Centrum voor Wiskunde en Informatica
REPORT*RAPPORT*

Filosofische aspecten van machinaal leren

H.A.N. van Maanen

Computer Science/Department of Algorithmics and Architecture

CS-N9501 1995

Report CS-N9501
ISSN 0169-118X

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

Filosofische Aspecten van Machinaal Leren

Jeroen van Maanen

CWI

Postbus 94079, 1090 GB Amsterdam

jeroenm@cwi.nl

Abstract

In de dertiger jaren heeft Alan Turing een formalisme bedacht waarvan inmiddels algemeen wordt aangenomen dat het alle machinaal gedrag kan beschrijven. Voor lerend gedrag ligt de zaak moeilijker. Er zijn binnen de Psychologie meerdere definities van leren in omloop, die of moeilijk formeel te maken zijn, of niet algemeen genoeg zijn. In dit artikel zal getracht worden aannemelijk te maken dat een formalisering van lerend gedrag op basis van Turing's model mogelijk is. Hierbij spelen overwegingen over de begrippen kennis en onzekerheid een centrale rol.

AMS Subject Classification (1991): 00A30, 00A71, 03A05, 60G05, 68Q30, 68T05
CR Subject Classification (1991): A.0, E.4, I.6.5
Keywords & Phrases: Machine learning, Philosophy of Science, Probability theory, Kolmogorov complexity, Minimum Description Length principle
Note: Partially supported by the European Union through NeuroCOLT ESPRIT Working Group Nr. 8556.

Table of Contents

1. Introductie

Een groot verschil tussen machines zoals we die nu kennen en mensen is dat mensen in staat zijn om van hun ervaringen te leren. Met name door gebruikers van computers wordt regelmatig geklaagd dat de gebruikte computerprogammatuur niet adaptief genoeg is. Aangezien het buitengewoon moeilijk blijkt te zijn om deze klachten te verhelpen, is binnen de Informatica de vraag ontstaan of het in principe mogelijk is om een machinaal systeem lerend gedrag te laten vertonen. Binnen de Informatica wordt de vraag of een machinaal systeem een bepaald type gedrag kan vertonen, vertaald naar de vraag of een Turing machine dat gedrag kan vertonen. Een Turing machine is niet zozeer een concrete machine, maar een formeel model waarvan wordt aangenomen dat het alle machinaal reproduceerbaar gedrag kan beschrijven. Voordat de vraag of een Turing machine lerend gedrag kan vertonen moeten

we ons eerst buigen over de vraag: wat *betekent* het voor een machine om lerend gedrag te vertonen? Over deze vraag gaat dit artikel.

2. Doelstelling

In tegenstelling tot de Filosoof John Searle zal ik niet proberen om een theorie over de *aard* van leren te presenteren. Searle [Sea92] beweert dat iemand die dat zou proberen, daarmee ook een theorie over het bewustzijn moet meeleveren. Ik voel me niet ter zake kundig om uitspraken over het bewustzijn te doen en ik zal me daar dan ook verre van houden. Wat ik wel zal doen is uiteenzetten hoe ik denk dat het mogelijk is om lerend gedrag in zijn algemeenheid te *beschrijven* en daarmee te meten. Ik zal ook betogen dat de vorm van deze beschrijving de mogelijkheid biedt om op termijn machines te construeren en te programmeren die binnen de voorgestelde beschijvingsmethode vallen en waarvan het gedrag in mijn ogen ook als lerend zal moeten worden aangemerkt.

Het in kaart brengen van de mogelijkheden en beperkingen van dergelijke machine lijkt me voor zowel Psychologisch als Filosofisch onderzoek de moeite waard, omdat het licht werpt op de systematiek van gecompliceerd gedrag en op met name de beperkingen van pogingen de werkelijkheid (ervaringswereld) in een eindige beschrijving te vangen. De suggestie van John Searle om de invalshoek van de Informatica bij het bestuderen van cognitieve verschijnselen af te schrijven als irrelevant, is in mijn ogen dan ook voorbarig. Zijn voorstel om de hoop te vestigen op het breinonderzoek dat zou moeten leiden tot de ontdekking van 'breinstof', de substantie die ons in staat zou stellen on te denken, is in mijn ogen nog veel naïver dan de, inderdaad vrij primitive, methoden waarmee onderzoekers binnen de Artificiële Intelligentie proberen menselijk gedrag na te bootsen. De argumenten waarmee Searle de onmogelijkheid van Artificiële Intelligentie probeert aan te tonen doen mij denken aan de argumenten waarmee middeleeuwse denkers probeerden te bewijzen dat het alleen vogels gegeven is om te vliegen.

De doelstelling van dit artikel zal zijn om een definitie van lerend gedrag in Psychologische termen te vertalen naar een beschrijving van lerend gedrag in termen van machinaal reproduceerbare processen. Daarbij is de bedoeling dat de resulterende methodes om het leerkarakter van processen te beschrijven de vorm aannemen van een formeel systeem. Laatst kwam ik bij het lezen van een boek over Romeinse stedenbouw [Mac81] het woord 'formeel' tegen als de naam van een houten staketsel waarop een boog wordt gebouwd. Als de sluitsteen van de boog geplaatst is kan het formeel verwijderd worden, omdat de boog dan opzichzelf blijft staan. Dat komt overeen met wat ik met een formeel systeem voor lerend gedrag voor ogen heb: een constructie die voorkomt dat een theorie al tijdens het opstellen ervan in elkaar valt.

