nederlands akoestisch genootschap
NAG

journaal
nr. 134
nov. 1996

Van welk station naar welk station wilt u reizen? REISINFORMATIE VIA EEN GESPROKEN-DIALOOGSYSTEEM

dr. Helmer Strik
Vakgroep Taal & Spraak
Katholieke Universiteit, Postbus 9103, 6500 HD Nijmegen

From which station to which station do you want to travel? TRAVEL INFORMATION BY A SPOKEN DIALOGUE SYSTEM

Summary: This paper reports on the development of a spoken dialogue system for providing information about public transport in the Netherlands. The starting point of our research was a German prototype. It is explained how this German prototype was adapted for Dutch. For the development of the spoken dialogue system a specific approach was chosen to collect the speech material and gradually improve the system. The emphasis in the current paper is on the speech recognition component of the spoken dialogue system.

1 INLEIDING

Onder automatische spraakherkenning (ASH) verstaan we het machinaal omzetten van een spraaksignaal in geschreven tekst. Automatische spraakherkenning kent vele toepassingen, die variëren van het automatiseren van allerlei telefonische diensten die informatie verstrekken over het weer, het telefoonverkeer, de beurs en het openbaar vervoer tot het bedienen van machines door middel van de stem of het telefonisch winkelen en het telefonisch bankieren.

Wat voor vele van deze toepassingen echter nodig is, namelijk het herkennen van spontane spraak, blijkt om verschillende redenen nogal problematisch te zijn. De belangrijkste oorzaak van problemen is de enorme variatie in de manier waarop een en dezelfde uiting gerealiseerd kan worden door verschillende sprekers of door dezelfde sprekers op verschillende momenten of onder verschillende omstandigheden.

Om automatische spraakherkenning toch bruikbaar te maken voor sommige toepassingen, wordt vaak geprobeerd een deel van de problemen uit te sluiten, door de taak eenvoudiger te maken. Zo worden bijvoorbeeld automatische spraakherkenners gebouwd die alleen maar losse woorden kunnen herkennen of de spraak van slechts één spreker. Een voorbeeld van spreker-afhankelijke herkenners zijn dicteersystemen, die dan ook voor iedere gebruiker afzonderlijk getraind moeten worden. Een andere mogelijkheid om de taak makkelijker te maken is door het aantal te herkennen woorden te beperken. Ook voor spraakherkenners met een klein vocabulaire zijn vele nuttige toepassingen te bedenken.

Voor toepassingen van ASH (met name voor informatiediensten) worden vaak de volgende twee soorten interactie strategiëen gebruikt. In een menu-gebaseerde strategie wordt een vast menu doorlopen. Op ieder punt in het menu stelt de computer een vraag, de gebruiker antwoordt (meestal slechts met een of enkele woorden), en de computer

probeert het antwoord te herkennen. Voor meer complexe toepassingen is dit systeem echter niet toereikend. Hiervoor worden dan gesproken-dialoogsystemen (GDS) gebruikt, waarin zowel de gebruiker als de computer het initiatief kunnen nemen. In deze systemen is de spraakherkenner slechts een onderdeel van het gehele systeem.

Op de Vakgroep Taal en Spraak van de Katholieke Universiteit Nijmegen wordt sinds enige tijd gewerkt aan de ontwikkeling van een GDS voor het Nederlands. Als applicatie is in eerste instantie gekozen voor een GDS dat informatie kan verstrekken over het openbaar vervoer. Dit systeem wordt daarom Openbaar Vervoer Informatie Systeem (OVIS) genoemd. Een eerste versie van dit systeem (OVIS-1) is ontwikkeld in het Europese MLAP project MAIS (Multilingual Automatic Inquiry Systems) en het NWO Prioriteitsprogramma Taal- en Spraaktechnologie. Binnen het kader van dit laatste programma en het Europese LE project ARISE (Automatic Railway Information Systems for Europe) zal de komende jaren gewerkt worden aan de verbetering van het GDS.

Over OVIS-1 wordt in deze bijdrage gerapporteerd, waarbij de nadruk ligt op de spraakherkenningscomponent van het systeem. Het artikel is als volgt ingedeeld. Allereerst wordt in paragraaf 2 een overzicht gegeven van het gehele GDS. De ASH component van het GDS wordt beschreven in paragraaf 3. Bij het ontwikkelen van het GDS is uitgegaan van een prototype van een Duits GDS. De eerste stap in het onderzoek was daarom het converteren van het GDS van het Duits naar het Nederlands (paragraaf 4). Meer details over dit systeem en over het converteren van dit systeem van het Duits naar het Nederlands kunnen gevonden worden in Strik et al. (1996a en 1996b; zie ook referenties aldaar). Enkele slotopmerkingen staan in paragraaf 5.

