

KU LEUVEN

FACULTEIT PSYCHOLOGIE EN
PEDAGOGISCHE WETENSCHAPPEN

Onderwijs en digitalisering

Een exploratief onderzoek naar de sturende werking van
algoritmes

Masterproef aangeboden tot het
verkrijgen van de graad van
Master of Science in de
pedagogische wetenschappen
Door
Stijn Coussement

promotor: prof. dr. Maarten Simons

2018

KU LEUVEN

FACULTEIT PSYCHOLOGIE EN
PEDAGOGISCHE WETENSCHAPPEN

Onderwijs en digitalisering

Een exploratief onderzoek naar de sturende werking van
algoritmes

Masterproef aangeboden tot het
verkrijgen van de graad van
Master of Science in de
pedagogische wetenschappen

Door

Stijn Coussement

promotor: prof. dr. Maarten Simons

2018

Samenvatting

Digitalisering dringt door tot ieder domein van de wereld en beïnvloedt onvermijdelijk dat domein. In het onderwijs is dit niet anders. Hoewel de digitalisering ondertussen ingeburgerd is, is het opvallend dat de cruciale rol van algoritmes hierin onderbelicht bleef. De laatste decennia is hier groeiende aandacht voor in internationaal onderzoek maar dit is in 'pedagogenland' nog relatief onbekend.

De voorliggende masterproef heeft een tweeledige opzet. Het *eerste deel* rapporteert over het resultaat van een literatuuronderzoek waarin de onderzoeksvraag 'Wat is de werking van een algoritme?' richtinggevend was. De onderzoeksvraag ontstond uit de interesse om de ontpoppende onderzoeksliteratuur in kaart te brengen. Het eerste hoofdstuk situeert algoritmes op conceptueel niveau en reikt hiervoor definities aan. Verder plaatst het algoritmes in relatie tot digitale apparaten. Het hoofdstuk eindigt met de vaststelling dat algoritmes socio-technisch zijn. Het tweede hoofdstuk start met een beschrijvende opsomming van de kenmerken van algoritmes om vervolgens het concept van een black box toe te lichten. Een black box is iets waar enkel de input en output van gekend is, zonder dat de werking gekend is. Het hoofdstuk eindigt met een aftoetsing van de kenmerken van algoritmes aan het concept van een black box om vast te stellen dat een algoritme volgens sommige onderzoekers een black box is en volgens andere niet. Het derde hoofdstuk behandelt de macht en werking van het algoritme. Dit hoofdstuk beschrijft de macht van het algoritme als een nieuwe vorm van macht die mogelijk wordt door de digitale technologie. Het geeft ook een situering van waar deze macht geanalyseerd kan worden. De werking van het algoritme verwijst naar algoritmische technieken die het concrete functioneren van bepaalde algoritmes beschrijven. Het vierde hoofdstuk sluit het eerste deel af door in te gaan op gebruikte onderzoeksmethodes in de literatuur. Het *tweede deel* rapporteert over een exploratief, empirisch onderzoek dat de volgende onderzoeksvraag beantwoordt: Welke vormen van sturing ontstaan er in een klaspraktijk onder invloed van een algoritme? Het onderzoek is exploratief van aard omdat hierover weinig onderzoek voor handen is waarop we ons konden baseren. De gebruikte onderzoeksmethode (namelijk ANT en visuele netwerk analyse) liet toe om de klaspraktijk enerzijds te analyseren als een compositie aan de hand van vijf dimensies en anderzijds operaties en mechanismen op te sporen. Dit laatste beschrijft de sturing door algoritmes. Het onderzoek toont een onderzoekspiste aan met nog veel potentieel voor verder onderzoek in het onderwijskundig en pedagogisch veld. Bovendien toont het een vorm van sturing aan die vandaag meer dan ooit aanwezig is, namelijk een sturing door algoritmes.

Dankwoord

Het indienen van deze masterproef vormt het sluitstuk van mijn masteropleiding waarin ik twee jaar aan deze masterproef mocht werken. Gelukkig stond ik er tijdens dit werken niet alleen voor. Daarom neem ik hier graag de plaats om een aantal mensen te bedanken voor wat ze hebben gedaan – maar nog meer voor wie ze zijn.

In de eerste plaats gaat mijn dank uit naar mijn promotor professor Simons. Bedankt voor uw tijd en uw geduldige en vertrouwensvolle houding waardoor ik na ieder gesprek de gang verliet met een nieuwe portie vertrouwen in mijn masterproef. Ook had ik na ieder gesprek ‘goesting’ om meteen verder te werken aan mijn masterproef. Uw begeleiding door mee na te denken en vragen te stellen lagen hiervoor ongetwijfeld aan de basis. Bedankt om – zonder enige twijfel – het eindresultaat naar een hoger niveau te brengen.

Daarnaast gaat mijn dank uit naar mijn metgezellen, Tine en Liam. Samen reisden we van de lerarenopleiding via de KULAK naar Leuven om onze honger naar kennis over het onderwijs te stillen. Bedankt om dit een aangename en boeiende reis te maken. Liam, mijn ‘partner in crime’ op het gebied van (lezingen en debatten over) onderwijs, bedankt voor al de filosofische gesprekken en discussies die we hebben gevoerd over onderwijs en in het laatste jaar bijzonder veel over de digitalisering ervan.

Graag bedank ik ook de externe lezers van mijn masterproef: Michèle, Laura en Liam. Bedankt voor de inhoudelijke en taalkundige feedback.

Yana, ook jij bedankt voor mij toe te laten tot jouw klas en mijn vragen in geuren en kleuren te beantwoorden. Daarnaast, wil ik ook Anton Derks en Bart Colpaert bedanken om tijd vrij te maken voor een verkennend interview. Ook al heeft de masterproef gaandeweg het traject een andere richting ingeslagen.

Elizabeth, bedankt voor de pauzes tussen het werken aan de masterproef zoveel gezelliger en grappiger te maken.

Aan het thuisfront bedank ik graag:

Mijn vrienden Pieter, Gert-Jan en Tom. Bedankt voor mij gedurende twee jaar ieder weekend te tonen dat studierichtingen en/of beroepen geen criteria zijn voor vriendschap. Bedankt om het zo af en toe eens één avond niet over onderwijs te hebben.

Mijn grote broer, Steven. Bedankt voor steeds een voorbeeld van werklust te zijn.

Mijn ouders. Bedankt voor de kans die jullie mij gaven om deze studies aan te vatten en er mij op tijd en stond aan helpen herinneren dat ik niet altijd zo streng moet zijn voor mezelf. Bedankt voor alles.

Eigen inbreng en toelichting aanpak

Tijdens het eerste kennismakingsgesprek wees prof. dr. Simons me op een aantal interessante artikels en auteurs die de thematiek van de masterproef behandelen. Dit was heel breed waardoor ik vanaf het eerste gesprek meteen vrij was om zelf richting te geven aan de masterproef. De aangeboden literatuur ging hoofdzakelijk over digitaal onderwijsbeleid en algoritmisch bestuur waarbij ik gefascineerd raakte door de rol van algoritmes hierin. Prof. Simons gaf me het vertrouwen om hier verder rond te werken en liet me toe hierop te focussen in deze masterproef. Ik schuimde bijgevolg de kroegen van het internet af om mij volledig te verliezen in de wereld van algoritmes.

Prof. Simons bracht me in contact met mensen die werken op de dienst Strategisch Onderwijsbeleid en gaf me zo de kans om de rol van algoritmes op beleidsniveau te onderzoeken. Ondanks deze interessante opportuniteit, bleef ik (waarschijnlijk als gevolg van mijn vooropleiding aan de lerarenopleiding) gefascineerd door de klaspraktijk zelf. Opnieuw kreeg ik hiervoor het vertrouwen van prof. Simons om de klaspraktijk te onderzoeken. Methodologisch vond ik hiervoor inspiratie bij een voormalige doctoraatsstudent van prof. Simons waarbij data verzameld moest worden in een klaspraktijk. Hiervoor contacteerde ik kennissen uit de lerarenopleiding die mij graag te woord stonden.

Inhoudsopgave

Algemene Inleiding.....	1
Deel 1: Algoritmes: literatuurstudie	2
1.1 Inleiding	2
1.2 Situering en definiëring algoritme.....	3
1.2.1 Situering.....	3
1.2.2 Definiëring	6
1.2.3 Sociotechnische opvatting.....	7
1.3 Kenmerken van algoritmes.....	8
1.3.1 Kenmerken	9
1.3.2 Black box.....	11
1.3.3 Algoritme als black box?.....	12
1.4 Algoritmische werking.....	14
1.4.1 Macht van het algoritme	14
1.4.2 Algoritmische technieken.....	18
1.5 Algoritmes onderzoeken	20
1.5.1 Fenomenologisch	20
1.5.2 Etnografisch.....	21
1.5.3 Discursief	22
1.6 Conclusie	24
Deel 2: Digitale klaspraktijk: empirisch onderzoek	25
2.1 Inleiding.....	25
2.2 Theoretische benadering.....	26
2.3 Methodologisch.....	28
2.3.1 Dataverzameling.....	28
2.3.2 Data visualiseren	30
2.3.3 Analyse	33
2.3.4 Kwaliteit.....	34

2.4	Resultaten.....	35
2.4.1	Sociodigitale assemblage.....	35
2.4.2	De compositie van het netwerk	37
2.4.3	Werking van de klaspraktijk	43
2.5	Conclusie	44
	Algemeen besluit.....	45
	Samenvatting.....	45
	Discussie	47
	Reflectie.....	48
	Referenties	50
	Bijlages.....	56
	Bijlage 1: Observatieformulier	56
	Bijlage 2: Voorbeelden interviewvragen	56
	Bijlage 3: Overzicht instellingen en parameters Gephi	57
	Bijlage 4: Resultaat Modularity Class	58

Lijst van tabellen

Tabel 1: Operaties, interacties en actanten van het sociodigitaal netwerk.....	38
--	----

Lijst van figuren

Figuur 1. Deel van tijdlijn over onderzoek naar algoritmes in Gillespie en Seaver (n.d).	4
Figuur 2. Algemene situering van algoritmes binnen een digitaal apparaat.	5
Figuur 3. Sociodigitaal assemblage	36
Figuur 4. Actor Netwerk van digitaal bord	37
Figuur 5. Sociodigitaal assemblage: clusters en regio's	42
Figuur 6. Sociodigitaal assemblage: infrastructuur	43

Algemene Inleiding

Algoritmes of *Weapons of math destruction* zoals O'Neil (2016) ze apocalyptisch benoemt, zijn niet meer weg te denken uit het dagelijks leven. De wereld is vandaag steeds meer (ook) een *wereld onder de duim* (Serres, 2014) of een *code/space* (Kitchin & Dodge, 2011) waarmee de 'vanzelfsprekende' aanwezigheid van computers en het digitale wordt aangeduid. Voorbeelden zijn pushmeldingen van het weer of het verkeer via een smartphone, het tracken van onder meer lopen via een *running app*, de zoekresultaten van Google, de toegang tot een (universiteits)bibliotheek, het krijgen van een *Magic Band* (of een *Tomorrowland bracelet*) als toegangsbewijs voor Disney-parken (of het festival *Tomorrowland*)... Een onmiskenbare component in ieder van bovenstaande voorbeelden van het digitale is 'het algoritme'. Het algoritme is een inherent onderdeel van het digitale dat de laatste decennia meer aandacht geniet. Het is immers steeds het algoritme dat iets doet (en doet doen). Voor het weer en het verkeer doet het algoritme probabilistische uitspraken over mogelijks relevante informatie. Voor het lopen visualiseert het algoritme de loopprestaties of 'pusht' het gebruikers om te gaan lopen. Voor Google geeft het algoritme de resultaten weer in een welbepaalde volgorde. De laatste twee voorbeelden zijn van een andere orde. Deze verzamelen hoofdzakelijk data waarop het algoritme wordt losgelaten om vervolgens 'het beleid' aan te sturen. Deze evidente aanwezigheid van het algoritme dat iets doet en iets doet doen geniet de laatste jaren aandacht in de humane wetenschappen (bv. Beer, 2017; Ziewitz, 2016) maar ook in de ruimere publieke opinie en media. Voorbeelden van dit laatste zijn te vinden in buzzwoorden zoals *big data*, *artificial intelligence*, *fake news*, *filter bubble*, ... maar ook in spraakmakende gebeurtenissen zoals de beïnvloeding van de Amerikaanse presidentsverkiezingen in 2016, het 'privacy-schandaal' van Facebook en Cambridge Analytica in het voorjaar van 2018.

Het zijn precies die voorbeelden, buzzwoorden en gebeurtenissen (als uitingen van 'de digitalisering') die mij intrigeren en die aan de basis liggen van de tweeledige opzet van deze masterproef. Beide interesses ontspringen uit de fascinatie voor de sturing die uitgaat van algoritmes. Een eerste interesse is om de ontpoppende onderzoeksliteratuur over dit nieuw onderzoeksveld in kaart te brengen en zo te introduceren in het brede domein van de pedagogie(k). Dit komt aan bod in het eerste deel van de masterproef waarin gerapporteerd wordt over een literatuuronderzoek naar het concept 'algoritme' en de werking van en sturing door het algoritme. Een tweede interesse is om te kijken of en hoe algoritmes sturen in een onderwijspraktijk. Hiervoor werd een empirisch onderzoek uitgevoerd waar we in het tweede deel over rapporteren. Aangezien het onderzoeksveld nieuw is, zijn er weinig onderzoeken en/of onderzoeksmethodes voorhanden waarop het onderzoek gebaseerd kon worden. Daarom is het empirisch onderzoek exploratief van aard waarbij we aftoetsen of een bepaalde

onderzoeksmethode geschikt is om de sturing door algoritmes te onderzoeken. Een gedetailleerdere omschrijving van de onderzoeksvragen en methodologie worden per deel in de inleiding aangegeven.

