Programmatuur-sectie

Text processing in Nederland.

In april 1978 werd onder leiding van professor D.E. Bailey aan het Technisch Centrum FSW van de Universiteit van Amsterdam een kursus georganiseerd in de informatika. De kursus bestond uit een reeks van 4 middagen en was gewijd aan "computer processing of text". Van deze kursus wordt hier verslag gedaan, terwijl het verslag tevens wordt aangegrepen om een overzicht van automatische analyse van tekst in Nederland te geven. Tot het schrijven van dit overzicht werd besloten om de volgende redenen: Ten eerste was de kwaliteit van de door Bailey opgezette kursus dusdanig hoog dat plaatsing ervan in een breder kader eenvoudigweg noodzakelijk is.
Ten tweede ontbreekt tot op heden ieder overzicht van de aktiviteiten en resultaten die in Nederland tot nu toe werden opgezet, resp. behaald voor zover het automatische analyse van teksten betreft.
Dit artikel zal inhoudelijk bestaan uit twee deelgebieden: een overzicht van de aktiviteiten die in het kader van de kursus van Bailey werden ontplooid, en een systematisch overzicht van de problemen in de automatische behandeling van teksten evenals de antwoorden die in Nederland, hetzij door de inleiders in de kursus, hetzij elders werden gegeven. De twee delen zullen niet streng gescheiden worden behandeld.

Doel van de kursus was een algemeen overzicht te geven van de toepassingen van automatische analyse van teksten in de sociale wetenschappen. Een overzicht werd gegeven van computertalen en systemen die voor dit toepassingsgebied zijn ontwikkeld. Kort werden de belangrijke gebieden van automatische analyse van tekstmateriaal als "artificial intelligence", "cognitive psychology" en "content analysis" betreden. Het hoofdaksent viel echter op de procedurele kant van de zaak, d.w.z. op computer-talen en -procedures voor behandeling van text per computer.
De kursus was verdeeld in 4 porties die resp. door Bailey, C.v.d. Wijgaart (FSW), D. Champeaux (economische wetenschappen), G. van der Steen (fonetische wetenschappen) werden gegeven. Champeaux en van der Steen gaven naast een overzicht van het gebied waarop ze werken een verslag van eigen werkzaamheden op dit gebied. Bailey en v.d. Wijgaart besteedden hun aandacht vooral aan het geven van een overzicht. Dit overzicht betrof vooral programmeertalen, d.w.z. Pascal, Algol 68 en SNOBOL. Bij het overzicht van SNOBOL kon ook een echte toepassing worden gedemonstreerd, omdat het beroemde ELIZA van Weizenbaum (Weizenbaum, 1977) op de SARA Cyber in SNOBOL4 aanwezig is. Als kommentaar op dit gedeelte moet worden gegeven dat ze uitblonken door helderheid en dat het inderdaad lukte in goed twee uur een beklijvend inzicht in de