2 HET GESPROKEN-DIALOOGSYSTEEM

In figuur 1 is te zien dat het GDS naast een telefooninterface bestaat uit 4 componenten: de continue-spraakherkenner (CSH), de natuurlijke-taalverwerking (NTV), de dialoog-modellering (DM) en de tekst-naar-spraak omzetter (TNS). De werking van ieder van deze componenten wordt hier kort beschreven. Dit doen we aan de hand van een voorbeeld. Stel dat iemand het GDS opbelt.

Systeem: "*Van welk station naar welk station wilt u reizen?*"

De openingszin is belangrijk, want als gewoon "goedemorgen" gezegd wordt, zijn mensen geneigd om allerlei lange verhalen te gaan vertellen. Maar als het GDS begint met een gerichte vraag, geven de bellers vaker een bondig antwoord.

Gebruiker: "Ik wil morgen van Maarn naar Amsterdam."

De ***CSH*** zet het binnenkomende spraaksignaal om in een reeks van woorden. Hoe dat gebeurt wordt in paragraaf 3 besproken.

De ***NTV*** zoekt naar concepten in de herkende woorden. Dat wil zeggen dat niet wordt geprobeerd om hele zinnen (grammaticaal) te analyseren, maar dat wordt gezocht naar betekenisdragende stukjes tekst in een zin. De NTV zoekt daarbij naar alle veelvoorkomende concepten in de antwoorden van de bellers. Dus er wordt niet alleen gezocht naar concepten die te maken hebben met tijdstippen en plaatsaanduidingen, maar er wordt ook naar beleefdheidsfrases gezocht. Veel mensen zullen hun antwoorden immers beginnen met een uitdrukking als "ik wil graag" of "goedemorgen". Door deze frases als concept op te nemen, wordt het minder waarschijnlijk dat ze verkeerd herkend worden als bijvoorbeeld "Den Haag" respectievelijk "morgen". Dergelijke fouten kunnen voor deze

Figuur 1 Systeemarchitectuur van het gesproken-dialoogsysteem.

applicatie namelijk erg vervelend zijn. Als de NTV bepaald heeft welke concepten er in de zin zitten, wordt de betekenis van de relevante concepten aan de DM doorgegeven.

De ***DM*** slaat deze informatie op en kijkt of er nog informatie ontbreekt. Bijvoorbeeld in het boven genoemde antwoord ontbreekt tijdstip van vertrek of aankomst. Omdat de door de beller gegeven informatie nog niet compleet is, zal het systeem om aanvullende informatie moeten vragen. De DM formuleert de vraag (in tekstvorm).

De ***TNS*** zet deze vraag (tekst) om in spraak. Hiervoor wordt concatenatie van opgeslagen spraakfragmenten gebruikt. Deze gesproken vraag wordt via de telefoon naar de gebruiker gestuurd.

Systeem: "*Hoe laat wilt u morgen van Maarn naar Amsterdam reizen?*"
Door de vraag aldus te formuleren doet het GDS twee dingen tegelijkertijd, namelijk om de ontbrekende informatie vragen en tevens de herkende gegevens aan de gebruiker presenteren. Op deze manier wordt (indirect) gecontroleerd of de herkende concepten correct zijn. De gebruiker heeft nu de mogelijkheid om correcties aan te brengen. Als het GDS bijvoorbeeld 'Baarn' in plaats van 'Maarn' zou hebben herkend, dan had de gebruiker kunnen zeggen: "Ik wil morgenavond om 8 uur vanuit MAARN vertrekken". Hier is 'MAARN' met hoofdletters geschreven om aan te geven dat de gebruiker dit woord waarschijnlijk zal accentueren, wat de herkenning meestal makkelijker maakt. Maar in dit geval waren alle herkende gegevens correct, en dus zal de gebruiker het GDS niet hoeven te corrigeren.

Gebruiker: "*Ik wil morgenavond om 8 uur vertrekken.*"
De CSH herkent weer woorden, en de NTV zoekt naar concepten. In dit geval 'morgen-avond' en 'om 8 uur'. Ofschoon eerder al 'morgen' herkend was, is het concept 'morgen-avond' toch nog belangrijk, namelijk om te weten of de gebruiker om 8 uur ''s morgens' of ''s avonds' wil vertrekken. De DM ziet dat de vraag nu compleet gespecificeerd is, zoekt het antwoord op in de database en formuleert het antwoord (in tekstvorm). Dit antwoord wordt met de TNS weer in spraak omgezet en naar de gebruiker gestuurd. Ten slotte vraagt het systeem of de gebruiker nog andere informatie wil. Als de gebruiker ontkennend antwoordt, bedankt het systeem voor het gebruik, neemt afscheid en wordt de verbinding verbroken. Kort samengevat gaat het dus als volgt:

1. De ***CSH*** herkent woorden in het spraaksignaal.
2. De ***NTV*** zoekt naar concepten in de herkende woorden.
3. De ***DM*** is de centrale module die zorgt voor de interfacing tussen de andere modules: de DM slaat gevonden concepten op, kijkt welke informatie ontbreekt, weet wanneer de informatie compleet is, zoekt dan het antwoord op in de database en formuleert boodschappen voor de gebruiker (in tekstvorm).
4. De ***TNS*** zet deze tekstuele boodschappen van de DM om in spraak, om ze vervolgens via de telefoon naar de gebruiker te sturen.

3 DE CONTINUE-SPRAAKHERKENNER

Vrijwel alle hedendaagse automatische spraakherkenners zijn probabilistische machines: wat ze doen komt neer op het berekenen van de kans dat het binnenkomende geluid veroorzaakt is doordat iemand een rij woorden uitgesproken heeft. De woordenrij die de grootste kans heeft om tot het waargenomen geluid te leiden wordt geselecteerd, en dit is dan de herkende uiting. Voordat een CSH gebruikt kan worden voor spraakherkenning moet hij eerst getraind worden. In training wordt 'geleerd' hoe groot de kans is dat een bepaald geluid bij een bepaalde reeks van woorden hoort. Voor training en herkenning wordt in principe hetzelfde algoritme gebruikt: het Viterbi algoritme. Voordat een getrainde spraakherkenner gebruikt wordt, zal hij vaak eerst getest worden. Een test is een herkenning waarbij achteraf gecontroleerd wordt of de herkende woorden correct zijn. Voor zowel training als test zijn een corpus en een lexicon nodig. In deze paragraaf worden achtereenvolgens lexicon, corpus, Viterbi algoritme, training en herkenning besproken. Allereerst zullen we echter ingaan op de akoestische modellen die gebruikt worden in de CSH.

Tabel 1 De 38 gebruikte basiseenheden, met per basiseenheid een voorbeeld van een woord waarin de klank voorkomt (de desbetreffende klank is vet en schuin weergegeven).

klinkers				*mede-klinkers*					
i	l***ie***p	I	l***i***p	p	***p***ut	n	***n***at	s	***s***ap
e:	l***ee***g	E	l***e***g	b	***b***ad	l	***l***at	z	***z***at
a:	l***aa***t	A	l***a***t	t	***t***ak	r	***r***at	S	***sj***aal
o:	b***oo***m	O	b***o***m	d	***d***ak	L	ba***l***	j	***j***as
y	b***uu***t	Y	p***u***t	k	***k***at	R	ba***r***	x	li***ch***t
2:	d***eu***k	@	g***e***lijk	N	la***ng***	f	***f***iets	h	***h***ad
Ei	w***ij***s	u	b***oe***k	m	***m***at	v	***v***at	w	***w***at
9y	h***ui***s	Au	k***ou***d	n= niet-spraak geluiden					

Op dit moment worden in de CSH 38 basiseenheden gebruikt (zie tabel 1). Voor iedere basiseenheid bestaat een akoestisch model. Een basiseenheid wordt gebruikt om alle niet-spraak geluiden te modelleren, die vaak aanwezig zijn bij telefoongesprekken (bijv. ademhaling, het smakken van de lippen, stofzuiger, baby). Van de 37 andere basiseenheden zijn er 33 fonemen (klanken) van het Nederlands (alles behalve l, r, L en R). In sommige gevallen is het echter beter om meerdere akoestische modellen per foneem op te nemen. Dit is bijvoorbeeld het geval als een foneem op duidelijk verschillende (akoestische) manieren gerealiseerd kan worden. Deze varianten van een foneem worden allofonen genoemd. Zo is het bekend dat een /l/ en een /r/ voor een klinker meestal anders geproduceerd worden dan een /l/ en een /r/ na een klinker. In het huidige systeem zijn dan ook allofonische varianten van de /l/ en /r/ opgenomen (voor een klinker: /l/ en /r/, na een klinker: /L/ en /R/, zie tabel 1). Behalve fonemen en allofonen kunnen echter ook andere basiseenheden gebruikt worden. Zo is het bijvoorbeeld bekend dat de articulatie en akoestische eigenschappen van spraakklanken sterk beïnvloed worden door de omringende klanken (de context). Daarom gebruikt men als basiseenheden ook wel difonen (een sequentie van 2 fonen: een klank met links of rechts een andere klank) of trifonen (een sequentie van 3 fonen: een klank met links en rechts een andere klank). Omdat de basiseenheden zo variabel kunnen zijn, gebruiken we hiervoor de term ‘foon’, die veel neutraler is dan foneem of allofoon.

