatie over de stammen *brand* en *ban* en de vorming van voltooide deelwoorden, de benoeming van *verbrand, verband* (geen voltooid deelwoord) en *verbannen* altijd problematisch blijven. Ook het tweede uitgangspunt (het dynamische lexicon) is aanvechtbaar. Ten eerste zijn onjuiste conclusies mogelijk, zoals bij *vereiste* ('verei' wordt als adjectief in het lexicon opgenomen vanwege de veronderstelde flexie '-ste'). Ten tweede zal de vulling van het lexicon afhankelijk blijven van het toevallig voorkomen van (mogelijkerwijze infrequente) vormen waaruit staminformatie kan worden geconcludeerd.
Op grond van de ervaringen met Lexical 1 is in het Hata-project extra aandacht besteed aan gedetailleerde morfologische (woordeinde) analyse, en aan de bouw van een adequaat lexicon.

De werkwijze van het Hata-project - een fase waarin de woorden van woordsoortcodes worden voorzien, gevolgd door een fase waarin de in de context juiste woordsoortcode wordt geselecteerd - is overgenomen uit het LOB-project. Dit project heeft tot doel een corpus Engelse teksten, dat door onderzoekers in Lancaster, Oslo en Bergen is verzameld, van woordsoortcodes te voorzien. Dit onderzoek bouwt voort op het Brown-Taggit-project (in de LOB-terminologie is een *tag* een woordsoortcode) waarin aan de woorden in het al oudere Brown-corpus woordsoortcoderingen zijn toegekend. In het Brown-Taggit-project zijn de woorden benoemd op grond van een lexicon en een

suffixenlijst. Deze werkwijze bleek tot veel fouten in de woordbenoeming te leiden; deze moesten alle met de hand worden hersteld. Het LOB-project maakt gebruik van dit voor een deel met de hand gecodeerde corpus. Codes voor woordbenoeming worden toegekend door de woordenlijst, waarin de woordsoortcodes van het Brown-corpus zijn toegevoegd, 'los te laten' op het LOB-corpus. Op deze manier kon een zeer groot deel van de woorden van één of meer woordsoortcodes worden voorzien. Vaak moet aan een woord meer dan één woordsoortcode worden toegekend: 'stone' is niet alleen nomen, maar kan ook verbum of adjectief zijn. De uitkomst van deze woordbenoemingsfase levert dus zinnen op waarin aan woorden - soms meerdere - woordsoortcodes zijn toegekend. Bijvoorbeeld:

mijn	*rechter*	*hand*
poss	adj	nomen
nomen	nomen	

Om te bepalen welke woordsoortcodering juist is, ondergaat de gecodeerde tekst een tweede bewerking, de disambiguering. In deze fase maakt men gebruik van zogenaamde 'context frame rules' die de meest waarschijnlijke woordsoortcode op basis van de context selecteren. Deze contextregels zijn afgeleid uit de al gecodeerde zinnen in het Brown-corpus. Voor het hier gegeven voorbeeld zijn dat bijvoorbeeld regels waarin de waarschijnlijkheid van de opeenvolging van poss - nomen en nomen - nomen wordt gegeven. Op grond van deze regels kan dan aan *mijn* de woordsoortcode possessivum en aan *rechter* de code adjectief toegekend worden.

2. Het Codeersysteem

In dit project vormt het Eindhovense corpus de belangrijkste hulpbron voor het inventariseren van mogelijke woordbenoemingen en het ontwikkelen van contextregels. Het lag daarom voor de hand om in het op te bouwen lexicon items te voorzien van woordsoortcodes zoals die zijn onderscheiden in het codeersysteem dat gehanteerd is voor het Eindhovense corpus. Met dit codeersysteem wordt aan elk woord een cijfercode toegekend waarmee de grammaticale functie en de relevante kenmerken worden gespecificeerd. Dit codeersysteem waarin codes van drie cijfers<2> worden gebruikt, staat bekend als het C3C-systeem. In principe kan het systeem duizend verschillende woordcoderingen onderscheiden. In de praktijk komen er ongeveer tweehonderd voor, verdeeld over de tien traditioneel onderscheiden woordsoorten. Met deze codes kunnen de woorden zeer precies worden benoemd. Het woordje *waar* bijvoorbeeld kent acht mogelijke woordbenoemingen in C3C:

000 (zelfstandig naamwoord: *waar voor je geld krijgen*)
100 (gewoon adjectief: *het is me een waar genoegen*)
241 (persoonsvorm: *ik waar niet rond, zei het spook*)
520 (vragend bijwoord: *waar ben je?*)
530 (betrekkelijk bijwoord: *de plaats waar ik woon*)
550 (vragend voornaamwoordelijk bijwoord: *waar ga je heen*)
560 (betrekkelijk voornaamwoordelijk bijwoord: *een plaats waar je heen kunt*)
710 (onderschikkend voegwoord: *Waar wij ons best doen, moet jij er niet je gemak van nemen*)

Ondanks mogelijke kritiek op onderscheidingen binnen het C3Csysteem (zie Renkema, 1981) is toch voor dit systeem gekozen. De redenen hiervoor zijn de volgende. Er is geen alternatief voorhanden en in de Nederlandse corpuslinguistiek is C3C een standaard. Bovendien moeten de resultaten van het woordbenoemingsprogramma getest worden met behulp van teksten. Het samenstellen en coderen van een test-corpus zou veel tijd en energie in beslag nemen. De teksten en de frequentielijsten in Uit den Boogaart (1975) vormen uitstekend testmateriaal om resultaten van automatische woordbenoeming te evalueren.
Het C3C-systeem is door Hata niet ongewijzigd overgenomen. De belangrijkste reden hiervoor is dat onderscheidingen die door menselijke codeerders gemaakt kunnen worden, niet altijd in een algoritme kunnen worden opgenomen. Het Hata-codeersysteem en C3C verschillen daardoor op een aantal punten. De meeste verschillen zijn klein en van ondergeschikt belang. Toch zijn er een paar verschillen die consequenties hebben voor de gebruiksmogelijkheden van teksten waarvan de woorden door Hata zijn benoemd. Daarom zal hier kort worden ingegaan op deze verschillen. De veranderingen betreffen enkele inperkingen en één belangrijke uitbreiding.
De codeerders van het Eindhovense Corpus bleken in staat aan de hand van enkele simpele instructies bijna foutloos te beslissen over transitief of intransitief gebruik van werkwoordsvormen. Ook de herkenning van eigennamen is - afgezien van enkele details - bij handcodering niet problematisch. Voor een woordsoorttoekenningsalgoritme is het nemen van dit soort beslissingen niet weggelegd. De veelal semantische informatie die nodig is om dit soort beslissingen te nemen, kan niet in een algoritme worden opgenomen.
Uitbreiding van het codeersysteem heeft plaatsgevonden met betrekking tot de C3C-categorie 5.., de bijwoorden. In het bijzonder de coderingen 50., 51., 52. en 53. zijn nader bekeken<3>.