de problematiek te geven. Wel vraagt men zich af, waarom een taal als ALGOL68 die op het stuk van performance (Boot, 1978) op geen enkele wijze serieus kan konkureren met PASCAL, SNOBOL, LISP, PL/I, SIMULA of het door door van der Steen gebruikte DGL (een PL/I-achtige taal), moest worden behandeld. Dat deze taal ter sprake kwam, zal wellicht mede liggen aan het feit dat "text processing" dikwijls louter en alleen wordt gezien als een probleem van "character manipulatie", d.w.z. als een probleem dat alleen wordt gedefinieerd vanuit het gezichtspunt van de "computer memory". Daaruit kan dan de wat irrelevante diskussie ontstaan of de "character handling" vanuit puur theoretisch oogpunt d.w.z. voor de compilerbouwer wel "mooi" is gedefinieerd. In de literatuur (Addyman, 1977) b.v. vindt men deze merkwaardige diskussie ook, zonder dat duidelijk wordt gemaakt in hoeverre zulke overwegingen voor het analyseren van teksten in natuurlijke taal relevant zijn. Zonder dat deze kwestie als zodanig aan de orde werd gesteld kwam hij met name tevoorschijn uit de desiderata die van der Steen stelde bij de ontwikkeling van programmatuur voor "text processing". Terecht werd gesteld dat de volgende kriteria de belangrijkste zijn: de programmatuur moet interaktief zijn; de programmatuur moet "user friendly" zijn, d.w.z. er moet een simpele "command language" worden ontwikkeld, waarmee alle soorten van text manipulatie, niet alleen character handling, kunnen worden verricht door een gebruiker die doorgaans van de informatika-achtergrond geen weet heeft.
Moet er in het geval dat het gaat over "character handling" sprake zijn van een uitsluitende aandacht voor het gestruktureerde computergeheugen, van text processing kan pas sprake zijn wanneer ook de "linguistic data structure" (Boot, 1976) in het design van de text processor wordt betrokken. Hiermee is dan meteen de meest fundamentele vraag bij de automatische analyse van tekstmateriaal aan de orde gesteld. Deze vraag luidt: Hoe dient het design van soft-ware packages te worden ontworpen, opdat niet de uitsluitende aandacht valt op de technisch/informatika-achtige aspekten ervan. Het antwoord op de vraag kan slechts gevonden worden door het begrip tekst aan een nadere beschouwing te onderwerpen.
Een definitie van het begrip tekst impliceert aandacht voor 2 aspekten. Er is ter definitie een kwantitatief aspekt en een kwalitatief aspekt nodig. (Bense, 1969) Het kwantitatief aspekt lijkt op het eerste gezicht als losstaand van het kwalitatieve aspekt definieerbaar en behandelbaar. Men definieert dan "woorden", "zinnen", "regels". Deze eenheden worden gelijkgesteld aan de kenmerken die ze hebben in het gestruktureerd computergeheugen, d.w.z. als "woord" wordt gedefinieerd "iedere string tussen twee blanks" of "tussen een leesteken en een blank". Als zin "iedere string tussen twee leestekens"; als regel "iedere als zodanig gekenmerkte string". Kenmerking geschiedt dan

door extra-kodering of door het nemen van ponskaartbeelden als regel.
Bij deze manier van definiëren blijft men strikt binnen wat boven "character handling" genoemd werd. De volgende resultaten zijn met deze wijze van definiëren bereikbaar: men kan indexen vervaardigen op de strings die tussen blanks voorkomen ("woordvormen"); men kan woordvormen tellen; men kan woordvormen doen voorzien van enige woordvormen die er in de oorspronkelijke tekst omheen stonden en zodoende een zgn. "konkordantie" vervaardigen; men kan er een vraag-/antwoordsysteem mee opbouwen volgens de specifikaties van ELIZA. Aan dit vraag/antwoordsysteem ligt het volgende design ten grondslag: Men neme een serie uitdrukkingen als "goede morgen", "ik begrijp niet wat u bedoelt", "het zal wel weer regenen". Daarnaast geeft men een serie sleutelwoorden die op bepaalde plaatsen in de vaste uitdrukkingen kunnen worden ingezet. B.V. "zeer", "echt", "hoe", "weer", "vanmiddag", "avond", "middag", "vanmiddag", "zijn". Men geeft dus frames mee en een "lexicon". Hiermee is dan de volgende dialoog mogelijk:

ROB: Goede morgen, hier is Rob de robot. Laten we over het weer praten!
Gebr.: Beste Rob, wat denk je van het weer?
ROB: Het zal wel weer regenen.
Gebr.: Hoe kan dat nu?
ROB: Ik begrijp niet wat u bedoelt.
Gebr.: Hoe is het weer vanmiddag?
ROB: Het zal een goede middag zijn.
enz. enz.

Het is duidelijk dat ook zinnen als "hier is Rob de robot" vooraf in het geheugen zijn ingevoerd. Wordt een vraag gesteld die niet in het repertoire voorkomt dan luidt het antwoord steevast: "Ik begrijp niet wat u bedoelt". Een dergelijk antwoord maakt een "intelligente" indruk. ELIZA was in dit opzicht van erg leuke grapjes voorzien b.v.: de zinnen: Do you believe our conversation will help you to ....?
What makes you think I ....?
Op de plaatsen van de puntjes wordt dan de string ingevuld die net aan de orde is geweest. (b.v. "walk").
De konklusie die men nu kan trekken, is dat het kennelijk mogelijk is om met pure "character handling" resultaten te bereiken die heel wat lijken. Het was precies Weyzenbaum's doelstelling dat met ELIZA aan te tonen. Hieruit nu dient weer de konklusie te worden getrokken dat text processing niet op de kwantitatieve kant alleen moet worden gebaseerd: de resultaten die daaruit voortvloeien zijn schijnresultaten. Het centrale probleem van automatische tekstanalyse is de definitie van de samenhang tussen kwantitatieve en kwalitatieve aspekten binnen de tekst.