Deel 1: Algoritmes: literatuurstudie

“*What actually is an algorithm?*” (Ziewitz, 2016, p. 4).

1.1 Inleiding

Uit de voorbeelden in de algemene inleiding blijkt dat algoritmes bijna permanent aanwezig zijn in het alledaagse leven. Dat is opmerkelijk aangezien er nog weinig geweten is over de precieze werking van een algoritme. Daarom start dit deel met een socio-technische onderzoeksvraag, namelijk: wat is de werking van een algoritme? Het is een technische onderzoeksvraag omdat het gaat over de werking van het algoritme (wat is het en hoe zit het in elkaar?) en het is een sociale onderzoeksvraag omdat het niet enkel gaat over de technische werking maar ook over de sturing die uitgaat van algoritmes bij mensen en de macht van algoritmes.

Om de eerste onderzoeksvraag te beantwoorden, werd er een literatuuronderzoek uitgevoerd. In het pedagogisch en onderwijskundig onderzoek is er weinig expliciet onderzoek voor handen over algoritmes (bv. Williamson, 2015; Perrotta & Williamson, 2018; Introna, 2016). Vandaar dat de geselecteerde literatuur voornamelijk het vakgebied van *Science en Technology Studies of Science, Technology en Society (STS)* representeert. De keuze hiervoor is gemaakt omdat in dat vakgebied expliciet ingegaan wordt op zowel de technische als de sociale kant van het algoritme. In STS wordt een algoritme met andere woorden socio-technisch opgevat (Williamson, 2017a). Dat impliceert dat het algoritme zowel een product als een proces/producer is (Kitchin & Dodge, 2011; Williamson, 2017a). De resultaten van het literatuuronderzoek weerspiegelen dit doordat deze rapporteren over de vermenging van ‘het sociale’ met ‘het technologische’.

De zoektocht¹ naar literatuur gebeurde via Google Scholar en Limo waarbij zoektermen als *algorithm**, *algorithm* AND education*, *algorithm* AND power*, *algorithm* AND govern*...* werden ingegeven. Zo kwam ik uit bij twee special issues over algoritmes namelijk *the social power of algorithms* (Beer, 2017a) en *governing algorithms* (Ziewitz, 2016a). Vervolgens paste ik de sneeuwbalmethod toe op deze twee special issues om andere relevante literatuur op te sporen. Hierin refereerden de auteurs vaak naar de boeken ‘code/space’ van Kitchin en Dodge (2011) en ‘the black box society’ van Pasquale (2015) waardoor deze ook deel uitmaken van het literatuuronderzoek. Aangezien algoritmes

¹ Gegeven het onderwerp van deze masterproef is het niet oninteressant te vermelden dat (een deel van) de literatuur ook mij ‘zocht’ via het *recommendation system* van ResearchGate en Mendeley. Ook Google Scholar bouwde een profiel op van mij waardoor ik op openbare computers andere zoekresultaten kreeg dan op mijn persoonlijke computer. Dit heeft uiteraard niet-onbelangrijke invloeden op de gevonden literatuur en het onderzoek.

voortkomen uit wiskunde en computerwetenschappen, werden ook een aantal (hoofdstukken in) boeken en encyclopedieën gelezen in die discipline. De geselecteerde boeken hiervoor waren eerder inleidende boeken die geschreven zijn voor nieuwkomers in de digitale wereld en zijn onder andere ‘algorithmics: the spirit of computing’ van Harel en Feldman (1987) en ‘introduction to algorithms’ van Cormen, Leiserson, Rivest en Stein (2009). De encyclopedieën waren ‘Encyclopaedia of mathematics’ van Hazewinkel (1990)’ en ‘Encyclopedia of Computer Science and Technology’ van Henderson (2009). De genoemde boeken en encyclopedieën dienden hoofdzakelijk als achtergrondinformatie en de onderzoekartikels vormden als het ware “de data” van het literatuuronderzoek.

Op basis van de gelezen literatuur kunnen er een aantal aspecten onderscheiden worden die een antwoord bieden op de onderzoeksvraag: wat is de werking van een algoritme? In wat volgt lichten worden er vier aspecten toegelicht die regelmatig aan bod komen in de literatuur. De aspecten zijn het gevolg van de intentie om een algemene inleiding weer te geven van wat algoritmes zijn. Bijgevolg zijn de aspecten representatief voor dat deel van het onderzoekdomein dat vaak terugkomt in de literatuur. Een eerste aspect behandelt de situering en definiëring van algoritmes. Het gaat hierbij om waar algoritmes zich situeren en hoe verschillende onderzoekers deze definiëren. Een tweede aspect behandelt de kenmerken die toegeschreven worden aan algoritmes. Deze kenmerken verschillen van wat in de definitie aan bod komt aangezien het gaat over kenmerken die toegeschreven worden aan algoritmes. Deze laatste kenmerken leunen eerder aan bij ‘het sociale’ dan bij ‘het technische’. Een derde aspect is de algoritmische werking waarin een aantal algoritmische technieken aan bod komen. Hierin gaat het eerder om het ‘technische’ dan het ‘sociale’. Hieraan gekoppeld wordt kort ingegaan op het concept ‘macht’. Een laatste aspect staat expliciet stil bij hoe algoritmes onderzocht worden. Hoe iets onderzocht wordt, weerspiegelt namelijk bepaalde opvattingen over wat iets is en hoe iets werkt.

1.2 Situering en definiëring algoritme

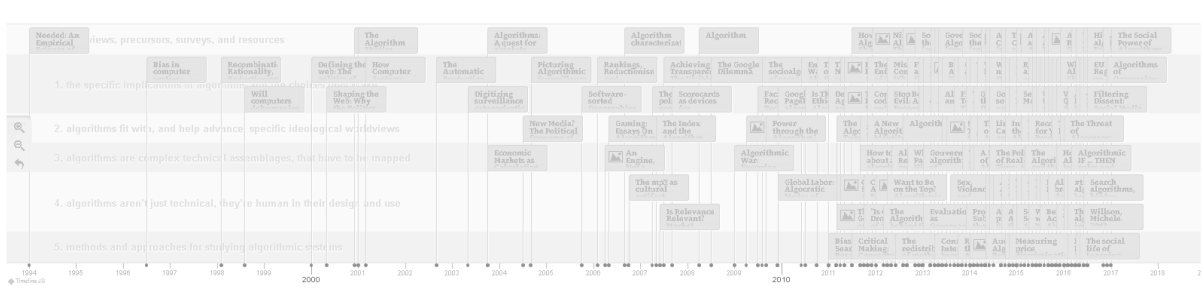
1.2.1 Situering

Het gebruik van algoritmes is geen nieuw gegeven, want de geestelijke vader ervan, Al-Chwarizmi, leefde in de periode van de 8^e en 9^e eeuw (Bauer, 2010; Dasgupta, Papadimitriou & Vazirani, 2006). Al-Chwarizmi’s werk ligt namelijk aan de basis van het huidige decimaal-positioneel getallensysteem (Hazewinkel, 1990). De achterliggende idee van een algoritme gaat nog verder terug in de tijd volgens Christian en Griffiths (2016) en is er al sinds het stenen tijdperk. Wat er toen onder verstaan werd, had – volgens Christian en Griffiths (2016) – vooral betrekking tot het volgen van een procedure. Dit laatste is de opvatting zoals die terugkomt in de bekeken encyclopedieën (Hazewinkel, 1990; Henderson, 2009). Hierdoor is het moeilijk om ‘de uitvinding’ van het algoritme te herleiden tot een bepaalde

periode aangezien het verwijst naar een procedure². Dit betekent dat het algoritme stapsgewijs omschrijft hoe een bepaald soort probleem opgelost moet worden.

Dit hoofdstuk hanteert de algemene opvatting over het algoritme. De opvatting over het algoritme als oplossingsprocedure verwijst immers naar de idee van een algoritme zoals het nu toegepast wordt in computerprogramma's. Het gaat dus niet over concreet geïmplementeerde algoritmes zoals bijvoorbeeld *PageRank* (Google) of *EdgeRank* (Facebook) of een algoritmische techniek of methode zoals bijvoorbeeld *Bayes Classifier* (Rieder, 2017) om data te classificeren of *BubbleSort* om data te sorteren. Het gaat eerder over de algemene achterliggende rationale. Algoritmes worden algemeen belicht en er wordt niet gefocust op één welbepaald algoritme of algoritmische techniek (Hazewinkel, 1990).

Gegeven dat enerzijds, algoritmes al een tijdje bestaan en dus *in se* niets nieuws onder de zon zijn, en anderzijds, algoritmes verwijzen naar algemene en alledaagse procedures, is het opmerkelijk dat de interesse voor algoritmes in de sociale en humane wetenschappen de laatste drie decennia exponentieel aan het stijgen is (Zie Figuur 1) (Gillespie & Seaver, 2016). Naast deze stijgende interesse in algoritmes in de sociale, humane wetenschappen en computerwetenschappen ontstaan er ook nieuwe onderzoeksvelden (e.g. *Digital Humanities*) of onderzoeksmethodes (e.g. visuele netwerkanalyse) door de mogelijkheden die algoritmes met zich meebrengen. De reden dat algoritmes veel aandacht genieten is te verklaren doordat deze een essentiële component vormen van digitale apparaten die sedert 1975 aan een opmars bezig zijn (Kitchin & Dodge, 2011). Hoewel men bij digitale apparaten typerend aan computers denkt, zijn deze twee termen niet tot elkaar te versmelten. Digitale apparaten omvatten naast computers immers ook persoonlijke computers, smartphones, tablets...

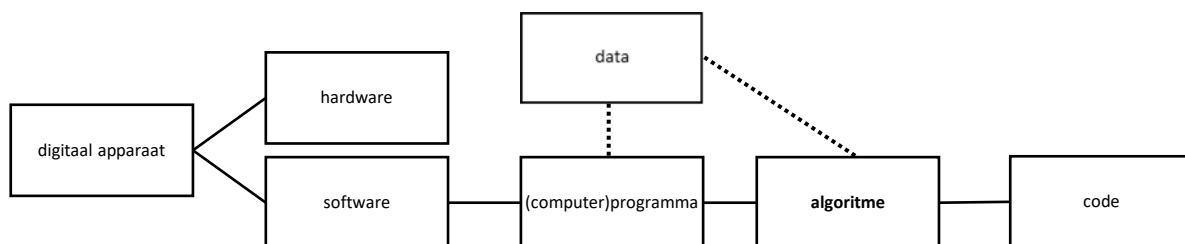


Figuur 1. Deel van tijdslijn over onderzoek naar algoritmes in Gillespie en Seaver (n.d.).

Aangezien de aandacht voor algoritmes in sociale en humane wetenschappen samenhangt met (de opkomst van) digitale apparaten, starten we met algoritmes te situeren binnen deze digitale apparaten

² Bepaalde algoritmes kunnen wel degelijk uitvindingen zijn (bv. de Euclidische deling die gebaseerd is op het algoritme van Euclides, het maken van een gerecht dat gebaseerd is op een recept, ...), maar dit is niet het 'niveau' waarop hier gedacht wordt. Hier gaat het om de algemene opvatting van algoritmes als allerlei handelingsprocedures en gaat in dat opzicht ook om allerlei en alledaagse algoritmes.

en hoe deze zich verhouden tot elkaar (Zie Figuur 2³). Digitale apparaten vormen vanwege hun alledaagse herkenbaarheid en concreetheid een degelijk startpunt om algoritmes te situeren. Deze digitale apparaten zoals een laptop, smartwatch, wasmachine, auto...⁴ vormen ‘het geheel’ en zijn datgene dat we feitelijk gebruiken en daarom het meest (her)kenbaar is. Het is de concrete en alledaagse uitwerking van het digitale dat we kennen. Deze apparaten bestaan enerzijds uit hardware (i.e. dat wat we kunnen vastnemen) en anderzijds uit software (i.e. dat wat we niet kunnen vastnemen). Voorbeelden van hardware zijn een beeldscherm, een harde schijf, een sluimerknop... De software bestaat op zijn beurt uit programma’s. Grofweg bestaan er twee soorten programma’s: enerzijds zijn er besturingsprogramma’s zoals bijvoorbeeld Windows, iOS, Android ... en anderzijds zijn er toepassingsprogramma’s zoals bijvoorbeeld Google Chrome, MS Word, Spotify... Al deze programma’s bevatten (en verzamelen) data en zijn opgebouwd uit algoritmes. Die algoritmes zijn de instructies die ervoor zorgen dat een bepaalde taak uitgevoerd wordt en hebben hiervoor data nodig. Ze worden geschreven in een bepaalde taal, namelijk een programmeertaal. Wat geschreven wordt in deze programmeertaal heet code. De code is met andere woorden de representatie van een algoritme (Hazewinkel, 1990). Bekende voorbeelden van programmeertalen waarin code getypt wordt, zijn C++, Java, Python ...



Figuur 2. Algemene situering van algoritmes binnen een digitaal apparaat.

Bovenstaande algemene weergave toont dat een algoritme zich situeert ‘tussen’ een programma en code (Kitchin & Dodge, 2011). Dit betekent dat algoritmes tussen een ‘groot’ programma (bestaande uit veel algoritmes) staan en kleine(re) code(s) die samen een algoritme beschrijven. Wat Figuur 2 toont is de situering van een algoritme binnen een digitaal apparaat. Wat het niet toont maar wel cruciaal is, is dat een algoritme een *autonome* ‘tussengrond’ vormt. Dat betekent dat een algoritme eigenlijk op zichzelf staat. Dat wil zeggen dat de programmeertaal waarin het algoritme geschreven wordt en het programma waarin het algoritme gebruikt wordt er voor een algoritme *in se* niet toe doet (Goffey, 2008; e.g. Rieder, 2017). Het algoritme is met andere woorden het belangrijkste. De

³ De figuur is gebaseerd op inzichten uit het werk van Kitchin & Dodge (2011, pp. 1-65).

