

INHOUD

Waarom dit boek er moest komen 7

Inleiding 9

Vervangen door een computer • Je zou niet de eerste zijn • Dit boek

1. Wat een computer is 17

Alan Turing • Input, compute, output • De transities zijn zelf een vorm van input • De RAM-machine • Nullen en enen • Booleaanse algebra • Rekenen met transistors • Toepassingen • Conclusie

2. Hoe een computer je verstaat 40

Talen • Verschillen tussen programmeertalen • ‘Hello world’ • Variabelen, operaties en condities • Functies en objecten • Goede code is leesbaar én efficiënt • Conclusie

3. Hoe computers iets onthouden 65

De basis van dataopslag • Van variabelen naar geheugen • Relatieve en niet-relatieve databases • Data herstructureren • Indexen • Hashing • Encryptie • Nog meer encryptie • De blockchain • Conclusie

4. Hoe computers met elkaar praten 92

Vragen en antwoorden • De backend • GET en POST • API-calls en schaalbare software • De cloud • Websites, apps en het Internet of Things • Hacks • Conclusies

5. Hoe computers leren – de basics 117

Recommendersystemen • De kracht van formalisatie • Uitgebreidere recommendersystemen • Omgaan met onzekerheid: kansberekening • Drie manieren van leren • Conclusies

6. Hoe computers leren door ervaring 143

Voorspellen • Modellen bouwen • Categorische uitkomsten modelleren • Uitbreidingen naar andere soorten modellen • De bomen en het bos • Neurale netwerken • Conclusies

7. Hoe computers leren door patronen te herkennen 170

Data zonder labels • Patronen herkennen op basis van afstanden • Patronen herkennen op basis van kansen • Andere manieren van patronen herkennen • Zekerheid • Conclusies

8. Hoe computers interactief leren 188

Exploreren en exploiteren • Efficiënt leren • Bayes brengt uitkomst • Context • Staten, acties en rewards • Causaliteit • De potentie en het gevaar • Conclusies

9. Big data 208

Wat is big data? • Omgaan met big data • Streaming modelleren • Onzekerheid in datastromen • Gaat iemand nog iets met al die data doen? • Risico's • Conclusies

10. De zelfrijdende auto 229

Stanleys hardware • Stanleys softwarearchitectuur • Hoe Stanley de weg 'ziet' • Hoe Stanley de route bepaalt • De Urban Challenge • De toekomst van zelfrijdende auto's • Conclusies

Nawoord 245

Dankwoord 249

Boter-kaas-en-eieren 251

Over de auteur 259

Literatuur 261

Register 267

INLEIDING

Ik schrijf dit boek op mijn laptop, terwijl mijn huis op temperatuur wordt gehouden door mijn 'slimme' thermostaat en mijn tablet 'speciaal voor mij' muziek selecteert en die draadloos naar mijn stereo stuurt. Dit zijn allemaal computers. In ons privéleven spelen ze een steeds grotere rol. Zo rekenen we op ze om ons op tijd wakker te maken, om ons naar onze bestemming te navigeren en om ons te adviseren over hoe we een gezond leven moeten leiden. En ook op ons werk kunnen we niet om ze heen.

Maar hoe werken computers eigenlijk? Hoe werken de onderliggende netwerken? Hoe kunnen computers op basis van data beslissingen nemen? Best goede vragen. Vooral omdat een goed begrip van de antwoorden inzicht geeft in hoe onze huidige wereld in elkaar zit.

Misschien denk je: zolang die technologie maar werkt, maakt het mij niet uit hóé die werkt. Die houding is echter op z'n best niet zo verstandig en in het ergste geval gevaarlijk: je hebt waarschijnlijk een baan waarin je werkt met technologie, waarin je beslissingen neemt over technologie of waarin je af en toe technologie van anderen inkoop of gebruikt. En anders doe je dat wel in je privéleven: als je een nieuwe telefoon of een router voor je draadloze internet koopt, als je surft op internet of een zoekmachine gebruikt, als je een bestand verstuurt via WeTransfer of deelt via Dropbox, als je sms't of whatsapppt. En wat dacht je van die aankoop die je gisteren online deed? Of van je auto? Op dit moment stuur je waarschijnlijk nog zelf, maar het lijkt erop dat dat in de


komende jaren gaat veranderen en je de verantwoordelijkheid uit handen moet geven aan een computer. Dan is het wellicht goed om te weten hoe dat ding werkt.

