Universiteit Utrecht

Bachelor Kunstmatige Intelligentie

Eindwerkstuk (7,5 ECTS)

Overzicht van Oude en Nieuwe Technieken voor de Ontwikkeling van Chatbots

R.J. (Rens) van Vliet
Studentnummer: 4022548

Begeleider
Dr. G.A.W. Vreeswijk

Tweede beoordelaar
Dr. J. Korbmacher

20 juni 2018

Samenvatting

Dit werk geeft een overzicht van oude en nieuwe technieken die worden gebruikt in chatbots en beschouwt en vergelijkt deze met oog op toepassingen en toekomstig onderzoek. Chatbots lijken in ten minste twee categorieën verdeeld te kunnen worden: gescripte chatbots die voorgeschreven regels gebruiken en lerende chatbots die gebruik maken van machine learning en data. Door de beperkingen van gescripte chatbots is onderzoek op dit gebied gestagneerd. Lerende chatbots werden lange tijd beperkt door de beschikbaarheid van data. Met de opkomst van microblogdiensten kwam hier een einde aan, waardoor in recenter onderzoek nieuwe technieken toegepast konden worden.

Inhoudsopgave

Hoofdstuk 1

Inleiding

Het idee van denkende en sprekende robots spreekt bij veel mensen tot de verbeelding en komt dan ook vaak terug in sciencefiction. Ook lang voor de ontwikkeling van digitale computers werd al nagedacht over denkende machines (Oppy & Dowe, 2018). Gepaard met de ontwikkeling van computers kwam de vorming van een nieuw onderzoeksgebied: de kunstmatige intelligentie. Turing (1950) veranderde de vraag of machines kunnen denken in een praktisch probleem en daarmee voor sommigen in een uitdaging om een systeem te ontwikkelen dat intelligent genoeg klinkt om een gesprek gaande te kunnen houden (Ji, Lu & Li, 2014).

In de korte geschiedenis is veel vooruitgang geboekt op het gebied van kunstmatige intelligentie. Hoewel het oorspronkelijke doel van dit gebied de ontwikkeling van algemene kunstmatige intelligentie (*artificial general intelligence*) was, is de focus over de jaren verschoven naar praktischer en haalbaarder doelen (Goertzel & Pennachin, 2007). Dit werk richt zich op een vorm van communicatie tussen mens en computer en in het bijzonder op chatbots die communiceren in de vorm van schrift.

Complexe dialoogsystemen die gebruik maken van logica en kennis hebben inmiddels de weg naar de praktijk gevonden (Lester, Branting & Mott, 2004). Deze systemen hebben toepassingen in klantenservice, zoals hulp bij websitenavigatie of gebruik van een webshop of door het beantwoorden van vragen. Hierbij kunnen dialoogsystemen zowel als aanvulling als vervanging voor personeel dienen, met als resultaat toegenomen klanttevredenheid of kostenbesparing.

Er is ook een directe aanpak van Turings uitdaging: in plaats van het maken van algemene kunstmatige intelligentie worden chatbots gemaakt die net genoeg kunnen om een gesprek te kunnen houden. Deze chatbots hebben niet intelligentie, maar de schijn ervan ten doel (Wallace, 2009). Dergelijke chatbots zijn meestal meer een vermakelijke dan een praktische functie, maar kunnen mogelijk inzicht geven in de mechanismen die spelen in gesprekken in natuurlijke taal (Ji e.a., 2014).

Deze tekst is bedoeld als introductie voor diegenen die geïnteresseerd zijn in wetenschappelijk onderzoek op het gebied van chatbots en voor iedereen die

geïnteresseerd is in het ontwikkelen van chatbots als hobby of met een praktisch doel. Voor een aspirant-chatbotbouwer is het aan te raden om niet direct in de praktijk te werk te gaan, maar zich eerst te verdiepen in de theorie. Door te onderzoeken welk werk al is gedaan en welke aanpakken het beste werken kan een hoop moeite worden bespaard. Overzicht kan bij deze oriëntatie ontbreken. Literatuuronderzoek kan zeer tijdrovend zijn en de drang om zo snel mogelijk aan de slag te gaan is hoog; een inleidende tekst die overzicht schept is daarom wenselijk.

Dit werk heeft als doel op basis van wetenschappelijke bronnen inzicht te geven in de werking van chatbots en de ontwikkelingen die deze hebben doorgemaakt. Hiervoor worden na een kort overzicht van de historische ontwikkeling een aantal recente voorbeelden toegelicht. Met chatbots worden in dit werk steeds systemen bedoeld die door middel van tekst in natuurlijke taal in gesprek kunnen gaan met gebruikers. Het overzicht dat dit werk schept is niet volledig, maar dient om gevoel te geven voor de in de technieken die worden gebruikt in het ontwikkelen van chatbots. De hoop is dat aspirant-chatbotbouwers dit werk als beginpunt kunnen gebruiken en aan de hand van de uitgewerkte voorbeelden een keuze kunnen maken voor het soort technieken waarin zij zich willen verdiepen. Om te helpen in deze keuze worden de voorbeelden aan het einde van hoofdstukken steeds vergeleken.

Er zijn vergelijkbare overzichten van relevante technieken, maar dit werk voegt iets toe door in het bijzonder in te gaan op chatbots en hierbij steeds wetenschappelijke onderbouwing te geven. Jurafsky en Martin (2009) geven een uitvoerig overzicht van natuurlijke taalverwerking, waaronder veel technieken en ontwikkelingen, maar hierin wordt niet in detail ingegaan op chatbots. Daarnaast zijn er inmiddels interessante nieuwe mogelijkheden waarover beschouwend onderzoek ontbreekt. Er is ook de mogelijkheid om hulp te vragen via online fora en er zijn tal van blogs die nuttig kunnen zijn, maar het overzicht dat hier geschept wordt is vaak oppervlakkig en veelal slecht onderbouwd.

Opbouw

Ter introductie wordt in Hoofdstuk 2 een beschrijving gegeven van de ontwikkelingen in de korte geschiedenis van chatbots. In Hoofdstuk 3 wordt aandacht gegeven aan enkele mechanismen van gesprekken die van belang kunnen zijn om begrip te krijgen van de werking en het ontwerp van chatbots. Chatbots lijken in ten minste twee categorieën verdeeld te kunnen worden: gescripte chatbots en lerende chatbots. Aangezien de eerste chatbots gescript waren en lerende chatbots een latere ontwikkeling zijn, worden gescripte chatbots eerst behandeld in Hoofdstuk 4 met chatbot ELIZA als uitgangspunt. In Hoofdstuk 5 wordt aandacht gegeven aan alternatieven voor gescripte chatbots en in Hoofdstuk 6 worden lerende chatbots besproken aan de hand van drie chatsystemen voor korte-tekstconversatie. In de conclusie in Hoofdstuk 7 wordt een samenvatting gegeven en wordt de lezer een keuze voorgelegd tussen de behandelde technieken voor een mogelijk project of onderzoek.

Hoofdstuk 2

Ontwikkeling

Dit hoofdstuk beschrijft zeer kort enkele ontwikkelingen die belangrijk zijn geweest voor chatbots. Dit heeft slechts ten doel enige context te geven aan de voorbeelden die in de rest van dit werk zullen worden besproken.

2.1 Turingtest

Al voor het bestaan van computers werd er gefilosofeerd over denkende en sprekende machines. Zo beweert René Descartes in het *Discours de la Méthode* dat het onmogelijk is dat een machine passend antwoord zou geven op iets dat er tegen wordt gezegd (Oppy & Dowe, 2018). Met de ontwikkeling van digitale computers kwam het idee van een computer die zou kunnen denken steeds dichterbij. Alan Turing (1950) stelde als alternatief op de vraag of machines kunnen denken een experiment voor dat tegenwoordig bekend staat als de Turingtest. De Turingtest is een experiment waarbij een mens en een computerprogramma beide in gesprek gaan met een derde, die moet bepalen welke van de twee de mens is en welke de computer. In dit experiment staat imitatie centraal, waardoor de vraag niet langer is of machines kunnen denken, maar of ze een gesprek gaande kunnen houden en daarmee de indruk kunnen wekken menselijk te zijn.

Hoewel Turing de test niet bedoelde als intelligentietest, wordt deze vaak wel zo geïnterpreteerd en toegepast. Het is daarom ook een omstreden onderwerp. Een implementatie van de test in wedstrijdvorm, de Loebnerprijs, lijkt bijvoorbeeld voornamelijk het gebruik van trucjes te stimuleren en weinig bij te dragen aan de ontwikkeling van kunstmatige intelligentie (Oppy & Dowe, 2018). Ondanks de kritiek is het duidelijk dat de Turingtest zeer invloedrijk is geweest. De Turingtest wordt door sommigen immers beschouwd als het ultieme doel in onderzoek naar communicatie in natuurlijke taal tussen mens en computer (Ji e.a., 2014).

2.2 Opkomst chatbots

Niet lang na Turings beroemde artikel werden systemen ontwikkeld die konden omgaan met natuurlijke taal. In 1966 presenteerde Joseph Weizenbaum de chatbot ELIZA. ELIZA gebruikt *pattern matching* om tekst die wordt ingevoerd door een gebruiker om te zetten in een antwoord. Elke handeling die het programma doet is voorgeschreven. ELIZA gebruikt geen semantische verwerking en is niet in staat tot leren, maar wekte bij sommige gebruikers de illusie van menselijke eigenschappen (Weizenbaum, 1966). Er werden enorme verwachtingen geschapen over de toekomst van kunstmatige intelligentie, tot weerzin van Weizenbaum (1976). In navolging van ELIZA kwamen vergelijkbare chatbots zoals PARRY (Colby, Hilf, Weber & Kraemer, 1972), Elizabeth (Millican, 2002) en ALICE (Wallace, 2009). Door de technische en praktische beperkingen die gepaard gaan met deze gescripte chatbots bleven veel beloftes echter uit.

Het op gang komen van de ontwikkeling van computers ging gepaard met toenemend onderzoek op het gebied van theoretische informatica. Dit onderzoek resulteerde in inzichten die belangrijk zouden zijn voor de ontwikkeling van chatbots. Zo werd onder andere onderzoek gedaan naar formele talen en werden probabilistische taalmodellen ontwikkeld. Gelijktijdig werd onderzoek gedaan naar de structuur van taal: zowel op gespreks- als zinsniveau. In het volgende hoofdstuk zal extra aandacht worden gegeven aan een aantal gespreksanalytische inzichten die relevant zijn voor chatbots. Met de ontwikkeling van systemen die konden omgaan met formele logica en een toegenomen begrip van natuurlijke taal ontstond de mogelijkheid om taalverwerking en logica te combineren in computerprogramma's (Winograd, 1972).

De combinatie van natuurlijke taalverwerking en formele logica vormt de basis van veel dialoogsystemen (Reiter, 1994), die daardoor betekenis kunnen geven aan nieuw ingevoerde tekst. Deze nieuwe informatie kan met behulp van logica gecombineerd worden met kennis uit een database om een antwoord te vormen. Deze systemen zijn hierdoor geschikter voor praktische toepassingen dan de gescripte chatbots die eerder ontwikkeld werden. Daarnaast maken veel dialoogsystemen gebruik van spraakherkenning en -synthese. Het ontwikkelen van dergelijke systemen is echter niet eenvoudig door de noodzaak van een databank met geformaliseerde informatie, waardoor ze vaak beperkt worden tot een enkel domein.