3. Benaderingen

Om te komen tot een beschrijving van lerend gedrag zullen we vanuit een aantal invalshoeken het vraagstuk in ogenschouw nemen. Aan de Psychologie zullen we een definitie van leren ontlenen. De kansrekening geeft ons de mogelijkheid om afleidingen te maken uit onzekere en mogelijk conflicterende aannamen. De Informatica geeft ons een formele definitie van machinaal gedrag. En met Kolmogorov complexiteit kunnen we meten hoeveel regelmaat de omgeving tot op een zeker moment vertoond heeft. In paragraaf 4 zal ik schetsen hoe, met het volgens deze benaderingen verkregen materiaal, een formeel model voor lerend gedrag eruit zou kunnen zien. Maar we zullen eerst de verschillende benaderingen uitwerken.

3.1 Psychologie

Het begrip 'leren' valt binnen het terrein van de Psychologie, omdat het over menselijk gedrag gaat. Aan de andere kant gaat het om gedrag dat de kennis over de omgeving — de wereld — vergroot. In die zin valt leren ook binnen het onderzoeks gebied van de Filosofie. Het verschil tussen deze disciplines is vooral gelegen in het standpunt van waaruit het gedrag beschreven wordt. Voor de Psycholoog is een lerend organisme het onderwerp van onderzoek. De onderzoeker zelf staat buiten de leersituatie. Vaak staat de Psycholoog zelfs buiten de wereld waarin het leerproces zich afspeelt. Bij veel experimenten kijkt de observator toe zonder zelf zichtbaar te zijn.

Voor de Filosoof echter is het gedrag van de Psycholoog zelf onderwerp van onderzoek. Het feit dat een Psycholoog zich buiten de wereld van haar onderwerp van onderzoek plaats om het risico op verstoringen van de meetresultaten te verkleinen is gedrag dat Filosofen proberen te verklaren.

We kunnen nog opmerken dat het gedrag van de Filosoof als lerend gedrag weer onderwerp van onderzoek voor Psychologen vormt. Hiermee is de cirkel rond. Dit betekent dat een formele beschrijving van lerend gedrag zowel de Psychologische als de Filosofische aspecten ervan zal moeten kunnen bevatten.

Een algemene definitie van leren in Psychologische termen is: *leren is het aanpassen van gedrag op basis van ervaring ter bevrediging van interne behoeften.* Hierbij moet worden aangemerkt dat voor het bevredigen van behoeften vaak energie, geld, vrijheid of algemeen uitgedrukt mogelijkheden nodig zijn. Het creëren van deze mogelijkheden wordt vaak een afgeleide behoefte, een doel op zichzelf. Een van de belangrijkste voorwaarden voor effectief gedrag is kennis over de omgeving. Vandaar dat we ons zullen concentreren op het verzamelen van kennis als hoofddoel van een lerend systeem. Andere criteria als het verzamelen van voedsel en het zoeken of construeren van onderdak kunnen worden gezien als reacties op beperkingen in de interactie tussen lerend systeem en omgeving. (Als het organisme sterft daalt de hoeveelheid kennis over de omgeving in het systeem in korte tijd naar nul.)

3.2 Kansrekening

Het is binnen de Artificiële Intelligentie gebruikelijk om de kennis van een systeem over de omgeving te beschrijven in termen van predicaten. Het systeem redeneert over deze predicaten met behulp van de klassieke predicatenlogica, eventueel uitgebreid met modale operatoren of andere constructies die onvolledige kennis en/of gedeeltelijk incorrecte feiten toestaan. Deze aanpak brengt echter de nodige problemen met zich mee. Aangezien leren inhoudt dat de kennis van een systeem over de omgeving toeneemt met de tijd wil dat zeggen dat, zolang het systeem nog leert, de kennis over de omgeving onvolledig is. Een strategie die mensen hanteren om daarmee om te gaan is om waar nodig aannames te doen en die te herzien zodra meer gegevens bekend zijn. Het vinden van aannames die logisch consistent zijn met de huidige kennis is buitengewoon moeilijk. Veel op predicatenlogica gebaseerde systemen zijn daarom, of erg langzaam, of gebruiken heuristieken voor het omzeilen van inconsistenties, die vaak al te eenvoudig tot absurde afleidingen leiden. Een ander probleem is dat de logische implicatie incompatibel is met de causaliteitsrelatie. Het blijkt zeer moeilijk te zijn om causaliteit goed te formaliseren binnen de predicatenlogica.