Het is prettig om te snappen hoe dit allemaal kan, wie er wel en niet meekijkt met je transactie en hoe je je veiligheid en privacy daarbij enigszins kunt waarborgen. Als je meer weet over technologie, kun je net iets beter met de huidige wereld overweg, en al helemaal met die van morgen.

Wat doe je bijvoorbeeld als de programmeur die je hebt gevraagd een extra functie voor een betalingssysteem voor je te programmeren zegt: ‘Dat kost 2 ton om te ontwikkelen’? Leg je je daarbij neer en maak je klakkeloos het geld over? Of zeg je: ‘Als we een hash maken van de html van de pagina en die via een REST-call toevoegen aan onze logging zijn we er toch? Dat lijkt me nog geen vijf minuten werk.’ Zo zou je in een paar seconden duizenden euro’s besparen. Alleen dat al lijkt me de moeite waard.

Dit boek leert je de basiswetten van digitale technologie, zodat je zelfstandig, zonder blind te hoeven vertrouwen op een of andere CTO of ICT’er, beslissingen kunt nemen over deze technologie. Ik bouw de kennis stap voor stap op; ik begin bij de werking van computers, ga vervolgens naar netwerken, dan via data naar *machine learning* en uiteindelijk naar kunstmatige intelligentie. Ik schroom daarbij niet af en toe de diepte in te duiken. De technologie die wij tegenwoordig als vanzelfsprekend beschouwen, is gestoeld op een aantal van de baanbrekendste wetenschappelijke inzichten, opgedaan door genieën als Alan Turing, Claude Shannon, John von Neumann en dominee Thomas Bayes.* Als het erg ingewikkeld wordt, geef ik dat in de kantlijn aan (met het volgende

* Je kunt natuurlijk ook zelf lezen wat deze mensen hebben geschreven. In de voetnoten geef ik soms extra informatie, maar verwijs ik vooral naar origineel materiaal. Een introductie op Turings werk over computers vind je in Petzold, 2008, Von Neumanns computer wordt besproken in Godfrey & Hendry, 1993, Bayes legt zijn theorie uit in Bayes & Price, 1763, en Shannon die van hem in Shannon & Weaver, 1963. Alle de moeite waard, niet allemaal makkelijk.

mooie plaatje: , zodat je weet dat je je op dat stukje even extra moet concentreren (of het eventueel kunt overslaan). Moeilijk? Wellicht, her en der, maar niemand heeft ooit gezegd dat het makkelijk is om in een paar seconden duizenden euro's te besparen.

Ik kan je niet leren om computers vanuit het niks zelf te bouwen. Maar ik kan je wel laten begrijpen wat er gebeurt als digitale technologie in actie komt. Daardoor leer je beter inschatten wat een computer wel en niet kan, wat daarvoor nodig is en hoeveel tijd het ongeveer kost om hem iets te laten doen. Je kunt dan ook beter inschatten waar je moet beginnen als de technologie niet doet wat je dacht dat ze zou doen. En je kunt de charlatans – en daar zijn er veel van – die dingen roepen als: 'Als we big data gebruiken, wordt het een succes', of: 'Wat jij nodig hebt: de cloud!' van repliek dienen.