Binnen C3C gelden de volgende subcategorieën:

50. gewone bijwoorden (*heel 500* groot, de kachel is *uit 500*, *dicht 500* bij huis, twee jaar *geleden 500*). Verdere onderscheidingen worden gehanteerd voor 'overige verbogen vormen' (ten *enenmale 503*), 'comparatieven onverbogen' (*vaker 504*), 'superlatieven onverbogen' (het *dichtst 507* bij) en 'overige vormen' (ten *zeerste 509*).
51. aanwijzende en onbepaalde bijwoorden (*er 510, zodoende 510, indertijd 510, uiteraard 510*)
52. vragende bijwoorden (*wanneer 520, waarom 520*, in *hoeverre 520*)
53. betrekkelijke bijwoorden (de stad *waar 530*, de reden *waarom 530*, in het jaar *dat 530* de oorlog uitbrak)

Het belangrijkste argument voor verfijning van het systeem is de diversiteit binnen met name de categorie 500. Deze klasse is in C3C negatief gedefinieerd: woorden die niet tot een van de andere categorieën onder 5.. behoren, worden benoemd als 'gewoon bijwoord'. Het gevolg is dat veel woorden die een duidelijk verschil in distributie en/of functie hebben, deze benoeming toegekend krijgen. De disambiguering van reeksen waarin bijwoorden voorkomen zal

hierdoor beïnvloed worden, juist omdat de distributie en functie van een aantal bijwoorden wel degelijk verschillen vertoont. Zo is de distributie van het bijwoord *uit* (de kachel is uit) duidelijk anders dan die van *heel* (heel groot). Bovendien kan een specifiekere bijwoordcodering van nut zijn bij de disambiguering van anderszins onbeslisbare gevallen, bijvoorbeeld in *.. was ik gisteren* waar de disambiguering van *was* (nomen, v-imperfectum, v-praesens) vereenvoudigd wordt door de benoeming van *gisteren* als bijwoord van (verleden) tijd. Het Hata-codeersysteem onderscheidt bijwoorden van *plaats, tijd, modaliteit* etc. Al deze subcategorieën worden op hun beurt weer verder onderverdeeld. De onderverdeling van de bijwoorden van *plaats* bijvoorbeeld kent twee hoofdgroepen: *locale* en *aanduidende* bijwoorden. De laaste categorie vertakt zich als volgt: *naar* en *vanaf*. Deze vertakkingen kennen op hun beurt weer subcategoriseringen als *hier* en *daar*. Zo is het bijwoord *daarheen* geklassificeerd als *plaats*bijwoord van *richting, naar, daar*. Het bijwoord *heen* daarentegen is benoemd als *plaats*bijwoord van *richting*, zonder verdere specificaties. Onafhankelijk van bovengenoemde specificeringen is binnen het codeersysteem plaats ingeruimd om informatie te specificeren over het *aanwijzend, vragend* of *relatief* karakter van het bijwoord.

3. Het Lexicon

De doelstelling van het Hata-project houdt in dat het lexicon aan de volgende twee eisen moet voldoen. Het moet aan elk Nederlands woord in al zijn verschijningsvormen de mogelijke woordsoorten toekennen. Het moet relatief klein zijn opdat de programmatuur in een micro-configuratie gebruikt kan worden, zodat de bereikbaarheid van het systeem maximaal is.
Uiteraard kan men als lexicon een woord(vorm)enlijst nemen waarin ook verbogen en vervoegde vormen opgenomen zijn. Maar zo'n lijst voldoet niet aan de eisen. Het is niet mogelijk om alle woorden op te sommen, omdat er steeds nieuwe woorden en vooral nieuwe samenstellingen ontstaan. Zo'n woordenlijst is per definitie niet volledig en zou ook te veel ruimte in beslag nemen. Wel kan men aan de tweede eis voldoen door een selectie te maken uit een woordenboek of een frequentielijst, maar dan kan niet meer élk Nederlands woord worden benoemd. Bovendien is het niet duidelijk op basis van welke criteria een woord in zo'n lexicon opgenomen moet worden. Frequentie is in ieder geval geen goed criterium. Immers, in een tekst komen ook tekstspecifieke woorden voor. Deze zullen problemen geven bij een lexicon dat is gecompileerd op basis van andere teksten.
Voor het Hata-project is een lexicon ontworpen dat een omvang heeft van ongeveer 7600 items. Een dergelijk lexicon kan in een micro-configuratie nog gemakkelijk gehanteerd worden. Met dit lexicon kunnen aan elk woord in een tekst de mogelijke woordsoorten worden toegekend. Het zal duidelijk zijn dat dit lexicon weinig overeenkomst vertoont met de traditionele lexica. Het Hata-lexicon is gebaseerd op het enig mogelijke criterium: volledigheid. In het vervolg zal deze claim uitvoerig worden toegelicht. In deze paragraaf zal uiteengezet worden uit welke onderdelen het lexicon bestaat (3.1), hoe het is opgebouwd (3.2) en hoe de woordsoort-

toekenning met behulp van dit lexicon verloopt (3.3). In (3.4) wordt de werking van het lexicon geïllustreerd aan de hand van de analyse van een voorbeeldzin.
Het totaal van de verschillende onderdelen wordt aangeduid met de term *lexicon*, de onderdelen heten *componenten* en vertonen oppervlakkige overeenkomst met een woordenboek. De componenten bestaan uit *items*, die op hun beurt te vergelijken zijn met de woorden in een woordenboek. Een belangrijk verschil tussen woordenboekwoorden en Hata-items is dat het formaat waarin ze zijn opgenomen verschilt. De items zijn ontdaan van elke vorm van flexie: een woordenboekwoord als 'lopen' wordt opgenomen als 'loop', zonder de '-en'.

3.1 Componenten

De Nederlandse woordvoorraad kan verdeeld worden in woorden mèt en zonder mogelijke flexie. Woorden zonder mogelijke flexie zijn invariant en kunnen worden opgesomd. Een lijst die al deze woorden bevat met hun woordsoortcode is voldoende om elk voorkomen van die woorden in een tekst te benoemen. Woorden met mogelijke flexie moeten eerst tot een lexiconformaat worden teruggebracht.

Het lexicon bestaat uit vier componenten. De eerste component bevat de invariante woorden; in de andere drie componenten worden woorden met flexie verantwoord. De vier componenten zijn: 1. een lijst woorden zonder flexie, 2. een reeks instructies om geflecteerde vormen te lemmatiseren, 3. een reeks filters voor uitzonderingen, en 4. een lijst basisvormen.

1. Component met woorden zonder flexie.
Tot de categorie van woorden zonder flexie behoren de woorden uit de zogenaamde gesloten woordklassen: pronomina, demonstrativa, prepositities, conjunctiva, bijwoorden enz. Deze woorden zijn op basis van Uit den Boogaart (1975) eenvoudig op te sommen. De woorden met flexie behoren tot de zogenaamde open woordklassen, overwegend nomina, adjectiva en verba; deze woorden zijn niet opsombaar.
De groep woorden zonder flexie is in zijn geheel opgenomen in de eerste component. Het gaat hier om ongeveer 1200, meestal zeer frequente, woorden. In het algemeen is de woordsoorttoekenning van deze invariante woorden eenvoudig. Als een woord hier wordt gevonden is de woordsoorttoekenning volledig. Meestal is dan één woordsoort aan het woord toegekend. Een aantal woorden in deze lijst heeft echter meer dan één woordsoortcode: *een* is lidwoord en telwoord. Ook bevat deze lijst woorden, die in andere componenten terugkomen: *vier* is in deze component slechts als telwoord en *pas* alleen als bijwoord opgenomen. Dit soort woorden is gemarkeerd voor het feit dat de benoeming incompleet is (*vier* is ook verbum en *pas* ook nomen en verbum); ze worden 'doorgestuurd' naar de andere componenten voor verdere benoeming.