Niet alleen ELIZA en de daarmee verwante problematiek van het begrijpen van natuurlijke taal tonen dit centrale probleem aan. Ook voor de bovengenoemde indexen en konkordanties of de automatische inhoudsanalyse blijkt bij nader toezien de definitie van het verband tussen kwaliteit en kwantiteit het centrale probleem. Immers, wat is het nut van frekwentietellingen op een tekst waarin wordt opgenomen dat b.v. de woordvorm "was" 500x voorkomt, wanneer men niet weet of het gaat om de verleden tijd van "is", "bijenwas", "vuil goed" etc. Toch laat men precies dit aspekt buiten beschouwing in de normaal door de computer geleverde indexen op tekstmateriaal. Als noodoplossing voert men dan de konkordantie in. Dit is immers een index waarin behalve de woordvorm ook nog een stukje van de omgevende string wordt meegeleverd. Hierdoor wordt men in de gelegenheid gesteld, zelf de betekenis te bepalen. Alleen is deze oplossing uiterst onelegant: precies dat wat met echte "menselijke" tekstprocessing te maken heeft, namelijk de beslissing over dubbelzinnige woordvormen, wordt met de hand gedaan. Bovendien geschiedt het op uiterst tijdrovende, dikwijls veel output opleverende wijze.
In de automatische inhoudsanalyse heeft men de volgende oplossing voor dit probleem bedacht: Naast de character string analyse wordt ook een zinsontleder ("parser") ingevoerd. Deze parser werkt op de omgevende string en veroorzaakt een aardig percentage van juiste keuze bij dubbelzinnige woordvormen. Het verst ontwikkeld is het model dat werd voorgesteld door Kelly/Stone (Kelly/Stone, 1975). De parser is hier gebaseerd op het model van de z.g. phrase structure grammar, een kontekstvrije beschrijving van de konstituenten struktuur van zinnen in natuurlijke taal. Natuurlijk moest dit model, de z.g. IC grammatika, voor computergebruik worden aangepast, reden waarom ik het "AIC-model" noem (Boot, 1978a). Voordeel van de IC grammatika is dat hij zeer geschikt is voor de computer, in feite is de Bachus Naur Form zo'n IC grammatika. Het nadeel is dat hij slechts geschikt is voor de beschrijving van een gedeelte van de struktuur van natuurlijke talen.
De konklusies die hieruit werden getrokken zijn verschillend van aard. Men heeft gezocht naar modellen die vanuit linguïstisch oogpunt krachtiger zijn. Dit zoeken mondde uit in een pleidooi voor het gebruik van het TGG model voor automatische tekstanalyse. Sinds 1967 (Hays, 1967) echter was al bekend dat TGG modellen niet bruikbaar zijn in automatische tekstanalyse. Een tweede voorstel was dan ook te proberen een eigen model te ontwikkelen voor automatische tekstanalyse. De tweede richting die ontstond ging daarom niet uit van een vooropgezet linguïstisch plan, maar zocht het meer in simulatie-achtige modellen. Binnen het kader van de kursus "text processing" was het met name Champeau die een dergelijk simulatiemodel hanteerde in zijn lezing "Text processing from an Artificial Intelligence point of view". Sinds het verschijnen van "Semantic Information Processing" (Minsky, 1968) ligt de methode van text processing

vanuit de "artificial intelligence" wel ongeveer vast. Men kiest een gebied van onderzoek ("tekst") dat inhoudelijk koherent is, b.v. het oplossen van algebraïsche vergelijkingen die worden gesteld in natuurlijke taal. (zie STUDENT van Bobrow in Minsky, 1968). Via een serie omzettingen ontstaat uit de lopende tekst in natuurlijke taal de algebraïsche vorm. De oplossing van deze vorm levert met de know how in de moderne computer geen moeilijkheid op.