Er was een tijd dat de meeste mensen niet konden lezen. Inmiddels is dit voor iedereen een onmisbare vaardigheid geworden. Voor basiskennis over digitale technologie geldt tegenwoordig hetzelfde: zonder dat begrip gaat de wereld voor een belangrijk deel aan je voorbij. Er is alleen wel een groot verschil tussen lezen en digitale technologie begrijpen. Lezen kun je op een gegeven moment, en dan kun je voortaan alle geschreven tekst ontcijferen. Er wordt echter telkens nieuwe technologie ontwikkeld; computers kunnen steeds meer, en de slimste mensen ter wereld werken continu aan nieuwe manieren om ze tot nog meer in staat te stellen. Bijblijven is daarom lastig, en het wordt steeds lastiger: als je de basis al niet hebt, is het steeds moeilijker om de nieuwe ontwikkelingen te plaatsen. Velen van ons zijn eigenlijk al te laat: we zijn wel gebruikers van technologie, maar we begrijpen er weinig van. Dit boek geeft je de basis waardoor je vanaf nu weer mee kunt doen.

Vervangen door een computer

Veel mensen hebben moeite om te begrijpen wat computers kunnen en hoe ze werken. Dat is niets om je voor te schamen. Ons begrip van digitale technologie is tot nog toe vooral gestoeld op ana-

logieën. Een bekende is om de desktopcomputer te zien als een bureaublad met mappen waar allerlei bestanden in zitten. Deze analogieën zijn bedacht om ons de technologie te laten gebruiken, niet om ons de technologie daadwerkelijk te laten begrijpen. Ooit was dat handig, maar nu de hele wereld digitaal wordt, staat de analogie een goed begrip soms juist in de weg. Het resultaat is dat slechts enkelen van ons diepgaande kennis hebben van specifieke onderdelen van de technologie. En er is eigenlijk niemand die precies kan vertellen hoe alle afzonderlijke delen van bijvoorbeeld een zelfrijdende auto werken.

Dat is een behoorlijk beangstigende gedachte, zeker als je bedenkt dat computers onderling kennis veel sneller delen dan wij: als er één computer zo geprogrammeerd is dat hij kan navigeren, dan kunnen vanaf dat moment alle computers in principe navigeren. De gevolgen hiervan worden vaak onderschat.

Ik mag af en toe voor een clubje mensen (een branchevereniging of iets dergelijks) vertellen over de impact van technologie en kunstmatige intelligentie op hun banen. De laatste tijd deed ik dat bijvoorbeeld bij groepen accountants, verzekeraars, bankiers en piloten. En steevast zei men: 'Ja, de technologie kan al veel, en delen van mijn werk zijn al overgenomen door computers, maar mijn baan is zo complex en vraagt bovendien om intuïtie, kennis en gevoel, dat zal een computer nóóit kunnen.'

Dit is een heel gevaarlijk wereldbeeld. Ik bestrijd niet dat accountants, verzekeraars, bankiers en piloten lastige banen hebben. En ik bestrijd ook niet dat ze allerlei complexe beslissingen nemen, beslissingen die op dit moment nog niet uit te besteden zijn aan computers. Maar stel nu dat een groep geniale ingenieurs een paar jaar aan het werk zou gaan om een computer te maken die een accountant kan vervangen, of een verzekeraar, of een piloot, en dat ze na vele jaren hard werken slagen in hun missie. Vanaf dat moment kunnen alle computers in de hele wereld, per direct, alle accountants, piloten en verzekeraars vervangen. Anders dan mensen hoeven computers niet te worden opgeleid of ervaring op te doen: als de technologie er is, dan kan deze verspreid (simpelweg gekopieerd) worden. Dus zo gauw één com-

puter beter is in iets dan mensen, dan zijn ze het in theorie allemaal.