Een mogelijke uitweg uit deze problematiek is het gebruik van kansrekening in plaats van predicatenlogica. De kansrekening is van meet af aan opgezet om te redeneren over waarnemingen die onzeker zijn omdat ze nog gedaan moeten worden en waarnemingen waarvan onzeker is hoe ze tot stand zijn gekomen. Verder staat de kansrekening toe dat verschillende

aannames die elkaar onderling uitsluiten naast elkaar bestaan. Het is bijvoorbeeld denkbaar dat iemand in een casino met zichzelf afspreekt: als ik nu win speel ik met het gewonnen kapitaal verder, als ik verlies ga ik naar huis. Aan de mogelijke uitkomsten van de waarneming (winst, verlies) worden waarden toegekend die maatgevend zijn voor de gemiddelde verwachtte uitkomst bij herhaling van het experiment.

De overstap van predicatenlogica naar kansrekening is niet zo groot als het op het eerste gezicht misschien lijkt. De axiomatisering van de kansrekening lijkt erg op die van de predicatenlogica. De predicatenlogica kan zelfs worden opgevat als een limietgeval van de kansrekening (*Cf.* Popper [Pop68, §48] die Wittgenstein's *Tractatus* [Wit18, prop. 5.152] citeert).

Een probleem van de predicatenrekening dat niet direct door het gebruik van kansrekening wordt opgelost is het vraagstuk van de elementaire feiten. Het is al een oud vraagstuk binnen de Filosofie hoe het mogelijk is om op grond van een noodzakelijk beperkte hoeveelheid waarnemingen iets met *zekerheid* vast te kunnen stellen. In de kansrekening vertaalt dit probleem zich naar het vraagstuk hoe het mogelijk is om op grond van een beperkte hoeveelheid waarnemingen de kans op een verschijnsel vast te stellen.

Hume stelde vast dat dit probleem, het inductieprobleem, voor de predicatenlogica onoplosbaar is. Voor de kansrekening moeten we eerst vaststellen wat het getal dat we de kans op een verschijnsel noemen betekent. Pogingen om 'kans' op te vatten als een objectieve meetbare eigenschap van een reeks waarnemingen [Mis39, Pop68] zijn tot op heden gestrand op de constatering dat extra informatie de kans op een verschijnsel kan beïnvloeden [Ris89]. Bijvoorbeeld als we over satelietfotos van het weer over de hele aardbol beschikken kunnen we de kans dat het morgen regent in Amsterdam beter voorspellen (de kans ligt dichter bij 1 of 0), dan wanneer we het zonder die informatie moeten doen. Daarom zullen we ons richten op de vraag: hoe zeker kunnen we van een geobserveerde regelmatigheid zijn, gegeven een bepaalde hoeveelheid waarnemingen?

Ik zal proberen aan te geven dat het mogelijk is hierop een universeel antwoord te geven. Daarmee bedoel ik dat er vele mogelijkheden zijn om op basis van dezelfde waarnemingen kansen toe te kennen, die welliswaar van elkaar kunnen verschillen, maar waarvan de toegekende kansen naarmate lijst waarnemingen langer wordt, dichter bij elkaar komen te liggen. Dit suggereert dat het voor de kansrekening niet mogelijk is om objectieve kansen toe te kennen. Net zomin als er voor de predicatenlogica een methode bekend is om objectief de waarheid van predicaten vast te stellen. Het is wèl mogelijk om een referentiekader te kiezen en binnen dat kader subjectieve kansen toe te kennen. Het redeneren op basis van deze kansen zal op de lange termijn tot dezelfde conclusies leiden als redeneren op basis van kansen toegekend binnen het kader van een willekeurig ander universeel referentiekader.

Deze opvatting van het begrip 'universeel' is ontleend aan de stelling van Alan Turing dat er machines bestaan die elke andere machine (en dus ook elkaar) kunnen simuleren. Toegepast op het inschatten van subjectieve kansen komt deze stelling neer op de constatering dat een universele schatter zich in het standpunt van elke andere schatter kan inleven. Deze opmerking is puur als analogie bedoeld en niet als argument om aan te tonen dat een universele machine enige cognitieve faculteit heeft (dat zou ook in strijd zijn met de in de introductie gestelde uitgangspunten). Het kenmerkend vermogen van een bewust schepsel om zich in te leven in een ander schepsel wordt uitgebreid aan de orde gesteld in *The Mind's I* [DH81, Hst. 24]. Voordat we dieper ingaan op de fundering van de kansrekening als methode om onvolledige kennis te modelleren zal de formele beschrijving van machinaal gedrag nader uitgewerkt worden.

3.3 Machinaal gedrag

Al in de dertiger jaren formuleerde Alan Turing een model voor het beschrijven van machinaal gedrag: de Turing machine. Het belang van dit model is vervat in de zogenaamde Church-Turing these. Die stelt dat alle systematische regelmatigheden in gedrag nagebootst kunnen worden met een Turing machine. Dit model is om twee redenen relevant voor het formaliseren van lerend gedrag.

Ten eerste betekent de Church-Turing these dat een beschrijving van lerend gedrag gegoten moet kunnen worden in de vorm van een algoritme, dat wil zeggen een programma voor een Turing machine. Dat wil zeggen dat dit programma in principe uitgevoerd zou moeten kunnen worden door een gewone computer. Of een computer dat een leer-algoritme uitvoert ook leert is een vraag waaraan ik, zoals aangekondigd in de inleiding, in dit artikel geen aandacht zal besteden.