2.3 Machine learning

Door de beperkingen van complexe dialoogsystemen en simpele gescripte chatbots rees ook de interesse in lerende chatsystemen. *Machine learning* wordt namelijk in toenemende mate toegepast op taalverwerkingsproblemen. Er is met het toepassen van statistische methoden bijvoorbeeld veel succes geboekt op het gebied van machinevertaling (Jurafsky & Martin, 2009). Sterk toegenomen gebruik van het internet maakt het bovendien veel makkelijker om de datasets die hiervoor nodig zijn te verzamelen. De grote populariteit van microblogdiensten

maakte een eerder ongekend grote hoeveelheid gespreksdata beschikbaar (Ritter, Cherry & Dolan, 2010). Hierdoor ontstond de mogelijkheid om machine learning te gebruiken om datagedreven chatsystemen te ontwikkelen.

Ook wanneer machine learning wordt gebruikt, is het ontwikkelen van chatbots niet probleemloos. Om van verzamelde gespreksdata te kunnen leren is het van belang om te kunnen meten hoe geschikt een antwoord is. Dit blijkt niet makkelijk te automatiseren, waardoor tijdrovende, handmatige beoordeling wordt gebruikt, zoals later besproken zal worden.

Hoofdstuk 3

Gespreksanalyse

In dit hoofdstuk wordt kort aandacht gegeven aan de structuur van gesprekken en de regels waaraan mensen zich (onbewust) houden, zodat hiermee rekening gehouden kan worden bij het ontwerpen en begrijpen van chatbots. Dit hoofdstuk is deels gebaseerd op Jurafsky en Martin (2009), die een overzicht geven van de structuur in gesproken conversatie tussen mensen. Voor verschillende gespreksvormen gelden verschillende regels; hier zal alleen aandacht worden gegeven aan de mechanismen die relevant zijn voor gesprekken in geschreven taal. Bovendien wordt in dit hoofdstuk zoals in de rest van dit werk steeds uitgegaan van precies twee gesprekspartners.

3.1 Taalhandelingen

In een gesprek heeft bijna elke uiting een functie en een doel; met spreken of schrijven wordt niet alleen taal geproduceerd, maar ook een handeling verricht die een bepaald effect op de wereld ten doel heeft (Wittgenstein, 1953). Een uiting wordt daarom ook wel een taalhandeling genoemd. Austin (1962) breekt taalhandelingen in drie delen:

- Locutie of directe taalhandeling: de geproduceerde taal op zich, bijvoorbeeld in de vorm van geluid of schrift.
- Illocutie of indirecte taalhandeling: de soort handeling die wordt gedaan.
- Perlocutie: het beoogde effect van de uiting.

Met taalhandeling wordt soms alleen de illocutie bedoeld. Deze indirecte taalhandelingen zoals vragen, verwelkomen, beloven, enzovoort kunnen worden ingedeeld in verschillende categorieën (Searle, 1975). Opeenvolgende uitingen voltrekken vaak taalhandelingen die op elkaar aansluiten (Schegloff, 1968). Vraag-antwoordparen zijn een belangrijk voorbeeld van deze afstemming van taalhandelingen: wanneer iemand een vraag stelt is er de ongeschreven regel en de verwachting dat de ander hier antwoord op geeft.

3.2 Maximes van Grice

In gesprekken worden niet alleen verwachtingen maar ook veel informatie impliciet gelaten. Om deze implicaties te kunnen ontrafelen heeft de ontvanger regels nodig waarvan verwacht kan worden dat de zender zich eraan houdt. De ontvanger gebruikt deze regels bewust of onbewust bij het interpreteren van een uiting, erop rekenende dat de zender zich ook aan deze regels houdt. Grice (1975, 1978) geeft hiervoor vier maximes:

1. *Maxim of quantity*
2. *Maxim of quality*
3. *Maxim of relevance*
4. *Maxim of manner*

Het *maxim of quantity* houdt in dat er niet meer of minder informatie moet worden gegeven dan nodig is. Het *maxim of quality* stelt dat de zender voldoende overtuigd moet zijn dat deze informatie ook waar is. Het *maxim of manner* gaat niet over beleefdheid, maar over de netheid van een uiting. Dit houdt onder andere in dat het helder en niet ambigu moet zijn. Het *maxim of relevance* houdt in dat uitspraken relevant moeten zijn. De combinatie van deze maximes stelt een toehoorder in staat om betekenis te vinden die niet expliciet aanwezig was in een uitspraak.

3.3 Grounding

Zoals een uiting kan worden gezien als handeling, kan een gesprek gezien worden als een vorm van samenwerking tussen de deelnemers. In gesprekken is er een grote behoefte aan bevestiging, juist ook omdat er veel informatie impliciet wordt gelaten. Gesprekspartners zijn steeds bezig om overeenkomst te vinden in overtuigingen en ideeën, de zogeheten *common ground* (Stalnaker, 1978). Dit wordt behaald door middel van *grounding*, waarbij de ontvanger duidelijk maakt dat de uiting ontvangen en begrepen is. *Grounding* heeft verschillende vormen die door Clark en Schaefer (1989) worden verdeeld in categorieën met verschillende sterktes. In chats kan grounding de vorm hebben van een korte bevestiging zoals 'ja' of 'oh'. Een sterkere vorm van *grounding* is het parafraseren van de uiting. De sterkste is volgens Clark en Schaefer het letterlijk herhalen van (delen van) de ontvangen uiting. Veel chatbots kopiëren tekst en maken hiermee gebruik van deze laatste vorm van *grounding*, wat mogelijk bijdraagt aan de illusie van begrip die sommige gebruikers ervaren.

3.4 Beschouwing

In gesprekken worden regels, verwachtingen en veel informatie impliciet gehouden. Aangezien deelnemers aan gesprekken zich hier vaak niet bewust van zijn,

kan het storend zijn wanneer een chatbot zich niet aan dezelfde regels houdt. Zo blijkt dat *grounding* van belang is voor dialoogsystemen en dat het ontbreken ervan gebruikers verward (Stifelman, Arons, Schmandt & Hulteen, 1993; Yankelovich, Levow & Marx, 1995). Het kan dus nuttig zijn om de onderliggende mechanismen van conversaties in overweging te nemen bij het ontwerp van dialoogsystemen en chatbots. Sommige van deze mechanismen zijn makkelijker toe te passen dan andere. Zo blijkt *grounding* redelijk eenvoudig te behalen door woorden te kopiëren. Voor zowel gescripte als lerende chatbots kan het nuttig zijn rekening te houden met het soort spraakhandeling dat wordt gedaan. Een systeem kan bijvoorbeeld worden geprogrammeerd om een vraag altijd te laten volgen door een antwoord. Voor het ontrafelen van impliciete informatie lijkt een systeem nodig dat niet alleen met logica kan omgaan, maar ook een databank heeft met voldoende kennis van de wereld. Dit laatste is een belangrijke horde voor de ontwikkeling van dialoogsystemen, zoals later besproken zal worden.

Hoofdstuk 4

Gescripte chatbots

In dit hoofdstuk worden drie voorbeelden van gescripte chatbots beschreven om een indruk te geven hoe deze werken en waartoe deze in staat zijn. Met gescripte chatbots worden hier chatsystemen bedoeld die werken met een verzameling regels en voorgeschreven antwoorden in een script. Aan de hand van de beschreven voorbeelden worden de sterke en zwakke punten van gescripte chatbots uitgelegd. Tot slot worden enkele uitdagingen bij het schrijven van scripts genoemd.

4.1 ELIZA

Misschien wel het belangrijkste voorbeeld van gescripte chatbots en chatbots in het algemeen is ook één van de oudste: ELIZA werd in 1966 gepresenteerd door Joseph Weizenbaum als instrument voor onderzoek op het gebied van communicatie met computers in natuurlijke taal (Weizenbaum, 1966). ELIZA is een redelijk eenvoudig programma, maar werd door sommige gebruikers al kenmerken zoals begrip toegeschreven. Gezien de praktische aard van dit werk zal in deze sectie veel aandacht worden gegeven aan de technische werking van ELIZA en minder aan de psychologie achter deze illusie.

Scripts & regels

ELIZA is een programma dat aan de hand van voorgeschreven regels kan antwoorden op ingevoerde zinnen. Het programma bevat zelf geen regels, maar laadt deze uit een script. Een script is een bestand met regels die aangeven hoe de input verwerkt wordt tot output. Deze regels bepalen hoe woorden in de invoer vervangen, verwijderd, toegevoegd of verplaatst worden. Daarvoor heeft iedere regel een input- en een outputpatroon. Het inputpatroon geeft aan hoe de input moet worden opgedeeld. Het outputpatroon geeft aan hoe de onderdelen van de ontlede input worden gebruikt in de output. Een voorbeeld van een mogelijke regel voor ELIZA is als volgt:

(IK BEN 0, WAAROM DENK JE DAT JE 3 BENT)

De 0 in het inputpatroon staat voor een willekeurig aantal woorden; er kan ook aangegeven worden dat er een specifiek aantal woorden moet staan. De 3 in het outputpatroon staat voor het derde deel van de input, het deel dat is aangegeven door 0. De regel kopieert dit deel van de input en plaatst dit in het outputpatroon om zo de output te vormen. De input "IK BEN EEN BEETJE ZIEK" resulteert dus in de respons "WAAROM DENK JE DAT JE EEN BEETJE ZIEK BENT". Regels staan in het script ingedeeld bij steekwoorden die worden gebruikt om te bepalen welke regel moet worden toegepast. Daarnaast bevat een script ook de zin waarmee ELIZA een gesprek opent.

ELIZA is niet in staat tot semantische verwerking van de input, maar het is wel mogelijk om in het script een basale verbinding tussen bepaalde woorden vast te leggen. Dit kan door middel van categorieën. Zo kunnen bijvoorbeeld de woorden 'vader' en 'moeder' beiden gekoppeld worden aan de categorie 'familie'. Als 'vader' of 'moeder' geregistreerd wordt in de input zal ELIZA de regels voor 'familie' kunnen gebruiken.

Algoritme

De input wordt woord voor woord doorzocht naar steekwoorden. Daarbij kunnen bepaalde woorden al worden vervangen, zodat bijvoorbeeld de eerste en tweede persoon verwisseld worden. Om het zoeken efficiënter te laten verlopen, zijn alle steekwoorden opgeslagen in een boom, die wordt geconstrueerd bij het laden van het script. Dit voorkomt dat alle bekende woorden moeten worden nagelopen wanneer de input een onbekend woord bevat. De positie van ieder woord in de boom is namelijk gebaseerd op dat woord zelf, waardoor slechts één tak doorlopen hoeft te worden. Met een score die in het script is vastgelegd, wordt bijgehouden wat het belangrijkste woord in de zin is. Vervolgens wordt bij het belangrijkste steekwoord een passende regel gezocht. De regels zijn geordend, waarbij specifiekere regels boven algemene regels staan. Ware dit niet zo, dan zouden algemenere regels altijd de voorkeur krijgen over specifiekere, ondanks dat deze misschien wel beter aan zouden sluiten op de input.

Wanneer er geen steekwoorden gevonden worden, kan er gebruik worden gemaakt van het geheugen. De geheugenfunctie slaat alle input met een bepaald patroon op in een lijst, zodat deze later verwerkt kan worden tot antwoord. Op deze manier houdt ELIZA het gesprek gaande, ook als er geen geschikte reactie is op de gegeven input.