Stel dat je pas gaat letten op wat een computer kan op het moment dat geniale ingenieurs al bezig zijn jouw baan te digitaliseren. Misschien ben je arts en heb je totaal niet meegekregen hoe IBM met supercomputer Watson probeert om langzaam maar zeker een computer te programmeren die betere diagnoses kan stellen dan jij.* Het zal nog wel even duren voordat het zover is, maar dan kun je als arts ook meteen inpakken. Zelfs fysiek contact en een luisterend oor zijn door kunstmatige intelligentie en andere slimme technologieën te vervangen.† Met wat meer begrip van bestaande en toekomstige technologie kun jij tenminste op tijd op zoek gaan naar een alternatief tijdverdrijf. Onderschat hierbij vooral niet waar computers toe in staat zijn. John von Neumann zei al: ‘U blijft erbij dat er iets is wat een machine niet kan. Maar als u mij precies kunt vertellen wat het is dat een machine niet kan, dan kan ik er een maken die precies dat kan.’‡

Je zou niet de eerste zijn

Goed, je moet dus weten hoe technologie werkt om in te kunnen schatten wat het effect ervan zal zijn op je huidige baan. Maar niet alleen voor jou persoonlijk is het heel belangrijk te weten in welke wereld we leven; juist in het bedrijfsleven en bij overheden is het van essentieel belang dat degenen die de beslissingen nemen over technologie weten waar ze het over hebben.

Te vaak wordt er door technologiebedrijven en door een leger consultants voor veel te veel geld technologische onzin verkocht.

* Pogingen in deze richting zijn al jaren gaande, dus als arts ben je er echt laat bij als je nu pas mee gaat kijken. Zie bijvoorbeeld Weaver, Ball, Kim, Kiel e.a., 2016 voor een recent overzicht (inclusief Watson).

† Wederom geen toekomstmuziek; zie: Broekens, Heerink & Rosendal, 2009.

‡ De quote uit 1948 is naar verluidt: ‘*You insist that there is something that a machine can’t do. If you will tell me precisely what it is that a machine cannot do, then I can always make a machine which will do just that.*’ Tot mijn spijt heb ik geen schrijven van Von Neumann zelf kunnen vinden waarin de uitspraak staat.

ICT-projecten lopen schandalig uit de hand omdat de mensen die de projecten managen essentiële kennis missen. Recentelijk maakte *Nieuwsuur* een inventarisatie van mislukte ICT-projecten bij de overheid. Daarin waren bijvoorbeeld de problemen met Diginotar opgenomen. Deze dienst verkocht veiligheidscertificaten voor een fors aantal overheidssites, maar werd in juli 2011 zelf gekraakt door Iraanse hackers. De overheid zegde het vertrouwen op en moest halsoverkop op zoek naar andere certificaten. Ook waren er veiligheidsissues met DigiD, de vingerafdrukken in paspoorten, het systeem dat de politie gebruikt om processen-verbaal in vast te leggen en de ov-chipkaart.*

Het is overigens zeker niet alleen de overheid waar nieuwe technologie wel eens problemen oplevert. Het is voor iedereen belangrijk een kloppend beeld te hebben van hoe computers werken.

Dit boek

In dit boek neem ik je mee op reis door de moderne technologie: van de ontwikkeling van de eerste computers tot big data, kunstmatige intelligentie en machine learning.

We beginnen met wat je met nullen en enen kunt doen. Alan Turing bedacht ooit dat machines konden rekenen, waarna door middel van de transistor en nullen en enen de basis werd gelegd voor onze huidige computers. Dit is de basis van de computer waar we zelf eigenlijk nooit meer over hoeven na te denken, maar die wel nuttig is om te kennen: waarom werken computers eigenlijk met nullen en enen? En zijn alle computers tot hetzelfde in staat? Wellicht moet je er even in komen – houd vol, het is het waard!

* Voor de volledige lijst zie: <http://nos.nl/nieuwsuur/artikel/2029208-mislukte-ict-projecten-en-andere-ict-missers.html>, en voor het parlementair onderzoek naar het functioneren van ICT-projecten binnen de overheid zie: www.tweedekamer.nl/sites/default/files/field_uploads/33326-5-Eindrapport_tcm181-239826.pdf.

Voortbouwend op deze basis bekijken we hoe we überhaupt praten met computers. In dit tweede hoofdstuk leg ik uit hoe we computers tegenwoordig programmeren en wat er wel en niet met een computer te bespreken valt. Dit hoofdstuk geeft je handvatten om mee te praten met programmeurs, mee te denken over softwarestructuren en geïnformeerde beslissingen te nemen over programmeerprojecten.