Ten tweede houdt de Church-Turing these een formele definitie van regelmaat in die, met enige moeite, omgezet kan worden in een kwantitatieve definitie van regelmaat. Het resultaat is een complexiteitsmaat die gebruikt kan worden om een bruikbare schatting van kansen op elementaire gebeurtenissen te genereren. De stap van Turing's model van machinaal gedrag naar het schatten van kansen is niet triviaal en zal worden uitgewerkt in paragraaf 3.4.

Turing ging er van uit dat elk mechanisch systeem vertaald kan worden in een symbolen verwerkend systeem. Hij bedacht dat het mogelijk is om de begintoestand van bijvoorbeeld een stoommachine te beschrijven in termen van pijpdiameters, coördinaten, materiaaleigenschappen enzovoort. Vervolgens kan de configuratie van de machine op een later tijdstip berekend worden met behulp van rekenregels die afgeleid zijn van de natuurwetten. In de tijd van Turing was dit een gedachtenexperiment, maar inmiddels zijn dit soort berekeningen voor ingenieurs dagelijkse kost. Bij het berekenen van latere toestanden van de machine wordt uitsluitend gewerkt met de gegeven beschrijving van de begintoestand en de rekenregels. De aanname dat elke machine equivalent is met een symboolverwerkend systeem, ligt aan de basis van de Church-Turing these.

De volgende stap is om een minimaal symboolverwerkend systeem te definiëren dat nog in staat is om alle machinaal gedrag te beschrijven. De werkwijze van Turing was om naast de begintoestand van de machine ook de rekenregels volledig uit te schrijven. Hij bewees dat het voldoende is om een lijst te maken met regels van de vorm: als in toestand p, het symbool a zichtbaar is, schrijf dan symbool b, doe een stap naar links of rechts en neem toestand q aan. Deze regels moeten dan worden uitgevoerd op een in vakjes verdeelde strook papier vaarvan steeds slechts één vakje zichtbaar is. Elk vakje kan precies één symbool bevatten. We eisen dat de lijst uit slechts eindig veel regels bestaat en dus maar eindig veel toestanden beschrijft. Een van deze toestanden is de begintoestand. Op de strook papier kan aan het begin van de berekening een reeks symbolen worden geschreven. Deze reeks heet de *invoer* van de berekening. We zorgen ervoor dat het eerste vakje van de strook zichtbaar is. We kunnen dan herhaaldelijk de regel die van toepassing is uitvoeren. We nemen aan dat er altijd hoogstens één regel van toepassing is. Op het moment dat voor de nieuwe toestand van het systeem en het op dat moment zichtbare symbool geen enkele regel van de lijst van toepassing is stopt de berekening. Als de strook papier gedurende de berekening te kort dreigt te zijn, moet er een stuk aangeplakt worden. De reeks symbolen die op de strook papier achterblijft als de machine stopt is de *uitvoer* van de berekening. Het is evident dat het niet uitmaakt of de regels door een mens of door een automaat worden uitgevoerd. De uitvoer wordt volledig

bepaald door de invoer, de lijst regels en de begintoestand. We noemen de lijst regels samen met de specificatie van de begin toestand een Turing machine.

De exacte formulering van the Church-Turing these is: elke machine is te beschrijven in termen van een Turing machine. Hoewel deze these niet wiskundig bewezen kan worden heeft ze binnen de Informatica de status van een natuurwet. Het vertrouwen van Informatici heeft zelf zoveel indruk gemaakt dat een vooraanstaand Fysicus als Richard Feynman de hypothese geopperd heeft dat het gedrag van het hele Fysische universum tot op quantum-mechanisch niveau *exact* te reproduceren is met een universele machine [Fey82]. (Feynman gaat uit van een ander, maar equivalent, model van machinaal gedrag dan Turing.)

De grote ontdekking die Turing deed is dat het mogelijk is om een Turing machine te definiëren die alle Turing machines kan simuleren. De reden daarvoor is dat het mogelijk is om de toestanden en symbolen te nummeren. Als we er dan voor zorgen dat de begintoestand een vast nummer krijgt, bijvoorbeeld 1, dan kunnen we een willekeurige Turing machine beschrijven als een rij getallen gescheiden door spaties. Als we dan ook de invoer voor een berekening met dezelfde nummering voor de symbolen coderen als een rij getallen gescheiden door spaties, dan kunnen we de regels en de invoer van de berekening op de strook papier schrijven van een machine die de regels aan het begin van de strook uitvoert op de gecodeerde beschrijving van de strook van de originele machine. Turing liet zien dat een universele machine die dat doet ook werkelijk bestaat. De codering van de regels van een Turing machine voor gebruik met op een universele Turing machine heet een *programma* voor die universele Turing machine. Een universele Turing machine die start met een dergelijk programma en een codering van de oorspronkelijke invoer op de strook, simuleert het gedrag van de oorspronkelijke machine. Turing bewees ook dat er oneindig veel universele Turing machines zijn, meestal met verschillende coderingen voor de regels. De meeste programmeertalen zoals we die nu kennen zoals Pascal, C, Lisp en Prolog zijn theoretisch te gebruiken als coderingssytemen voor universele Turing machines. De universaliteit van deze talen blijkt uit het feit dat er vertalers zijn tussen deze talen. Het is bijvoorbeeld mogelijk om een Pascal programma naar een C programma te vertalen zonder enig verlies aan betekenis. Het resulterende C programma doet precies hetzelfde als het origineel. De meeste machinale vertalers vertalen programmas die zijn gescheven in een abstacte programmeertaal naar een taal die meer lijkt op de lijst regels waarmee we Turing machines gedefiniëerd hebben en die met machinetaal aangeduid wordt. Machinetaal kan direct door een digitale computer uitgevoerd worden.