⁴ De voorbeelden zijn niet uitsluitend digitale apparaten maar zijn ook mechanische apparaten, elektrische apparaten... Het benoemen van die apparaten als digitale apparaten betekent dat die apparaten vanuit hun digitale component bekeken worden en op die digitale componenten focussen.

programmeertaal waarin het geschreven wordt en het programma waarin het gebruikt wordt, zijn “implementatiedetails” volgens computerwetenschappers (Goffey, 2008).

1.2.2 Definiëring

De voorgaande situering geeft reeds vanuit een bepaalde invalshoek (namelijk computerwetenschappen) een idee over de werking van een algoritme door te omschrijven wat het is en waar het zich situeert. Zoeken we vervolgens een definitie vanuit de computerwetenschappen over een algoritme, dan constateren we dat hier geen eensgezinde definitie over is (Moschovakis, 2001; Buss, Kechris, Pillay, & Shore, 2001). Dat is echter niet per se problematisch vermits er wel degelijk een relatief duidelijk beeld is over wat een algoritme is. Dit zal in het hoofdstuk over de kenmerken verduidelijkt worden. Alvorens daarop in te gaan, wordt er in deze paragraaf gerapporteerd hoe algoritmes gedefinieerd worden door sociale en humane wetenschappers waarbij gelijkenissen en verschillen opgespoord worden. De aangehaalde definities zijn hierbij selectief gekozen op basis van de verschillende accenten die in de definitie liggen. Dit betekent dat er geen systematisch overzicht wordt gegeven van al de gehanteerde definities (dit vormde wel een tussenstap) maar dat de definities een keuze representeren.

Een eerste definitie vinden we bijvoorbeeld bij MacCormick (2013) die een algoritme omschrijft als: *“a precise recipe that specifies the exact sequence of steps required to solve a problem”* (p. 3). In deze definitie staat expliciet de term van de vaak terugkerende gemaakte analogie met een recept. Een algoritme wordt hierbij vergeleken met een recept waarbij welbepaalde instructies stapsgewijs gevolgd moeten worden om een bepaald ‘probleem’ op te lossen (bijvoorbeeld een chocoladecake maken). Het accent ligt met andere woorden op het algoritme als een procedure die volgens vaste stappen verloopt waarbij de toepassing van de procedure een probleem moet oplossen. MacKenzie (2006) legt hetzelfde accent maar kleedt dit aan met een digitaal jasje: *“[Algorithms are] recipes or sets of steps expressed in flowcharts, code or pseudocode”* (p. 43). Door code en pseudocode mee op te nemen in de definitie verbindt MacKenzie (2006) algoritmes expliciet met hun huidige rol en plaats in de (digitale) wereld. Deze invulling van een algoritme ligt in lijn met Gillespie (2014) die stelt dat: *“[algorithms] need not be software: in the broadest sense, they are encoded procedures for transforming input data into a desired output, based on specified calculations”* (p. 167). Deze laatste definitie haalt een niet onbelangrijk nieuw aspect aan vermits het algoritme gerelateerd wordt aan data die getransformeerd wordt in een gewenste output. Deze omschrijving sluit aan bij de definiëring zoals in de encyclopaedia of Mathematics (Hazewinkel, 1990) omschreven is: *“[A] detailed instructions defining a computational process [...] which begins with an arbitrary input [...] and with instructions aimed at obtaining a result (or output) which is fully determined by the input”* (p. 131). In tegenstelling tot de eerste definitie gaat het hier niet om de oplossing van een probleem, maar om het omzetten

van een input naar een output. MacKenzie (2006) en Gillespie (2014) nemen de gecodeerde representatie van het algoritme op en stappen hiermee impliciet af van de analogie met het recept (een niet-gecodeerd, niet-digitaal algoritme). Dat betekent dat de huidige opvatting over een algoritme volgens MacKenzie (2006) en Gillespie (2014) is dat het algoritme steeds geschreven is in code en bijgevolg in digitale apparaten voorkomt.

Besluitend kunnen we stellen dat een algoritme algemeen een procedure is om een input om te zetten naar een output waarbij de procedure de oplossing omschrijft voor een probleem.

1.2.3 Sociotechnische opvatting

Gillespie (2014) geeft in de definitie aan dat het gaat om een wenselijke output. De vraag naar wat wenselijk is, is een normatieve vraag (naast een economische, culturele, politieke, filosofische... vraag) die beantwoord moet worden door mensen. Dat is niet zomaar een bijzaak maar vormt een wezenlijk aspect van de werking van een algoritme, namelijk dat algoritmes tot stand komen door mensen in een bepaalde context. Het gaat er met andere woorden om dat algoritmes een *sociale creatie* zijn en dat bijgevolg het soort output van een algoritme door mensen wordt bepaald. Of nog, de algoritmes zijn een *product* van mensen (en de wereld) die gebruikt worden en daardoor ook een *proces* zijn (i.e. deel uitmaken van die wereld) (Kitchin & Dodge, 2011) of *producer(end)* zijn (Williamson, 2017a). Een algoritme sociotechnische benaderen betekent volgens Gillespie (2014): “[we] must not conceive of algorithms as abstract, technical achievements, but must unpack the warm human and institutional choices that lie behind these cold mechanisms” (p. 169). Concreet voor een algoritme betekent dit dat het niet ‘zomaar’ een techniek of onderdeel van een technologie is, maar dat dit een *product* is dat door mensen binnen een bepaald instituut of organisatie gemaakt wordt.

Uit de gelezen literatuur blijkt dit dat we het ontwerpen van algoritmes (en de invloeden hiervan) op drie niveaus kunnen beschrijven⁵.

Een eerste niveau heeft betrekking op degene die zelf de code typt en hierbij impliciet dan wel expliciet deel uitmaakt van een collectief (door een bepaalde programmeertaal te gebruiken, een bepaalde stijl van coderen te hanteren, bepaalde criteria voorop te stellen ...).

Een tweede niveau verwijst naar de brede invulling van een organisatie (een profit bedrijf, een non-profit bedrijf, een ngo, een kabinet...) waarbinnen het algoritme tot stand komt. Dit betekent *de facto* dat code (en in extenso algoritmes) typen altijd binnen een bepaald kader gebeurt zoals Kitchin en Dodge (2011) aangeven: “It [production of code] is embedded within workplace or hacker cultures, personal interactions and office politics, relationships with customers/users, and the wider political and

⁵ Deze niveaus zijn gedistilleerd uit Kitchin en Dodge (2011, pp. 32-38).

cultural economy" (p. 37). Hierbij wordt een groter, gemeenschappelijk doel gerechtvaardigd. Een interessant voorbeeld en onderzoek hierover vinden we bij Neyland (2016) die een projectgroep gevolgd heeft die als doel had om een ethisch algoritmisch systeem te ontwikkelen dat aansprakelijk gesteld kan worden (*accountability*). Hoewel Neyland (2016) focust op het ethische aspect van het algoritme, toont het onrechtstreeks ook de vele invloeden van een organisatie die erbij horen, wil men een algoritme ontwikkelen. De doelen waren in dit geval om mensen op te sporen in verboden zones, die bewegen in de verkeerde richting en bagage in een luchthaven achterlaten. Ook Williamson (2017b) toont dit aan op basis van een documentanalyse: "*how a particular set of assumptions about the use of learning analytics and machine learning algorithms is circulating within the institutional context of the company [IBM]*" (p. 86). Verder stelt Williamson: "[algorithms] *are the product of complex sociotechnical practices and are embedded in the methodological commitments, assumptions, values, and styles of thinking of their designers, such as those associated with Smarter Education at IBM*" (p. 86). Degene die een algoritme ontwikkelen zijn in dit geval steeds ingebed in een organisatie (in dit geval IBM, een internationaal IT-bedrijf).

Een derde niveau gaat nog een stapje verder doordat algoritmes als product tot stand komen binnen een groter discours en gelijkaardige, geformaliseerde kaders. Een voorbeeld van dit laatste vinden we opnieuw bij Neyland (2016) waarbij het project pas van start kon gaan nadat de financiering door de Europese Unie (*7th Framework Programme*) werd goedgekeurd en toegekend. Door financiering aan bepaalde projecten te koppelen stuurt de Europese Unie de totstandkoming en ontwikkeling van bepaalde algoritmes.

Samenvattend kan gesteld worden dat algoritmes niet louter als een technische kwestie opgevat kunnen worden waarbij de rol van de individuele softwareontwikkelaar als louter uitvoerend gezien wordt. Daarnaast betekent het ook dat algoritmes niet in het luchtledige gemaakt worden en dus altijd beïnvloed zijn door een specifieke tijd en ruimte die geladen is met politieke, economische, culturele, maatschappelijke waarden, visies, ontwikkelingen, invloeden... De vraag naar de werking van algoritmes gaat dus onverbiddeijk ook over deze (achterliggende) sociale processen van productie (Williamson, 2017a; Beer, 2017b).

1.3 Kenmerken van algoritmes

Het ontbreken van een eenduidige, formele definitie vormt geen drempel om algoritmes te onderzoeken. Omdat het eenduidig definiëren niet mogelijk is, beschrijven we in dit hoofdstuk een aantal kenmerken die toegeschreven worden aan algoritmes. Het samenbrengen en categoriseren van deze kenmerken laat toe om de (sociotechnologische) werking van algoritmes op een andere manier te vatten en weer te geven. Deze andere 'kijkwijze' bouwt verder op de inzichten uit het vorige hoofdstuk waarbij het algoritme als sociaal product werd omschreven. De kenmerken die we zullen

bespreken zijn nauw verwant met de idee van een black box zoals omschreven door Latour (1987). Na een toelichting van dit concept, zullen we algoritmes daaraan aftoetsen.

1.3.1 Kenmerken

Zoals eerder aangegeven is een algoritme geschreven in **code** waardoor een leek (de mensen die de codetaal niet machtig zijn) geen vat kan krijgen op de code en in extenso het algoritme. Bovendien is het niet zo dat een algoritme één lijn code is, maar meestal bestaat uit honderden of duizenden lijnen code (Neyland & Möllers, 2017). Zo stelt Introna (2016) bijvoorbeeld dat Windows Vista (een besturingssysteem) uit vijftig miljoen lijnen broncode bestaat. Net omwille van deze omvangrijke codes en complexiteit gelden metaforen voor code als ‘a Big Ball of Mud’ (Foote & Yoder, 1997; Kitchin & Dodge, 2011) of spaghetti code (Kitchin & Dodge, 2011). Het is vaak zelf voor de ontwikkelaars van de code (en algoritmes) bijna nog onmogelijk om te omschrijven hoe en waar een algoritme in een bepaalde code is getypt⁶.

De reden dat een algoritme zelf voor de ontwikkelaars vaag is (Kitchin, 2017; Ziewitz, 2016b), is tweeledig. Ten eerste, omdat een algoritme meestal verwijst naar een (heterogeen) **algoritmisch systeem** (Gillespie, 2014; Seaver, 2013; Hazewinkel, 1990). Wat betekent dat een algoritmisch systeem opgebouwd is uit verschillende, op elkaar inspelende algoritmes. Die vaststelling hangt samen met de vorige paragraaf waarbij gesteld werd dat algoritmes niet uit één lijn van code bestaan. Zo verwijst Bucher (2016) naar PageRank (een algoritme ontwikkeld door Google om webpagina’s te ordenen) dat niet één algoritme is, maar bestaat uit een netwerk van algoritmes (i.e. een algoritmisch systeem). Ten tweede, omdat vaak meerdere mensen – al dan niet onafhankelijk van elkaar in tijd en ruimte – aan een algoritme werken (Seaver, 2013). Dat betekent dat het niet één persoon is die een algoritme maakt, maar dat dit meestal wordt gemaakt binnen een **team** waarbij verschuivingen van personeel, visie, middelen... mogelijk zijn. Dat hangt nauw samen met het feit dat een algoritme meestal een algoritmisch systeem is waardoor het te veel werk is voor één persoon om zo een systeem te ontwikkelen of te onderhouden. Het algoritme wordt dus ontwikkeld door verschillende mensen die elk aan een stukje van het algoritme werken waardoor niemand een volledig zicht heeft op het algoritmisch systeem. Bovendien bestaat het ontwikkelen en onderhouden van een algoritme hoofdzakelijk uit ‘**bug fixing**’. Een ‘bug’ verwijst naar een foute of onverwachte output die ontstaat door een fout in het algoritmisch systeem. Het proces van het herstellen (*fixen*) van een bug omschrijft Crowstone (1994) als: “*a microcosm of coordination problems and solutions*” (p. 157). Dat impliceert dat het onderhouden van een algoritme neerkomt op het oplossen van *huidige* problemen (bugs)

⁶ Hoewel dit in de geselecteerde literatuur niet expliciet uitgewerkt aan bod komt, kan wel gesteld worden dat dit nog moeilijker wordt indien een licht geworpen wordt op algoritmes vanuit *deep learning*, *machine learning* en *artificial intelligence*.

waarbij geen volledig zicht verkregen moet worden op de volledige genealogie of technische werking van het algoritme. Ceglowski (2016) omschrijft dit proces als: *“we will instrument, then we will analyze, then we will optimize”* waarbij dit optimaliseren in lijn ligt met het herstellen van bugs. Verder betekent dit ook dat een algoritme nooit ‘af’ is en steeds in de maak, in wording is (i.e. ontogenetisch)⁷. Niemand kent hierbij de volledige genealogie of werking van ‘het’ algoritme en dit wordt ook niet relevant geacht.