DOCTOR

Mogelijk het bekendste script voor ELIZA is DOCTOR, dat een Rogeriaanse psychotherapeut nabootst. In Rogeriaanse psychotherapie staat zelfactualisering centraal (Lang & Molen, 2016). Het idee is dat de patiënt zelf tot inzichten komt, mits daarvoor de ruimte wordt gegeven. De therapeut stelt zich daarom accepterend op en grijpt zo min mogelijk in. Dit is voor ELIZA een ideale rol,

omdat de therapeut weinig aan het gesprek hoeft worden toegevoegd (Weizenbaum, 1966). Het script geeft het initiatief steeds terug aan de gebruiker en laat deze zo het hart luchten.

ELIZA werd bij gebruik van het DOCTOR-script door de eerste gebruikers al begrip toegeschreven: één gebruiker vroeg zelfs om een privégesprek met ELIZA Weizenbaum, 1966. Weizenbaum (1976) en Kuipers, McCarthy en Weizenbaum (1976) gaan verder in discussie over de psychologie en filosofie achter deze misleiding. Gezien het meer praktische doel van dit werk wordt hier verder niet op ingegaan, maar het feit dat deze misleiding plaatsvindt is wel interessant. De illusie laat namelijk zien dat ELIZA met een passend script menselijke interacties kan nabootsen, ondanks het ontbreken van interne logica of kennis van de wereld.

4.2 ALICE

ALICE is een chatbot die speciaal is ontworpen voor deelname aan een door Alan Turing beschreven experiment (Wallace, 2009). In dit experiment zou onderzocht worden hoe vaak een ondervrager het geslacht van een man en een vrouw correct kan achterhalen als één van hen zich voordoet als het andere geslacht (Turing, 1950). Dit experiment moet niet verward worden met wat bekend staat als de Turingtest, die in hetzelfde artikel werd beschreven, waarbij een computer en een persoon allebei proberen een ondervrager te overtuigen dat ze een mens zijn. ALICE is dus niet alleen ontworpen met als doel over te komen als mens, maar in het bijzonder als vrouw. (Wallace, 2009). ALICE bleek ook succesvol te zijn in de minder specifieke taak: in 2000, 2001 en 2004 won het de Loebnerprijs voor de meest menselijke chatbot.

ALICE gebruikt de opmaaktaal AIML (een variant op XML) die speciaal is ontwikkeld als standaard voor chatbotscripts (Wallace, 2009). Inputpatronen, bijbehorende antwoorden en optionele context zijn georganiseerd in categorieën, die opgeslagen zijn in een boomstructuur. De in- en outputpatronen lijken op die van ELIZA. De outputpatronen kunnen daarnaast ook informatie bewaren, andere programma's aanroepen of andere categorieën toepassen door recursie. Als context kan het vorige bericht van de chatbot gebruikt worden of een onderwerp dat bestaat uit een verzameling categorieën. Wallace (2009) beschrijft twee benaderingen voor het schrijven van AIML-scripts. De eerste probeert te voorspellen waar een gesprek over zal gaan en de tweede kijkt terug op gesprekken om te zien waar verbetering kan worden behaald. De terugkijkende benadering kan ook deels automatisch worden uitgevoerd. Dit kan bijvoorbeeld door gebruikte inputpatronen met een wildcard aan te vullen met waargenomen woorden om het inputpatroon specifieker te maken (Wallace, 2009). Voor dit nieuwe inputpatroon moet handmatig een nieuw antwoord worden gemaakt.

Antwoorden worden zoals bij ELIZA opgezocht door de input te vergelijken met patronen in een script. Het algoritme zoekt met backtracking een zo lang mogelijke reeks van woorden die overeenkomt met de input. Het langste overeenkomende patroon wordt gezien als het beste en het bijbehorende outputpatroon

wordt gebruikt om het antwoord te vormen (Shawar & Atwell, 2002).

4.3 Elizabeth

Elizabeth is een variant op ELIZA die werd ontwikkeld om als voorbeeld te dienen bij introductiecursussen over kunstmatige intelligentie (Millican, 2002) en later ook voor onderzoek is gebruikt (Kolb, Afrika & Millican, 2006). Naast de gewijzigde opmaak en indeling van scriptdocumenten voegt Elizabeth onder andere de mogelijkheid toe om het script automatisch aan te passen tijdens het gesprek. Er kunnen namelijk regels worden toegevoegd en verwijderd als onderdeel van een reactie. Als de gebruiker 'mijn zus' zegt, kan bijvoorbeeld een reactie worden toegevoegd die niet van toepassing zou zijn als het systeem niet wist of de gebruiker een zus had.

Daarnaast heeft Elizabeth een uitgebreidere geheugenfunctie dan ELIZA. De geheugenfunctie kan worden gebruikt om specifieke woorden met een unieke index op te slaan zodat hier later naar kan worden verwezen. Dit staat bijvoorbeeld toe een gemoedstoestand in het script te verwerken of correcte voornaamwoorden te gebruiken als er wordt verwezen naar eerder genoemde onderwerpen.

Tot slot staat Elizabeth het gebruik van recursie toe in het vormen van antwoorden. De input kan hierdoor onder andere worden gesplitst om ieder onderdeel apart te behandelen als input en de antwoorden samen te voegen. In de instellingen van Elizabeth kan worden aangegeven of het herhaald toepassen van regels is toegestaan. Daarbij kan ook worden ingesteld hoeveel mutaties maximaal toegepast kunnen worden, om oneindige recursie te voorkomen. Deze toevoegingen maken Elizabeth veel flexibeler dan ELIZA (Shawar & Atwell, 2002). Met het juiste script kan Elizabeth dan ook worden gebruikt voor zinsontleding, het oplossen van rekensommen of resolutie in propositielogica (Millican, 2002).

4.4 Beschouwing

ELIZA toonde aan dat met een simpel chatsysteem bij gebruikers de illusie kan worden gewekt dat ze spreken met een persoon (Weizenbaum, 1966). Systemen die na ELIZA kwamen bouwden verder op dit succes om deze illusie nog sterker te maken (Wallace, 2009). Er is echter ook kritiek op het misleiden van gebruikers als hoofddoel. Zo zou de Loebnerprijs kandidaten stimuleren om trucjes te gebruiken (Floridi, Taddeo & Turilli, 2009). Opmerkelijk is bijvoorbeeld dat ELIZA's DOCTOR-script en andere gescripte chatbots vaak (delen van) een bericht herhalen, wat als sterkste vorm van grounding (Sectie 3.3) wordt gezien. Wallace (2009) omarmt imitatie als doel en merkt op dat

> [...] the theme and strategy of deception and pretense [...] can be traced through the history of Artificial Intelligence research.

Opvallend is dat juist Joseph Weizenbaum, die ELIZA creëerde, één van de felste critici op chatsystemen en kunstmatige intelligentie in het algemeen was (Weizenbaum, 1966, 1976). Hij was bijvoorbeeld fel gekant tegen het toeschrijven

van menselijke kenmerken aan computers. Hij merkte op dat sommige gebruikers zelfs na uitleg over de werking van ELIZA geloofden dat het systeem hen begreep en vergeleek deze irrationaliteit met het geloof in waarzeggers.

Het om de tuin leiden van gebruikers is niet het enige waar gescripte chatbots goed in zijn. De uitbreidingen van geheugenfuncties en context, het toepassen van recursie en het automatisch aanpassen van het script op gespreksonderwerpen maken ALICE en Elizabeth veel flexibeler dan ELIZA. Dit maakt het mogelijk deze chatbots te gebruiken voor praktischer doeleinden dan het misleiden van gebruikers. Kolb e.a. (2006) verzamelen informatie over klanttevredenheid over telecombedrijven door Elizabeth te programmeren om door te vragen na open vragen. Dit doorvragen kan belangrijk zijn, omdat het eerste antwoord op een open vraag vaak oppervlakkig is. Er kan op deze manier niet alleen meer informatie worden verkregen, maar een gesprek kan hierdoor ook natuurlijker lopen (Kolb e.a., 2006).

Het voorschrijven van scripts maakt het mogelijk dat er een zekere logica achter een antwoord zit die ontbreekt in de lerende chatsystemen die later besproken zullen worden. Scripts zijn echter ook de zwakte van deze chatbots. Sommige systemen geven beperkte mogelijkheden voor het schrijven van een script. Zo kan ELIZA maar op één zin of zinsdeel antwoord geven en negeert alle tekst die voor of na interpunctie komt (Weizenbaum, 1966). Elizabeth en ALICE hebben beperkingen voor gebruikte tekens, zoals regels over het gebruik van hoofdletters (Millican, 2002; Wallace, 2009).

Het schrijven van een script blijkt lastig en tijdrovend. Hierdoor kan een gescripte chatbot vaak maar een beperkt domein behandelen. Als de gebruiker over een onderwerp begint waarvoor de chatbot niet is geprogrammeerd, wordt dit door het gebruik van vage of ontwijkende antwoorden al snel duidelijk (Floridi e.a., 2009). Eén manier om dit probleem aan te pakken is door de taak van het schrijven van scripts te verdelen over verschillende personen: scripts voor ALICE worden geschreven door honderden vrijwilligers (Wallace, 2009). Shawar en Atwell (2003) maakten een script voor ALICE door gesprekken uit het Dialogue Diversity Corpus (Mann, 2003) om te zetten in AIML-formaat. Dit corpus is een verzameling corpora van transcripties uit allerlei verschillende situaties. Shawar en Atwell hoopten hiermee een chatbot te trainen die kan omgaan met een groter domein. Het omzetten van tekst uit corpora in een geschikt script verliep niet zonder problemen: er stonden notities tussen gesproken tekst, sprekers waren soms lang aan het woord en er werd door elkaar heen gesproken. Dergelijke onregelmatigheden werden verwijderd en de paren van uitspraken werden omgezet in input- en outputpatronen. Het resulterende script bevatte geen inputpatronen met wildcards, waardoor de input precies moest overeenkomen met een inputtekst in het script. Hiermee werd één van de sterkste instrumenten van gescripte chatbots, patroonherkenning, onbenut gelaten. Shawar en Atwell (2003) noemden het vervangen van minst gebruikte woorden door patronen wel als mogelijkheid voor vervolgonderzoek. In de volgende hoofdstukken zal worden besproken hoe corpora op effectievere wijze gebruikt kunnen worden om chatsystemen te ontwikkelen.

Hoofdstuk 5

Alternatieven voor scripts

In het eerste deel van dit hoofdstuk wordt een korte indruk gegeven van de structuur en werking van dialoogsystemen. Het modelleren van volledige gesprekken blijkt complex en er is voor het trainen een zeer beperkte hoeveelheid data beschikbaar. Daarom wordt in het tweede deel van dit hoofdstuk ingegaan op *short-text conversations* waarbij slechts één bericht-antwoordpaar wordt behandeld.