Omdat we het later behalve over berekeningen ook over machinale processen willen hebben, zullen we ook Turing machines bekijken met drie stroken papier. Op de bovenste strook kunnen tijdens het proces invoersymbolen geschreven worden. De middelste strook dient als kladpapier en wordt op dezelfde manier gebruikt als de strook van een eenvoudige Turing machine. Op de onderste strook kan de machine tijdens het proces uitvoersymbolen schrijven. Welke regel van toepassing is, hangt af van de symbolen die zichtbaar zijn op de bovenste en de middelste strook. Op de bovenste en de onderste strook kan alleen naar rechts bewogen worden of pas op de plaats gemaakt. Naar links bewegen op deze stroken is uitgeloten. Aangezien de bovenste strook nu de invoer bevat, kunnen we de middelste en de onderste strook leeg veronderstellen aan het begin van het proces. Als we een eindig aantal symbolen op de invoerstrook schrijven kunnen we regels toepassen tot de machine een symbool probeert te lezen dat we nog niet geschreven hebben. De uitvoer die tot dan toe geschreven is kan gezien worden als het begin van de uitvoer van het proces op een oneidige rij symbolen waarvan we alleen en beginstuk hebben behandeld. We kunnen meer symbolen op de bovenste strook

schrijven en de machine verder laten rekenen met als resultaat de uitvoer van het proces op het verlengde beginstuk.

In wiskundige terminologie heet een Turing machine met één strook een discrete Turing machine. Een Turing machine met drie stroken met de bovenstaande eigenschappen heet continu. De reden hier voor is dat de invoer van een discrete Turing machine eindig te beschrijven is en, als de machine stopt, de uitvoer ook. Terwijl de uiteindelijke invoer van een continue Turing machine een oneindige rij is, waarop de machine continu kan doorrekenen. Het resultaat kan dan ook weer een oneidige rij symbolen zijn. De verzameling van alle oneindige rijen over een bepaald alfabet is in wiskundige zin een continuüm. Een continue Turing machine wordt ook wel aangeduid met de term monotone Turing machine omdat de uitvoer monotoon toeneemt met de invoer.

Net als universele discrete Turing machines zijn er ook universele continue Turing machines. De enige extra procedure die een universele continue Turing machine moet uitvoeren is het kopieëren van het programma van de invoerstrook naar de middelste strook.

Als we een continue Turing machine gebruiken om het gedrag van een systeem of organisme te modelleren, coderen we de waaremingen van het systeem en schrijven die op de invoerstrook, vervolgens laten we de machine werken tot hij probeert een symbool te lezen dat we nog niet geschreven hebben. De symbolen die in de tussentijd door de machine op de uitvoerstrook geschreven zijn vatten we op als beschrijvingen van acties. Vervolgens bepalen we welke waarnemingen volgen op de beschreven acties. De beschrijving van deze waarnemingen schrijven we weer op de invoerstrook en we laten de machine verder rekenen.

Bij wijze van gedachtenexperiment kan ook de hele wetenschap opgevat worden als lerend systeem. Op de invoerstrook schrijven we dan alle waarnemingen van alle experimenten uit de historie tot nu toe. Als de machine een goede representatie is van het wetenschapsbedrijf, schrijft het op de uitvoerstrook alle experimenten die er gedaan zijn. (Toegegeven, dit is een onwaarschijnlijke aanname. Tenzij de werkelijkheid waarin we leven geïmplementeerd is op een Turing machine. Een aanname die falsifieerbaar is, maar nog niet gefalsifieerd [Fey82].) Niet alleen de interne overwegingen van wetenschappers, maar ook hun communicatie onderling wordt gerepresenteerd door de bewerkingen op de middelste strook. Het publiceren van een theorie is dus een actie van een individuele wetenschapper, maar geen actie van de wetenschap als geheel.

Als we een continue Turing machine gebruiken om een theorie over de omgeving van een lerend systeem te modelleren, coderen we de waarnemingen van het systeem in het alfabet van de *uitvoerstrook* van de Turing machine. We kijken dan of er een reeks symbolen is zodat als we die op de invoerstrook schrijven, de machine exact de beschrijving van de ervaringen reproduceert. Als de beschrijving van de ervaringen eindig is, mag de continue Turing machine op de gegeven invoer meer uitvoersymbolen genereren. Zolang de beschrijving van de ervaringen maar een beginstuk is van de uitvoer. De reeks symbolen op de invoerstrook kan dan opgevat worden als de codering van de randvoorwaarden en eventuele correcties.