2. Component met lemmatiseerinstructies.
Wanneer men een geflecteerd woord wil terugbrengen tot de basisvorm dan moet men altijd één of meer van de volgende letters van het woordeinde afhalen: *d, r, e, n, s* of *t*.
De lemmatiseerinstructies specificeren voor specifieke woordeinden welke letters of combinaties van letters achtereenvolgens verwijderd moeten worden<4>, of bijvoorbeeld consonant-reductie (*knoppen - knop*), vocaal-verlenging (*knopen - knoop*) of een andere aanpassing (*graven - graaf*; *huizen - huis*) nodig is, en welke component voor de zo ontstane basisvorm geraadpleegd moet worden. Bovendien vermelden de lemmatiseerinstructies ook welke conclusies over de woordsoort getrokken moeten worden als bepaalde eindletters zijn verwijderd. Voor het woord *denkt* bestaat er bijvoorbeeld een instructie die de 't' verwijdert; in een andere component wordt de vorm 'denk' herkend als een werkwoord; vanwege de verwijderde 't' wordt nu geconcludeerd dat het een werkwoordsvorm betreft: de derde persoon praesens singularis.
Deze component bevat ongeveer 150 instructies. Iedere instructie heeft betrekking op woorden met dezelfde eindletters en bestaat uit meerdere delen omdat herleiding tot woordenboekformaat meestal niet eenduidig is. Indien bijvoorbeeld een woord eindigt op '-ste', (*vereiste, rotste, botste*) dan moet worden ondezocht welk gedeelte van dit woordeinde flectief is, en welk gedeelte deel uitmaakt van de basisvorm. De lemmatiseerinstructie voor '-ste' beregelt dit door achtereenvolgens de verschillende mogelijke basisvormen aan te bieden aan de filtercomponent en/of de basisvormencomponent.
Vereiste moet als niet-geflecteerd nomen, als verbogen voltooid deelwoord (*vereis-t-e*) en als imperfectum (*vereis-te*) worden herkend; *rotste* als verbogen superlatief (*rot-ste*); *botste* als verbogen superlatief (*bot-st-e*) en als imperfectum (*bots-te*).

3. Filtercomponent.
Het grote probleem bij lemmatisering is dat de flexieletters niet altijd flexie impliceren. Bij woorden als *regen, molen, leugen,* krijgt men geen basisvorm door '-en' af te kappen, bij *handen* wel. Voor deze uitzonderingen zijn een aantal filters geconstrueerd. Omdat flexies altijd zijn opgebouwd uit bepaalde letters of lettercombinaties ('-d', '-er', '-e', '-en', '-s', '-t') kunnen zes filters, één voor elke uitgang, elke mogelijke pseudoflexie afvangen. *Regen, molen* en *leugen* zijn woorden die in zo'n filter worden benoemd. Regelmatige woorden, zoals *klank-en* en *bord-en*, passeren de filters, en worden elders benoemd.
Deze zes filters, waarin aan uitzonderingen op regelmatige flexie woordsoorten worden toegekend, bevatten in totaal ruim 3200 items. Naar verhouding is deze component dus vrij omvangrijk: ongeveer 40 procent van het totale lexicon.

4. Basisvormencomponent.
Wanneer een woord in aanmerking komt om gelemmatiseerd te worden en als niet door een van de filters de woordsoorttoekenning wordt voltooid, krijgt het in de component basisvormen één of meer woordsoortcodes toegekend. Bij het opstellen van deze component is uitgegaan van het feit dat woordeinden vaak informatie geven over de woordsoort. Woorden op '-ing' bijvoorbeeld zijn bijna allemaal

nomina. Men behoeft dus niet elk woord dat eindigt op '-ing' op te nemen met de toevoeging 'n(omen)'. Wanneer men alleen '-ing' opneemt en de uitzonderingen verantwoordt, kan veel ruimte bespaard worden. Tot de uitzonderingen behoort onder andere *zing* dat de toevoeging 'v(erbum)' krijgt, en een uitzondering op een uitzondering is hier *lezing* dat wel een nomen is. Op basis van het retrograde-woordenboek (Nieuwborg, 1978) konden veel van dergelijke regelmatigheden met hun uitzonderingen worden opgespoord. De strategie maakte het mogelijk de omvang van de basisvormencomponent te beperken tot ongeveer 3200 items. In de volgende subparagraaf zal in detail aandacht worden besteed aan een gedeelte van de basisvormencomponent.

3.2 Opbouw

De lexiconcomponenten zijn opgebouwd volgens de filosofie dat woordsoorttoekenning plaats kan vinden volgens regels. Dit betekent dat alle uitzonderingen op de regels moeten worden verantwoord. Soms gebeurt dit in de lemmatiseringscomponent maar meestal in een van de andere drie componenten. Aan de hand van een stukje uit de basisvormencomponent zal deze opbouw worden verduidelijkt.
Het principe dat woordeinden, op een aantal uitzonderingen na, de lexicale categorie van een woord bepalen, is tot in zijn uiterste consequentie doorgevoerd. Door een uitputtende inventarisatie te maken van woordeinden van Nederlandse woorden, zijn niet alleen voor de hand liggende regelmatigheden gevonden, zoals de regel dat woorden met suffixen als '-heid' en '-ing' nomina zijn, maar ook bijvoorbeeld dat woorden op '-g' voor het merendeel nomina zijn. Deze regelmatigheden maken het mogelijk met behulp van ongeveer 3200 woordeinden alle woorden te verantwoorden waarvan de eventuele flexieuitgangen zijn verwijderd. Misschien ten overvloede wordt hier nog opgemerkt dat de basisvormencomponent in principe geen woordeinden bevat die eindigen op een van de letters 'd', 'r', 'e', 'n', 's' of 't'; deze letters signaleren immers flexie of pseudoflexie. Een voorbeeld van een deel van deze component wordt hieronder gegeven. De zoekvolgorde is aangegeven door de oplopende cijfers. De organisatie van deze component en het zoekproces is strikt alfabetisch. Een woord wordt gevonden in een component als, afgezien van een aantal nog te bespreken restricties, het woordeinde met een item overeenkomt. Het zoekproces wordt beëindigd als een woord wordt aangetroffen dat alfabetisch groter is dan het woord waarmee gezocht wordt. De consequentie van deze wijze van zoeken is, dat bij een woord vaak meerdere items worden gevonden. Alleen het langste item blijft bewaard en dat item bepaalt de woordsoorttoekenning.

1	g N	7	#heug N V
2	schurg V	8	!plug N
3	wurg V	9	vlug A
4	-urg N	10	#overbrug V
5	deug V	11	stug A
6	!verheug	12	!spuug N

Lexiconinspectie voor een woord als *meug* resulteert in de toekenning van de grammaticale categorie n(omen) op basis van het eerste item "g N". Er wordt namelijk geen langer item gevonden voor het woord *meug*. Zo zullen aan het woord *heug* op basis van het item "#heug N V" de woordsoorten n(omen) en v(erbum) toegekend worden. De bedoeling van de tekens "#", "-" en "!" is om beperkingen aan te kunnen brengen in de groep woorden die door zo'n item benoemd kunnen worden, en om conclusies te kunnen trekken over de grammaticale categorieën waartoe een woord behoort als het overeenkomt met een item. Zo is de betekenis van "#" dat een woord om met een item overeen te stemmen exact hieraan gelijk moet zijn. Daarentegen betekent "-" dat aan een woord alleen via dit item woordsoorten kunnen worden toegekend als het over de lengte van het item identiek is met het item en bovendien langer dan het item. Alle woorden op "urg" behalve *schurg* en *wurg* zijn dus n(omina). De betekenis van "!" is gecompliceerder. Dit teken legt niet zozeer beperkingen op aan het woord dat tot aan het teken "!" overeenkomt met het item, maar zorgt voor verschillende woordsoorttoekenningen afhankelijk van het eerste deel van het woord (dus dat deel dat niet overeenkomt met het item). Wordt het door het item gedekte deel van het woord voorafgegaan door een werkwoordelijk prefix (voornamelijk voorzetsels als *in* en *over*) dan wordt de woordsoort verbum toegekend. Is dit voorste deel van het woord niet zo'n werkwoordelijk prefix dan wordt de woordsoort van het item overgenomen (bv. nomen bij "!plug N"). Als het woord dat tegen zo'n item getest wordt precies even lang is als het item dan wordt de categorie verbum praesens toegevoegd aan de categorie(ën) die bij het item gegeven worden<5>. Als bijvoorbeeld achtereenvolgens de woorden (a) *muurplug*, (b) *inplug* en (c) *plug* worden opgezocht in de basisvormencomponent, leidt dit tot respectievelijk de toekenningen (a) nomen, (b) verbum en (c) nomen plus verbum. Merk op dat sommige "!" items geen eigen woordsoortcodes kennen (bijvoorbeeld "!verheug"). In een dergelijk geval falen de vergelijkingen met woorden die geen werkwoordelijk prefix hebben, en wordt voor woorden met zo'n werkwoordelijk prefix of voor woorden die precies gelijk zijn aan het item alleen maar de categorie verbum toegekend. Merk ook op dat aan een (in principe niet uit te sluiten) nieuwvorming als *oeverheug* niet op grond van het "!verheug"-item woordsoorten worden toegekend, maar op basis van de enkele "g N", de laatste letter van het woord.