In hoofdstuk 3 bekijken we hoe computers kennis onthouden. Hoe kunnen we informatie opslaan in databases en hoe verschillen bepaalde soorten databases van elkaar? Dit hoofdstuk behandelt SQL- en NoSQL-databases, en alle huidige buzz-termen – Hadoop, Spark, Map/Reduce – die vaak door consultants gebezigd worden. Na het lezen van dit hoofdstuk begrijp je hoe lastig het is om informatie veilig op te slaan, te koppelen of te veranderen.

In hoofdstuk 4 bekijken we hoe computers tegen elkaar praten. Nu behandelen we de moderne infrastructuur: is die cloud nu echt zo bijzonder? En hoe werkt internet eigenlijk? In dit hoofdstuk wordt duidelijk hoe servers verschillen van laptops, hoe apps verschillen van websites, en wat er nu wezenlijk nieuw is aan het Internet of Things (IoT). Een goed besef van de structuur van dit netwerk van computers helpt je om te begrijpen waar de zwakheden zitten.

In de hoofdstukken 5 tot en met 8 gaan we ons bezighouden met machine learning: hoe kan het dat computers dingen ‘leren’? Hoe kan een computer bedenken wat onze voorkeuren zijn, bijvoorbeeld welke muziek we leuk vinden? En hoe kan het dat er computers zijn die beter kunnen schaken dan grootmeesters? Deze hoofdstukken leren je wat een computer wel en niet kan leren, en hoe dit werkt.

We beginnen in hoofdstuk 5 met wat simpele wiskundige formalisaties van problemen (en wat statistiek, sorry) om te laten zien wat de kracht is van zelfs heel simpele algoritmen. Daarna bespreken we in hoofdstuk 6 hoe computers kunnen leren van voorbeelden en in hoofdstuk 7 hoe computers patronen herkennen, om in hoofdstuk 8 te eindigen met *reinforcement learning*: de huidige manier waarop veel computers zelf interactief leren door

trial-and-error.* Natuurlijk focussen we in dit deel van het boek ook op kunstmatige intelligentie (of *artificial intelligence*, kortweg AI). Dit is een veelbesproken onderwerp: de groten der aarde (Stephen Hawkins, Elon Musk et cetera) waarschuwen ons stervelingen voor de impact van deze technologie.† We sluiten dit deel van het boek af door te kijken hoe computers omgaan met big data; dit doen we in hoofdstuk 9.

In hoofdstuk 10 zien we hoe dit alles samenkomt. Hier bekijken we een van onze mooiste toepassingen van moderne computers: de zelfrijdende auto. Als je zover bent gekomen, kun je al het voorgaande integreren en begrijpen hoe we een computer onbegrijpelijke dingen kunnen laten doen.

Ten slotte kijken we in het nawoord terug op wat we hebben besproken, en gaan we in op de maatschappelijke impact: willen we wel dat ‘onze’ data gebruikt wordt door computers? Is onze data daar wel veilig? En wordt onze privacy niet geschonden?

Na het lezen van dit boek heb je een redelijk idee van de technologieën die gebruikt worden om het internet draaiende te houden en om de zelfrijdende auto op de weg te houden. Je kunt dan nog lang niet zelf zo’n auto bouwen, maar hopelijk wel inschatten waar de technologie naartoe gaat.

* Wil je al direct meer diepgang, dan hierbij alvast een paar interessante boeken over machine learning en kunstmatige intelligentie; verderop in het boek verwijs ik naar specifieke artikelen die de methoden die ik bespreek presenteren. Maar voor nu zie bijvoorbeeld: Bishop e.a., 2006; Hastie, Tibshirani & Friedman, 2013; Sutton & Barto, 1998.

† Voor de indringende waarschuwing, ondertekend door velen, zie: <https://futureoflife.org/ai-open-letter/>.