We zullen in de volgende paragraaf zien dat deze opzet de mogelijkheid biedt om op een verstandige manier kansen toe te kennen aan theorieën en toekomstige waarnemingen.

3.4 Willekeur

Voor het schatten van kansen is willekeur het centrale begrip. Iets dat zeker is krijgt kans 1, iets dat onmogelijk is krijgt kans 0. Hoe zekerder iets is, hoe dichter bij 1 de geschatte kans zal liggen en hoe onwaarschijnlijker iets is, hoe dichter de geschatte kans bij 0 zal liggen. De afstand van de geschatte kans tot de absolute grenzen 1 en 0 wordt bepaald door de mate

van willekeur in het verschijnsel. Elke regelmaat in het optreden van een verschijnsel leidt tot een betere voorspelling van dat verschijnsel en trekt dus de geschatte kans naar één van beide uitersten toe.

Om kwantitatieve uitspraken te kunnen doen over kansen op verschijnselen op grond van een beperkte hoeveelheid waarnemingen, stellen we ons voor dat we een beschrijving van alle waarnemingen van het systeem tot op een bepaald moment tot onze beschikking hebben. Een *theorie* die deze waarnemingen verklaart kan dan gegoten worden in de vorm van een algoritme, een programma voor een universele Turing machine, dat de beschrijving van de waarnemingen exact reproduceert. In het geval van een theorie die de waarnemingen niet exact voorspelt, kan het algoritme bestaan uit een sub-algoritme dat de voorspelling in het ideale geval uitrekent, aangevuld met aanpassingen die verwerkt worden in het resultaat van het sub-algoritme. Samen moeten deze delen alsnog exact de beschrijving van de waarnemingen reproduceren. Een triviale toepasing hiervan is een theorie die helemaal niets voorspelt. In dat geval bevat de lijst met uitzonderingen een exacte beschrijving van de waarnemingen. Het zal duidelijk zijn dat in dat geval de lengte van een beschrijving van het sub-algoritme dat deze simpele theorie plus de lengte van een beschrijving van de lijst met uitzonderingen ongeveer gelijk is aan de lengte van de beschrijving van de waarnemingen die we proberen te reproduceren. Aan de andere kant als we een niet triviale theorie hebben, bijvoorbeeld de theorie van Newton over de bewegingen van hemellichamen, en we proberen een lijst met waargenomen planeetstanden te reproduceren, dan kunnen we volstaan met een beschrijving van het sub-algoritme (dat hangt niet af van het aantal waarnemingen en kunnen we negeren als het om erg veel waarnemingen gaat) plus een lijst met data en planeetnamen en kleine correcties. De feitelijke planeetstanden hoeven we niet letterlijk in het algoritme op te nemen, want die kunnen uit de overige gegevens gegenereerd worden met behulp van het sub-algoritme dat Newton's mechanica beschrijft. Dat wil dus zeggen dat het algoritme dus significant korter zal zijn dan de oorspronkelijke letterlijke beschrijving van de waarnemingen.

Als we de oorspronkelijke beschrijving en het algoritme beide in het binaire alfabet opschrijven (dat wil zeggen dat we ons tot twee symbolen beperken), dan kunnen we het verschil van de lengte van de oorspronkelijke beschrijving en de lengte van het kortste algoritme dat deze beschrijving exact reproduceert opvatten als de mate van regelmaat die de beschrijving bevat. De lengte van het korste algoritme noemen we de Kolmogorov complexiteit van de (beschrijving van de) waarnemingen [LV93]. Als geen er geen enkel algoritme is dat de beschrijving van de waarnemingen letterlijk reproduceert en dat echt korter is dan de oorspronkelijke beschrijving, dan zeggen we dat de waarnemingen volstrekt willekeurig zijn; we kunnen er geen regelmaat in ontdekken. Dit oordeel hangt overigens af van de specifieke universele Turing machine waarmee we de algoritmen interpreteren. Een reeks waarnemingen kan volstrekt willekeurig zijn in termen van een universele Turing machine maar toch regelmatig zijn in termen van een andere. Het is wel zo dat als een oneindige rij symbolen oneindig veel beginstukken heeft die willekeurig zijn in termen van een universele Turing machine machine, dan heeft die rij oneindig veel willekeurige beginstukken in termen van elke andere universele Turing machine.

Voor een continue universele Turing machine kunnen we nu de kans bekijken dat als we kop-of-munt gooien voor de invoersymbolen, de uitvoer een (eventueel verlengde) exacte reproductie van de beschrijving van de waarnemingen is. Deze kans ligt dicht bij twee tot de macht min de Kolmogorov complexiteit van de beschrijving van de waarnemingen in termen van de gegeven continue Turing machine. Deze kansmaat kan met behulp van Bayesiaanse statistiek gebruikt worden voor het maken van inductieve afleidingen. Om deze reden heeft Ray Somo-

noff onafhankelijk van Kolmogorov dezelfde complexiteitsmaat gedefiniëerd [Sol64]. Aangezien Gregory Chaitin [Cha87] ook weer onafhankelijk van Kolmogorov en Solomonoff dezelfde definitie opstelde, is de volle naam van deze complexiteitsmaat: Kolmogorov-Solomonoff-Chaitin complexiteit.