5.1 Dialoogsystemen

Dialoogsystemen (ook wel *conversational agents*) benaderen gesprekken op gestructureerde wijze door gebruik te maken van logica en geformaliseerde kennis. Dergelijke systemen hebben veel toepassingen, maar individuele systemen zijn meestal gebonden aan één specifieke taak (Lester e.a., 2004). Een veel gebruikt voorbeeld is een systeem dat reizigers helpt bij het boeken van vluchten (Jurafsky & Martin, 2009). Virtuele assistenten zoals Alexa en Siri kunnen ook onder deze noemer worden geschaard. Dialoogsystemen hebben veelal een vergelijkbare opbouw (Jurafsky & Martin, 2009; Lester e.a., 2004; Rambow, Bangalore & Walker, 2001):

1. De input wordt geïnterpreteerd en geformaliseerd.
2. Een dialoogmanager bepaalt het soort antwoord en de informatie die hierin moet worden verwerkt.
3. Het antwoord wordt omgezet in natuurlijke taal.

Voorafgaand aan het interpreteren wordt vaak een spraakherkenner gebruikt en na het genereren van een antwoord wordt dit vaak omgezet in geluid. In deze tekst zijn deze onderdelen niet van belang en het modulaire ontwerp van dialoogsystemen staat het focussen op andere onderdelen ook toe (Rambow e.a., 2001; Reiter, 1994). Daarnaast kan een taakmanager een deel van de functie van

de dialoogmanager overnemen door aan de hand van een strategie het huidige doel aan te geven (Jurafsky & Martin, 2009).

De verschillende onderdelen van een dialoogsysteem moeten foutloos en efficiënt informatie kunnen uitwisselen (Lester e.a., 2004). Er wordt daarom intern een formalisatie van informatie gebruikt. Dit kan bijvoorbeeld door waarden op te slaan in objecten of aan de hand van predicatenlogica of een semantisch web (Jurafsky & Martin, 2009). Zo kan een reisplanner vluchtinformatie zoals vertrek- en aankomsttijden en -locaties en informatie over de gebruiker bevatten. De input wordt geïnterpreteerd, in de gekozen formalisatie omgezet en opgeslagen, zodat nieuwe informatie kan worden gebruikt. Doordat alle informatie in dezelfde vorm wordt omgezet kunnen vervolgens de berekeningen voor het vormen van een antwoord worden uitgevoerd.

Aan de hand van de huidige doelen en de beschikbare informatie maakt de dialoogmanager een besluit over de te nemen handeling. Dit kan een vraag om informatie of het beantwoorden van een vraag zijn, maar er kunnen ook taken worden uitgevoerd die geen spraakhandelingen zijn. In een reisplanner zou bijvoorbeeld een ticket geboekt kunnen worden na bevestiging van de gebruiker. De strategieën die gebruikt worden door de dialoog- of taakmanager kunnen worden geleerd met *reinforcement learning* om handmatig ontwerp te omzeilen (Misu, Georgila, Leuski & Traum, 2012; Schatzmann, Weilhammer, Stuttle & Young, 2006). Het kan bij het bepalen van een strategie van belang zijn dat een systeem steeds het initiatief houdt of dit ook aan de gebruiker geeft. Nadat de dialoogmanager heeft bepaald welke informatie het antwoord moet bevatten, wordt deze door de taalgenerator omgezet in natuurlijke taal.

Door de opbouw van dialoogsystemen en de noodzaak van een databank met geformaliseerde informatie, zijn deze systemen vaak zeer domeinspecifiek (Litman, Singh, Kearns & Walker, 2000; Misu e.a., 2012). Daarnaast ontbreekt vaak de specifieke data in corpora die nodig is voor het trainen van (de onderdelen van) dialoogsystemen (Rambow e.a., 2001). In de rest van dit hoofdstuk wordt daarom een simpeler probleem geïntroduceerd waarbij slechts één keer antwoord wordt gegeven.

5.2 Short text conversation

Mens-computerconversatie wordt gezien als één van de grootste uitdagingen in kunstmatige intelligentie (Ji e.a., 2014; Shang, Lu & Li, 2015; Wang, Lu, Li & Chen, 2013). Om dit probleem behapbaar te maken, wordt onderzoek soms beperkt tot *Short Text Conversation* (korte tekst-conversatie, STC). Ji e.a. (2014) definiëren STC als

> *[...] one round of conversation via two short texts, with the former being a message from [a] human and the latter being a response to the message given by the computer.*

De doelstelling bij STC is het automatisch geven van een zo goed mogelijk antwoord op één bericht, zonder verdere context. Onderzoek over dit probleem

zou inzicht kunnen geven in conversaties in natuurlijke taal (Ji e.a., 2014).

Een belangrijk voordeel van de beperking tot STC is de beschikbaarheid van data. Corpora die voor STC-onderzoek worden gebruikt, zijn gebaseerd op microblogdiensten zoals Twitter (Ritter e.a., 2010) of het Chinese Weibo (Wang e.a., 2013). Deze datasets kunnen zeer groot zijn in vergelijking met eerder beschikbare datasets en kunnen dan ook zeer gevarieerde onderwerpen omvatten. Hierdoor zijn ze uiterst geschikt voor de ontwikkeling van chatsystemen in een open domein (Wang e.a., 2013). Daarnaast beperken de microblogdiensten het aantal karakters per bericht, waardoor chatachtige gesprekken ontstaan en het aannemelijk is dat er per bericht maar één spraakhandeling wordt gedaan (Ritter e.a., 2010).

Ritter e.a. (2010) creëerden een dataset van Twitterberichten en -antwoorden. Hiervoor werden actieve gebruikers van de publieke tijdlijn geselecteerd. Gedurende 2 maanden werden alle gesprekken waarin deze gebruikers antwoordden opgeslagen. De berichten werden verder niet verwerkt of gefilterd.

Wang e.a. (2013) verfijnden hun dataset van Weiboberichten en -antwoorden wel en beperkten daarnaast de bronnen van de berichten. De berichten van 3200 voldoende actieve gebruikers en de antwoorden daarop werden 2 maanden lang verzameld. De gevolgde gebruikers waren voornamelijk professoren, onderzoekers en studenten op het gebied van natuurlijke taalverwerking. Berichten en antwoorden die te kort waren, werden verwijderd om het aantal te algemene of vage berichten terug te dringen. Slechts een beperkt aantal antwoorden op ieder bericht werd bewaard, omdat latere antwoorden steeds minder relevant zijn voor het originele bericht. Daarnaast werden mogelijke advertenties verwijderd door identieke antwoorden op verschillende berichten te verwijderen. Interpunctie en emoticons werden verwijderd en tot slot werden woorden gesegmenteerd met ICTCLAS (Zhang, Yu, Xiong & Liu, 2003), omdat het begin en einde van woorden in Chinees schrift ambigu kan zijn door het ontbreken van spaties.

Van een deel van de verzamelde berichten en antwoorden maakten Wang e.a. (2013) een train- en testset. Een deel van de berichten werden gekoppeld aan ieder ongeveer 30 nieuwe antwoorden uit de data. Van ieder bericht-antwoordpaar werd handmatig beoordeeld of het antwoord passend was. Antwoorden werden gelabeld aan de hand van 3 criteria:

- Semantische relevantie: Het antwoord moet inhoudelijk relevant zijn en de onderwerpen moeten overeenkomen of aan elkaar gerelateerd zijn.
- Logische consistentie: Het antwoord mag niet in strijd zijn met het originele bericht.
- Afstemming van spraakhandelingen: De spraakhandeling van het antwoord moet passen op die van het bericht (zoals besproken in Sectie 3.1).

Het labelen van bericht-antwoordparen is tijdsintensief en werd daarom maar op een kleine selectie toegepast.

Door microblogdiensten zoals Twitter en Weibo te gebruiken als bron kunnen zeer grote datasets worden gemaakt (Ritter e.a., 2010; Wang e.a., 2013). De

beschikbaarheid van deze datasets maakt het mogelijk om verschillende machine learning-technieken toe te passen op STC (Ji e.a., 2014; Ritter, Cherry & Dolan, 2011; Shang e.a., 2015; Wang e.a., 2013). Deze datasets zijn echter niet vrij van beperkingen. Zo voldoen de originele, door mensen geschreven antwoorden vaak niet aan de eisen die onderzoekers stellen aan geschikte antwoorden, omdat ze te vaag, algemeen of irrelevant zijn (Ji e.a., 2014; Shang e.a., 2015). Daarnaast is het taalgebruik in berichten van Twittergebruikers vergelijkbaar met dat in smstaal en bevat de tekst vaak grammatica-, spel- of typefouten (Ritter e.a., 2010). Om te voorkomen dat ongewenst gedrag wordt aangeleerd moeten dergelijke ongeschikte voorbeelden eerst worden gefilterd (Ji e.a., 2014; Shang e.a., 2015).

Wang e.a. (2013) kozen ervoor gebruikers te volgen die allemaal actief zijn in hetzelfde onderzoeksgebied. Dit kan invloed hebben op de onderwerpen die in de resulterende dataset behandeld worden, waardoor ook de prestaties van systemen die getraind worden aan de hand van deze data kunnen worden beïnvloed (Wang e.a., 2013). Deze beperking van het domein is opmerkelijk, omdat Ritter e.a. (2010) juist potentie zien om deze corpora toe te passen in natuurlijke taalverwerkingsproblemen in een open domein.

Het kan bij het samenstellen van nieuwe datasets van belang zijn rekening te houden met de karakterlimiet, die voor Twitter inmiddels is verhoogd tot 280 en voor Weibo zelfs tot 2000. Het chatachtige karakter van de gesprekken kan hierdoor veranderd zijn en het is ook minder aannemelijk dat een bericht slechts één spraakhandeling bevat (Ritter e.a., 2010).

Hoofdstuk 6

Lerende chatbots

In dit hoofdstuk wordt ingegaan op lerende chatbots. Met lerende chatbots worden hier systemen bedoeld die gebruik maken van technieken uit *machine learning* en natuurlijke taalverwerking om uit data te leren gesprekken te houden. In het bijzonder worden hier drie verschillende benaderingen voor Short-Text Conversation behandeld aan de hand van Ji e.a. (2014), Ritter e.a. (2011) en Shang e.a. (2015). Tot slot worden de drie genoemde systemen met elkaar vergeleken om een indruk te geven van de successen en uitdagingen van ieder systeem.

6.1 Information retrieval

Eén benadering die gretig gebruik maakt van deze hoeveelheid data is de Information Retrieval-methode, waarbij antwoorden uit de data gezocht en gekopieerd worden. Om op elk mogelijk bericht antwoord te kunnen geven zijn namelijk zeer veel mogelijke antwoorden nodig. Information Retrieval-systemen (informatie opzoeken, IR) worden gebruikt om teksten te vinden die overeenkomen met een bepaalde zoekopdracht (Voorhees, 1999). Voorbeelden hiervan zijn zoeksystemen in bibliotheken of internetzoekmachines zoals Google. Op basis van kenmerken van de zoekopdracht, worden documenten gezocht die deze kenmerken (zoals bepaalde woorden) delen. Een vergelijkbare aanpak kan worden gebruikt om antwoorden te zoeken voor *short-text conversation*. In deze sectie wordt de IR-methode voor STC beschreven op basis van Ji e.a. (2014). De gevonden antwoorden zijn exacte kopieën van (door mensen geschreven) antwoorden uit de dataset. Om tijdrovende analyse van alle mogelijke antwoorden te voorkomen wordt eerst een kleine groep kandidaat-antwoorden gezocht op basis van simpele kenmerken. Deze kandidaat-antwoorden worden vervolgens geordend naar geschiktheid en de beste kandidaat wordt gekozen als antwoord.