Het bestaande algoritme wordt met andere woorden telkens aangepast en ‘is er gewoon’. Dat brengt ons tot het volgende kenmerk, namelijk het algoritme werkt op de **achtergrond**. Dat betekent dat een algoritme tijdens het alledaagse leven op de achtergrond aanwezig is (Willson, 2017). Bucher (2017) toont bijvoorbeeld deze alledaagse rol aan door de momenten te onderzoeken waarbij mensen (plots) wél bewust zijn van de rol van het algoritme (in dit geval EdgeRank, het Facebook-algoritme). Dat alledaagse gaat verder dan enkel softwareplatformen zoals Kitchin en Dodge (2011) aantonen door te wijzen op de dagelijkse aanwezigheid en rol van code in bijvoorbeeld de supermarkt of een huis en wat Thrift (2005, in Beer, 2009) benoemt als: *“‘technological unconscious’, the operation of powerful and unknowable information technologies that come to ‘produce’ everyday life”* (p. 988). Of nog anders gesteld gaat het om: *“technological environments that operate without the knowledge of those upon whom they are taking an effect”* (Beer, 2009).

Een laatste “kenmerk” gaat over de **eigendomsrechten** van een algoritme (Pasquale, 2015). Dat kenmerk is van een andere orde dan de voorgaande kenmerken omdat het in een bepaalde zin geen inherent kenmerk van een algoritme is, maar eerder een soort keuze van het beleid om onder andere competitief ‘aan de top’ te blijven en het *gamen* van het algoritme te vermijden (Crawford, 2016; Diakopoulos, 2015). Het gamen van een algoritme benoemen Souto-Otero en Beneito-Montagut (2016) als een weerstandstrategie (*resistance* strategie) en lichten dit als volgt toe: *“the social actor plays within the proposed field, but not under the set rules”* (p. 27). Dat betekent dat een gebruiker iets of zichzelf aanpast aan het algoritme om zo bijvoorbeeld meer zichtbaarheid te verwerven. Een bekend voorbeeld hiervan is *search engine optimization*.

Deze kenmerken verwijzen naar een zekere (technische) complexe, heterogene en obscure of vage werking van een algoritme dat meestal onopvallend in de achtergrond afspeelt (Bucher, 2017). Bovendien gaat het bij een algoritme, zoals uit de definities bleek, om de gewenste soort output te krijgen gegeven een bepaalde input. Vandaar dat geregeld claims worden gemaakt over het algoritme als een black box die geopend dient te worden (e.g. Pasquale, 2015).

⁷ Een concreet voorbeeld hiervan is het updaten van software waarbij de gebruiker vaak ingelicht wordt over de veranderingen van het algoritme (i.e. over het algoritme in de maak, in wording).

1.3.2 Black box

Het openen van een black box is een veelgebruikte benadering binnen STS (waarvan Latour de grondlegger is). Alvorens dit openen te omschrijven, verdient het concept van een black box verduidelijking. Hiervoor grijpen we terug naar Latour (1987) die een black box omschrijft (en zo introduceert in STS) als:

whenever a piece of machinery or a set of commands is too complex. In its place they [cyberneticians] draw a little box about which they need to know nothing but its input and output. [...] That is, no matter how controversial their history, how complex their inner workings, how large the commercial or academic networks that hold them in place, only their input and output count (pp. 2-3).

Bij een black box staat precies de werking van *iets*⁸ op het spel door het proces (van input naar output) buiten beschouwing te laten vanwege de complexiteit. Niet *hoe* iets werkt is van tel, maar wel welke input nodig is en welke output daarmee verkregen wordt. Of nog, zolang men de input en output kent, is het irrelevant hoe iets werkt. Een sprekend voorbeeld binnen een veelgeciteerd boek van Ashby (1956) hiervan is het (leren) openen van een deur. Hierbij dient het handvat van de deurklink (input) om het sluitmechanisme of de grendel (output) van de deur te openen. Wat aan het zicht onttrokken is, is hoe dit precies werkt (Ashby, 1956).

Waar het openen van een black box dan naar doelt, is precies die werking van iets blootleggen. Het poogt het proces tussen de input en output te begrijpen door net dat proces te onderzoeken. Dit proces beperkt zich tevens niet enkel tot de werking van iets – zoals uit het tweede deel van de definitie voortvloeit – maar ook naar de sporen van de (controversiële) geschiedenis en/of het netwerk dat samenhangt met de black box. Dit laatste draagt een tweede niet onbelangrijke invulling van een black box met zich mee: namelijk dat iets ook een black box kan zijn doordat het deel uitmaakt van een netwerk en gedragen wordt door een netwerk. Een gevolg hiervan is dat iets vanzelfsprekend (geobjectiveerd, genormaliseerd, gestabiliseerd) wordt en daardoor ook als black box benoemd kan worden (iets wordt met andere woorden '*geblack-boxed*'). Een illustratief onderzoek hierover vinden we terug bij Ceulemans (2015) die het beroepsprofiel van de leraar traceert als iets dat geblack-boxed is doordat onder andere iedereen het begint te gebruiken en ernaar verwijst. Het beroepsprofiel is met andere woorden een black box waarvan de precieze werking niet meer duidelijk is.

Later komt Latour (1987) terug op deze invulling van een black box door te stellen dat iets een black box is: "*when many elements are made to act as one*" (p. 131). Verder betekent dit dat een black box

⁸ Waar Latour over machines en reeksen commando's spreekt in de definitie, laat het concept van een black box zich niet beperken tot machines en reeksen commando's. Het gaat telkens om *iets* (een computer, een document, de weergave van de structuur van DNA, ...).

tot stand komt aangezien: “*machinations [...] turn a gathering of forces into a whole that then may be used to control the behaviour of the enrolled groups*” (p. 131). Latour verduidelijkt dit met het voorbeeld van een fototoestel waarbij veel verschillende elementen – slechts door samen te ‘werken’ – een geheel vormen. Zo vormt het fototoestel het geheel en vormen de onderdelen van het fototoestel de elementen. Bovendien – zo stelt Latour – is het fototoestel door een complexer commercieel netwerk gedragen maar blijft het als één geheel werken. Iemand die met andere woorden een fototoestel koopt, koopt het geheel van elementen (hetzij onderdelen, hetzij het commercieel netwerk) dat samenwerkt.

Concluderend kan gesteld worden dat een black box meerdere betekenissen bevat die voldoende onderscheiden kunnen worden. Zo gaat het bij een black box over (1) iets waarvan we de werking niet precies kennen en/of (2) iets dat zich als vanzelfsprekend laat verschijnen en/of omschrijven en (3) pas als geheel ‘werkt’ door een samenspel van onderliggende mechanismen.

1.3.3 Algoritme als black box?

Gegeven dat er in de literatuur geregeld naar algoritmes als black box verwezen wordt (e.g. Kitchin, 2017; Pasquale, 2015; Ziewitz, 2016b), gaan we kort in op deze relatie tussen algoritmes en de idee van een black box. Leggen we de kenmerken van een algoritme naast de idee van een black box, dan is dat in een bepaald opzicht ook niet verwonderlijk. Algoritmes laten zich onder andere door hun kenmerken immers nét omschrijven als (1) iets waarvan we de werking niet precies kennen, (2) iets dat vanzelfsprekend lijkt en (3) iets dat als geheel werkt door onderliggende elementen/mechanismen.

De meest aangehaalde reden waarom een algoritme als black box opgevat wordt, is omdat we de werking niet precies kennen. Dat komt omdat, ten eerste, de code van het algoritme vaak niet toegankelijk is voor onder andere onderzoekers door bijvoorbeeld eigendomsrechten. Waar deze toegang tot de code als een feitelijke beperking gezien kan worden om de werking van het algoritme te kennen, rijst nog maar de vraag of dit daadwerkelijk een beperking vormt. Stel dat we (sociale en humane onderzoekers) toegang krijgen tot de code van het algoritme, dan is het nog steeds plausibel dat we de werking het algoritme niet zullen kunnen achterhalen door gebrek aan specifieke scholing en dus het ontbreken van de expertise van een softwareontwikkelaar (Kitchin, 2017). De tweede reden waarom een algoritme als een black box opgevat wordt is het feit dat een algoritme voor de softwareontwikkelaars zelf vaag en obscuur is en daaruit volgt dat hetzelfde geldt voor sociale en humane onderzoekers. Bovendien is de idee achter het herstellen van bugs, gelijkaardig met de idee van een black box waarbij niet de werking centraal staat maar enkel de input en output.

Een reden die we minder aantreffen in de literatuur, is de vanzelfsprekende aanwezigheid van een algoritme. Een eerste voorbeeld vinden we bij Kitchin en Dodge (2011) die aanduiden hoe code (en in

extenso algoritmes) ruimtes zoals een huis, een supermarkt en een luchthaven heeft omgevormd tot *“coded assemblages”* (p. 7) en *“code/spaces”* (p. x). Naast deze vanzelfsprekende aanwezigheid van software in ruimtes, gaat het ook om de aanwezigheid van algoritmes in ‘alledaagse’ software. Zo omschrijft Willson (2017) deze aanwezigheid als volgt: *“increasing dependence on and engagement with and through the online [...] render these relationships and the ‘algorithmisation’ of everyday practices as commonplace and unremarkable”* (p. 143).

Een laatste reden die stelt dat algoritmes black boxes zijn, is ingegeven vanuit het feit dat algoritmes pas als geheel werken door de onderliggende, samenhangende elementen en mechanismen. Het kenmerk van het algoritme dat hier het best bij aansluit is dat algoritmes in werkelijkheid eigenlijk gaan over algoritmische systemen. Een algoritmisch systeem werkt pas indien ieder deel(algoritme) samenwerkt. Zo stellen Perrotta en Williamson (2018): *“LA [learning analytics] might be conceived as an ‘algorithmic assemblage’ in which methods (which include cluster analysis) are interwoven with expert knowledge, mathematical mechanisms, technical practices and political and economic objectives”* (p. 4). Hiermee wijzen Perrotta en Williamson ook op het feit dat de elementen breder zijn dan enkel dat wat in het algoritme zit zoals bijvoorbeeld expertise, economische doelen... Het is pas door het samenspel van al deze elementen dat een algoritme als geheel werkt.

Toch gaan niet alle onderzoekers per se akkoord met de analogie tussen een black box en het algoritme. Hierbij neemt Bucher (2016) een expliciet standpunt in door te stellen dat algoritmes *“neither black nor box”* zijn. De reden waarom algoritmes niet per se black boxes zijn, is volgens Bucher (2012) omdat: *“what is important is not necessarily to know every technical detail of how a system works, but to be able to understand some of the logics or principles of their functioning”* (p. 1177). Het gaat met andere woorden niet om het technisch openen van een black box, maar wel om de logica's ervan te begrijpen. Dit betekent dat de metafoor voor het algoritme als black box dan wel – deels (Bucher, 2016) – klopt, maar dat dat geen probleem is vermits deze niet op die technische manier geopend moet worden. Hierbij bekijkt Bucher (2012) een algoritme hoofdzakelijk vanuit het code-kenmerk waardoor het algoritme zich slechts deels als black box laat omschrijven. Vanuit Latour (1987) bleek immers dat een black box meer omvat dan enkel de werking niet kennen. Daarnaast schrijft Gillespie ook dat de black box-metafoor niet geldt voor algoritmes net omdat het een algoritmisch systeem is dat constant geüpdatet wordt waardoor de werking van het algoritme kneedbaar (*malleable*) is.

Besluitend kunnen we stellen dat gegeven de kenmerken van algoritmes, deze zich vrij vaak en gemakkelijk laten vallen onder de idee van een black box zoals omschreven door Latour (1987). Toch bleek uit het literatuuronderzoek dat hierover enigszins onenigheid is.

1.4 Algoritmische werking

In de algemene inleiding wezen we op de aanwezigheid van algoritmes in het dagelijks leven en in de kenmerken van algoritmes rapporteerden we dat algoritmes in de achtergrond werken. Dat kan enige verbazing wekken aangezien algoritmes door verschillende onderzoekers als black box omschreven worden. We worden, volgens hen, met andere woorden gestuurd door iets (i.e. algoritmes) waarvan niemand volledig begrijpt hoe het werkt. Een aantal onderzoekers bogen zich over die sturing door algoritmes om hier vat op te krijgen. In dit hoofdstuk worden de bevindingen van die onderzoeken gerapporteerd.

1.4.1 Macht van het algoritme

1.4.1.1 *Macht in een post-hegemonie*

De macht [kan nooit] worden beschouwd als een op zichzelf staand principe, noch als een verklarende waarde die zich van meet af aan doet gelden. Met de term 'macht' wordt slechts verwezen naar een gebied van onderlinge relaties die vragen om een volledige analyse. (Foucault, 2004, pp. 247-248).

Dit is het uitgangspunt van Foucault voor macht en hoewel Foucault zelf geen aandacht besteedde aan algoritmes, zijn er veel onderzoeken over de macht van algoritmes gebaseerd op zijn werk (bv. Beer, 2017b; Cheney-Lippold, 2011; Introna, 2016).

Een werk waar veel naar geciteerd wordt in de sociaalwetenschappelijke literatuur over de macht van algoritmes, is het onderzoek van Lash (2007). Lash traceert een shift van een *hegemonie* (denk hierbij in extreme vorm aan Orwells *1984* of Huxley's *brave new world*) naar een *post-hegemonie* onder andere door de opkomst van digitale technologie. Dit traceert Lash aan de hand van een aantal aspecten waarvan er twee worden toegelicht met betrekking tot algoritmes.

Een eerste aspect is dat de macht in een post-hegemonie van binnenuit (het subject) werkt als "*potentia*" (Lash, 2007, p. 59). Met deze term verwijst hij naar de kracht van iets of iemand (ding of mens) om dit iets of iemand te worden (*becoming of the thing-itself*). Het gaat hier met andere woorden niet om een macht die van buitenaf werkt (bv. soevereine macht of disciplinerende macht), maar om een macht die van binnenuit werkt in de vorm van zichzelf organiseren of zelforganisatie (*self-organization*). Lash geeft hier zelf geen voorbeeld van maar verwijst naar het werk van Foucault die deze vorm van macht beschrijft. Een voorbeeld van deze macht in een post-hegemonie met betrekking tot algoritmes wordt gegeven en uitgewerkt in het onderzoek van bijvoorbeeld Lupton (2016) of Williamson (2014) naar *self-tracking*. Het concept self-tracking verwijst naar het gebruik van technologie door mensen om hun leven of bepaalde aspecten ervan te monitoren, te managen en te optimaliseren (Lupton, 2016). In het onderzoek van Lupton (2016) wordt aangegeven dat self-tracking

gaat om “self-responsibilized practices of dataveillance” (p. 118). Williamson (2014) toont in zijn onderzoek het belang van algoritmes aan in *self-tracking* en hoe die algoritmes daardoor een rol spelen in het zichzelf organiseren.