In het kader van STUDENT en dergelijke wordt wel de term "transformation" (Minsky, 1976) gebruikt, een linguïstisch belaste term die men beter door de term "transpositie" zou kunnen vervangen om verwarring te voorkomen. Na Bobrow, Raphael en Minsky kwamen de voorstellen van Winograd en Woods waarin de taalkundige analyse een wat belangrijker plaats inneemt. (Winograd, 1972; Woods, 1970) In deze voorstellen wordt de syntaktische komponent van tekstanalyse apart behandeld, waardoor men meer mogelijkheden krijgt dan bij de strenge transpositietechniek. Hier kan men immers werken met groepen woorden waarvan de lengte niet a priori vastligt. De parser vindt - voor zover gedefinieerd - de funktie van de woordgroepen, b.v. onderwerp, gezegde enz. Deze kunnen op zich weer worden vertaald in funktionele kategorieën in de semantiek van de op te lossen vraagstelling ("tekst"). Champeaux's "tekst" is het uitvoeren van alle rekenmachinefunkties bij aanbieding in het Nederlands.
Zo krijgt men dus vragen als: Wat is vier plus vijf?
Hoeveel is dat oktaal?
Wat is de wortel van vier?
Wat is de som van duizend en honderd gedeeld door het produkt van vier en vijf?
Als parser werd een aangepaste versie van Wood's ATN (Woods, 1970) gebruikt, als taal onvermijdelijk LISP, aangezien de artificial intelligence groep die zich met dit soort kwesties bezighoudt nu eenmaal het basismateriaal in LISP heeft ontworpen en ter beschikking gesteld. In algemene zin kan men dus stellen dat deze tak van "text processing" in Nederland "helemaal bij" is. Jammer is wel dat ook nu weer werd gekozen voor een tekst ("algebraïsche vergelijkingen") waarvan de problematiek al bekend en beschreven was. Dit is temeer jammer, omdat het werk van Bobrow al eens in Nederland, zij het voor een kleiner gedeelte van de "tekst", werd overgedaan (Brandt Corstius, 1970).
De benadering vanuit de artificial intelligence is vooral gericht op semantische problemen. Semantiek is van oudsher, maar in versterkte mate sinds de formalistische linguïstiek à la Chomsky zijn intrede heeft gedaan, een gebied van studie dat raakpunten met wetenschappen vertoont die ver buiten de linguïstiek liggen. Artificial intelligence-problemen worden met name serieus genomen in de psychologie, of zo men wil in de psycholinguïstiek. Hier wordt ook het probleem van het automatisch voortbrengen van zinnen bestudeerd. Wat Nederland en het Nederlands

betreft, wordt op dit gebied boeiend werk verricht door Kempen in Nijmegen. Met name interessant is de door Kempen voorgestelde en door Hoenkamp uitgewerkte parser die op uiterst elegante wijze gebruik maakt van het begrip SCOPE voor de definitie van de konstituenten. Hierin ligt een elegant en kreatief gebruik van het puur informatika-achtige aspekt van de computer en de linguïstische ("kwalitatieve") komponent van tekstbehandeling. Met andere woorden, hier ligt een aanpak van het al zo lang bekende probleem dat de door de moderne linguïstiek gehanteerde modellen voor taalanalyse en taalproduktie niet erg succesvol kunnen worden nagebootst op de computer. De poging van Kempen kan dan ook worden beschouwd als een belangrijke bijdrage aan de modelvorming in de computerlinguïstiek. (zie verder Kempen, 1975; Kempen, 1977).
Behalve vanuit artificial intelligence, een benaderingswijze die men zou kunnen kenmerken als een benadering van bovenaf, kent de computerlinguïstiek als automatische tekstanalyse, ook een benadering van onder af. Gedurende de kursus kwam deze benaderingswijze en de problematiek ervan het duidelijkst tot uiting in de bijdrage van G.v.d. Steen, genaamd "Automatic Text-processing".
De benaderingswijze van onderaf vat een tekst niet in de eerste plaats op vanuit het standpunt van de mens, d.w.z. vanuit het standpunt dat een tekst een kommunikatief instrument is, maar vanuit het standpunt dat een tekst bestaat uit fysieke eenheden waarvan de enkele letter de kleinste eenheid is. De benaderingswijze van onderaf is met andere woorden de kwantitatieve benadering. De meest kruciale vraag is hier: hoe brengt men het kwalitatieve aspekt ("betekenis") binnen?
In zeer algemene zin behandelde van der Steen deze vraag door erop te wijzen dat text processing uit twee hoofddelen bestaat:
- "operations for creating and modifying sequences of characters"
- "operations depending on the structure which has a user in mind".