Selectie kandidaat-antwoorden

Voor het kiezen van de kandidaat-antwoorden worden door Ji e.a. (2014) drie lineaire vergelijkingsmodellen gebruikt. De tekst in berichten en antwoorden wordt voor deze berekeningen omgezet in vectoren. Hierbij wordt *Term Frequency-Inverse Document Frequency* (termfrequentie-inverse documentfrequentie, TF-IDF) gebruikt. TF-IDF is een manier om in een getal uit te drukken hoe belangrijk een woord in een tekst is, in vergelijking met andere teksten in een corpus (Rajaraman & Ullman, 2011). Alle berichten en antwoorden worden omgezet in vectoren van TF-IDF-waarden. Als kandidaat-antwoorden selecteren Ji e.a. de tien antwoorden die het hoogst scoren voor ieder van de volgende drie basiskenmerken:

1. De cosinus van de hoek tussen de vectoren van het nieuwe bericht en berichten uit de data. Het resultaat van deze berekening geeft aan hoeveel woorden de teksten delen, waarbij sommige woorden zwaarder wegen dan anderen.

2. De cosinus van de hoek tussen de vectoren van het nieuwe bericht en antwoorden uit de data.

3. Het inproduct van de vectoren van het nieuwe bericht en antwoorden uit de data geprojecteerd op een vectorruimte met minder dimensies.

Als twee teksten woorden delen die in het corpus niet vaak voorkomen, wordt dit gezien als teken dat ze semantisch aan elkaar gerelateerd zijn. Het eerste model gebruikt deze aanname om relevantie tussen twee berichten te bepalen. De gedachte achter het tweede model is dat antwoorden die woorden delen met het bericht relevanter zijn dan antwoorden die dat niet doen. Het derde model heeft als doel een semantische vergelijking tussen de teksten te geven, ook als er niet veel gedeelde woorden zijn. De projecties die hiervoor worden gebruikt, moeten eerst worden geleerd zoals bij Wu, Lu en Li (2013). De gevonden kandidaten worden geordend aan de hand van complexe kenmerken, die in de volgende secties worden besproken, om te bepalen welke het beste antwoord is.

Vertaalmodel

Ji e.a. (2014) gebruiken een vertaalmodel om het mogelijk te maken dat antwoorden die weinig woorden gemeen hebben met het bericht toch hoog worden gescoord. Het vertaalmodel leert de kansen voor voorkomens van woorden in het antwoord afhankelijk van woorden in het bericht en omgekeerd. Het vertaalmodel leert ook de kansen voor identieke woordparen. Ji e.a. (2014) gebruiken de toolkit GIZA++ (Och & Ney, 2003) voor het leren van het vertaalmodel. Het model is gebaseerd op Xue, Jeon en Croft (2008), die een vergelijkbaar model toepassen op vraag-antwoordparen. De kansen van alle woordparen die voorkomen tussen het bericht en een kandidaat-antwoord worden gecombineerd om aan te geven hoe sterk het verband tussen bericht en antwoord is. Xue e.a. (2008) geven als voorbeeld dat wanneer een vraag gaat over ‘vreemdgaan’ het

antwoord vaak het woord 'vertrouwen' bevat. Ondanks het ontbreken van identieke woorden kan door het vertaalmodel een antwoord met 'vertrouwen' goed scoren als antwoord op een bericht over 'vreemdgaan'.

Diep neuraal netwerk

Om gebruik te maken van eigenschappen van tekst die niet goed te vangen zijn door alleen lineaire vergelijkingsmodellen, gebruiken Ji e.a. (2014) een diep neuraal netwerk zoals beschreven door Lu en Li, 2013. Hierbij worden zogenaamde lokale beslissingen genomen op basis van enkele woorden uit het bericht en het kandidaat-antwoord. Deze lokale beslissingen worden genomen aan de hand van lineaire projecties op een vectorruimte die zoals bij het derde model voor kandidaatselectie dienen als semantische representaties. De lokale beslissingen worden in het neurale netwerk gecombineerd om een beslissing te nemen over de geschiktheid van het hele antwoord.

De parameters van dit model worden geleerd door geschikte en ongeschikte bericht-antwoordparen te vergelijken. Een bericht-antwoordpaar uit de data wordt gekozen als geschikt voorbeeld en de combinatie van hetzelfde bericht met een willekeurig antwoord wordt gekozen als ongeschikt voorbeeld.

Onderwerpwoorden

Om niet alleen semantische relevantie, maar ook overeenkomst in onderwerp te meten, gebruiken Ji e.a. (2014) een model voor het herkennen van onderwerpwoorden Het model gebruikt simpele kenmerken van woorden in een bericht (bijvoorbeeld hoe vaak ze herhaald worden) om te bepalen of ze onderwerpwoorden zijn. Om het model te trainen werden woorden in berichten en antwoorden eerst met de hand gelabeld. LIBLINEAR (Fan, Chang, Hsieh, Wang & Lin, 2008) werd gebruikt om met deze gelabelde data het model te leren. Dit model wordt gebruikt om voor berichten en antwoorden een vector te maken vergelijkbaar met de vectoren op basis van TF-IDF. Om de overeenkomende onderwerpwoorden te bepalen wordt de cosinus van de hoek tussen de vectoren van beide teksten berekend.

Overige kenmerken

Ji e.a. (2014) testten verschillende combinaties van kenmerken voor het ordenen. Naast de beschreven complexere modellen gebruikten zij voor het ordenen van de kandidaat-antwoorden ook een aantal simpelere kenmerken. Deze kenmerken zijn: de lengte van de langste gemeenschappelijke reeks tekens tussen bericht en antwoord, het aantal overeenkomende woorden, het percentage overeenkomende woorden, de totale inverse documentfrequentie (IDF) van gemeenschappelijke woorden, en de gemiddelde IDF van gemeenschappelijke woorden. Samen met de drie basismodellen die gebruikt worden voor kandidaatselectie, vormen deze kenmerken de basis voor ieder model dat zij testten.

Ordening en resultaten

Het ordenen van kandidaat-antwoorden gebeurt op basis van een totaalscore die gelijk is aan de gewogen som van de scores voor alle gebruikte kenmerken. De gewichten worden geleerd met behulp van een lineair *RankingSVM* (ordenende steunvectormachine) (Herbrich, Graepel & Obermayer, 1999). De gelabelde bericht-antwoordparen worden gebruikt als trainingsdata. Er wordt steeds een passend en een niet-passend paar vergeleken, zodat het ordenmodel leert hoe alle features bijdragen aan de geschiktheid van een antwoord.

Verschillende combinaties van kenmerken werden vergeleken om invloed van ieder kenmerk te meten. Per getraind IR-model werd bepaald hoe vaak het hoogst beoordeelde antwoord passend is en hoe goed de hele lijst geordend is. Hieruit blijkt dat van de complexe modellen het vertaalmodel het meeste bijdraagt aan de prestaties van het het volledige model, het onderwerpwoorden-model blijkt tweede in bijdrage en het diepe neurale netwerk draagt het minste bij. Het basismodel geeft in 57,4% van de tests een passend antwoord de hoogste score. Het model dat gebruik maakt van alle kenmerken geeft in 63,7% van de gevallen een passend antwoord.

6.2 Statistical machine translation

In tegenstelling tot de IR-methode kan de Statistical Machine Translation-methode (statistische machinevertaling, SMT) wel antwoorden genereren. In deze sectie wordt de SMT-methode uitgelegd aan de hand van Ritter e.a. (2011). Om SMT toe te passen op short-text conversation wordt het antwoord beschouwd als een vertaling van het bericht. In plaats van paren van teksten in twee verschillende talen wordt geleerd van bericht-antwoordparen.

Het vertalen van berichten in antwoorden wordt gedaan aan de hand van een taalmodel en een vertaalmodel. Deze modellen geven de kans op een bepaald woord in een antwoord en worden gebruikt om stap voor stap het meest waarschijnlijke antwoord te genereren. Hiervoor wordt met de decoder van SMT-systeem Moses (Koehn e.a., 2007) een *beam search* uitgevoerd, waarbij steeds nieuwe woorden worden toegevoegd en een vast aantal van de meest waarschijnlijke antwoorden wordt bewaard.

Het taalmodel schat de kans op een antwoord onafhankelijk van het bericht. Dit model heeft als doel een grammaticaal correcte zin te vormen. Het taalmodel wordt getraind op basis van n-grams uit de antwoorden in de dataset. Het geeft daardoor aan hoe antwoorden er in het algemeen meestal uitzien.

Het vertaalmodel schat de kans op een antwoord gegeven het bericht en omgekeerd met als doel een relevant en passend antwoord te geven op het bericht. Dit gebeurd aan de hand van fraseparen die een frase uit een bericht en een 'vertaling' voor het antwoord bevatten. Het geeft daarmee aan hoe een relevant antwoord in het bijzonder eruitziet.

Ritter e.a. (2011) probeerden GIZA++ (Och & Ney, 2003) te gebruiken voor het koppelen van frasen, maar concludeerden dat dit in veel ruis resulteert, door-

dat er vaak geen een-op-een-relatie is tussen frasen in bericht-antwoordparen. Voor het bepalen van fraseparen werd daarom gekeken naar alle mogelijke paren van frasen met vier of minder woorden. Van ieder paar werd geteld hoe vaak de ene frase voorkomt met of zonder een voorkomen van de andere en omgekeerd. Met Fishers exacte toets, die ook goed werkt als er weinig voorkomens zijn (Johnson, Martin, Foster & Kuhn, 2007), werd vervolgens bepaald hoe sterk het verband is tussen voorkomens van de twee frasen.

Om de efficiëntie van de berekeningen aan de hand van het vertaalmodel te bevorderen werden fraseparen met een lage correlatie verwijderd. Johnson e.a. (2007) tonen aan dat dit vaak niet resulteert in prestatieverlies en anders slechts in zeer beperkte mate. Fraseparen met dezelfde woorden in bericht en antwoord behalen meestal de hoogste score, waardoor het vertaalmodel als antwoord vaak het bericht herhaalt (Ritter e.a., 2011). Fraseparen waarvan één van de frasen een onderdeel is van de andere (inclusief paren van identieke frasen) werden daarom verwijderd. De overige paren werden in score beboet voor overeenkomende woorden.

Ritter e.a. (2011) gebruikten Amazons Mechanical Turk voor handmatige beoordeling bij het testen van het Statistical Machine Translation-model. Beoordelaars kregen steeds een bericht en twee antwoorden van verschillende systemen te zien en hadden de opdracht om het beste antwoord te kiezen. Ze kregen hierbij de instructie dat een passend antwoord over hetzelfde onderwerp moest gaan als het bericht en dat het goed moet klinken.

Het SMT-model werd vergeleken met twee simpele IR-modellen (vergelijkbaar met de lineaire vergelijkingsmodellen die door Ji e.a. (2014) werden gebruikt voor kandidaatselectie), willekeurige antwoorden die minstens twee keer voorkomen in de data, een model dat fraseparen vindt met GIZA++ en echte antwoorden uit de data. Er werd steeds bepaald welk antwoord door een meerderheid van de beoordelaars als beter werd beschouwd. De antwoorden van het SMT-model werden vaker als het betere beoordeeld dan die van alle andere modellen. Het verloor wel vaker tegen de door mensen geschreven antwoorden uit de data, hoewel in 14,5% van de vergelijkingen tussen het SMT-model en originele antwoorden een antwoord van het SMT-model als beter werd beschouwd.