Een interessant detail van deze component dient nog vermeld te worden. Het lijkt arbitrair of bij de inventarisatie van het Nederlandse woordeinde voor een bepaalde letter (bijvoorbeeld "g") de categorie nomen, verbum of eventueel nog een andere gekozen wordt. Ogenschijnlijk heeft de keuze alleen maar consequenties voor de uitzonderingen die opgenomen moeten worden. In veel gevallen zal er waarschijnlijk wel een optimale keuze zijn omdat het aantal uitzonderingen op de geformuleerde regel kleiner is dan het aantal uitzonderingen op de alternatieve regel(s). Maar er is nog een ander argument dat een rol speelt. Door consequent te kiezen voor de 'slotregel' "letter N", moeten alle andere woordsoorten, en daarmee alle werkwoorden, in een der componenten opgenomen worden. Doorslaggevend voor deze keuze was de problematiek rond het Nederlandse voltooide deelwoord. Deze categorie die

gekenmerkt wordt door een discontinue structuur zal hieronder nog uitgebreid aan de orde komen.

3.3 Zoekprocedure

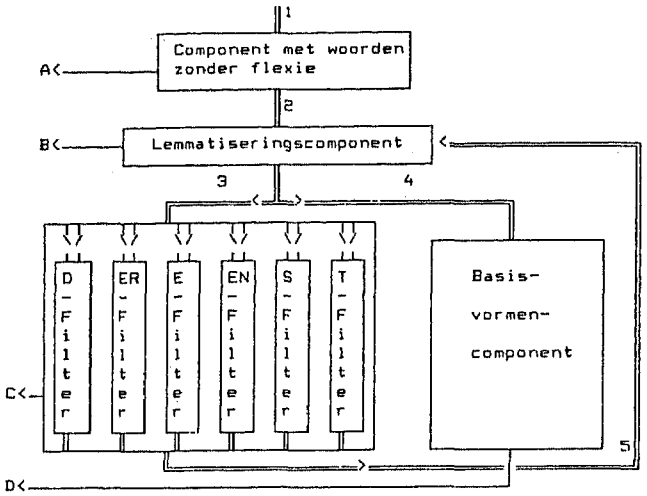
Het belangrijkste kenmerk van het lexiconsysteem is dat lexiconinspectie stopt als de benoeming van een woord volledig is. Het systeem bepaalt zelf, met behulp van informatie bij een bepaald item of met informatie uit de lemmatiseringscomponent, wanneer een woord compleet benoemd is. Dit betekent dus dat soms in meer dan één component gezocht wordt.
Figuur 1 geeft een overzicht van de zoekprocedure. Alle woorden worden eerst opgezocht in de component met woorden zonder flexie. Voor de meeste woorden die hier gevonden worden geldt dat de woordbenoeming compleet is. Uitzonderingen zijn hierboven al aan de orde geweest; het betreft hier woorden zoals *vier* en *pas*. Dat de benoeming voor deze woorden nog niet compleet is, wordt gesignaleerd door het lexicon zelf: één van de toegevoegde codes is dan een zogenaamde stuurcode, een "*". Overigens is het gevolg van het feit dat alle woorden eerst door de component met woorden zonder flexie gaan, dat grote aantallen woorden, de hoog frequente functiewoorden, slechts één van de componenten doorlopen.
Alle woorden die nog niet of niet compleet zijn benoemd, kunnen geflecteerd zijn. Daarom gaan de woorden eerst naar de lemmatiseringscomponent. Als er geen lemmatiseerinstructie op het woord van toepassing is (bijvoorbeeld *wurg*), wordt verder gezocht in de basisvormencomponent en wordt de woordbenoeming voltooid. Elk woord vindt in deze component altijd een passend item, minstens "letter LEXICALE CODE", bijvoorbeeld "g N". Als er wel een lemmatiseerinstructie wordt gevonden, bepaalt deze instructie het verdere verloop van de woordsoorttoekenning. Zo'n instructie bestaat eventueel uit meerdere delen en is specifiek voor een bepaalde groep woorden, bijvoorbeeld woorden op '-eren'. De verschillende delen van een instructie worden achtereenvolgens uitgevoerd. Indien een van deze delen tot een complete woordsoorttoekenning leidt, stopt het zoekproces. Elk deel van een instructie specificeert hoe het woord moet worden aangepast om een lemmavorm te verkrijgen, bijvoorbeeld door aan te geven welke letters van het woord moeten worden afgehaald. Bovendien bevat elk instructiedeel een opgave welke woordsoorten acceptabel zijn, gegeven de verwijderde eindletters. Zo kan bijvoorbeeld, na verwijdering van een eind-e de toekenning nomen verworpen worden. Er bestaat geen flexie van nomina met een '-e'<6>.

3.4 Analyse van een voorbeeldzin

Om te illustreren hoe de lexiconcomponenten samenwerken en welke zoekprocedures doorlopen moeten worden zal bij een zin voor elk woord worden aangeven hoe de woordbenoeming tot stand is gekomen. De cijfers (1-5) en letters (A-D) corresponderen met figuur 1. We kiezen hiervoor een zin waarbij alle componenten aan bod komen, en waarbij problemen die in de volgende paragraaf worden besproken nog geen roet in het eten gooien:

Figuur 1: de opbouw van het lexicon

In het volgende schema worden de relaties tussen de vier componenten weergegeven:



De filters zijn in figuur 1 in één blok weergegeven. Elk filter heeft echter zijn eigen 'in-' en 'uitgang'; er bestaat geen verbinding tussen verschillende filters binnen het blok in de figuur.
Alle woorden worden allereerst gezocht in de Component met flexieloze woorden, 1. Als een woord hier volledig benoemd wordt (de), dat wil zeggen geen van de toegekende codes is *, verlaat het woord via A het systeem. De overige vervolgen hun weg via 2. Een enkel woord wordt hier benoemd (*wordt*) en verlaat via B het systeem. De Lemmatiseringscomponent bepaalt voor ieder woord welke weg vervolgens doorlopen moet worden, 3 (woorden, die na toepassing van een instructie-deel eindigen op een der 'DRENST'-letters, *ketenen* -> *keten*) of 4 (woorden, die na toepassing van een instructiedeel niet eindigen op een der 'DRENST'-letters, *apen* -> *aap*). Als in een filter of in de Basisvormencomponent de benoeming wordt voltooid, dan verlaat het woord het systeem via C of D. Als in een filter de benoeming niet wordt voltooid, een van de toegekende codes is *, dan wordt een volgend instructiedeel geprobeerd (5). Als er niet meer instructiedelen op een woord van toepassing zijn, dan is de benoeming ook voltooid (B).

De soldaten groeven zich in, dronken een borreltje en aten iets.