Onder de aanname van de Church-Turing these kunnen we nu een willekeurige universele continue Turing machine zien als een theorie voor het beschrijven van waarnemingen. De kans op een specifieke waarneming a gegeven een historie d is dan de conditionele kans die volgens de standaard formule uit de kansrekening gelijk is aan $\mu(a \mid d) = \mu(d\,a)/\mu(d)$. Waarbij μ de eerder beschreven kansmaat is. Merk op dat volstrekt tegenstrijdige waarnemingen een positieve kans kunnen hebben, zolang de som van de kansen maar gelijk is aan 1.

De zwakke plek van deze opzet is niet zozeer dat de keuze van de universele machine arbitrair is. Dat heeft nauwelijks effect op de geschatte kansen op de lange termijn. De moeilijkheid is dat de Kolmogorov complexiteit voor alle behalve een eindig aantal beschrijvingen onberekenbaar is. Het is principieel onmogelijk om het kortste algoritme te berekenen dat een bepaalde uitvoer oplevert. Een voor de hand liggende methode om dit probleem te omzeilen is om te eisen dat het begin van de invoerstrook een algoritme beschrijft dat binnen een analyseerbare klasse van algoritmen valt. Een dergelijke klasse noemen we dan een *hypotheseklasse*. Bij veel programma's als BACON, MYCIN en SOAR is een dergelijke hypothese klasse expliciet of impliciet gegeven.

De kans op een reeks waarnemingen gegeven een hypotheseklasse kan met behulp van de kansrekening eenvoudig bepaald worden. Als de hypotheseklasse overzichtelijk genoeg is kan de kans zelfs uitgerekend worden. Inductieve inferentie is dan de taak om het programma uit de hypotheseklasse te vinden waarvoor de resulterende beschrijving van de waarnemingen zo kort mogelijk is. Als er dan meerdere mogelijkheden zijn verdient het aanbeveling om meer waarnemingen te doen. Deze methode om kansen in te schatten op basis van een overzichtelijke hypotheseklasse is ontwikkeld door Jorma Rissanen [Ris78, Ris89] en onafhankelijk van Rissanen door de informatie-theoretici Wallace en Boulton [WB68]. Het onderliggende principe om de lengte van de codering van een beschrijving van de waarnemingen zo kort mogelijk te maken wordt in het Engels aangeduid als het *Minimum Description Length Principle* (MDL).

De allermooiste variant zou zijn om de hypotheseklasse langzaam ingewikkelder te laten worden. Langzaam genoeg om de rekentijd overzichtelijk te houden en snel genoeg om in de limiet dezelfde kansen aan waarnemingen toe te kennen als de universele Turing machine. De haalbaarheid van deze mogelijkheid is voor zover ik weet nog onderwerp van studie.

4. Machinaal Leren

Op basis van de voorgaande paragraaf zal ik nu schetsen hoe een formele beschrijving van lerend gedrag er uit zou kunnen zien.

In paragraaf over willekeur lag de nadruk op het beschrijven van waarnemingen. Echter een zeer klein deel van de mensheid legt zich geheel toe op het beschrijven van waarnemingen. En zelfs dat deel publiceert deze beschrijvingen in plaats van deze geheim te houden. Een belangrijk deel van een lerende systemen is belast met het genereren van uitvoer. Uit Psychologisch onderzoek kan worden afgeleid dat leren nog niet meevalt als werkelijke interactie met de omgeving niet mogelijk is.

De moeilijkheid bij het analyseren van acties van een lerend systeem is het te hanteren criterium. In onderzoek binnen de Artificiële Intelligentie is het gehanteerde criterium vaak extern. Hoe vaak geeft een medisch expert systeem een correcte diagnose? Is de ontleed-boom

van een natuurlijke taal herkenner correct? Aan de andere kant is het criterium voor mensen die leren uiteindelijk intern. Mensen vinden bepaalde omstandigheden prettiger dan andere en proberen zich zo lang en zo vaak mogelijk prettig te voelen. Het lijkt niet mogelijk om het begrip 'prettig' in een zelfde formeel jasje te duwen als we met het begrip 'kans' gedaan hebben. Aan de andere kant kunnen we wel zeggen dat voor een lerend systeem 'leren' belangrijk is. Dit suggereert dat we voor een lerend systeem de toename van de lengte van de optimale hypothese als criterium voor de acties van dat systeem zouden kunnen hanteren. In deze context is het denkbaar dat er voor een gunstige hypotheseklasse een algoritme denkbaar is dat acties genereereert zodat de verwachte toename van de lengte van de optimale hypothese maximaal is. Dat wil zeggen de acties van het systeem zijn gericht op het blootleggen van mogelijke regelmatigheden die nog niet in de waarnemingen uit het verleden besloten liggen.

Als een dergelijk algoritme gevonden zou worden voor een langzaam in complexiteit toenemende hypotheseklasse zoals beschreven aan het eind van paragraaf 3.4, dan zou dat algoritme met recht een universeel leer-algoritme mogen heten.