Daaruit worden de twee volgende desiderata voor text processing afgeleid:
"In automatic text-processing we have to translate:
A. the text into a form convenient for processing (not only into a machine-readable form).
B. the operations in an easy-to-adapt form (not only in an ad hoc computer program)."

Om deze zaken te bereiken stelt v.d. Steen voor, een meer fundamenteel gebruik van kontekst vrije grammatika's te maken bij het design van soft ware packages voor automatische tekstbehandeling. Daarmee sluit hij zich dus aan bij zeer recente ontwikkelingen op dit gebied (zie Boot, 1978a). Een dergelijk gebruik van b.v. BNF-notaties voor de definitie van soft ware is overigens op vergelijkbare gebieden van automatisering minder nieuw. Zo heeft men voor de automatische partituuranalyse het systeem DARMS ontworpen, waarvan de definitie in

BNF notatie is geschied. (Gomberg, 1975) Hoe noodzakelijk het is tot een dergelijke manier van werken als hier door van der Steen voorgesteld over te gaan, bleek ook op het internationale congres voor linguistic and literary computing dat van 2-9 april in Birmingham werd gehouden. (ALLC, 1978) Dit kongres dat ook een Amerikaanse variant heeft kan worden beschouwd als de plaats waar men zich het meest specifiek bezighoudt met wat hier genoemd wordt text processing van onderaf. Ook hier vindt men als bij van der Steen uitgebreide verhandelingen over indexen en konkordanties, in het vokabulair van v.d. Steen "verrijkte" teksten. Onder de verrijking van teksten rekent hij ook de lemmatisering van woordvormen. Dat wil zeggen: het samenbrengen van de verschillende woordvormen uit één paradigma tot één vorm b.v. de stam:

Hij poot aardappelen.
Hij heeft een houten poot.
Hij kan knollen poten.
Hij heeft twee houten poten.

Om te kunnen lemmatiseren moet men kunnen onderscheiden tussen de 2 woordvormen "poot" en "poten". Respektievelijk zijn er twee paren "poot/poten" die ieder onder een lemma "POOT" vallen.
Het is duidelijk dat het probleem dat hier gesteld wordt een probleem van kwalitatieve aard is, het gaat immers om woordbetekenissen. Verder heeft men grammatikale kennis nodig om te onderkennen dat "poten" in het ene geval "meervoud van een zelfstandig naamwoord", in het andere geval "onbepaalde wijs van een werkwoord" is. Deze kennis is niet zonder meer afleidbaar uit de "sequence of characters", maar moet op de een of andere wijze worden ingevoerd. Van der Steen verwees deze kennis naar het gebied "operations depending on the structure which has the user in mind" en demonstreerde vervolgens hoe men zoeksystemen met behulp van de definitie van een tekst als binaire boom aanmerkelijk kan verbeteren tegenover de tot nu toe ontworpen systemen van query languages en batch processing. (Voor een voorbeeld van een dergelijke wijze van definiëren zie Boot, 1978an). Uit deze wijze van benadering kan worden gekonkludeerd dat zaken als lemmatisering worden opgevat als gebruiker-afhankelijk en/of dat het voor onmogelijk wordt gehouden deze soort "kwalitatieve" text processing in het design van de text processor zelf op te nemen. Deze opvatting is algemeen: men houdt het voor onmogelijk de dubbelzinnigheid van woordvormen ook op dit niveau automatisch te doen verrichten. Ook in kringen van de association voor literary and linguistic computing, in kringen van computerlinguïsten, is deze opvatting algemeen. De houdbaarheid van de stelling dat dubbelzinnigheid van woordvormen in geen enkel opzicht automatisch kan worden opgelost, wordt enkel en alleen vanuit Nederland bestreden. Hier werd namelijk een model voor automatische woordklastoekenning ontwikkeld.