6.3 Neural responding machine

Hoewel met SMT antwoorden kunnen worden gegenereerd, heeft het belangrijke beperkingen. Het voornaamste hiervan is het feit dat het niet is ontworpen voor de semantische inequivalentie tussen bericht en antwoord, maar juist van equivalentie uitgaat. De *Neural Responding Machine* (neurale antwoordmachine, NRM) werd bedacht door Shang e.a. (2015) om de beperkingen van Information Retrieval en Statistical Machine Translation te omzeilen. De NRM vindt net als de SMT-methode zijn oorsprong in technieken uit machinevertaling (Auli, Galley, Quirk & Zweig, 2013; Bahdanau, Cho & Bengio, 2014; Kalchbrenner & Blunsom, 2013; Sutskever, Vinyals & Le, 2014). Gegeven een bepaalde context bepaalt de NRM de kansen op bepaalde woorden in het antwoord om met een

beam search een antwoord te genereren. Eerst wordt het inputbericht door een encoderend neuraal netwerk omgezet in een verborgen vector, die dient als de semantische representatie van de input. Daarna wordt deze vector door een decoderend neural netwerk gebruikt voor het bepalen van de kans op een nieuw woord.

Er werden door Shang e.a. (2015) drie encoders getest: een lokale, een globale en een hybride. Bij alle encoders worden verborgen toestanden gebruikt voor het bepalen van de context die als input dient voor de decoder. De verborgen toestand wordt steeds berekend als functie van de voorgaande toestand en het meest recente woord. De globale encoder geeft als output alleen de verborgen toestand van het laatste woord. De decoder krijgt hierdoor steeds dezelfde context. Deze laatste toestand heeft daarom als taak het hele bericht samen te vatten. De lokale encoder geeft als context een gewogen som van alle verborgen toestanden. De gewichten worden gegeven door het aandachtssignaal, dat een functie is van de verborgen toestanden van de encoder en de decoder. Hierdoor kan deze encoder op dynamische wijze nadruk leggen op de belangrijkste woorden in de input. De hybride encoder combineert de globale en lokale encoders door de verborgen toestanden simpelweg te concateneren. De decoder krijgt in dit geval zowel de globale samenvatting als de lokale context voor het genereren van het antwoord.

De decoder is een recurrent neuraal netwerk met als taak het bepalen van de kans op een woord gegeven de context, een verborgen toestand en het voorgaande woord. De verborgen toestand wordt berekend als functie van de vorige toestand, het vorige woord en de huidige context. Voor het trainen van de NRM worden bericht-antwoordparen uit de data gebruikt. Het doel van het trainen is het maximaliseren van de kans op een antwoord gegeven het bijbehorende bericht. De rol van het globale deel van de hybride encoder blijkt niet genoeg nadruk te krijgen wanneer het als geheel wordt getraind. De parameters van de globale en lokale NRMs worden daarom gebruikt als beginwaarden voor de hybride versie, die daarna nog wordt getraind om de parameters te verfijnen.

Shang e.a. (2015) lieten vijf ervaren Weibogebruikers de antwoorden van de Neural Responding Machine-varianten, een Information Retrieval-model en een Statistical Machine Translation-model labelen. Antwoorden werden gelabeld met een score van 0 (niet-passend), 1 (neutraal) of 2 (passend). Dit gebeurde op basis van de vijf volgende criteria:

1. Grammatica en vloeiend taalgebruik: De antwoorden dienen in natuurlijke taal te zijn en mogen geen grammaticale fouten bevatten.

2. Logische consistentie: Antwoorden mogen niet strijdig zijn met het bericht.

3. Semantisch relevantie: Antwoorden moeten relevant zijn voor het bericht.

4. Scenario-afhankelijkheid: Antwoorden mogen afhangen van een bepaald scenario, maar mogen niet in strijd zijn met de eerste drie criteria.

5. Algemeenheid: Antwoorden mogen algemeen zijn zolang ze niet in strijd zijn met de eerste drie criteria.

Als een antwoord niet voldeed aan de eerste drie criteria moest het als niet-passend worden gelabeld. Als criteria 4 en 5 van toepassing waren, moest een bericht als neutraal worden gelabeld. Om als passend beoordeeld te worden moest een antwoord dus voldoen aan de eerste drie criteria, maar niet scenario-afhankelijk of te algemeen zijn. Van de drie NRMs bleek de hybride versie het beste te scoren, de lokale als tweede en de globale het slechtst. In 39,3% van de door de hybride NRM gegenereerde antwoorden werd als passend gelabeld en 23,6% als ongeschikt. Het IR-model scoorde slechter met 29,8% geschikt en 31,5% ongeschikt. Het Statistical Machine Translation-model scoorde het slechtst met 5,6% geschikte antwoorden en 74,4% ongeschikt.

6.4 Beschouwing

In deze sectie worden de behandelde systemen voor short-text conversation van Ji e.a. (2014), Ritter e.a., 2011 en Shang e.a., 2015 met elkaar vergeleken aan de hand van het modelontwerp en de methoden die gebruikt worden voor het meten van de resultaten. Hierbij wordt steeds gekeken waar de sterke en zwakke punten van ieder systeem liggen. Ook wordt een aantal interessante resultaten genoemd en toegelicht.

6.4.1 Ontwerp

Eén van de belangrijkste verschillen tussen de behandelde systemen is dat de Information Retrieval-methode de antwoorden kopieert uit de data, waar de Statistical Machine Translation-methode en de Neural Responding Machine de antwoorden zelf genereren. Beide benaderingen hebben voor- en nadelen. Door antwoorden te kopiëren kan de kans worden vergroot dat het antwoord grammaticaal correct is en geen spel- of stijlfouten bevat (Shang e.a., 2015). Hiermee wordt echter wel aan flexibiliteit ingeleverd ten opzichte van gegenereerde antwoorden. Deze hebben namelijk als voordeel volledig aangepast te zijn op het bericht, zodat de relevantie zo hoog mogelijk is (Ritter e.a., 2011).

Een belangrijk deel van de mate waarin een antwoord past op een bericht is de semantische relevantie (Ji e.a., 2014; Shang e.a., 2015). Het lijkt daarom belangrijk om inzicht te hebben in de semantische lading van een bericht. Dit kan bijvoorbeeld behaald worden door de betekenis van het bericht samen te vatten in een abstracte vertegenwoordiging of door voorkomens van specifieke woorden te vergelijken. De besproken systemen pakken dit probleem ieder op een eigen manier aan.

Ji e.a. (2014) proberen met de IR-methode de betekenis van een tekst onder andere te vangen door de vectorvertegenwoordiging te projecteren op een vectorruimte met minder dimensies, die daardoor dient als semantische ruimte. Kandidaat-antwoorden worden ook geselecteerd op basis van overeenkomende woorden. Om te voorkomen dat alleen antwoorden met veel overeenkomende

woorden hoge scores krijgen, worden extra modellen gebruikt om overeenkomst in betekenis te bepalen. Zo wordt een vertaalmodel gebruikt om veel voorkomende fraseparen te vinden, zoekt een neuraal netwerk naar semantische hiërarchie en worden onderwerpen van berichten en antwoorden vergeleken. Deze informatie wordt echter niet gebruikt om zelf een antwoord te genereren, maar om er één te kiezen uit een lijst van kandidaten. Dit resulteert in een beperking in de flexibiliteit van antwoorden ten opzichte van methoden die tekst genereren (Ritter e.a., 2011).

Ritter e.a. (2011) pogen met de SMT-methode de relevantie tussen bericht en antwoord te vangen in (niet-identieke) fraseparen. In vergelijking met het neurale netwerk dat Ji e.a. (2014) als één van de kenmerken voor de IR-methode gebruiken, wordt relevantie alleen op lokaal niveau bepaald. Er wordt dus geen semantische hiërarchie of samenvatting van de hele tekst gebruikt. Bij het toepassen van machinevertaling op STC ontbreekt de semantische equivalentie die er wel is bij vertaling normaal wel is (Ritter e.a., 2011). Shang e.a. (2015) noemen de machinevertalingsmethode daarom inherent ongeschikt voor het genereren van antwoorden. Zij benoemen hierbij dat er op een enkel bericht veel verschillende passende antwoorden zijn en demonstreren dat de Neural Responding Machine bij kleine variaties wel zeer verschillende antwoorden kan genereren.

Shang e.a. (2015) laten de NRM een semantische representatie te geven in de vorm van context. Zij illustreren de NRM als zandloper, waarbij de context in het midden staat; de encoder wordt gedwongen het bericht samen te vatten, zodat de decoder een antwoord kan genereren. Dit heeft als belangrijk voordeel dat er niet klakkeloos woorden worden gekopieerd uit het bericht, zoals bij Ritter e.a. (2011) ontmoedigd moest worden. Het heeft hierdoor ook de mogelijkheid om identieke woorden te gebruiken, zonder dat het bericht volledig gekopieerd wordt.

Een probleem dat gerelateerd is aan semantische representatie zit in de details die in antwoord en bericht voorkomen. Naast relevantie stellen Ritter e.a. (2011), Ji e.a. (2014) en Shang e.a. (2015) allen de eis dat een antwoord logisch moet zijn. Dit is voornamelijk problematisch voor de IR-methode, omdat hierbij namen, tijden, locaties en dergelijke precies worden overgenomen uit de data (Shang e.a., 2015). Het blijkt niet voldoende om namen te beschouwen als normale woorden in het vergelijken van vectorvertegenwoordigingen van bericht en antwoord. Het beboeten van logische inconsistenties, zoals het verwarren van locaties en tijd, is lastig, omdat hiervoor meer inzicht nodig is in de betekenis van het bericht en het antwoord (Ji e.a., 2014).

Waar voornamelijk de IR-methode moeite heeft met logische inconsistenties, hebben de generatieve systemen meer moeite met grammatica, spelling en vloeiend taalgebruik. Bij het genereren van tekst kunnen taalfouten namelijk makkelijk ontstaan (Ritter e.a., 2011). Dit blijkt voornamelijk voor de SMT-methode een groot struikelblok te zijn (Shang e.a., 2015). De gebruikte data is echter ook niet vrij van spelfouten en grammaticale fouten en heeft veel weg van sms-taal (Ritter e.a., 2010). Gekopieerde teksten zullen deze fouten dus ook bevatten, wat bij paarsgewijze vergelijking in het voordeel kan werken van

gegenereerde tekst (Ritter e.a., 2011).

6.4.2 Meetbaarheid

Om effectiviteit van een systeem empirisch te kunnen aantonen is het noodzakelijk dat succes en falen meetbaar zijn. Succes betekent in de context van short-text conversation dat er een passend antwoord wordt gegeven. Om succes te meten moet dus de geschiktheid van antwoorden worden beoordeeld. Aangezien er in STC met grote hoeveelheden data gewerkt kan worden is het wenselijk een automatische beoordelingsmethode te gebruiken. Hoewel de technieken voor het genereren van antwoorden worden geleend uit de machinevertaling, blijken de beoordelingsmethoden die bij machinevertaling worden gebruikt niet geschikt voor beoordeling van antwoorden in STC (Ritter e.a., 2011; Shang e.a., 2015). Ritter e.a. toetsten hun resultaten zowel handmatig als met BLEU (Papineni, Roukos, Ward & Zhu, 2002) dat veel gebruikt wordt voor evaluatie van machinevertaling. Zij concludeerden dat BLEU niet geschikt is voor beoordeling van gegenereerde antwoorden, maar zagen wel de potentie voor een vergelijkbare methode die meer gefocust zou zijn op dialogen. Shang e.a. noemden het automatisch beoordelen van gegenereerde antwoorden een open probleem. Door het ontbreken van een heldere definitie van het probleem lijkt het gebruik van deze term niet volledig op zijn plaats. Het ontbreken van een helder gedefinieerd probleem lijkt echter juist een indicatie dat een automatische beoordelingsmethode voor STC voorlopig ontbreekt. Daarnaast concluderen Shawar en Atwell (2007) dat er geen unieke beoordelingsmethode is voor alle chatbots, maar dat deze op maat gemaakt moeten worden afhankelijk van het systeem.