Een tweede aspect is dat er in een post-hegemonie met “*pervasive media and ubiquitous coding*” (Lash, 2007, p. 71) een nieuw type van regels bestaat. Naast wetgevende regels (bv. de regels van een spel; die als een soort toegangsticket of paspoort werken) en reglementaire regels (bv. reguleert je activiteit tijdens het spel) ontstaan er algoritmische of voortbrengende, groeiende (*generative*) regels. Deze omschrijft Lash (2007) als:

Virtuals that generate a whole variety of actuals. They are compressed and hidden and we do not encounter them in the way that we encounter constitutive and regulative rules. Yet this third type of generative rule is more and more pervasive in our social and cultural life of the post-hegemonic order. They do not merely open up opportunity for invention, however. They are also pathways through which capitalist power works. (p. 71)

Hoewel Lash dat amper uitwerkt, pikt Beer (2009) hierop in en past dat idee van algoritmische regels toe op het Web 2.0. Beer geeft aan dat het bij het Web 2.0 niet enkel gaat om communicatie of vormen van zelfexpressie, maar ook om plaatsen waar informatie over mensen verzameld wordt. Die informatie of data wordt gebruikt om kennis te verwerven en voorspellingen te maken (Beer, 2009). Deze algoritmische regels zijn volgens Lash (2007) het gevolg van een nieuwe vorm van macht die volgens Beer (2009) gebaseerd zijn op: “*the central premise that algorithms have the capacity to shape social and cultural formations and impact directly on individual lives*” (p. 994).

Kitchin en Dodge (2011) rapporteren gelijkaardige bevindingen. Ze omschrijven het niet als een hegemonie en een post-hegemonie (concepten van in culturele studies) maar omschrijven het als een “*surveillance model*” (p. 89) en een “*capture model*” (p. 89). Het capture model ontstaat uit de mogelijkheden van software en digitale technologie om te sturen en sluit aan bij een post-hegemonie zoals beschreven door Lash (2007). Het mechanisme van de macht is hierbij niet meer disciplineren door zelfdiscipline, maar managen/disciplineren door het moduleren van ervaringen. Een voorbeeld hiervan is het concept *The Filter Bubble* (Pariser, 2011). De term duidt op een gevolg van het fenomeen van gepersonaliseerde zoekresultaten. De ervaringen van het surfen op het internet zijn hierbij aangepast aan de gebruiker. Naast dat mechanisme, geven Kitchin en Dodge (2011) ook aan dat de zichtbaarheid in het capture model meestal verborgen en geheim werkt aangezien het verzamelen van informatie of data inherent deel uitmaakt van het capture model. Kitchin en Dodge (2011) geven geen voorbeeld maar cookies op het internet zijn hier een duidelijk voorbeeld van. In het capture model gaat het met andere woorden om: “*new forms of surveillance seek to produce objectified individuals*

where the vast amount of capta harvested about them is used to classify, sort, and differentially treat them, and actively shapes their behaviour” (Kitchin & Dodge, 2011, p. 86). Toegepast op het moduleren van ervaringen betekent dit dat er data verzameld wordt over gebruikers. Op basis van die data ontstaan ‘geobjectiveerde’ individuen die geclassificeerd worden. Vervolgens wordt op basis van die classificatie of categorie de ervaring van een individu gemoduleerd, wat op zijn beurt een invloed heeft op het gedrag van dat individu.

1.4.1.2 *Situering van macht*

In de vorige paragraaf werd gerapporteerd over een algemene opvatting van macht in een post-hegemonie. In deze paragraaf worden aantal onderzoeken die de macht van algoritmes analyseren besproken. Een toonaangevende auteur wat dat betreft is Beer (2017b) die de macht van algoritmes op drie verschillende ‘plaatsen’ beschrijft. De drie plaatsen of categorieën zijn: in de algoritmische techniek, in de context en in het discours. We nemen deze drie categorieën over en vullen die aan met een aantal voorbeelden van onderzoeken.

De eerste categorie situeert de macht van het algoritme eerder ‘in’ het algoritme. Dat betekent dat het algoritme macht verwerft door wat het kan en doet. Bucher (2012) toont bijvoorbeeld hoe de macht van het algoritme ingebouwd is in een algoritmische techniek die beslist wat zichtbaar gemaakt wordt. De algoritmische techniek is gebaseerd op een logica die items rangschikt volgens hun relevantie. Hoewel dit niet precies geweten is voor Facebook (EdgeRank-algoritme), is er wel een idee over wat de relevantie van een item bepaald zoals bijvoorbeeld de tijd (hoe ouder een bericht, hoe lager de index), het ‘gewicht’ (hoe meer interactie, hoe hoger de index waarbij dit afhankelijk is van het soort interactie [‘like’, ‘comment’, ‘share’, ...])... De macht van het algoritme zit met andere woorden in de formule die aangeeft wat ‘zichtbaar’ gemaakt zal worden waarbij uiteraard een input van de gebruikers aan voorafgaat. Op een vergelijkbare, maar iets technischere wijze breidt Yeung (2017) het concept *nudge* uit naar *hypernudge* door algoritmische technieken. Een typisch voorbeeld van ‘*nudging*’ is voetstappen kleven op de trap zodat mensen eerder de trap dan de roltrap zullen nemen. Net zoals de voetstappen de techniek zijn voor *nudging*, zijn algoritmes de techniek voor *hypernudging*. De mensen worden onbewust gestuurd om iets te doen als gevolg van een algoritme. De macht van het algoritme zit met andere woorden in de algoritmische techniek zelf.

De tweede categorie vertrekt vanuit de opvatting dat een algoritme pas macht verwerft door de inbedding in een context. Een algoritme los van die context onderzoeken is volgens Beer (2017b) niet wenselijk:

The problem comes if we try to detach the algorithm from the social world in order to analyse its properties and powers – seeing it as a technical and self-contained object that exists as a distinct presence is likely to be a mistake (p. 4).

Het is typerend aan deze categorie dat de onderzoeken sterk geïnspireerd zijn door ANT (bv. Latour, 2005). Zo onderzoeken Neyland en Möllers (2017) bijvoorbeeld de rol van algoritmes in videotoezicht waarbij ze vaststellen dat algoritmes door een netwerk van mensen en dingen opereren die in interactie zijn met elkaar en daaruit macht verwerven. De macht zit dus niet zoals in de eerste categorie in het algoritme maar de macht komt voort uit het netwerk van het algoritme. Introna (2016) stelt ook dat algoritmes in hun netwerk van associaties gezien moet worden: *“it is an algorithm because it is enacted as such by a heterogeneous assemblage of actors”* (p. 23). De macht van een algoritme moet bijgevolg altijd begrepen worden vanuit dat assemblage van alledaagse praktijken. Het algoritme in zijn context of in het netwerk bekijken, betekent dan ook dat een algoritme op conceptueel niveau een actant met agency is (Tufekci, 2015).

De derde categorie bestaat uit onderzoeken of onderzoekers die stellen dat de macht van het algoritme in de notie en het idee van het algoritme zit. Hierbij gaat het niet over de techniciteit noch over de context maar over: *“the concept of the algorithm [is] holding powerful and convincing sway in how things are done or how they should be done”* (Beer, 2017b, p. 2). De macht van het algoritme in de notie en het idee van een algoritme komt volgens Beer (2017b) via twee manieren tot stand. De eerste manier is om te kijken naar de rol van algoritmes in het creëren van waarheid. Dat wil zeggen dat algoritmes een rol kunnen spelen in het creëren van waarheid (i.e. de output van een algoritme als waarheid). De tweede manier gaat over de notie van het algoritme zelf. Hoe wordt er over een algoritme gesproken in een bepaalde discours en wat zijn de mogelijkheden van algoritmes volgens dit discours? Het is precies dit dat Williamson (2015) bevestigt in zijn onderzoek naar de macht van algoritmes: *“the power of software is not in its technical instruction but in how it sinks into collective discourse”* (pp. 85-86). Er is nog weinig onderzoek voorhanden die de macht van algoritmes op deze manier onderzocht. Een voorbeeld beschrijven Kitchin en Dodge (2011) binnen *automated management*⁹ waarbij een discours gebruikt wordt dat gerelateerd is aan veiligheid, efficiëntie, productiviteit, betrouwbaarheid...

In de literatuur worden drie plaatsen aangeduid waar de macht van algoritmes geanalyseerd wordt. In de eerste plaats is de macht van het algoritme te analyseren in het algoritme zelf. Dat betekent dat

⁹ *“Automated management is the regulation of people and objects through processes that are **automated** (technologically enacted), **automatic** (the technology performs the regulation without prompting or direction), and **autonomous** (regulation, discipline, and outcomes are enacted without human oversight) in nature”* (Kitchin & Dodge, 2011, p. 85).

het algoritme *in zich* de mogelijk heeft om macht uit te oefenen (bv. door wat het mogelijk maakt door bepaalde algoritmische technieken). In de tweede categorie is de macht van het algoritme een effect van het netwerk waarin het voorkomt. De macht van het algoritme is hier 'gesitueerd' in de associaties van het netwerk. Om de macht van het algoritme te analyseren, dient bijgevolg het netwerk geanalyseerd te worden. De laatste categorie stelt dat de notie en het idee van een algoritme macht bevat en deel uitmaakt van een bepaald discours. Om deze macht van het algoritme te analyseren, is een analyse van het discours nodig.

1.4.2 Algoritmische technieken

In de vorige paragraaf werd gesitueerd waar de macht van algoritmes geanalyseerd kan worden. We rapporteerden over die macht in een post-hegemonie. In de inleiding werd die macht van algoritmes ook als het doen doen omschreven. Om verder zicht te krijgen op de werking van algoritmes (i.e. het doen van algoritmes) wordt in deze paragraaf het doen vanuit een eerder technische invalshoek bekeken. Dat doen we aan de hand van een aantal specifieke onderzoeken naar algoritmische technieken. Er wordt gestart met twee concrete voorbeelden van deze technieken te bespreken (*Bayes classifier* en clusteranalyse) die terug te vinden zijn in de overzichtstabel van Kitchin en Dodge (2011) over *data-mining*-technieken. Naast deze twee voorbeelden, wordt nog een derde techniek besproken die niet zozeer gerelateerd is aan *data-mining* maar dient om data te *indexeren* en *ranken*.

De term algoritmische techniek omschrijft Rieder (2017) als: "*standardized yet plastic methods*" (p. 101) en "*units of knowledge and expertise in the domain of software making*" (p. 102). Het gaat met andere woorden om een bepaalde techniek die zich op een niveau tussen het concreet gebruik van een algoritme en de algemene theorie van een algoritme plaatst. Hiermee plaatst Rieder softwarespecialisten (i.e. algoritmici) op dezelfde voet als bijvoorbeeld statistici die ook over een breed gamma aan methodes en procedures beschikken om toe te passen op verschillende soorten situaties.

Aangezien Rieder (2017) de term algoritmische techniek, naar eigen zeggen, als eerste introduceert in het veld, starten we met zijn onderzoek naar een algoritmische techniek. Meer bepaald, de *Bayes classifier*. Deze techniek wordt, zoals uit het tweede deel van de naam af te leiden valt, gebruikt om een input te classificeren. Deze classificering is ontstaan uit *content-based filtering*. Hierbij wordt een bepaalde input geclassificeerd op basis van de inhoud van die input. Vervolgens wordt de tekst gebruikt om de betekenis (i.e. 'de inhoud') van een input te achterhalen. Ten slotte wordt de inhoud gecategoriseerd op basis van de relevantie (Rieder, 2017). Een concrete toepassing van deze algoritmische techniek is het categoriseren van binnenkomende mail. Op basis van de afzender of de inhoud van de e-mail, kan de mail als spam gecategoriseerd worden. Bepaalde woorden krijgen dus de betekenis van spam toegekend door de *Bayes classifier*. Initieel diende de *Bayes classifier* hierin 'getraind' te worden door per gewenste categorie een aantal voorbeelddocumenten te hebben. Die

voorbeelddocumenten worden toegevoegd door mensen. Het trainingsproces verliep als volgt: *“if a word appears very often in the training documents assigned to a certain category, but is rare for others, it becomes a strong clue or indicator for that category”* (Rieder, 2017, p. 108). Eenmaal dat de woorden (i.e. indicatoren voor de categorie) per categorie aangeleerd waren, dan kan de *Bayes classifier* zelf nieuwe documenten categoriseren (aangezien het hierin ‘getraind’ is). Rieder (2017) omschrijft dat het systeem zelf nog een stap verder gaat aangezien het algoritmisch systeem kon bijleren. Het systeem leerde met andere woorden meer dan in het trainingsproces werd aangeleerd. Dit op basis van nieuwe documenten.