In dit model is het dus niet zo dat de interne processen van een lerend systeem in de voorgaande zin altijd tot uitdrukking komt in het gedrag, de uitvoer, van het systeem. Het is heel goed mogelijk dat de omgeving van dien aard is dat zelfs een wiskundig optimaal lerend systeem niet genoeg systematiek kan ontdekken om een verandering van gedrag te rechtvaardigen. Bijvoorbeeld, als de reeks invoersymbolen vanaf een zeker moment constant wordt, dan zal het optimale gedrag bestaan uit het volkomen willekeurig genereren van acties. Dit gedrag zal pas veranderen zodra de reeks invoersymbolen een andere regelmaat gaat vertonen, die correlaties vertoont met de gegenereerde uitvoer. Het voorgestelde model van lerende systemen is dus geen *behavioristisch* model. Het feit dat een systeem zich aanpast aan een gecompliceerde en veranderende omgeving maakt het tot een lerend systeem. Let wel, een omgeving die willekeurige waarnemingen genereert is net zo min een gecompliceerde omgeving als een omgeving die steeds dezelfde waarnemingen genereert.

Het toetsen van dit model in Psychologische zin aan lerende organismen lijkt vooralsnog moeilijk. Daarvoor is een theorie nodig die de stimuli die van invloed zouden kunnen zijn op het gedrag van een organisme vertaalt in een overzichtelijke gegevensstroom. Gezien het feit dat zowel Kolmogorov complexiteit en het MDL principe uitgaan van het *exact* reproduceren van deze stroom ligt het evenaren van het menselijk leervermogen op een dergelijke stroom gegevens nog ver buiten bereik. Mogelijkerwijs kunnen binnen sterk gereduceerde omgevingen zinvolle vergelijkingen tussen werkelijk lerende organismen en lerende algoritmen gemaakt worden.

Het voorgestelde model gaat er vanuit dat kennis verzamelen over de omgeving de basis vormt van lerend gedrag. Onder de aanname dat de Church-Turing these correct is kan alle kennis over de omgeving gegoten worden in de vorm van een algoritme dat de informatie over de omgeving reproduceert gegeven een beschrijving van randvoorwaarden en correcties. Evaluatie criterium voor een lerend systeem is dan de effectiviteit van het algoritme dat de kennis representeert. Hoe korter het algoritme plus de beschrijving van de randvoorwaarden en correcties is, hoe meer regelmaat de waarnemingen blijken te vertonen. De acties van het lerende systeem kunnen vervolgens zo gekozen worden dat de verwachting van de toename van de lengte van het algoritme plus be beschrijving van de randvoorwaarden en correcties gemaximaliseerd wordt.

Voor het wetenschapsbedrijf betekent dat dus dat experimenten waarvan de uitkomsten onvoorspelbaar zijn de meeste kans op nieuwe informatie over de werkelijkheid zijn. Verder moet binnen de wetenschap continu gezocht worden naar theoriën waarmee het totaal van

alle resultaten van wetenschappelijke experimenten tot nu toe korter beschreven kan worden dan met de bestaande theoriën. Dat kan door een theorie bondiger te formuleren, door theoriën te integreren, door een theorie te ontwikkelen die randvoorwaarden genereert uit andere randvoorwaarden zodat ze niet meer gespecificeerd hoeven te worden, of door een theorie te ontwikkelen die minder correcties behoeft. Voorwaarde is wel dat de voorgestelde verbetering al met al niet langer mag zijn dan het origineel. Een ander belangrijk punt is dat bij het vergelijken van twee theoriën *eerst* overeenstemmig bereikt wordt over 1) de te reproduceren reeks gegevens en 2) het onderliggende begrippenkader waar beide theoriën in geformuleerd (desnoods vertaald) moeten worden. Anders zijn paradoxen niet te vermijden.

REFERENTIES

[Cha87] Gregory J. Chaitin. *Algorithmic Information Theory.* Cambridge University Press, 1987.

[DH81] Daniel C. Dennet and Douglas R. Hofstädter, editors. *The Mind's I.* Basic Books, 1981.

[Fey82] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6/7):467–488, 1982.

[LV93] Ming Li and Paul M.B. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications.* Springer Verlag, 1993.

[Mac81] David Macaulay. *De Stad: Het verhaal van de Romeinse stedebouw.* Ploegsma, Amsterdam, 1981.

[Mis39] Richard von Mises. *Probability, Statistics and Truth.* William Hodge & Co., London, 1939.

[Pop68] Karl Popper. *The Logic of Scientific Discovery.* Harper Torch Books, New York, NY, 1968.

[Ris78] Jorma J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978. Journal of the IFAC.

[Ris89] Jorma J. Rissanen. *Stochastic Complexity in Statistical Enquiry.* World Scientific, Singapore, 1989.

[Sea92] John R. Searle. *The Rediscovery of the Mind.* MIT Press, Cambridge, MA, 1992.

[Sol64] Ray J. Solomonoff. A formal theory of inductive inference, part I. *Information and Control*, 7:1–22, 1964.

[WB68] C.S. Wallace and D.M. Boulton. An information measure for classification. *Computer Journal*, 11:185–195, 1968.

[Wit18] Ludwig Wittgenstein. *Tractatus Logico-Philosophicus.* ?, 1918.