Het model wordt gebruikt in Utrecht en Groningen. Uit de publikaties blijkt dat automatische woordklastoekenning als module van een algemene text processor wel mogelijk is. De dubbelzinnigheid die men via woordklastoekenning oplost kan met de volgende voorbeelden worden gedemonstreerd:

Hij poot aardappelen.

Hij heeft een houten poot.

Bij toepassing van het model voor automatische woordklastoekenning (het ICS-model, zie Boot, 1978a), is het mogelijk via computerprogramma's vast te stellen dat de woordvorm "poot" in de eerste zin een werkwoordsvorm is, de woordvorm "poot" in de tweede zin echter een zelfstandig naamwoord. (zie voor het Nederlands, Boot, 1977 en 1977br). Hieruit blijkt dat het probleem van lemmatisering voor dit soort gevallen in feite kan worden opgelost zonder menselijke tussenkomst. Aangezien woordklastoekenning voor zeer veel andere doelstellingen ook kan worden gebruikt, men denke b.v. aan automatische zinsontleding, automatische stilistische analyse, automatische "authorship studies" (ALLC-78) en automatische inhoudsanalyse kan men rustig spreken van een belangrijke Nederlandse bijdrage aan "text processing" wat dit betreft.

Voor de volledigheid moet dan nog worden gewezen op het gebruik van de computer bij klankanalyse van input in natuurlijke taal. Met name in Groningen (De Graaf, 1978) wordt aan dit aspekt van text processing gewerkt.
Uit dit alles kan men de konklusies trekken dat text processing in Nederland een bloeiende tak van wetenschap is die resultaten oplevert die internationaal gezien mogen worden.

Bibliografie

- Addyman: "A language for Literary Data Processing", ALLC Bulletin, 1977.
- ALLC: "Computers in Literary and Linguistic Research". Department of Modern Languages University of Aston in Birmingham, 1978 (pre-prints).
- Bailey, D.: Computer Processing of Text Course Outline. Amsterdam, 1978.
- Bense, M.: Einführung in die informationstheoretische Ästhetik. Hamburg, 1969.
- Boot, M.: "Linguistic Data Structure, Reducing Encoding by Hand and Programming Languages" in Churchhouse/Jones "Computers in Literary and Linguistic Research", Cardiff, 1976.
  : 1977: An experimental Design for Automated syntactic encoding of natural language Texts" in ALLC Bulletin, 1977, pp. 237-248.
  : 1977br: "Een model voor automatische woordklastoekenning" in: "Handelingen van het 31e Vlaams Filologencongres", Brussel, 1977
  : 1978: Comparable Computer Languages for linguistic and literary Computing: Performance" in: ALLC Bulletin, 1978.
  : 1978a: "Homographie. Ein Beitrag zur automatischen Wortklassenzuweisung in der Computerlinguistik". Utrecht, 1978.
  : 1978an: "Key words in natural languages, a problem of system analysis" in Annals of System Research, 6(1977) pp 111-121.
- Brandt Corstius, H.: "Exercises in Computational Linguistics", Amsterdam, 1970.
- Gomberg, D.:"A computer-oriented system for music printing", 1975.
- Graaf, T.de: "Analyse de Voyelles avec des Méthodes Digitales". 9ièmes Journées d'étude sur la Parole, Lannion, 1978.
- Hays, D.: "Introduction to Computational Linguistics", 1967.
- Kelly/Stone: "Computer Recognition of English Word Senses", 1975.
- Minsky, M.: "Semantic Information Processing", MIT, 1968.
- Weizenbaum, J.: "Computer Power and Human Reasoning", 1976.
- Winograd, T.: "Understanding Natural Language" in Cognitive Psychology, 1972.
- Woods, W: "Transition Network grammars for natural language analysis". CACM, 1970, 591-606.