De CSH kan alleen woorden herkennen die in zijn lexicon staan. In het lexicon staan voor ieder woord twee vormen: [1] De orthografische vorm, oftewel het woord zoals het geschreven wordt. Dit is een reeks van grafemen (= schrifttekens). [2] Een transcriptie in basiseenheden, oftewel het woord zoals het uitgesproken wordt. Dit is een reeks van fonen (= uitspraaktekens), en wordt daarom een foontranscriptie genoemd.

In ons onderzoek wordt de foontranscriptie voor ieder woord uit het lexicon als volgt verkregen. Allereerst wordt gecontroleerd of het woord voorkomt in bestaande corpora die foontranscripties bevatten. Dit zijn het Nederlandse ONOMASTICA corpus (voornamelijk voor eigennamen; zie Konst & Boves 1994) en het CELEX corpus (Baayen et al. 1993). CELEX is een groot elektronisch woordenboek dat o.a. de meest frequente

woorden van het Nederlands bevat. Als het woord niet in deze twee corpora aanwezig is, wordt een grafeem-foon conversieprogramma (Kerkhoff et al. 1984) gebruikt voor het genereren van de foontranscriptie.

De corpora die nodig zijn voor training en test moeten naast de spraaksignalen ook de bijbehorende translitteraties bevatten. Een translitteratie is een orthografische beschrijving van de inhoud van een uiting. Voor een bepaalde uiting kunnen alle woorden uit de translitteratie opgezocht worden in het lexicon, ieder woord wordt vervangen door de bijbehorende foontranscriptie en zo wordt een foontranscriptie voor de hele uiting verkregen.

Vervolgens wordt het Viterbi algoritme gebruikt om de optimale oplijning te vinden tussen het spraaksignaal en de foontranscriptie. Deze oplijning is in feite een segmentering, omdat in het spraaksignaal de grenzen van ieder van de elementen uit de foontranscriptie bepaald worden. Voor de gevonden oplijning berekent het Viterbi algoritme ook een kans. Deze kans kan geïnterpreteerd worden als de kans dat het spraaksignaal en de foontranscriptie bij elkaar horen. Het Viterbi algoritme berekent de oplijning met de grootste kans: de optimale oplijning.

In de trainingsfase worden het Viterbi algoritme en een trainingslexicon gebruikt om een trainingscorpus geheel te segmenteren. Na segmentatie kan voor iedere basiseenheid (foon) opgezocht worden welke stukken spraaksignalen uit het trainingscorpus hierbij horen. Voor iedere foon worden alle bijbehorende stukken spraaksignaal statistisch verwerkt en wordt een stochastisch model berekend: een “Hidden Markov model” (HMM). Hiervoor is het belangrijk dat het trainingscorpus groot genoeg is, om voldoende realisaties van elke foon te hebben. Door alle woorden uit het trainingscorpus op te nemen in het trainingslexicon wordt het trainingscorpus optimaal gebruikt. De berekende foonmodellen worden vervolgens opgeslagen.

Daarnaast worden er taalmodellen getraind. De taalmodellen die in het huidige systeem gebruikt worden zijn een unigram (de kans op ieder woord) en een bigram (de kans op een sequentie van 2 woorden). Deze taalmodellen worden getraind door in de translitteraties te tellen hoe vaak ieder woord respectievelijk iedere combinatie van 2 woorden voorkomt. Deze waarden worden dan gedeeld door het totaal aantal woorden respectievelijk het totaal aantal woordparen.

De CSH bestaat uit de foonmodellen, de taalmodellen en het herkenningslexicon. Bij herkenning weet men vooraf niet wat de gebruikers gaan zeggen. Voor herkenning kan het lexicon dus niet bepaald worden door alle woorden uit een corpus te nemen (zoals dat bij training gebeurd is). Door onderzoek moet het optimale herkenningslexicon bepaald worden. Dit moet niet te klein zijn omdat anders veel woorden niet correct herkend kunnen worden, simpelweg omdat ze niet in het lexicon aanwezig zijn. Maar het mag ook niet te groot zijn, omdat er dan meer verwarringen mogelijk zijn en daardoor een groter aantal woorden verkeerd herkend zal worden. Grofweg kan gesteld worden dat het herkenningslexicon alleen de meest frequente en relevante woorden moet bevatten.