De toekenning van de woordsoorten aan de woorden van deze zin duurt ongeveer 500 milliseconden. Het resultaat is als volgt:

De	: lidwoord
soldaten	: nomen pluralis
groeven	: nomen pluralis<7>; verbum imperfectum pluralis; verbum praesens pluralis; infinitief
zich	: reflexief pronomen
in	: adverbium; prepositie; niet-verbaal deel scheidbare ww; 2e deel gesplitst vnw. bijwoord
,	: leesteken
dronken	: adjectief; nomen pluralis; verbum pluralis
een	: indefiniet pronomen; hoofdtelwoord
borreltje	: nomen
en	: conjunctie
aten	: verbum imperfectum pluralis
iets	: indefiniet pronomen
.	: leesteken

De benoeming van *de, zich, in, en, een* en *iets* is eenvoudig; deze woorden worden in de component voor woorden zonder flexie gevonden (1) en compleet benoemd (A). De ander woorden worden in deze component niet aangetroffen en in de lemmatiseringscomponent wordt gezocht naar toepasselijke instructies (2).
De lemmatiseerinstructie voor *soldaten* beregelt een grote groep woorden die eindigen met de flexieletters '-ten'. Voor andere groepen woorden, bijvoorbeeld eindigend op '-rten' is een andere instructie van toepassing. Een woord als *storten* kan dus niet dezelfde weg door het lexiconsysteem doorlopen. De woordbenoeming zou dan *nooit* succesvol kunnen zijn. Ook voor de lemmatiseerinstructies geldt dus dat uitzonderingen op een bepaalde regelmatigheid moeten worden verantwoord door aparte instructies. De instructie voor *soldaten* bestaat uit vier delen.
De eerste stap is inspectie van het EN-filter (3) met het ongewijzigde woord. Er worden geen restricties gesteld aan de opgeleverde benoemingen. Met andere woorden: als een woord gevonden wordt, is de benoeming legaal. Dit deel van de instructie is bedoeld voor woorden als *keten* en *verwaten*. Het EN-filter levert geen benoemingen op voor *soldaten* (5).
De volgende stap schrijft voor de '-n' te verwijderen en in het E-filter te zoeken (3). Met dit instructiedeel moeten woorden als *akte(n)* en *gedaante(n)* worden gevonden. Om nu geaccepteerd te worden moet het opgeleverde woord een nomen zijn en de instructie wijzigt deze benoeming in nomen pluralis. Dit is dus een voorbeeld van de restricties die in de lemmatiseerinstructies zijn gespecificeerd. De ratio achter deze restrictie is dat alleen nomina eindigend op een 'e' een flectieve 'n' kunnen hebben. Ook deze stap levert voor het woord geen acceptabele benoeming op (6). Merk op dat *soldaten* terecht niet wordt geaccepteerd als meervoud van *soldate*.
Vervolgens wordt '-en' verwijderd. Indien mogelijk wordt het woord nu tot lemma-formaat teruggebracht (*soldaat*). De speurtocht naar benoemingen wordt vervolgd in het T-filter (3), waar, zoals verderop in de tekst zal worden uitgelegd, voornamelijk werkwoorden zijn opgenomen en *soldaat* dan ook niet wordt gevonden (6).

De laatste stap instrueert met het woord zonder '-en' de basisvormencomponent te raadplegen. Ook hier wordt het woord zelf niet gevonden maar het laatste item van het 't'-deel van de component kent aan ieder woord op een '-t' dat tot hier weet door te dringen, de woordsoort nomen toe (D). Deze benoeming wordt door de instructie aan de flexie aangepast (verwijdering van 'en' betekent in dit geval dat het woord een meervoud is) en resulteert in nomen pluralis.

Voor het woord *groeven* wordt een driedelige lemmatiseerinstructie gevonden voor de flexie '-en'. Allereerst wordt, zonder aanpassingen aan het woord, in het EN-filter gezocht (3). Woorden die hier gevonden moeten worden, zijn bijvoorbeeld *leven* en *oneven*; woorden die uitzonderingen zijn op de regel dat '-en' flexie signaleert. *Groeven* wordt hier niet gevonden (5)<7>.
De tweede stap is identiek aan het tweede deel van de lemmatiseringsinstructie voor woorden op '-ten' en verwijdert de '-n'. Ook de component waarin gezocht wordt, het E-filter (3), en de restricties op wat acceptabele benoemingen zijn, zijn identiek. In dit deel wordt het item "groeve N *" gevonden. De toekenning nomen wordt vertaald in nomen pluralis en, vanwege de instructie door te zoeken ("*") weet het systeem dat de benoeming nog niet is voltooid (5).
De derde stap bestaat uit verwijdering van '-en', wijzigen van het woord tot lemma-formaat, dat wil zeggen 'v' wordt 'f' en zoeken in de basisvormencomponent. Voor de stam *groef* wordt de basisvormencomponent geraadpleegd (4). Dit deel van de instructie bevat een aantal restricties op mogelijke benoemingen. Acceptabele benoemingen zijn voor deze instructie alleen: nomina, praesens- of imperfectumstammen van verba, adverbia, en tegenwoordige en verleden deelwoorden. Het gelemmatiseerde woord *groef* wordt hier herkend als een nomen, een imperfectum- èn praesensstam, en deze benoemingen worden gewijzigd in nomen pluralis, verbum imperfectum pluralis, verbum praesens pluralis en infinitief (D). Merk op dat tweemaal de benoeming nomen pluralis wordt toegekend (op grond van *groeve* en *groef*). Slechts een van de twee wordt gehandhaafd.

De lemmatiseerinstructie voor *dronken* is dezelfde als voor *groeven*. De resultaten verschillen echter. De eerste stap, zoeken met het hele woord in het EN-filter (3), levert de benoeming adjectief op en de opdracht verder te zoeken (5). De tweede stap, *dronke(n)*, leidt niet tot resultaat in het E-filter (3", ">5), en de derde stap, *dronk(en)* (4), voegt nog de benoeming nomen pluralis en verbum imperfectum pluralis toe op basis van een imperfectumstam (D).

De woordsoorttoekenning van *borreltje* verloopt via de lemmatiseerinstructie voor woorden eindigend op '-e'. De eerste stap van deze tweedelige instructie zoekt met het hele woord in het E-filter (3). Hier wordt het woord compleet benoemd als nomen via het item "tje N (C<)". Het tweede deel van de instructie (aanwezig voor adjectiva en verba als *vrije* en *rije*) wordt niet meer uitgevoerd, omdat de benoeming compleet is.

Het laatste woord in de zin, *aten*, wordt beregeld door een eendelige lemmatiseerinstructie. Deze instructie is een andere dan die voor *soldaten*, ondanks het feit dat de woordeinden van beide woorden identiek zijn. De instructie die van toepassing is, is specifiek voor het werkwoord *aten* en werkwoorden die met *aten* gevormd worden, zoals bijvoorbeeld *overaten*. De speciale instructie voor deze groep woorden is noodzakelijk omdat *aten* een uitzondering is op een regelmatige lemmatisering. In de regel ondergaan woorden, die na verwijdering van bijvoorbeeld '-en' eindigen op een vocaal plus consonant, vocaalverdubbeling om de basisvorm te verkrijgen. Ingeval *aten* zou dit abusievelijk de vorm *aat* opleveren. De specifieke instructie voor (over)aten levert *at* op (3). Op dit moment van de analyse is al duidelijk dat alleen de stam van het werkwoord *at* een acceptabele woordsoort kan toekennen. Deze benoeming wordt door de instructie gewijzigd in verbum imperfectum pluralis (C).

3.5 Complicaties en oplossingen

Het tot nu toe gegeven exposé heeft zich beperkt tot een aanduiding van de grote lijnen van de gevolgde werkwijze. In deze subparagraaf zal nader worden ingegaan op enkele complicaties. Er is een zodanige selectie gemaakt dat meer inzicht wordt verkregen in de opbouw en de mogelijke zoekstrategieën in het lexicon. Het gaat hier om drie onderwerpen die alle te maken hebben met het werkwoord namelijk (a) de werkwoordsvorm eindigend op '-t', (b) de onregelmatige voltooide deelwoorden en (c) woorden eindigend op '-eren'.