De problemen rond automatische beoordeling leidden ertoe dat de besproken onderzoekers voor handmatige beoordeling kozen. Het handmatig labelen van de data is zeer tijdsintensief, waardoor het praktisch gezien niet op hele datasets toe te passen is (Wang e.a., 2013). Om toch voldoende data te hebben voor het trainen, kan worden aangenomen dat originele antwoorden op een bericht geschikt zijn en willekeurige andere antwoorden ongeschikt (Shang e.a., 2015). Het gebruik van handmatig gelabelde data geeft echter betere resultaten (Ji e.a., 2014).

De handmatige beoordeling van de geschiktheid van antwoorden is inherent subjectief (Ritter e.a., 2011). Ji e.a. (2014) beoordeelden antwoorden zelf; Ritter e.a. en Shang e.a. (2015) lieten dit over aan derden. Ritter e.a. gaven beoordelaars alleen de instructie dat een geschikt antwoord goed moest klinken en in onderwerp moest overeenkomen met het bericht. Ji e.a. en Shang e.a. gebruikten onderling vergelijkbare criteria voor handmatige beoordeling. In het eerste geval werden deze gebruikt voor het maken van zowel de train- als testset en in het tweede geval alleen om het eindresultaat te beoordelen. In beide gevallen werden semantische relevantie en logische consistentie als eis gesteld voor passende antwoorden. Dit lijkt een reflectie te zijn van het *maxim of relevance* van Grice (1975, 1978). Shang e.a. stelden daarnaast de eis dat een bericht niet te vaag of algemeen mocht zijn of juist te afhankelijk van een specifieke situatie en voegden vloeiend taalgebruik en taalfouten toe als criterium. Ji e.a. stelden

deze eisen niet, maar namen wel in de beoordeling mee of de spraakhandeling in het antwoord paste op die van het bericht. De gekozen criteria worden wel toegelicht, maar de keuze zelf wordt bij geen van de onderzoeken onderbouwd.

Ji e.a. (2014) labelden antwoorden als geschikt of ongeschikt. Ritter e.a. (2011) labelden antwoorden niet, maar vergeleken steeds twee antwoorden, om te bepalen welke geschikter was. Het gebruik van deze verschillende methoden vermoeilijkt directe vergelijking van de resultaten. Shang e.a. (2015) gebruikten de labels 'geschikt' en 'ongeschikt' en voegden voor vage of overspecifieke antwoorden een neutrale categorie toe. Het gebruik van het label 'neutraal' blijkt inzicht te kunnen geven in de sterke en zwakke punten van verschillende STC-benaderingen. Er zou naast een oordeel over geschiktheid ook aan beoordelaars gevraagd kunnen worden welke criteria van toepassing zijn, om hier nog meer inzicht in te krijgen.

Een universele beoordelingsmethode voor chatbots lijkt te ontbreken, waardoor het vergelijken van testresultaten bemoeilijkt wordt. Shawar en Atwell (2007) beargumenteren daarnaast dat de beoordeling van een chatbot afhankelijk moet zijn van de taak en dus in veel gevallen op maat moet worden gemaakt. Voor STC kan handmatige beoordeling door de ontwerper of door derden gebruikt worden. Het lijkt hierbij aan te raden om heldere criteria voor geschiktheid te hanteren. Bovendien kan het nuttig zijn om beoordelaars te vragen waarom een antwoord ongeschikt is, zodat in de ontwikkeling rekening kan worden gehouden met de zwaktes van het beoordeelde systeem, zoals ook blijkt in de volgende sectie.

6.4.3 Resultaten

In ieder van de drie besproken onderzoeken worden op een andere manier resultaten gemeten, waardoor de resultaten niet direct vergelijkbaar zijn. In deze sectie wordt dan ook niet beoordeeld welke methode objectief het beste is. Voor ieder onderzoek worden de interessantste resultaten benoemd en waar mogelijk vergeleken met de andere methoden om de zwakke en sterke punten te belichten.

Ji e.a. (2014) maten het succes van de verschillende combinaties van kenmerken voor Information Retrieval-modellen. Hiervoor gebruikten zij twee maten: gemiddelde geschiktheid van het hoogst scorende antwoord en de naar ordening gewogen som van alle kandidaat-antwoorden. Voor vergelijking met de andere methoden voor short-text conversation is vooral de geschiktheid van het beste antwoord interessant (Shang e.a., 2015). Van de hoogst scorende antwoorden van het basismodel is 57,4% geschikt (Ji e.a., 2014). De combinatie van alle kenmerken geeft 63,7% geschikte, hoogst scorende antwoorden. De combinatie van complexe kenmerken resulteert dus in 6,3% meer geschikte antwoorden ten opzichte van het basismodel. Shang e.a. bepaalden dat het volledige IR-model 29,8% geschikte en 38,7% neutrale antwoorden geeft. Het verdelen van antwoorden over drie categorieën van geschiktheid lijkt dus voornamelijk tot een afname van geschikte antwoorden te leiden. Er worden echter ook minder antwoorden als ongeschikt beoordeeld dan door Ji e.a. (respectievelijk 31,5% en 36,3%).

Ritter e.a. (2011) beweerden dat de Statistical Machine Translation-methode

op maat gemaakte antwoorden genereert die bij handmatige beoordelingen de voorkeur krijgen boven antwoorden die worden gevonden met IR-methoden. De antwoorden die hun SMT-model genereert werden dan ook in 64,5% van de gevallen als geschikter beoordeeld dan die van het IR-model op basis van overeenkomsten in het bericht en 59,3% bij het model op basis van overeenkomst tussen antwoorden. Deze IR-modellen zijn echter zeer simpel in vergelijking met zelfs het basismodel van Ji e.a. (2014). Uit de handmatige beoordeling van Shang e.a. (2015) blijkt een SMT-model slechts in 5,6% van de gevallen een goed passend antwoord te genereren; 74,4% van de antwoorden blijkt ongeschikt. Belangrijk is hierbij op te merken dat vloeiend taalgebruik volgens Shang e.a. een vereiste is om als geschikt of neutraal te worden beoordeeld. In paarsgewijze vergelijking weegt dit misschien minder mee dan relevantie (Ritter e.a., 2011). Bovendien werd het SMT-model getest dat gebruik maakt van GIZA++ voor het kiezen van fraseparen, in plaats van de methode aan de hand van de Fisher exacte toets die beter presteert (Ritter e.a., 2011). Die methode werd in 14,5% van de vergelijkingen zelfs als geschikter gezien dan door de originele (door mensen geschreven) antwoorden.

Uit de resultaten van Shang e.a. (2015) bleek dat de verschillende Neural Reponding Machines in meer dan 60% van de gevallen minstens neutraal scoren en daarmee als vloeiend, relevant en logisch consistent worden beoordeeld. De lokale NRM presteert beter dan de globale, waaruit blijkt dat een dynamische context geschikter is dan een vaste samenvatting. De combinatie in de hybride NRM gaf het beste resultaat met 39,3% geschikte en slechts 23,6% ongeschikte antwoorden, waaruit te concluderen is dat de globale samenvatting bruikbare informatie toevoegt die ontbreekt in de dynamische context. Opvallend is dat de globale NRM en de IR-methode vergelijkbare gemiddelde scores halen (respectievelijk 1,552 en 1,448) (Shang e.a., 2015). De IR-methode heeft relatief veel neutrale antwoorden en Shang e.a. concludeerden dat dit komt door inconsistente details die in gekopieerde antwoorden relatief vaak voorkomen, terwijl de NRMs dit soort details zelden genereren.

Hoewel een universele beoordelingsmethode voor short-text conversation ontbreekt, kunnen uit de verschillende resultaten conclusies worden getrokken. Zo lijkt de voornaamste zwakte van de IR-methode te zitten in het gebrek aan flexibiliteit en de resulterende inconsistenties. De vermeende ongeschiktheid van de SMT-methode lijkt te worden bevestigd door het lage aantal antwoorden dat als geschikt wordt beoordeeld. De hybride NRM behaalt de beste resultaten, waardoor het lijkt dat een generatief systeem met een doordachte, gespecialiseerde structuur het meest geschikt is om relevante, correcte antwoorden te geven.

Hoofdstuk 7

Conclusie

In dit hoofdstuk wordt eerst een samenvatting van de voorgaande tekst gegeven, gevolgd door een aantal suggesties voor de lezer.

Samenvatting

De ontwikkeling van chatbots begon met ELIZA, een prototypische gescripte chatbot die gedreven wordt door voorgeschreven regels in een script. Hoewel gescripte chatbots met redelijk eenvoudige technieken al overtuigend kunnen overkomen, hebben ze ook belangrijke beperkingen. Regels in scripts worden meestal stuk voor stuk geschreven door de programmeur, waardoor het uitbreiden van een script lang duurt. De meeste scripts worden daarom beperkt tot een enkel domein. Er zijn twee alternatieven voor gescripte chatbots besproken.

Ten eerste zijn er de dialoogsystemen die gebruik maken van logica, geformaliseerde kennis en natuurlijke taalverwerkingstechnieken. Deze systemen zijn flexibeler dan gescripte chatbots en hebben veel praktische toepassingen, zoals besproken in Sectie 5.1. Daarnaast staat het modulaire ontwerp van veel dialoogsystemen ontwerpers toe om voort te bouwen op werk van anderen door te focussen op één onderdeel. Geschikte data voor het trainen van (de onderdelen van) deze systemen ontbreekt veelal, waardoor ook deze systemen meestal beperkt worden tot toepassingen binnen één domein.

Ten tweede zijn er de lerende chatbots die gebruik maken van machine learning en grote hoeveelheden data. Deze systemen hebben als voordeel dat ze makkelijk uit te breiden zijn en daardoor geschikter zijn voor toepassingen in een open domein. De focus is bij de bespreking van lerende systemen gelegd op short-text conversation, waarbij slechts één antwoord hoeft worden gegeven. Er zijn drie verschillende soorten lerende systemen voor STC beschreven en vergeleken:

- Bij de *information retrieval*-methode wordt geleerd om aan de hand van verschillende kenmerken het meest geschikte antwoord uit het corpus te kiezen.

- Bij de *statistical machine translation*-methode wordt geleerd antwoorden te genereren door het antwoord te zien als vertaling van het bericht.
- De *neural responding machine* leert om een bericht om te zetten in een abstracte vorm en deze abstractie te vertalen naar een antwoord.

De IR-methode is het minst flexibel, doordat het antwoorden kopieert. De SMT-methode lijkt ongeschikt, omdat het uitgaat van semantische equivalentie, die ontbreekt tussen berichten en antwoorden. De NRM is flexibeler dan de IR-methode en is geschikter gestructureerd dan de SMT-methode, maar lijkt niet altijd vloeiende, grammaticaal correcte antwoorden te geven.