Een tweede algoritmische techniek omschrijven Perrotta en Williamson (2018) als clusteranalyse (*cluster analysis*). Opnieuw zit wat het algoritme doet in de naam vervat, het vormt namelijk clusters van gegeven data (input) zodanig dat de data die in één cluster staan meer gelijkenissen met elkaar vertonen dan met de data in andere clusters (Perrotta & Williamson, 2018). Een voorbeeld hiervan is terug te vinden in hoofdstuk drie van deze masterproef waarbij de klaspraktijk op basis van een clusteranalyse wordt gevisualiseerd en geanalyseerd. In tegenstelling tot de *Bayes classifier* zoals besproken door Rieder (2017) moet deze algoritmische techniek niet getraind worden aangezien het geen vooraf opgelegde categorieën (i.e. clusters) bepaalt. Deze vaststelling is echter voorbarig volgens Perrotta en Williamson (2018) vermits om clusters te vormen er criteria vereist is. Deze criteria (of dit criterium) weerspiegelt zich namelijk in de gevormde clusters. De data vormt zich met andere woorden steeds in clusters maar deze clusters zijn niet noodzakelijk de enige mogelijkheid om de data te clusteren. Dat wil dan ook zeggen dat de clusters: *“are not just ‘found’ but, at the same time, actively constituted”* (Perrotta & Williamson, 2018, p. 8). Dat beargumenteren ze vanuit de formule van de *K-means clustering technique* waarin het algoritme iteratief werkt totdat het ‘beste’ resultaat gevonden wordt. Het ‘beste’ resultaat is gebaseerd op een gelijkheids criterium. De clusters vormen zich dus afhankelijk van dit criterium en niet op basis van bijvoorbeeld vooraf bepaalde categorieën.

Een derde algoritmische techniek is *indexing* en *ranking* (Rieder, 2017). Deze twee ‘technieken’ worden samen besproken vermits deze meestal samen gebruikt worden waarbij de ranking bepaald wordt door de indexscore. Het meest sprekende voorbeeld dat (onder andere) gebruik maakt van deze technieken is het Google-algoritme. Dat algoritme dient om relevante zoekresultaten te onderscheiden van irrelevante zoekresultaten (Gillespie, 2017). Hiervoor maakt men gebruik van een *indexing* techniek die een indicatie geeft van de relevantie van documenten of webpagina’s. Een voorbeeld van een formule om die relevantie te bepalen vinden we in het onderzoek van Rieder (2017): *“ $P(\text{DocumentIsRelevant}) = (\text{WordsQuery} \cdot \text{WordsDoc}) * P(\text{DocumentUse})$ ”* (p. 107). Dit betekent dat de relevantie van een document afhankelijk is van de overeenkomst tussen een zoekterm (WordsQuery) en documenten met die zoekterm (WordsDoc) waarbij de relevantie stijgt indien een document vaak

gebruikt wordt. Deze formule geeft meteen ook inzicht in hoe het Google-algoritme (weliswaar is het een andere, onbekende formule) gemanipuleerd kon worden zoals aangetoond in het onderzoek van Gillespie (2017) waarin gerapporteerd wordt over een bepaalde zoekterm in een document waar plots veel naar werd verwezen (de $P(\text{DocumentUse})$ steeg).

1.5 Algoritmes onderzoeken

“Given the inevitably, confidential, proprietary and highly technical nature of the core algorithms that now socially sort so many key social domains, what research techniques and paradigms can offer any genuine assistance here?” (Graham, 2005, p. 576)

In dit hoofdstuk wordt dieper ingegaan op hoe algoritmes in de geselecteerde, sociaalwetenschappelijke literatuur onderzocht worden. Hierbij ligt de focus op hoe het algoritme methodologisch benaderd wordt en welke opvatting over het algoritme daarmee gepaard gaat. Om dit weer te geven, kozen we ervoor om de onderzoeksmethodes in dit hoofdstuk categorisch weer te geven. Deze categorieën kwamen tot stand via een (beknopte) systematische weergave van de onderzoeksmethodes van de *special issues: governing algorithms* (Ziewitz, 2016a) en *the social power of algorithms* (Beer, 2017a) waaruit vervolgens artikels waren gekozen die zich het meest duidelijk onder een categorie plaatsten. De artikels zijn met andere woorden exemplarisch en representatief voor de opgestelde categorieën.

De categorieën – fenomenologisch, etnografisch en discursief – zijn algemene onderzoeksmethodes of -opvattingen waaronder de (meeste) onderzoeken zich laat vallen en vormen op hun beurt de basis voor de opvatting van het algoritme. Merk vooraf op dat deze categorieën niet de intentie hebben om elkaar mutueel uit te sluiten of exhaustief te zijn. Dat betekent dat er een zekere mate van overlap mogelijk is tussen de categorieën en dat enkel de meest voorkomende soorten van onderzoek worden weergegeven. Iedere categorie wordt kort toegelicht met behulp van twee illustratieve onderzoeken die aantonen wat dit soort onderzoek ons leert over de werking van een algoritme.

1.5.1 Fenomenologisch

Een eerste categorie kunnen we plaatsen onder de noemer van interpretatieve fenomenologie. Hierbij gaat de aandacht van de onderzoeker uit naar de betekenis die gegeven wordt aan het algoritme door (gewone¹⁰) mensen. Bucher (2017) onderzoekt deze betekenisgeving bij een kleinschalige groep mensen die zich bewust werden van ‘het’ Facebook-algoritme. Op basis van deze bewustwording stelt Bucher (2017) een aantal categorieën van reacties op die voorbeelden zijn van een *algorithmic imaginary*. Een algorithmic imaginary definieert Bucher (2017) als volgt: *“ways of thinking about what algorithms are, what they should be and how they function”* (p. 30). Gelijkaardig met deze vorm van

¹⁰ Met gewone mensen worden mensen bedoeld die beroepsmatig niet over algoritmes nadenken.

betekenisgeving schrijven Souto-Otero en Beneito-Montagut (2016) over reactiviteitsstrategieën (*reactivity strategies*). Dat zijn strategieën die mensen toepassen als reactie op het feit dat ze gemeten of geëvalueerd worden. Deze strategieën kunnen opgevat worden als een vorm van betekenisgeving waarbij de betekenis die gegeven wordt aan het algoritme de reactiviteitsstrategie bepaalt. Souto-Otero en Beneito-Montagut (2016) onderscheiden twee vormen van strategieën namelijk *alignment* en verzet (*resistance*) waarbij de gebruikers het algoritme accepteren dan wel zich ertegen verzetten waarbij deze laatste uitgebreider aan bod komt namelijk *gaming, bordering, folding, en rebellion*.

Beide onderzoeken gaan over de betekenisgeving en hoe er op basis van die betekenisgeving gereageerd wordt door mensen op algoritmes. Het onderzoeken van de betekenisgeving toont wat (gewone) mensen op basis van bepaalde ervaringen over het algoritme vertellen. Dat betekent dat om een algoritme te 'kennen', de betekenisgeving van (gewone) mensen aan algoritmes onderzocht moet worden: Hoe denken/spreken mensen over algoritmes en hoe geven ze hieraan betekenis? Of nog, het algoritme is iets waar de gebruiker een betekenis aan kan geven. Het is dan ook opvallend dat de betekenisgeving voornamelijk pas komt nadat de eindgebruikers een zekere confrontatie met het algoritme ervaren en het algoritme als falend ervaren. Of zoals Bucher (2017) stelt: "*when algorithms do not behave in the way people expect, they tend to describe the system as broken*" (p. 36). Dat betekent dat de werking van het algoritme als falend ervaren wordt als het algoritme de gebruiker iets toont dat zij/hij niet had verwacht. Het gaat er met andere woorden om dat de eindgebruiker geconfronteerd is met hoe hij/zij door het algoritme wordt geprofileerd. Bucher (2017) geeft hierbij onder andere het voorbeeld van Eric Meyer die in het jaaroverzicht van Facebook geconfronteerd werd met het overlijden van zijn dochter datzelfde jaar waardoor hij het algoritme als wreed (of falend om de 'juiste' gevoelens op te roepen) omschrijft. Tegelijkertijd is deze bevinding (dat betekenisgeving voornamelijk tot stand komt als het algoritme als falend ervaren wordt) ook niet opvallend aangezien het net een kenmerk van een algoritme is dat het in de achtergrond moet werken.

1.5.2 Etnografisch

Een tweede groep kunnen we plaatsen onder de noemer etnografie. Dat verwijst naar de interesse van de onderzoeker in het langdurig volgen van iets of iemand in zijn of haar natuurlijke setting om diegene of datgene beter te begrijpen. Zo observeert Neyland (2016) bijvoorbeeld een projectgroep die een algoritme ontwikkelt dat ethisch moet zijn. Het gaat immers om een algoritme dat voor bewakingsdoeleinden wordt gebruikt in de publieke ruimte van een luchthaven. Deze ethiek vertaalt zich voor algoritmes volgens Neyland (2016) naar: "*being more open, transparent, available to be questioned, and even governed*" (p. 69). De projectgroep komt hiervoor telkens samen en moet verantwoording afleggen aan een ethische commissie waardoor het algoritme telkens aangepast en 'verbeterd' wordt. Waar Neyland (2016) de ontwikkelaars van het algoritme volgt, daar ligt de focus

van het onderzoek van Gillespie (2017) bij het algoritme zelf. Gillespie (2017) onderzoekt namelijk retrospectief hoe de resultaten (i.e. *output*) van het Google-algoritme veranderden voor een bepaalde zoekterm (i.e. *input*) door verschillende actoren te volgen die hierbij een belang hadden – een belang van zichtbaarheid verwerven. De gevolgde actoren waren het Google-algoritme, Dan Savage en Rick Santorum waarbij deze laatste twee met dezelfde zoekterm de hoogste zichtbaarheid wilden verwerven (i.e. bovenaan de zoekresultaten staan via Google) door in te spelen op het algoritme en het hierdoor aan te passen. Hierdoor constateerde Gillespie (2017) dat op die manier zicht gekregen kan worden op hoe een algoritme beïnvloed wordt door verschillende actoren.

Een algoritme volgen is zoals bovenstaande studies aantonen mogelijk door het algoritme zelf te volgen of de mensen rondom het algoritme te volgen. Uiteraard is het gamma aan onderzoeksmethodes en mogelijkheden om een algoritme te volgen nog veel uitgebreider. Voorbeelden hiervan zijn *technography* (Bucher, 2016), *reverse engineering* (Diakopoulos, 2015), *interested reading* (Rieder, 2017)... Maar de twee aangehaalde onderzoeken geven voldoende voeling weer waar het in deze categorie om gaat en wat hierdoor geïmpliceerd wordt over de opvatting van de werking van een algoritme. Het betekent namelijk dat algoritmes (en de mensen die het maken) *in principe*¹¹ (na)volgbaar zijn. Tufekci (2015) stelt zelf voor om algoritmes op te vatten als actanten – in lijn met Latour (2005). Op die manier wordt ook aangetoond dat algoritmes altijd in de maak zijn en dat (het maken van) een algoritme gevolgd kan worden.

1.5.3 Discursief

Een derde groep kunnen we omschrijven als discursief waarbij een analyse van het discours centraal staat (en/of de condities voor dit discours). Deze benadering gaat over hoe er over iets gesproken wordt (dit kan zowel schriftelijk als verbaal zijn) en welk soort spreken mogelijk gemaakt wordt. De manier van spreken gaat immers hand in hand met hoe er naar iets (in dit geval algoritmes) gekeken wordt.

Williamson (2017b) analyseert het discours van het *Smarter Education* programma van IBM (International Business Machines Corporation) waarbij het (neuro)discours geladen is met algoritmische leerprocessen. Dit uit zich enerzijds in de visie van IBM die gebaseerd is op cognitieve computers¹² zoals Williamson (2017b) aangeeft: “*IBM is taking a particular algorithmic model of brain functioning*” (p. 87). Anderzijds toont het discours aan hoe het brein omschreven wordt als een

¹¹ De studie van Neyland (2016) kan hierover wel een verkeerd beeld geven aangezien het algoritme hier rond een overheidsdienst circuleerde en niet binnen een privébedrijf met winstoogmerk. Deze laatste (Google, Facebook, ...) laten hoogstwaarschijnlijk immers geen externe onderzoeker toe.

¹² De idee hierachter omschrijft Turing (1950) als: “*instead of trying to produce a program to simulate the adult mind, why not rather try to produce one which simulates the child’s? If this were then subjected to an appropriate course of education one would obtain the adult brain*” (p. 457).

computer: “*big data processor*’, *brain-like computations*’, *algorithms that learn*’, *neural network learning algorithms*’, *brain-inspired algorithms*” (Williamson, 2017b, p. 87). Williamson gaat hierbij niet specifiek in op de techniciteit van de algoritmes maar gaat in op de assumpties, kaders en verwachtingen die ermee gepaard gaan vanuit de organisatie. Zo concludeert Williamson (2017b): “*IBM’s claims about the cognitive classroom represent a nexus of neuro knowledges and imagined neurofutures with technical expertise in learning algorithms, neural networks, cognitive computing, and neurosynaptic modelling*” (p. 94). Hieruit blijkt nogmaals dat algoritmes een fundament vormen in het discours van IBM.

Het discours dat iemand, zoals bijvoorbeeld IBM, hanteert hangt samen met het discours dat mogelijk wordt. Dit betekent concreet dat vandaag de dag algoritmes gepaard gaan met een bepaalde manier van spreken en handelen. Zo stelt Gillespie (2014): “*more than mere tools, algorithms are also stabilizers of trust, practical and symbolic assurances that their evaluations are fair and accurate, free from subjectivity, error, or attempted influence*”. Hiermee wijst hij vanuit een andere invalshoek naar het discours dat rond een algoritme hangt en door dat discours legitimiteit probeert te verwerven. Het gaat bij dit laatste immers om het algoritme dat onder andere als objectief en rechtvaardig gezien wordt.

Een derde onderzoek hierover vinden we bij Bucher (2016) die aangeeft dat een algoritme: “*an object [is] that provokes talk*” (p. 90). Dit betekent dat wat een algoritme doet dus niet per se ‘in’ het algoritme zit (Introna, 2016), maar dat het eerder zit in hoe: “*they [algorithms] are being articulated, discussed, and contested in the public domain*” (p. 90).