(a) Elke eind-t is een mogelijke werkwoordsvervoeging. De opbouw van de filtercomponent en de basisvormencomponent vereist dat stammen eindigend op een niet-flectieve '-t' in het T-filter worden opgenomen (woorden als *falikant*, maar ook werkwoorden als *spuit*). Door deze oplossing zou het T-filter enorm vergroot worden omdat veel woorden in het Nederlands eindigen op een niet-flectieve 't'. Om het filter klein te houden is gekozen voor een andere aanpak. Het aantal werkwoordstammen op '-t' is beperkt, ongeveer 200. Deze zijn opgenomen in het T-filter. Een woord op '-t' gaat eerst door dit filter. Wanneer een woord volledig kan worden benoemd, bijvoorbeeld *zet*, wordt de zoekprocedure stopgezet. Wordt een woord niet gevonden (*schuit*) dan wordt de basisvormencomponent geraadpleegd. Woorden als *schuit* worden niet in het T-filter gevonden omdat de filters, zoals eerder al is uitgelegd, geen items hebben die op basis van slechts één eindletter een woordsoort toekennen.
Bij een aantal items in het T-filter staan nog aanwijzingen voor mogelijke andere benoemingen of een opdracht om verder te zoeken. Een voorbeeld is "spuit". Het T-filter bevat het item "!spuit N *". Het uitroepteken geeft aan dat het woord onder bepaalde voorwaarden een verbum is, dus dat *spuit* een vorm is van het werkwoord *spuiten*. Achter het item staat de woordsoort n(omen) en de opdracht door te zoeken (*). De woordsoorttoekenning tot nu toe is dus '*spuit*: verbum enkelvoud (*spuiten*); nomen meervoud'. Het item *spuit* wordt nu volgens de lemmatiseerinstructies ontdaan van de eind-t, waarna de basisvormencomponent wordt geraadpleegd omdat de ontstane vorm niet meer eindigt op een flexieletter. In deze lijst wordt de

benoeming compleet gemaakt door de toevoeging, op basis van het item "!spui N", 'verbum 3e pers. enkelvoud (*spuien*)'. De uiteindelijke toekenning verwijst dus naar de beide werkwoorden *spuiten* en *spuien*.
Een voordeel van deze oplossing, opname van werkwoordstammen op '-t' in het T-filter, is dat het 't'-deel van de basisvormenlijst niet behoeft te worden aangesproken om een vorm als werkwoord te benoemen. In het 't'-deel hoeven dus alleen de mogelijkheden nomen en adjectief te worden opgenomen. Omdat de meeste niet-werkwoordelijke vormen op 't' nomina zijn, bestaat het 't'-deel voornamelijk uit adjectieven die eindigen op een 't' (ongeveer 300 stuks). Dit zijn dan de uitzonderingen; alle nomina op '-t' kunnen worden benoemd door het item "t N" op te nemen. De voordelen van deze aanpak zijn dat het lexicon, ondanks het grotere T-filter voor de werkwoordsvormen, klein kan blijven, omdat bij het 't'-deel van de basisvormenlijst slechts een beperkt aantal items behoefde te worden opgenomen.
Complicaties die zouden kunnen optreden als de flectieve of niet-flectieve eind-t niet aan het eind van het woord staat maar nog gevolgd wordt door andere flexieletters, worden door het aldus opgezette systeem probleemloos verwerkt. Woorden als *spuiten* en *barstte* doorlopen in principe dezelfde weg. Na verwijdering van 'en' respectievelijk 'te' worden de werkwoordstammen *spuit* en *barst* in het T-filter gevonden. Aan uitzonderingen op de aldus geformuleerde regel als *verwaten* en *geste* worden, voordat lemmatisering wordt geprobeerd, de woordsoorten toegekend in respectievelijk het EN-filter en het E-filter.

(b) Wanneer men voltooide deelwoorden benoemt op grond van het prefix 'ge' in combinatie met een werkwoordstam en een suffix '-d', '-t' of '-en', ontstaan veel foute woordbenoemingen. De woorden *geheid, geniet* en *geslacht* zijn geen echte voltooid deelwoorden of kunnen ook tot een andere categorie behoren. Een extra probleem is nog dat sommige voltooide deelwoorden eerder als adjectief of bijwoord functioneren, bijvoorbeeld *welgesteld(e), gemiddeld* of *bejaard*. Maar om tot de benoeming adjectief of bijwoord te besluiten moet wel eerst duidelijk zijn dat het om een voltooid deelwoord gaat. Om deze problemen het hoofd te bieden is de volgende constructie gekozen.
De voltooide deelwoorden op '-en' zijn als aparte categorie ondergebracht in het EN-filter. Deze groep bevat 200 woordeinden (zoals *lopen*; de *en* is opgenomen in het item) die alle zijn gemarkeerd voor specifieke kenmerken waaraan het woord moet beantwoorden wil de benoeming voltooid deelwoord legaal zijn. In dit filter worden alle deelwoorden van dit type benoemd. De markeringen bij de items beregelen de prefigering ('ge-', 'be-', 'ver-' of 'ont-'). In de regel moet een van deze prefixen aanwezig zijn. In een aantal gevallen echter mag het prefix wegblijven. In het algemeen geldt dit indien een van de prepositie 'door', 'over', 'mis', 'onder', 'achter' of 'voor' voorafgaat aan de deelwoordstam. De markeringen zijn specifiek voor elk item. Woorden als *misdragen, overkomen* en *doorlopen* krijgen de toevoeging voltooid deelwoord; *mislopen* en *doorkomen* niet. Bij een aantal items in deze lijst staat uiteraard ook nog de instructie verder te

zoeken, zodat een vorm als *doorlopen* na het toepassen van lemmatisering ook benoemd kan worden als praesens of infinitief.
Voor de voltooide deelwoorden op '-d' of '-t' bleek het nodig de woordsoorttoekenning in twee fasen te laten verlopen. De lemmatiseerinstructie voor woorden eindigend op '-d' of '-t' houdt rekening met de mogelijkheid dat de woordsoort voltooid deelwoord moet worden toegekend. Allereerst doorlopen deze woorden het T- of D-filter. Als hier volledige benoeming plaatsvindt, is de analyse beïndigd; het woord is geen voltooid deelwoord. Als in het filter niets gevonden wordt, vervolgt het woord, na verwijdering van de eindletter ('-d' of '-t') zijn weg in een andere component, doorgaans de basisvormencomponent. In de betreffende component worden aan het woord één of meer woordsoorten toegekend. Alleen indien een van deze woordsoorten verbum is, in een of andere vorm, volgt de tweede fase van de voltooid deelwoord analyse. In het woord wordt nu gezocht naar een voltooid deelwoord prefix ('ge-' etc. of een prepositie, bijvoorbeeld 'door-') dat aansluit op de gevonden werkwoordstam. Als een dergelijk prefix gevonden wordt, wordt in principe de woordsoort voltooid deelwoord toegekend. Een vorm als *ge-bel-d* krijgt dus alleen een voltooid deelwoord benoeming omdat de werkwoordstam *bel* wordt gevonden in de basisvormencomponent: Hierdoor wordt voorkomen dat veel woorden met 'ge-' en '-t' of '-d' (structureel niet te onderscheiden van voltooide deelwoorden) ten onrechte als voltooid deelwoord worden herkend. Een woord als *geld* krijgt dus niet de benoeming voltooid deelwoord (*ge-l-d*) omdat *l* niet als werkwoordelijk item in de basisvormencomponent staat.
Zelfs deze eis, een voltooid-deelwoordbenoeming moet gesteund worden door een werkwoordstam in de basisvormencomponent, bleek in sommige gevallen niet voldoende. Ook vormen als *doorloopt* en *voorstelde* zouden in deze opzet als voltooid deelwoord benoemd worden. Na prepositie is 'ge-' immers niet verplicht: *doordacht, voorvoeld*. Fouten van het type '*doorloopt* (*doorlopen*)' konden alleen voorkomen worden door in de basisvormencomponent bij elk werkwoordelijk item aan te geven of het al dan niet als stam van een voltooid deelwoord op '-d' of '-t' kan fungeren. Ook is bij niet-werkwoordelijke items aangegeven of ze eventueel wel kunnen fungeren als voltooid deelwoordstam; *lief* bijvoorbeeld is als zodanig gemarkeerd om *verliefd* te kunnen herkennen. Een andere complicatie is dat een woord als bijvoorbeeld *gepraat* naast de benoeming voltooid deelwoord ook nog de benoeming nomen moet krijgen. Ook dit probleem is opgelost. Zoals het systeem nu is opgebouwd, geschiedt de toekenning voltooid deelwoord zonder fouten.