In de herkenningsfase wordt geprobeerd een onbekende uiting te herkennen. Dit gaat ongeveer als volgt. De CSH genereert alle mogelijke sequenties van woorden. Omdat van te voren niet bekend is uit hoeveel woorden een uiting bestaat, is het aantal hypothesen gigantisch groot, zeker als de herkenner over een groot lexicon beschikt. Gelukkig worden alle hypothesen van begin af aan gescoord, dat wil zeggen dat bepaald wordt hoe waarschijnlijk iedere hypothese is gegeven het binnenkomende (onbekende) signaal. Hiervoor wordt weer het Viterbi algoritme gebruikt, dat de optimale oplijning en de bij-

behorende kans bepaalt. Het merendeel van de hypothesen blijkt dan al na een paar stappen zoveel minder waarschijnlijk te zijn dan de favorieten, dat ze zonder enig gevaar geschrapt kunnen worden uit de lijst van mogelijke oplossingen. Op die manier blijven geheugenbeslag en rekentijd voor het scoren van de hypothesen binnen redelijke grenzen. Dit is belangrijk omdat een spraakherkenner in de praktijk 'real-time' moet werken. Uiteindelijk wordt de sequentie van woorden met de grootste kans gekozen, en dit is de beste zin. Ter informatie kan dan nog vermeld worden dat tijdens herkenning vrijwel alleen spectrale informatie gebruikt wordt. Segmentele duur en prosodische informatie worden op dit moment nog niet gebruikt in deze spraakherkenner, zoals dat ook niet het geval is in vrijwel alle andere spraakherkenners.

De zojuist beschreven procedure wordt vaak gebruikt om de beste zin te vinden. Dit is de meest waarschijnlijke sequentie van woorden gegeven het binnenkomende onbekende spraaksignaal. Omdat spraakherkenning nog zeker niet perfect is, zal het correcte woord hier niet altijd tussen zitten. Daarom is de uitvoer van de CSH in het GDS niet alleen de meest waarschijnlijke sequentie van woorden, maar meerdere waarschijnlijke sequenties van woorden. De kans dat het correcte woord daartussen zit is namelijk groter. Deze sequenties van woorden zijn opgeslagen in de vorm van een netwerk van woorden, de zogenaamde woordgraaf. De woordgraaf is de invoer van de NTV, en het is de taak van de NTV om in deze woordgraaf de juiste concepten te vinden.

4 HET GDS OVERZETTEN VAN HET DUITS NAAR HET NEDERLANDS

Hierboven is al vermeld dat het uitgangspunt een Duits prototype voor een GDS was. Dit GDS moest overgezet worden naar het Nederlands. Om een GDS voor een bepaalde applicatie te trainen is een grote hoeveelheid data nodig, in de orde van tienduizenden uitingen die elk uit een of meerdere woorden kunnen bestaan. Deze data moeten eerst verzameld worden. Maar omdat deze data zeer applicatiespecifiek zijn, ontstaat er een probleem: om de applicatie te trainen heb je veel data nodig, maar om deze data te verzamelen heb je eigenlijk de applicatie zelf nodig.

De meest gebruikte oplossing voor dit probleem is een experimenteertechniek die in het Engels 'Wizard-of-Oz experiment' heet en in het Nederlands wel eens 'groen gordijn experiment' genoemd wordt. Het principe van deze techniek is dat de gebruiker denkt met een machine te communiceren terwijl hij/zij in werkelijkheid communiceert met een persoon (achter het 'Wizard-of-Oz scherm' of een 'groen gordijn'). Deze 'Wizard-of-Oz' neemt dan de werking van de CSH, NTV en DM over. Echter niet die van de TNS, omdat een gebruiker bij terugmeldingen in perfect menselijke spraak meteen door zou hebben dat hij/zij niet met een machine aan het communiceren is. Ofschoon deze methode goed werkt heeft hij een belangrijk nadeel, namelijk dat het verzamelen van een grote hoeveelheid data een tijdrovende en kostbare zaak is.

Daarom hebben wij gekozen voor een andere aanpak, die we verder prototypemethode zullen noemen. Deze methode bestaat uit 6 stappen:

[1] maak een versie van het GDS met beschikbare middelen en data
[2] laat dit systeem gebruiken door een beperkte groep gebruikers
[3] gebruik de aldus verkregen data om het GDS te verbeteren
[4] ga naar [5] als het systeem goed genoeg is, anders ga naar [2]

[5] maak de gebruikersgroep geleidelijk groter
[6] stop als het systeem goed genoeg is, anders ga naar [2].

Deze prototypemethode is gebruikt voor alle componenten van het GDS. Aangezien in dit artikel de nadruk op de ASH ligt, zullen we het converteren van de CSH-component het meest uitvoerig bespreken.