De grootste uitdaging voor het maken van een lerend chatsysteem blijkt niet het ontwerpen van het model; het verzamelen van geschikte data en het meten van succes en falen blijken grotere problemen. Het verzamelen van data blijkt voornamelijk een praktisch probleem, dat kan worden ontweken door beperking tot STC of een ander domein waarvoor data makkelijk beschikbaar is. Het meten van succes en falen is problematischer, omdat handmatige beoordeling voor STC in sterke mate subjectief is en er voor bepaalde systemen geen geschikte automatische beoordelingsmethode is.

Slotsom

In vergelijking met lerende chatbots lijkt niet veel meer te bereiken met onderzoek op het gebied van gescripte chatbots en er zijn weinig praktische toepassingen voor. Gescripte chatbots kunnen wel vermakelijk zijn en zijn relatief eenvoudig, waardoor ze zeer geschikt zijn als programmeeroefening of hobbyproject. Voor praktische en commerciële doeleinden zijn complexere dialoogsystemen geschikter. Qua wetenschappelijk onderzoek valt nog veel te bereiken op het gebied van lerende chatbots. Zo zouden de besproken systemen voor short-text conversation uitgebreid kunnen worden door rekening te houden met spraakhandelingen en sentiment. De lezer heeft na het lezen van dit werk hopelijk voldoende overzicht gekregen van de ontwikkelingen en de mogelijkheden op het gebied van chatbots om zelf een keuze te maken over de volgende te nemen stap.

Bibliografie

Auli, M., Galley, M., Quirk, C. & Zweig, G. (2013). Joint Language and Translation Modeling with Recurrent Neural Networks. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 1044–1054).

Austin, J. L. (1962). *How to do things with words: The William James lectures.* Cambridge, MA: Harvard University Press.

Bahdanau, D., Cho, K. & Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR*, *abs/1409.0473*. arXiv: 1409.0473. Verkregen van http://arxiv.org/abs/1409.0473

Clark, H. H. & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive science*, *13*(2), 259–294.

Colby, K. M., Hilf, F. D., Weber, S. & Kraemer, H. C. (1972). Turing-like indistinguishability tests for the validation of a computer simulation of paranoid processes. *Artificial Intelligence*, *3*, 199–221. doi:https://doi.org/10.1016/0004-3702(72)90049-5

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R. & Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *Journal of machine learning research*, *9*(Aug), 1871–1874.

Floridi, L., Taddeo, M. & Turilli, M. (2009). Turing's Imitation Game: Still a Challenge for Any Machine and Some Judges. *19*, 145–150.

Goertzel, B. & Pennachin, C. (2007). Contemporary Approaches to Artificial General Intelligence. In *Artificial General Intelligence* (pp. 1–30). doi:10.1007/978-3-540-68677-4_1

Grice, H. P. (1975). Logic and conversation. *Speech Acts: Syntax and Semantics Volume 3*, 41–58.

Grice, H. P. (1978). Further notes on logic and conversation. *Pragmatics: Syntax and Semantics Volume 9*, 113–128.

Herbrich, R., Graepel, T. & Obermayer, K. (1999). Large Margin Rank Boundaries for Ordinal Regression. In *Advances in Large Margin Classifiers* (Hfdstk. 7, pp. 115–132). The MIT Press. Verkregen van http://www.herbrich.me/papers/nips98%5C_ordinal.pdf

Ji, Z., Lu, Z. & Li, H. (2014). An Information Retrieval Approach to Short Text Conversation. *CoRR*, *abs/1408.6988*. arXiv: 1408.6988. Verkregen van http://arxiv.org/abs/1408.6988

Johnson, H., Martin, J., Foster, G. & Kuhn, R. (2007). Improving translation quality by discarding most of the phrasetable. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Jurafsky, D. & Martin, J. H. (2009). *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*. Upper Saddle River, N.J. London: Pearson Education.

Kalchbrenner, N. & Blunsom, P. (2013). Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 1700–1709).

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., ... Zens, R. e.a. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions* (pp. 177–180). Association for Computational Linguistics.

Kolb, C., Afrika, A. & Millican, P. (2006). Connecting with Elizabeth: Using artificial intelligence as a data collection aid. In *Connections, MRS annual conference* (pp. 22–24).

Kuipers, B., McCarthy, J. & Weizenbaum, J. (1976). Computer power and human reason. *ACM SIGART Bulletin*, (58), 4–13.

Lang, G. & Molen, H. v. d. (2016). *Psychologische gespreksvoering: een basis voor hulpverlening*. Boom.

Lester, J., Branting, K. & Mott, B. (2004). Conversational agents. *The Practical Handbook of Internet Computing*, 220–240.

Litman, D., Singh, S., Kearns, M. & Walker, M. (2000). NJFun: a reinforcement learning spoken dialogue system. In *Proceedings of the ANLP-NAACL 2000 Workshop on Conversational Systems* (pp. 17–20). Association for Computational Linguistics.

Lu, Z. & Li, H. (2013). A Deep Architecture for Matching Short Texts. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani & K. Q. Weinberger (Red.), *Advances in Neural Information Processing Systems 26* (pp. 1367–1375). Curran Associates, Inc. Verkregen van http://papers.nips.cc/paper/5019-a-deep-architecture-for-matching-short-texts.pdf

Mann, W. C. (2003). *The Dialogue Diversity Corpus*. Verkregen van http://www-bcf.usc.edu/~billmann/diversity/DDivers-site.htm

Millican, P. (2002). Elizabeth for Windows (Versie 2.04). Verkregen van http://www.philocomp.net/ai/elizabeth.htm

Misu, T., Georgila, K., Leuski, A. & Traum, D. (2012). Reinforcement learning of question-answering dialogue policies for virtual museum guides. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue* (pp. 84–93). Association for Computational Linguistics.

Och, F. J. & Ney, H. (2003). A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, *29*(1), 19–51.

Oppy, G. & Dowe, D. (2018). The Turing Test. In E. N. Zalta (Red.), *The Stanford Encyclopedia of Philosophy* (Spring 2018). Metaphysics Research Lab, Stanford University.

Papineni, K., Roukos, S., Ward, T. & Zhu, W.-J. (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 311–318). ACL '02. doi:10.3115/1073083.1073135

Rajaraman, A. & Ullman, J. D. (2011). *Mining of Massive Datasets*. doi:10.1017/CBO9781139058452

Rambow, O., Bangalore, S. & Walker, M. (2001). Natural language generation in dialog systems. In *Proceedings of the first international conference on Human language technology research* (pp. 1–4). Association for Computational Linguistics.

Reiter, E. (1994). Has a Consensus NL Generation Architecture Appeared, and is It Psycholinguistically Plausible? In *Proceedings of the Seventh International Workshop on Natural Language Generation* (pp. 163–170). INLG '94. Kennebunkport, Maine: Association for Computational Linguistics. Verkregen van http://dl.acm.org/citation.cfm?id=1641417.1641436

Ritter, A., Cherry, C. & Dolan, B. (2010). Unsupervised Modeling of Twitter Conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 172–180). HLT '10. Los Angeles, California: Association for Computational Linguistics. Verkregen van http://dl.acm.org/citation.cfm?id=1857999.1858019

Ritter, A., Cherry, C. & Dolan, W. B. (2011). Data-driven response generation in social media. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 583–593). Association for Computational Linguistics.

Schatzmann, J., Weilhammer, K., Stuttle, M. & Young, S. (2006). A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The knowledge engineering review*, *21*(2), 97–126.

Schegloff, E. A. (1968). Sequencing in conversational openings. *American anthropologist*, *70*(6), 1075–1095.

Searle, J. R. (1975). A taxonomy of illocutionary acts.

Shang, L., Lu, Z. & Li, H. (2015). Neural Responding Machine for Short-Text Conversation. *CoRR*, *abs/1503.02364*. arXiv: 1503.02364. Verkregen van http://arxiv.org/abs/1503.02364

Shawar, B. A. & Atwell, E. (2002). *A comparison between Alice and Elizabeth chatbot systems*. University of Leeds, School of Computing research report 2002.19.

Shawar, B. A. & Atwell, E. (2003). Using dialogue corpora to train a chatbot. In *Proceedings of the Corpus Linguistics 2003 conference* (pp. 681–690).

Shawar, B. A. & Atwell, E. (2007). Different Measurements Metrics to Evaluate a Chatbot System. In *Proceedings of the Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies* (pp. 89–

96). NAACL-HLT-Dialog '07. Rochester, New York: Association for Computational Linguistics. Verkregen van http://dl.acm.org/citation.cfm?id=1556328.1556341

Stalnaker, R. C. (1978). Assertion. *Pragmatics: Syntax and Semantics Volume 9*, 315–332.

Stifelman, L. J., Arons, B., Schmandt, C. & Hulteen, E. A. (1993). VoiceNotes: A Speech Interface for a Hand-held Voice Notetaker. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems* (pp. 179–186). CHI '93. doi:10.1145/169059.169150

Sutskever, I., Vinyals, O. & Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence & K. Q. Weinberger (Red.), *Advances in Neural Information Processing Systems 27* (pp. 3104–3112). Curran Associates, Inc. Verkregen van http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf

Turing, A. M. (1950). Computing Machinery and Intelligence. *Mind*, *59*(236), 433–460. Verkregen van http://www.jstor.org/stable/2251299

Voorhees, E. M. (1999). Natural Language Processing and Information Retrieval. In M. T. Pazienza (Red.), *Information Extraction* (pp. 32–48). Berlin, Heidelberg: Springer Berlin Heidelberg.

Wallace, R. S. (2009). The Anatomy of A.L.I.C.E. In R. Epstein, G. Roberts & G. Beber (Red.), *Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer* (pp. 181–210). Dordrecht: Springer Netherlands.

Wang, H., Lu, Z., Li, H. & Chen, E. (2013). A dataset for research on short-text conversations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 935–945).

Weizenbaum, J. (1966). ELIZA - Computer Program for the Study of Natural Language Communication Between Man and Machine. *Commun. ACM*, *9*(1), 36–45. doi:10.1145/365153.365168

Weizenbaum, J. (1976). *Computer power and human reason: From judgment to calculation.* San Fransisco: WH Freeman & Co.

Winograd, T. (1972). Understanding natural language. *Cognitive psychology*, *3*(1), 1–191.

Wittgenstein, L. (1953). Philosophical Investigations (Philosophische Untersuchungen).

Wu, W., Lu, Z. & Li, H. (2013). Learning bilinear model for matching queries and documents. *The Journal of Machine Learning Research*, *14*(1), 2519–2548.

Xue, X., Jeon, J. & Croft, W. B. (2008). Retrieval models for question and answer archives. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 475–482). ACM.

Yankelovich, N., Levow, G.-A. & Marx, M. (1995). Designing SpeechActs: Issues in Speech User Interfaces. In *Proceedings of the SIGCHI Conference on*

Human Factors in Computing Systems (pp. 369–376). CHI '95. doi:10.1145/223904.223952

Zhang, H.-P., Yu, H.-K., Xiong, D.-Y. & Liu, Q. (2003). HHMM-based Chinese lexical analyzer ICTCLAS. In *Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17* (pp. 184–187). Association for Computational Linguistics.