(c) Een apart probleem vormen de woorden op '-eren'. Ten eerste is de lemmatisering van deze woorden uiterst complex omdat veel varianten van flexies verscholen kunnen gaan achter de uitgang '-eren'. Een aantal voorbeelden kunnen dit duidelijk maken.

onteren moet gelemmatiseerd worden tot (*ont*)*eer*
groteren -- *groot*
scheren -- *scheer*
bieren -- *bier*
ambiëren -- *ambieer*
knipperen -- *knipper*

kipperen -- heeft al lemma-formaat
kinderen -- *kind*
blonderen -- zowel *blondeer* als *blond*
halveren -- *halveer* (en niet: *half*)

Bovendien moet voor deze groep woorden een uitzondering worden gemaakt op de, bij de voltooide deelwoorden gebruikte, strategie<8> dat werkwoorden in het systeem moeten worden opgenomen. Immers bijna alles kan 'ge-eer-d' of 'geïs-eer-d' worden. Deze klasse van '-eer' werkwoorden is in het Nederlands niet opsombaar. De gevolgde procedure voor '-eren'-woorden maakt gebruik van het feit dat de werkwoorden op '-eer' niet in het lexicon zijn opgenomen. Normaal wordt, als een woord niet wordt gevonden in één van de componenten, de woordsoort nomen toegekend op basis van een item als "f N". Bij woorden op '-eren' luidt de conclusie als er geen volledige lexicalisering plaats vindt daarentegen verbum.
De eerste stap in de analyse van '-eren'-woorden is het opzoeken van het woord in het EN-filter. Hier wordt aan nominale meervouden zoals *kinderen* de woordsoort toegekend. Vervolgens wordt de vorm ontdaan van de terminale '-en'. In het ER-filter wordt gezocht naar items die het woord benoemen als verbum en/of adjectief. Bovendien worden ook soort- en stofnamen geaccepteerd. Voorbeelden zijn '*knipperen*: verbum (*knipperen*)', '*donkeren*: adj (*donker*)', '*ijzeren*: stofnaam (*ijzer*)'. Vervolgens wordt het woord gemanipuleerd tot een '-eer'-stam. Met de nu ontstane vorm wordt opnieuw gezocht in het ER-filter en vervolgens in de basisvormencomponent. In het filter wordt gezocht naar nomina op '-eer' (*geweren*). De basisvormencomponent wordt geraadpleegd om woorden als *blinderen* en *blonderen* te relateren aan de werkwoordstammen *blindeer* en *blondeer*. Vervolgens wordt het woord ontdaan van de vier laatste letters: '-eren'. Van het resterende woorddeel wordt indien nodig een lemmavorm gemaakt (*groteren* wordt *groot*). In beide filters wordt gezocht naar adjectieven. Uiteindelijk, als al deze opties niets hebben opgeleverd, wordt de woordsoortcode werkwoordsvorm (praesens meervoud of infinitief) toegekend.

4. Evaluatie

Het Hata-systeem, zoals dat op dit moment geïmplementeerd is op een VAX-mainframe en een IBM-PC, werkt niet foutloos. Een aantal fouten is van structurele aard en kan dan ook niet eenvoudig worden opgelost. De structurele problemen betreffen de benoeming van werkwoorden als transitief en intransitief en de herkenning van eigennamen. Eigennamen zullen meestal door het systeem als nomina worden herkend, op zich een bevredigende oplossing. Afhankelijk van het 'uiterlijk' is het echter niet ondenkbaar dat een aantal eigennamen een andere benoeming zullen krijgen. Dit probleem kan opgelost worden door het systeem te voorzien van een tekstspecifieke uitbreiding in de component voor vormen zonder flexie, of in de vorm van een eigennamen-lexicon, dat door de gebruiker kan worden gevuld met voor een bepaalde tekst relevante namen. Alleen bij de bewerking van grote tekstbestanden zal dit een probleem zijn, omdat het inventariseren van alle eigennamen in zo'n bestand een tijdrovend karwei is.

Voor het transitief/intransitief probleem is een dergelijke praktische oplossing niet voorstelbaar. Weliswaar zou het in het Hata-systeem mogelijk zijn een groot aantal werkwoorden te coderen voor de mogelijkheid van transitief of intransitief gebruik, maar een dergelijke codering zou het probleem niet bevredigend oplossen. Niet alle werkwoorden zijn opgenomen in de lexica, bijvoorbeeld de werkwoorden op '-eren'. Ernstiger is echter dat transitiviteit of intransitiviteit een gebruiksaspect is van werkwoorden, dat alleen bepaald kan worden na syntactische/ semantische analyse van een zin. Een dergelijke analyse behoort niet tot de doelstellingen van het Hata-systeem.
De overige, niet structurele fouten kunnen in de loop van het project worden opgelost. Deze fouten worden veroorzaakt, maar kunnen ook worden hersteld, door de 'ontwerp-filosofie' van het systeem. Immers, met een beperkte hoeveelheid woordeinden en lemmatiseerinstructies moeten alle Nederlandse woorden kunnen worden benoemd. Ten gevolge van de complexe wijze waarop de verschillende componenten van het systeem met elkaar samenwerken is het van groot belang dat een woord, of een groep woorden, op de juiste plaats in het systeem benoemd worden. Gebeurt dit niet dan loopt zo'n woord verder in het systeem en wordt ergens benoemd via een regel die niet voor het woord of de groep woorden bedoeld is. Een simpel voorbeeld van zo'n probleem zou kunnen zijn een ommisie in het EN-filter. Als bijvoorbeeld een nomen met als meervoudsuitgang '-eren' (bijvoorbeeld *kinderen*) hier niet gevonden wordt, wordt aan dit woord uiteindelijk via de lemmatiseringsregels de benoeming verbum toegekend (zoals aan *amenderen*). De organisatie van de componenten vereist dus een hoge nauwkeurigheid; elke onregelmatigheid in termen van de elders (in de lexicale componenten of in de lemmatiseringscomponent) gedefinieerde regels moet verantwoord zijn.
Deze opzet maakt het ook mogelijk de aan te treffen fouten op eenvoudige wijze te verhelpen. Gaat het om een groep woorden, dan moet het samenspel tussen de componenten worden gewijzigd. Gaat het om een individueel woord, dan is de remedie nog eenvoudiger: het woord moet op de juiste plaats in het systeem verantwoord worden. De flexibele opzet van het Hata-systeem maakt dit mogelijk.
We vermelden hier nog enkele benoemingen die weliswaar correct zijn, maar toch enigszins vreemd aandoen. Zijn de woorden *blonderen* en *mankeren* comparatieve vormen van respectievelijk *blond* en *mank*? In de huidige configuratie van het lexicon wel. Het is niet ondenkbaar dat deze woorden in zeer speciale contexten ook inderdaad deze benoeming toegekend moeten krijgen. Waarschijnlijk is het echter niet.
Een vergelijkbaar fenomeen doet zich voor bij een aantal adjectieven, verbogen met 'e'. In veel van die gevallen is strikt genomen ook de benoeming conjunctief (van een praesens werkwoord) terecht: *dikke*, *korte*. Men kan zich afvragen of gezien de lage frequentie van de conjunctief deze benoeming niet geheel uit het systeem verwijderd zou moeten worden. Beslissingen als deze zouden, met verlies van volledigheid, het werk in de disambigueringsfase kunnen vergemakkelijken.