Allereerst moest beslist worden welke basiseenheden gebruikt zouden worden in de CSH. In paragraaf 3 is beschreven welke basiseenheden gekozen zijn (zie tabel 1). Vervolgens moest voor ieder van deze basiseenheden een akoestisch model getraind worden. De eerste versie van deze akoestische modellen werd verkregen met behulp van de Polyphone database (Damhuis et al. 1994; den Os et al. 1995). Deze database is geheel opgenomen via de telefoon, en bestaat uit voorgelezen en (semi-)spontane spraak. Voor ieder van 5000 proefpersonen zijn 50 items opgenomen. Vijf van deze 50 items zijn de zogenaamde 'fonetisch rijke uitingen'. Deze vijf uitingen zijn zo samengesteld dat alle fonemen van het Nederlands tenminste één maal voorkomen, en dat de meer frequente fonemen vaker voorkomen. De vijf 'fonetisch rijke uitingen' van 500 proefpersonen (tezamen 2500 uitingen) zijn gebruikt om de eerste foonmodellen te trainen.

In paragraaf 3 hebben we al gezegd dat voor herkenning een lexicon nodig is. Indien een voldoend groot corpus met applicatiespecifieke data aanwezig is, kan het lexicon vaak gebouwd worden door de meest frequente en relevante woorden uit het corpus te nemen. In dit geval hadden we aan het begin van het onderzoek niet de beschikking over applicatiespecifieke data en moesten we het lexicon op een andere manier samenstellen. Om te beginnen hebben we alle relevante woorden uit de database genomen (dit zijn voornamelijk de namen van alle stations in Nederland, want die moeten zeker herkend kunnen worden). Vervolgens hebben we proberen te bedenken wat de verschillende manieren zijn waarop mensen informatie kunnen vragen over treintijden.

We waren ervan overtuigd dat we door middel van introspectie er niet in zouden slagen om alle uitdrukkingen te vinden die mensen gebruiken bij het verkrijgen van dergelijke informatie. Daarom hebben we een spraakloze versie van het GDS gemaakt. Kort gezegd is dit een GDS zonder de spraakinterfaces. Gebruikers konden op een computer dit programma aanroepen, hun zinnen intypen op het toetsenbord en de reacties van het systeem verschenen op het scherm van de computer. Omdat mensen waarschijnlijk hun zinnen anders verwoorden als ze schrijven dan wanneer ze praten, werd de gebruikers gevraagd om hun zinnen zo te formuleren als ze zouden doen wanneer ze deze zinnen zouden uitspreken.

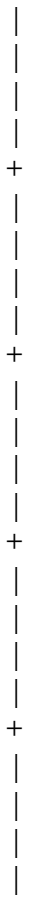
Behalve voor het verzamelen van (geschreven) materiaal, werd deze spraakloze versie tevens gebruikt om de eerste versies van de NTV en DM uit te testen. De sessies met deze spraakloze versie bleken zeer nuttig te zijn. Met behulp van de opgenomen schriftelijke dialogen was het mogelijk om de eerste versie van het GDS op vele plaatsen te verbeteren. Een goed voorbeeld hiervan is dat in de originele Duitse versie er 18 manieren waren om een bevestigend antwoord te geven en 7 manieren om een ontkennend antwoord te geven. Op basis van de opgenomen dialogen zijn er 34 bevestigende en 18 ontkennende antwoorden geformuleerd voor het Nederlands.

Tabel 2 De verschillende stadia in het opbouwen van database Convers-1.

database	aantal uitingen	bron	duur (min.)
DB0	2500	Polyphone	282
DB1	1301	applicatie	41
DB2	5496	applicatie	227
DB3	6401	applicatie	275
DB4	8000	applicatie	355
DB5	10003	applicatie	440
DB6	21288	applicatie	921
DB7	32971	applicatie	1405

De aldus verkregen tweede versie van het GDS werd in december 1995 in het publieke telefoonnetwerk geplaatst. Een kleine groep mensen ontving het telefoonnummer van dit GDS, en werd gevraagd om het systeem geregeld te bellen. Deze mondelinge dialogen werden opgenomen. De aldus verkregen database met applicatiespecifieke data heet Convers-1. De verschillende stadia in het verzamelen van database Convers-1 staan in tabel 2 vermeld. Voor alle opgenomen uitingen werd manueel een translitteratie gemaakt. Vervolgens werd automatisch gecontroleerd welke woorden in deze uitingen nog niet in het trainingslexicon aanwezig waren, de zogenaamde buiten-lexicon-woorden. Op deze manier konden ook typefouten in de orthografische transcripties opgespoord worden. De woorden die niet in het lexicon zaten en die geen typefouten bleken te bevatten werden dan voorzien van een foontranscriptie en werden toegevoegd aan het trainingslexicon, zodat alle verzamelde data gebruikt konden worden voor de training van het systeem. Niet alle nieuwe woorden werden echter toegevoegd aan het herkenningslexicon. Alleen woorden die gerelateerd zijn aan voor deze applicatie relevante concepten werden opgenomen in het herkenningslexicon. De reden hiervoor is dat het opnemen van vele niet essentiële woorden in het herkenningslexicon de gemiddelde prestaties van de CSH verslechtert, zoals in paragraaf 3 al is uitgelegd.