Deze laatste groep geeft een interessante piste weer over algoritmes. Het toont immers dat het gebruik van algoritmes gedocumenteerd en gepromoot wordt door een specifieke, nieuwe groep van experts (bv. IBM; Williamson, 2017b) – zo gaat het in onderwijs bijvoorbeeld niet meer (alleen) om (neuro-)psychologen of statistici maar om “*algoritmici*” (Williamson, 2016, p. 5). De expertise van algoritmici beschrijft Williamson (2016) als volgt: “*the expertise to turn educational data into useful intelligence that might shape the decision-making of policymakers, influence the interventions made by educational leaders or the pedagogic decisions made by teachers or even shape the choices of parents and learners themselves*” (p. 11). Dat betekent dat algoritmici het huidige discours beïnvloeden (door het überhaupt zelf mogelijk te maken).

De drie aangehaalde categorieën van onderzoeksmethoden tonen verschillende manieren om een algoritme te onderzoeken die elk op hun beurt een andere opvatting over de werking van het algoritme weergeven. Zo omschrijft de eerste categorie het algoritme als iets waar betekenis aan gegeven wordt en waarop men kan reageren. Deze opvatting insinueert dat een algoritme datgene is wat de gebruiker

ervaart. De tweede categorie geeft aan dat algoritmes gevolgd kunnen worden. Dit volgen kan zowel over het algoritme zelf gaan door te kijken naar de input en output (cf. reverse engineering¹³), als over het observeren van een projectgroep die een algoritme maakt. Wat deze onderzoeken tonen over de werking van een algoritme is dat ze altijd in de maak zijn en dat dit maken gevolgd kan worden. De derde en laatste categorie, discursief, toont hoe de makers van het algoritme zelf over een algoritme spreken en dit promoten. Dat toont aan dat een algoritme gepaard gaat met een bepaalde manier van spreken (en denken) maar ook een bepaalde manier van spreken (en denken) mogelijk maakt. Dat betekent dat de werking van een algoritme niet alleen een technische aangelegenheid, maar ook een talige aangelegenheid is.

1.6 Conclusie

Het blijkt vandaag de dag dat er geen ontkomen meer is aan algoritmes. Algoritmes zijn overal om ons heen en we maken er al dan niet bewust of gewild meerdere malen per dag gebruik van. Daarom zocht dit hoofdstuk een antwoord op de socio-technische onderzoeksvraag naar de werking van een algoritme. Om hierop een antwoord te bieden kozen we vier verschillende invalshoeken (situering en definiëring; kenmerken; werking en onderzoeksmethodes). Zo verschijnt het algoritme onder iedere invalshoek net iets anders. In de eerste invalshoek van situering en definiëring bleek een algoritme voornamelijk opgevat te worden als een procedure binnen een digitaal apparaat die een gegeven input tot een vooraf bepaalde soort output leidt. Deze vaak gehanteerde opvatting over wat een algoritme is, bleek al snel (in de geselecteerde literatuur) onrecht te doen aan de werkelijke complexiteit van een algoritme. Vandaar dat het algoritme in een tweede invalshoek benaderd wordt volgens de kenmerken die toegeschreven worden aan algoritmes. Hierin constateerden we dat het algoritme vaak als een black box omschreven wordt wat verwijst naar de kenmerken van een algoritme als (technisch) complex, heterogeen en vaag (i.e. 'op de achtergrond'). In het derde hoofdstuk van het eerste deel werd gerapporteerd over de macht van algoritmes in een post-hegemonie en werden een aantal onderzoeken toegelicht die deze macht op verschillende 'plaatsen' situeerden. Vervolgens werd de eerder technische werking van het algoritme beschreven door onderzoeken naar algoritmische technieken toe te lichten. In het laatste hoofdstuk categoriseerden we een aantal onderzoeksmethodologieën. De categorieën dienden als kapstukken om een aantal verschillende onderzoeken aan te hangen die vanuit een gelijkaardige methode vertrekken.

¹³ Reverse Engineering is een methode waarbij de onderzoeker zelf met het algoritme 'speelt' om zicht te krijgen op de werking door telkens de input en output te vergelijken (Diakopoulos, 2015, p. 404).

Deel 2: Digitale klaspraktijk: empirisch onderzoek

2.1 Inleiding

In dit hoofdstuk wordt gebruik gemaakt van de opgedane inzichten in het eerste deel om een concrete onderwijspraktijk te benaderen. Hierbij gaan we onderzoeken wat iets is of doet door het netwerk in kaart te brengen. Het netwerk in kaart brengen betekent dat de macht van het algoritme in het netwerk gesitueerd wordt. Om zicht te krijgen op de macht van het algoritme wordt het netwerk geanalyseerd. Dit betekent dat het algoritme als actant opgevat wordt in een netwerk dat navolgbaar is en geobserveerd kan worden. In deze onderwijspraktijk ligt de focus op het digitale en de eventuele sturing erdoor waar zicht op gekregen kan worden. Het gaat met andere woorden niet enkel om wat algoritmes doen in een bepaalde onderwijskundige praktijk (dit is de werking, het doen van algoritmes) maar ook om wat algoritmes doen doen (dit is hoe ze een centrale rol spelen en macht verwerven). Aangezien algoritmes een abstract concept zijn, spreken we in dit deel over het digitale waar algoritmes een inherent onderdeel van vormen zoals in het eerste deel aangetoond werd. De digitalisering van de klas kan met verschillende benaderingswijzen onderzocht worden. In wat volgt worden drie benaderingswijzen besproken, om vervolgens onze eigen benadering toe te lichten.

In een eerste benadering, gaat men er vanuit dat het digitale de klas verandert in een *digital classroom* (Gordon, 2000; John & Wheeler, 2008). Dit betekent dat het digitale bepaalde aspecten van een klaslokaal (bv. computer in klas) of het klasgebeuren (bv. agenda digitaal invullen) heeft veranderd. Kenmerkend aan deze benadering is dat de technologie als determinerend wordt gezien (de la Cruz Paragas & Lin, 2016). Dit technologisch determinisme betekent dat de technologie, waar het digitale ook onder valt, de (toekomst van de) maatschappij bepaalt, wat impliceert dat veranderingen een gevolg zijn van de technologie en als dusdanig daaraan toe te schrijven zijn. De reden hiervoor is dat technologie – volgens het technologisch determinisme – bepaalde intrinsieke eigenschappen heeft waardoor het *agency* heeft. Toegepast op het klaslokaal bijvoorbeeld betekent dit dat de digitalisering als input voor een verandering zorgt van het klaslokaal waarbij de verandering toe te schrijven is aan het digitale. Zo stelt Valenti (2000) dat de klas veranderd is in een *High-Tech Classroom* waarop leraren voorbereid moeten worden. Een ander voorbeeld is *Massive Open Online Courses* waarbij deze de fysieke ruimte van het klaslokaal zodanig verandert dat het als het ware verdwijnt door toedoen van digitalisering. Het klaslokaal als *worldwide classroom* (Breslow, et al., 2013).

Een tweede benaderingswijze richt zich op de betekenisgeving aan de digitalisering van de klas. De technologie wordt hierbij ook opgevat als een input. Weliswaar niet als een input die een klas verandert maar wel als een input waar leerkrachten betekenis aan geven of waarlangs leerkrachten aan zichzelf betekenis geven. Zo stellen Tondeur, De Bruyne, Van Den Driessche, McKenney en Zandvliet (2015) dat de digitalisering en de setting van het digitale in een klas deels afhankelijk zijn van

de pedagogische oriëntatie van de leerkracht. Of nog, op basis van de betekenis die leerkrachten geven aan het digitale, past de leerkracht zijn of haar klaslokaal en didactiek aan. Salavati (2016) komt tot gelijkaardige resultaten door onderzoek naar de betekenisgeving van leraren aan digitale technologieën die ze in hun dagelijkse praktijk gebruiken.

De toon is bij beide onderzoeksbenaderingen gelijk, namelijk het digitale staat los van de mensen waarbij het respectievelijk gaat om de (effectiviteit van de) technologie zelf of de betekenis die hieraan gegeven wordt. Hierbij krijgt de digitalisering een stem via de mensen binnen een onderwijssetting waardoor het digitale op zich *de facto* niet aan bod komt. De klas en het digitale spreken niet zelf, maar krijgen een stem via de menselijke actoren. Nochtans lijkt het ons essentieel om de klas zelf te laten spreken – dit is van binnenuit onderzoeken – waarbij het digitale al dan niet een stem krijgt. Hiervoor wordt een derde, relationeel-sociomateriële, benadering gebruikt. De derde benadering laat de klas zelf spreken door zich te richten op wat er precies gebeurt in de klas en hoe dit vormgegeven wordt door de relaties tussen mensen (i.e. socio) en dingen (i.e. materieel) (bv. Sörensen, 2009).

De onderzoeksvraag is hierbij: Welke vormen van sturing ontstaan er in een klaspraktijk onder invloed van een algoritme? Dit beantwoorden we via een empirisch onderzoek dat aansluit bij de derde benadering. Hier is nog weinig onderzoek voor handen waarop we ons konden baseren waardoor het ook een exploratief onderzoek is. Meer bepaald baseren we ons in dit hoofdstuk op de methode van Decuypere en Simons (2014a, 2014b) om dit onderzoek uit te voeren. Hierbij toetsen we met andere woorden ook af of deze methode zich leent om onze onderzoeksvraag te beantwoorden.

2.2 Theoretische benadering

Om een antwoord te bieden op de onderzoeksvraag, gebruiken we een relationeel-sociomateriële benadering, namelijk de Actor Network Theorie (ANT) (Latour, 2005). Decuypere en Simons (2016) omschrijven de benadering als volgt: *“refuse[s] to separate the human dimensions of educational practices from their material dimensions, and rather focus[es] on the relational composition of these practices”* (p. 27). De klas is, volgens deze benadering, een relationele compositie van klaspraktijken. Een relationele compositie betekent dat de klas bestaat uit (een samenkomen van) *relaties* tussen actanten (zowel mensen als dingen). Deze relaties tussen actanten ontstaan in een specifieke klaspraktijk waarbij er iets specifiek gebeurt. Voor de onderzoeksvraag betekent dit verder dat het digitale en algoritmes als actant (kunnen) omschreven worden (bv. Tufekci, 2015) binnen het netwerk van een klas waardoor specifiek gekeken kan worden naar hun (eventueel sturende) rol binnen dit netwerk.

ANT is niet zozeer een welomlijnd theoretisch kader dat *a priori* als bril kan dienen om een praktijk te onderzoeken, maar is eerder een theoretische *benadering* die, voor dit onderzoek, gestoeld is op de

volgende drie sensibiliteiten (of principes): symmetrie, relationaliteit en ‘enactment’¹⁴ (Decuyper & Simons, 2014a, 2014b). Deze drie sensibiliteiten vormen de ‘denkwijzen’ die aan de grond liggen van dit onderzoek. Het verwijst dus niet zozeer naar een concrete onderzoeksmethode maar eerder naar een aantal richtinggevend sensibiliteiten die gekend en in het achterhoofd gehouden moeten worden doorheen het hele onderzoek – het beoogt in dat opzicht dus meer een attitude (of theoretische *benadering*) te zijn dan een onderzoeksmethode of theoretisch kader.

De eerste sensibiliteit, *symmetrie*, heeft betrekking op de actoren die deel uitmaken van een klascompositie. Een actor (of actant) kan in principe alles en iedereen zijn en kan dus zowel menselijk als niet-menselijk zijn waardoor een heterogeen assemblage ‘ontstaat’. De heterogeniteit verwijst hierbij naar het feit dat de actanten zowel menselijk als niet-menselijk (kunnen) zijn. Dit betekent concreet voor de klas dat de klas bestaat uit mensen (bv. Leerkracht, leerling...) en dingen (tafel, papier...). Precies omdat beide belangrijk zijn voor een klas, is het niet wenselijk – volgens deze benadering – om op voorhand meer belang te hechten aan enkel de dingen of enkel de mensen. Vandaar dat Latour (2005) suggereert om het neutralere begrip ‘actant’ te gebruiken in plaats van het aan mensen geconnoteerde begrip ‘actor’. Deze neutralisering door de term actant sluit aan bij de eerste sensibiliteit van symmetrie. Het verschil tussen mensen en dingen wordt hierbij wel (h)erkend, maar dit maakt geen verschil voor het verzamelen en analyseren van de data. Met andere woorden, er wordt geen *a priori* verschil gelegd op de rol van het menselijke dan wel het materiële in een klas.

De tweede sensibiliteit, *relationaliteit*, is verwant met de sensibiliteit van symmetrie doordat het de relatie beschrijft tussen de actanten. Het gaat er namelijk om dat deze actanten nooit op zichzelf iets doen of staan, maar telkens in relatie tot iets of iemand iets doen of staan. Een actant is, met andere woorden, pas en slechts dan een actant als het in relatie staat met minstens één andere actant. De relationaliteit verwijst dus naar een fundamentele verschuiving van o.a. denken in termen van (f)actoren naar relaties (tussen actanten). Dit betekent dat wat een klas is, niet omschreven kan worden vanuit één enkele actant (zoals leerkracht of computer) maar pas omschreven kan worden vanuit de relaties tussen actanten. Volgend uit het principe van symmetrie waarbij alles en iedereen een actant kan zijn, betekent dit dat alles en iedereen in principe ook met elkaar in relatie kan staan. En dat dus iedere actant, als gevolg van relationeel denken, zelf uit een netwerk bestaat. Of nog, wat een actant is en doet kan niet intrinsiek aan deze actant worden toegeschreven maar kan pas beschreven worden in relatie tot andere actanten waarbij dit zijn of doen afhankelijk is van de relaties met andere actanten. Dit samenspel van (alle) relaties tussen (alle) actanten vormt een netwerk (of een assemblage) dat tot stand komt door operaties en mechanismen (Zie 2.3.3). Concreet voor de klas

¹⁴ In het Nederlands hebben we hier geen goede vertaling voor. Enactment verwijst naar het moment dat iets gebeurt, een vertaling die hier dichtbij aanleunt is ‘in-de-maak’.

betekent dit dat we bijvoorbeeld de leerkracht enkel maar kunnen omschrijven in relatie tot een leerling, het bord, een toets... en niet als iemand los van al deze dingen.