Deze uiteenzetting maakt duidelijk dat de meeste problemen met betrekking tot automatische woordbenoeming zijn opgelost. Ter indicatie van het succespercentage is een steekproef genomen op een tekst van 5946 woorden. Als materiaal werd een gedeelte van het Eindhovencorpus ontdaan van de coderingen, en als ruwe tekst aan Hata aangeboden. Slechts bij 200 (100 verschillende) woorden van de tekstwoorden was sprake van (meestal) onvolledige of (zelden) foutieve benoemingen, doorgaans te wijten aan incorrecte specificaties in de lemmatiseerinstructies of omissies en onzorgvuldigheden in een van de andere componenten. Het merendeel van de fouten kon dan ook worden hersteld door kleine veranderingen in de componenten door te voeren. Zo ontbrak *navolging* als nomen in de basisvormencomponent, met als gevolg de benoeming 'verbum imperfectum' (*na + vol + ging*; ook de nomina *opvolging, opeenvolging* en *achtervolging* zijn toegvoegd aan de basisvormencomponent). Ook kwam een fout aan het licht in de lemmatiseerinstructies voor woorden die eindigen op *-eerde* en *-eerden*, die er toe leidde dat geen verschil gemaakt kon worden tussen vormen als *adviseerde* en *geadviseerde*.

Het succespercentage van 96 zegt op zichzelf nog niet zoveel. In andere teksten verschijnen andere woordvormen, en daarmee andere problemen voor de woordbenoeming. Gelet echter op de aard van de fouten in de steekproef kan gesteld worden dat de ontwerpfilosofie van het Hata-project juist is gebleken. Hoe meer tekst aan het systeem wordt aangeboden, des te meer foutjes kunnen worden getraceerd en hersteld.

4.1 Voortzetting werkzaamheden Hata .

Het Hata-systeem zoals dat in bovenstaande tekst is besproken, is nog maar de helft van het uiteindelijke systeem. De tweede fase, de disambigueringsfase, maakt een essentieel onderdeel uit van het totale programma. In deze slotparagraaf zal nog kort ingegaan worden op de werkwijze die bij deze disambiguering zal worden gevolgd, om de lezer een - zij het schetsmatig - beeld te geven van de uiteindelijke gebruikswaarde van het Hata-systeem.
De disambiguering van de resultaten van de eerste fase zal op soortgelijke wijze worden aangepakt als de disambiguering in het LOB-project (zie o.a. Marshall, 1983). Deze benadering maakt gebruik van zogenaamde contextregels. Deze regels doen uitspraken over de waarschijnlijkheid van de opeenvolging van een aantal, meestal twee, woordsoortcodes. Deze waarschijnlijkheden zijn berekend aan de hand van het voorkomen van de woordsoorten in een omvangrijk corpus. De benadering is dus eerder probabilistisch dan taalkundig van aard. De uiteindelijke resultaten van disambiguering met behulp van contextregels hangt sterk af van de representativiteit van het gebruikte corpus. Het Hata-project maakt voor het opstellen van deze regels gebruik van het Eindhovense corpus.
Contextregels worden uiteraard alleen gebruikt als in een zin een woord voorkomt dat in de eerste fase van de verwerking meer dan één woordsoortcode heeft meegekregen. Als dit woord aan twee kanten begrensd wordt door niet ambigue woorden is de toepassing van de contextregels eenvoudig. In het geval dat een serie woorden ambigu is, treedt een mechanisme in werking dat de ambiguiteit van links

naar rechts probeert op te lossen, door de meest waarschijnlijke sequentie van codes te berekenen. De toegekende codes zijn dan de codes die deel uitmaken van deze meest waarschijnlijke sequentie. In de hierboven vermelde steekproef hebben 3301 van de 5946 woorden twee of meer codes. Opvallend daarbij is dat verreweg de meeste ambiguïteiten uit de component van invariante woorden komen: 2698 stuks. De open woordklassen leveren 603 woorden met meer dan een benoeming.

NOTEN

1. IBM-Nederland heeft ten behoeve van het project een IBM-AT micro-computer ter beschikking gesteld.

2. In C3C bepaalt het eerste cijfer de woordsoort; het tweede cijfer voegt informatie toe over subcategorieën en het derde cijfer geeft voornamelijk informatie over het al of niet verbogen zijn van een woordvorm.

3. De opsomming in Uit den Boogaart (1975) is op het Eindhovense corpus gebaseerd en daarom onvolledig. De lijst van bijwoorden is aangevuld op basis van de inventarisatie van Van Wijk en Kempen (1980), die het probleem al eerder constateerden en de lijst completeerden met de woorden die in het Nederlands niet anders dan als bijwoord voorkomen. De samenvoeging resulteerde in een lijst van zo'n 600 stuks.

4. Een lemmatiseerinstructie bestaat dus vaak uit meer delen. Deze delen specificeren de bewerkingen die het woord moet ondergaan en de componenten die doorzocht moeten worden.

5. Voor de (onregelmatige) imperfectumstammen is de markering "?" ingevoerd; zo bestaat het item "?liep". De interpretatie van deze items is analoog aan die van de "!"-items, met dien verstande dat op basis van een "?"-item nooit besloten kan worden tot de benoeming infinitief.

6. De uitzonderingen op deze regel zijn in het E-filter opgenomen.

7. Een complicatie met *groeven* is dat in eerste instantie het item "ven N" (i.v.m. *zoetwaterven, heideven, bosven,* en dergelijke) wordt gevonden. Hetzelfde probleem doet zich voor bij woorden eindigend op bijvoorbeeld *pen, den* en *ren*. Om fouten zoveel mogelijk te vermijden zijn de betreffende items voorzien van een extra markering ('@') die instrueert om door te gaan met zoeken. Indien ook een werkwoordelijke benoeming resulteert, is de @-benoeming ongeldig. Uitzonderingen op deze regelmatigheid moeten lexicaal worden verantwoord.

8. Deze strategie sluit overigens nauwkeurig aan bij de talige werkelijkheid; nieuwvormingen zijn in het algemeen nomina.

LITERATUUR

BAKEL, J. VAN, e.a. (1977), *Automatische Lexicalisering*. Grammarij 5, 1977.

JOHANSSON, S. (ed.) (1982), *Computer Corpora in English Language Research*. Bergen, Norwegian Computing Centre for the Humanities, 1982.

LEECH, G.N. & R.G. GARSIDE (1982), 'Grammatical Tagging of the LOB Corpus: General Survey'. In: *Johansson* (ed.) (1982).

MARSHALL, I. (1983), 'Choice of Grammatical Word-Class without Global Syntactic Analysis: Tagging Words in the LOB Corpus'. In: *CHUM* 17 (1983), pp. 139-150.

NIEUWBORG, E.R. (1978), *Retrograde Woordenboek van de Nederlandse Taal*. Deventer-Antwerpen 1978.

OPSTAL, T. VAN & H. KEMPFF (in press), 'Word Class Information to Dutch Words in Texts'. Een lezing op het Symposium voor Taaltechnologie Tilburg 1985, te verschijnen in symposiumbundel.

RENKEMA, J. (1981), *De Taal van 'Den Haag', een Kwantitatief-stilistisch Onderzoek naar Aanleiding van Oordelen over Taalgebruik*. 's-Gravenhage, Staatsuitgeverij, 1981.

RENKEMA, J. e.a. (1975), *HATA*. Halfautomatische Tekstanalyse, een vooronderzoek. Till-paper 47, 1984.

UIT DEN BOOGAART, P.C. (1975), *Woordfrequenties in geschreven en Gesproken Nederlands*. Utrecht, Oosthoek, Scheltema en Holkema, 1975.

WIJK, C. VAN, & G. KEMPEN (1980), 'Functiewoorden - een inventarisatie voor het Nederlands'. In: *ITL review of applied linguistics* 47 (1980), pp. 53-68.