Voor een spraakherkenner is het belangrijk om te weten hoeveel woorden niet in het lexicon aanwezig zijn, de zogenaamde buiten-lexicon-woorden. Immers, woorden die niet aanwezig zijn in het lexicon kunnen ook niet herkend worden. Als maat gebruiken we hier het relatieve aantal buiten-lexicon-woorden: het aantal buiten-lexicon-woorden gedeeld door het aantal uitingen in Convers-1. Het relatieve aantal buiten-lexicon-woorden als functie van het aantal uitingen in Convers-1 is te zien in figuur 2. Allereerst kan opgemerkt worden dat het relatieve aantal buiten-lexicon-woorden klein is. Blijkbaar zijn we er, met de hierboven beschreven methode, goed in geslaagd om een lexicon te maken dat de meeste woorden bevat. Voor een groot deel kan dit verklaard worden door het feit dat we hier te maken hebben met een redelijk beperkt domein. Verder is in figuur 2 te zien dat het relatieve aantal buiten-lexicon-woorden langzaam kleiner wordt als het aantal uitingen groeit van 1301 naar 6401. In het begin is het aantal buiten-lexicon-woorden ongeveer 3.8%. Omdat deze groep gebruikers in veel gevallen dezelfde woorden gebruikt



Figuur 2 Het relatieve aantal buiten-lexicon-woorden (BL-woorden) als functie van het aantal uitingen in Convers-1.

voor het verkrijgen van de gewenste informatie, neemt het aantal onbekende woorden langzaam af. Op dat moment (bij 6401 uitingen) werd het telefoonnummer bekend gemaakt aan een grotere groep gebruikers. De nieuwe mensen gebruikten andere woorden dan de mensen in de eerste groep, en daarom nam het aantal buiten-lexicon-woorden eerst toe. Aangezien ook binnen deze nieuwe, grotere groep vaak dezelfde woorden gebruikt werden, nam het relatieve aantal buiten-lexicon-woorden daarna weer langzaam af tot ongeveer 1.3%.

Een gedeelte van de verzamelde data werd gereserveerd als testcorpus, om de verschillende versies van het GDS te testen. Hier zullen we ons beperken tot de tests voor de CSH. Het gebruikte testcorpus bestond steeds uit dezelfde 500 zinnen, en het gebruikte testlexicon uit alle 299 woorden in die zinnen. Hierbij dient opgemerkt te worden dat het herkenningslexicon uit meer woorden bestaat. Zoals hierboven al vermeld is, is dit herkenningslexicon langzaam gegroeid naarmate de database Convers-1 groter werd. Op dit moment bestaat het herkenningslexicon uit 985 woorden (waaronder bijvoorbeeld alle stationsnamen). Om een spraakherkenner te testen zou je als testlexicon eigenlijk het herkenningslexicon zelf moeten nemen. Niettemin hebben we besloten om dat hier niet te doen. De belangrijkste reden is dat het herkenningslexicon langzaam groter wordt, en bijgevolg zou ook het testlexicon niet constant zijn. Dit maakt het vergelijken van verschillende versies van het systeem onmogelijk.

De rest van de data werd gebruikt als trainingscorpus. Steeds als een voldoende hoeveelheid nieuwe data verzameld was, werd dit (steeds groter wordende) trainingscorpus gebruikt om de CSH opnieuw te trainen. De verschillende databases die gebruikt zijn om de CSH te trainen staan in tabel 2. Vervolgens werd het testcorpus gebruikt om de nieuwe CSH te vergelijken met de oude. Hiervoor kunnen verschillende evaluatiecriteria gebruikt worden. De uitvoer van de CSH is normaal een woordgraaf (WG), maar het is ook mogelijk om alleen de beste zin (BZ) te berekenen. Voor zowel woordgraaf als beste zin kun je vervolgens bepalen hoe vaak woorden niet correct herkend zijn (WFP: woordfouten percentage) en hoe vaak gehele zinnen niet compleet correct herkend zijn (ZFP: zins-fouten percentage). Dit maakt een totaal van 4 evaluatiecriteria: WG-WFP, WG-ZFP, BZ-WFP en BZ-ZFP. Voor verschillende databases (verschillende versies van Convers-1) staan de waarden voor deze 4 evaluatiecriteria in tabel 3. Daarnaast staat in de tweede rij van tabel 3 de testcorpus-perplexiteit. Dit is een maat van de kwaliteit van het taalmodel berekend voor dit specifieke testcorpus. Te zien is dat de perplexiteit langzaam afneemt; het taalmodel wordt dus steeds beter. Ook te zien in tabel 3 is dat de fouten percentages langzaam afnemen (de enige uitzondering is WG-ZFP voor DB7). Omdat de nieuwe CSH meestal beter en nooit slechter presteerde dan de oude, werd in het GDS de oude CSH steeds door de nieuwe vervangen.

Tabel 3 Perplexiteit en fouten percentages (FP in %) voor verschillende versies van het GDS.

database	DB0	DB3	DB4	DB5	DB6	DB7
perplexiteit	317.90	65.84	50.30	48.20	35.24	33.44
WG-WFP	26.16	14.81	11.89	11.35	8.48	8.42
WG-ZFP	42.20	25.80	22.80	20.40	16.60	16.80
BZ-WFP	49.04	26.11	21.45	20.61	16.79	16.37
BZ-ZFP	66.00	37.40	32.20	30.20	25.80	25.60

5 CONCLUSIES

In dit artikel hebben we een beschrijving gegeven van de ontwikkeling van een gesproken-dialoogsysteem voor het verstrekken van openbaar vervoer informatie. Het uitgangspunt was een Duits prototype van een vergelijkbaar systeem. Voor het ontwikkelen van een GDS is veel data nodig. Voor het verzamelen van deze data wordt vaak de zogenaamde ‘Wizard-of-Oz methode’ gebruikt. Wij hebben echter een andere methode gebruikt, de prototype methode. Ons onderzoek heeft laten zien dat de prototypemethode zeer geschikt is voor dit doel. De aldus verzamelde database Convers-1 bestaat inmiddels uit 32.971 applicatiespecifieke uitingen.

Convers-1 werd gebruikt om de ASH component geleidelijk te verbeteren. De hierboven gepresenteerde cijfers laten inderdaad zien dat de fouten percentages van de CSH langzaam afnemen naarmate het aantal uitingen groter wordt. Daarnaast werd de prototypemethode ook gebruikt om de andere componenten van het GDS geleidelijk te

verbeteren, en zo werd het gehele GDS langzaam verbeterd. De DM en TNS hoefden echter nauwelijks verbeterd te worden, de initiële versies waren meteen voldoende goed. Naast de CSH moest aan de NTV wel veel verbeterd worden. De conclusie is dan ook dat het converteren van de DM en TNS veel eenvoudiger was dan het converteren van de CSH en NTV.

6 LITERATUUR

Baayen, R.H., R. Piepenbrock en H. van Rijn, 1993. *The CELEX lexical database* (on CD-ROM). Philadelphia, PA: Linguistic Data Consortium, University of Pennsylvania.

Damhuis, M., T. Boogaart, C. in 't Veld, M. Versteijlen, W. Schelvis, L. Bos en L. Boves, 1994. Creation and analysis of the Dutch Polyphone corpus. *Proceedings International Conference on Spoken Language Processing* (ICSLP) ’94, Yokohama, 1803-1806.

Kerkhoff, J., J. Wester en L. Boves, 1984. A compiler for implementing the linguistic phase of a text-to-speech conversion system. In Bennis H. en W.U.S. van Lessen Kloeke, red., *Linguistics in the Netherlands*, Dordrecht: Foris.

Konst, E.M. en L. Boves, 1994. Automatic grapheme-to-phoneme conversion of Dutch names. *Proceedings International Conference on Spoken Language Processing* (ICSLP) ’94, Yokohama, 735-738.

den Os, E.A., T.I. Boogaart, L. Boves en E. Klabbers, 1995. The Dutch Polyphone corpus. *Proceedings ESCA 4th European Conference on Speech Communication and Technology*: EUROSPEECH 95, Madrid, 825-828.

Strik, H., A. Russel, H. van den Heuvel, C. Cucchiarini, en L. Boves, 1996a. A spoken dialogue system for public transport information, *Proceedings of the Department of Language and Speech* 19, 129-142.

Strik, H., A. Russel, H. van den Heuvel, C. Cucchiarini, en L. Boves, 1996b. Localizing an automatic inquiry system for public transport information. *Proceedings International Conference on Spoken Language Processing* (ICSLP) ’96 Philadelphia.