De derde sensibiliteit, *enactment*, stelt dat een praktijk altijd in de maak is wat betekent dat een praktijk steeds een proces is dat bewerkstelligd wordt door en in een assemblage. Of nog, wat gebeurt – in de maak is – is het effect van dit (constant in de maak zijnde) samenspel van relaties tussen actanten (i.e. *ontology of becoming*). Hetzelfde geldt voor de ontstane clusters en regio's die niet *a priori* omschreven kunnen worden, maar telkens het effect zijn van het netwerk. Het zijn met andere woorden de interacties die de praktijk vormen waardoor de praktijk (in ons geval een klas) als in de maak wordt bestudeerd. Een implicatie voor het bestuderen van iets in de maak is dat dicht bij de alledaagse praktijk moet worden gebleven. Opnieuw betekent deze sensibiliteit dat er geen assumpties gemaakt kunnen worden van wat of wie typische actanten (zoals leerkracht, het bord...) zijn van een klas en dat er geen categorieën van wat een klas is of doet, opgelegd kunnen worden aan een praktijk aangezien het dagdagelijks maken van een klas door specifieke interacties gebeurt waarvan een klas het effect is.

Hoewel ANT meer omvat dan deze drie sensibiliteiten, is het niet het doel om deze volledige theorie uit te lichten. De drie sensibiliteiten volstaan immers om voldoende de achterliggende redenering van het onderzoek te kaderen waarbij het aantoont hoe wij een klas benaderden. Gegeven deze drie sensibiliteiten betekent dat dus dat de klas benaderd wordt als iets dat in de maak is – als effect van relaties tussen zowel menselijke als niet-menselijke actanten.

2.3 Methodologisch

Rekening houden met de drie sensibiliteiten heeft specifieke gevolgen voor (de kwaliteit van) het verzamelen en analyseren van de data. Vandaar dat in dit hoofdstuk wordt toegelicht hoe we de data hebben verzameld, gevisualiseerd en geanalyseerd om vervolgens kort in te gaan op de kwaliteit van deze drie stappen.

2.3.1 Dataverzameling

“Follow the actors themselves” (Latour, 2005, p. 12)

Berustend op de drie sensibiliteiten kunnen we de klas niet *a priori* omschrijven. Dit betekent dat we de (relaties tussen de) actanten moeten volgen om zo van binnenuit te beschrijven wat een klas is. Het volgen van de actanten geldt zowat als slogan voor ANT (Fioravanti & Velho, 2010; Ruming, 2009) en betekent dat de focus ligt op de actanten en de directe interacties (of relaties) tussen deze actanten. Deze interacties werden telkens genoteerd als een werkwoord en verwijzen naar een activiteit tussen twee actanten. Aangezien een actant pas een actant is als hij/zij in relatie staat met een andere actant, betekent dit dat er telkens een activiteit plaatsvindt tussen deze actanten. Bijgevolg worden enkel die activiteiten opgenomen die effectief iets doen – actief zijn. Deze precieze activiteiten laten zich telkens

omschrijven in werkwoorden (bv. de leraar *deelt* een tablet *uit*, Google Maps is *verbonden* met internet, leerkracht *maakt groepjes* via Group Maker...). Om de actoren te volgen kozen wij voor twee etnografische onderzoeksmethodes, namelijk de *registratie* (i.e. participerende observatie) en het *verhoor* (i.e. individueel interview).

2.3.1.1 *Setting*

Om de data te verzamelen werd doelbewust een klas gekozen waarbij een leerkracht aangaf geregeld met een digitaal bord te werken en gebruik te maken van tablets tijdens haar lessen. De keuze hiervoor is omdat we op die manier wisten dat we zeker ook digitale actanten zouden kunnen volgen.

De observatie vond plaats in de namiddag in een klas (15 leerlingen) van het zesde leerjaar (+/- drie uren geobserveerd) in een landelijk gelegen basisschool. De geobserveerde lessen waren wereldoriëntatie (90 minuten), verkeer (20 minuten), taal (40 minuten) en dans (10 minuten).

2.3.1.2 *Observatie – registratie*

De participerende observatie diende om de actanten te volgen tijdens het maken van een klas. Aangezien de focus lag op het volgen van actanten, kreeg de observatie vorm als een *registratie* (Decuyper & Simons, 2016). Dit betekent dat geregistreerd wordt, wat gezien wordt. Hiermee bedoel ik dat de notities van het observeren descriptief zijn. Hierbij werd gebruik gemaakt van een vooraf gestructureerd observatieformulier (zie OBijlage 1: Observatieformulier) gebaseerd op twee vragen. Door de observatie vooraf te structureren bepaalt het een bepaalde manier van kijken, wat helpt om gericht te kijken naar wat er gebeurt en zo meer kan worden gezien. Toch moet de onderzoeker voldoende openstaan voor wat er gebeurt waardoor het formulier niet als bindend geldt. Dit betekent dat de notities van de observator zich niet hoeven te beperken tot wat het formulier aangeeft, maar dat er ook 'vrije' observatienotities genomen mogen worden. Het formulier werd chronologisch en analoog (niet-digitaal) ingevuld en maakte gebruik van afkortingen (bv. Lk = leerkracht, Il = leerling, db = digitaal bord). Om praktische redenen werd in de mate van het mogelijke de leerling gespecificeerd tijdens de observatie maar dit was uiteraard niet volledig haalbaar gedurende de hele observatie waarbij geen volledige groep leerlingen voortdurend in de gaten kon worden gehouden. Een implicatie voor het beantwoorden van de onderzoeksvraag hierbij is dat het doen van een leerling in principe over 'iedereen en niemand' gaat. Aangezien een netwerk in principe oneindig doorloopt, kozen we om ons te beperken en het netwerk te af te snijden (*to cut*; Strathern, 1996) tot de sociale en digitale (directe) interacties.

2.3.1.3 *Interview – verhoor*

Een beperking van een klasobservatie is dat er enkel gekeken wordt naar wat zicht afspeelt in de fysieke ruimte van een klaslokaal waardoor niet kon worden geregistreerd wat reeds voorafgegaan was aan deze praktijk (bv. 's avonds lessen voorbereiden, presentatie maken, agenda invullen, ...). De

praktijk (van een klas) is immers meer dan louter wat zich afspeelt in die fysieke ruimte (bv. leerkracht die 's avonds lessen voorbereidt, de agenda nog invult...). Of nog, de praktijk – als effect – is aan meer toe te schrijven dan enkel de interacties die geregistreerd kunnen worden. Om deze bredere praktijk van een klas mee op te nemen, diende het individueel interview. Dat verliep als een soort “*verhoor*” (zie Bijlage 2: Voorbeelden interviewvragen) (Decuyper & Simons, 2014b) waarbij kort vragen gesteld werden over de (inter)acties die voorafgingen aan deze praktijk (bv. aanwezigheden opnemen, maaltijden doorgeven, mailen met een collega voor materiaal, lesvoorbereiding maken via Word, bordschema maken via PowerPoint...) en die nog zouden volgen (bv. De leraar kan thuis behaalde doelstellingen aanduiden, ouders krijgen een overzicht van de resultaten van wat hun zoon/dochter gedaan heeft, leerlingen krijgen automatisch aangepast huiswerk, leerkracht bekijkt overzicht resultaten en stelt lessen hierop af...). De interesse lag hierbij niet in de inhoud van de activiteiten of de betekenisgeving van de leerkracht maar in welke actanten hierbij aan bod komen en welke relaties daarmee aangegaan worden. Dit interview verliep direct na de lessen en duurde ongeveer 20 minuten en werd *verbatim* getranscribeerd.

2.3.2 Data visualiseren

Om de data te analyseren kozen we om de data visueel weer te geven op een topologische kaart. Het topologisch weergegeven van data ligt in lijn met onze theoretische benadering waarbij de topologische kaart toelaat om de praktijk in een relationele vorm te visualiseren. Een topologische kaart toont, met andere woorden, iets anders dan een meer traditionele, topografische kaart (bv. Tondeur, De Bruyne, Van Den Driessche, McKenney & Zandvliet, 2015; Vanden Bouverie & Simons, 2017). Een topografische kaart is er namelijk op gericht om zo waarheidsgetrouw de objecten, de infrastructuur, het landschap... weer te geven. Hierbij is de kaart bijvoorbeeld georiënteerd naar het Noorden, wordt het reliëf weergegeven door hoogtelijnen of kleuren, zijn de afstanden op schaal weergegeven... Dit is anders bij een topologische kaart. Een topologische kaart is immers een kaart die toont hoe een ruimte is samengesteld zonder vooraf te verwijzen naar absolute of metrische eigenschappen zoals bijvoorbeeld een x- en y-as (Venturini, Jacomy & Pereira, 2015).

In dit onderzoek toont de kaart hoe een klaspraktijk is samengesteld uit een geheel van relaties tussen actanten. Het toont met andere woorden hoe een praktijk in de maak is door de relaties tussen de actanten en niet door bijvoorbeeld de fysieke positie van de leerling ten opzichte van het digitaal bord. De praktijk benoemen als iets dat in de maak is, betekent dat de praktijk telkens het effect is van deze relaties tussen actanten en als dusdanig zal (h)erkend worden als die praktijk. Het gaat er met andere woorden om dat de praktijk niet opgevat wordt als iets dat op voorhand omschreven kan worden in categorieën.

Het zijn de relaties tussen de actanten die bepalen hoe de klaspraktijk topologisch wordt gevisualiseerd. Dit betekent dat de posities van de actanten en de relaties tussen de actanten telkens relatief ten opzichte van elkaar worden weergegeven. Het gaat er met andere woorden telkens om hoe (relaties tussen) actanten zich positioneren ten opzichte van elkaar en dus niet ten opzichte van een absolute of externe maatstaf (bv. schaal, x-as...). Bijgevolg kan in principe geen gebruik gemaakt worden van de gebruikelijke terminologie om iets te oriënteren op de kaart aangezien wat bovenaan op de kaart gesitueerd is net zo goed onderaan kan staan.

Alvorens de data topologisch gevisualiseerd kan worden, moeten eerst alle actoren die op een bepaald moment deel uitmaken van een interactie precies geïdentificeerd worden. Dit gebeurde in dit onderzoek op basis van de registraties. Tijdens het observeren werd dit direct opgeschreven en bij het interview werd dit uit het verbatim transcript gehaald. Hierbij werd telkens het netwerk van interacties van de digitale actor weergegeven (bijvoorbeeld als de leerkracht de functie 'volgende dia' van PowerPoint gebruikt, dan wordt MS PowerPoint, slideshow... ook mee opgenomen als actant vermits deze een interactie van 'verbonden' vormen). Iedere opgenoemde interactie wordt hier maar één keer gevisualiseerd. Met andere woorden, het gewicht van de interacties is allemaal even groot, het is dus niet de frequentie van de interactie die centraal staat maar het soort van interactie. Dit betekent dat een interactie die eenmalig voorkwam hetzelfde gewicht krijgt als een interactie die meerdere malen voorkwam en dat deze dus allemaal op dezelfde manier gevisualiseerd worden. Het aantal keer dat een interactie voorkomt visualiseren zou betekenen dat de edges tussen actanten dikker worden naarmate die interacties vaker voorkomen. Dit laatste werd in dit onderzoek dus niet mee opgenomen waardoor iedere edge hetzelfde gewicht heeft (i.e. iedere edge is even dik gevisualiseerd). De edges representeren wel telkens een bepaalde soort interactie die geregistreerd werd. De soort interactie diende om in de analysefase betekenis te geven aan de clusters.

Het tellen van hoeveel keer eenzelfde interactie voorkwam, heeft – volgens de gebruikte parameters en het ForceAtlas-algoritme – geen invloed op de relatieve *positie* van de actanten (i.e. knooppunten) op de topologische kaart. De relatieve positie van een actant op een topologische kaart wordt bepaald door de interacties die iedere actant aangaat (of niet) met (alle) andere actanten en niet door hoe vaak die interacties voorkomen. Bovendien baseerden we ons hierbij niet op het aantal keer dat die interactie voorkwam aangezien er een aantal 'interacties' continue aanwezig zijn. Dit laatste betekent dat deze interacties een constante activiteit van verbonden zijn en als dusdanig zo benoemd werden (bv. internet, toetsenbord-Desktop PC, Digitaal bord-iOmniWize...) en niet verwijzen naar discontinue interacties (zoals bijvoorbeeld tikken, *swipen*...).

Verder gaat het ook telkens om directe interacties. Zo is er tussen het beeldscherm en het digitaal bord *in se* geen interactie aanwezig vermits die interactie verloopt via de desktop pc. Maar ook omdat alles

en iedereen een actor kan zijn werd niet altijd enkel de 'kleinste' actor weergegeven aangezien dat een onvolledig, vertekend beeld van het netwerk zou weergeven. De interactie met de 'kleinste' actor is immers pas mogelijk doordat deze tot een 'grotere' actor behoort. Of nog, om tot een interactie met de kleinste actor te komen, dienen een reeks andere activiteiten plaats te vinden. Het is niet onbelangrijk om die reeks van andere activiteiten mee op te nemen aangezien op die manier zicht verkregen kan worden op eventuele centrale actoren. Ter illustratie kunnen we opnieuw verwijzen naar MS PowerPoint. Stel dat de leerkracht gebruik maakt van de actor 'Figuur invoegen', die in dit geval verwijst naar een functie van MS PowerPoint, dan wordt iedere interactie die hieraan voorafgaat en/of nodig is er ook aan gekoppeld. Zo is bijvoorbeeld de actor 'desktop PC' en 'muis' ook in relatie met de actor 'Figuur invoegen' en 'MS PowerPoint'. Door iedere interactie op die manier in te geven, kunnen bepaalde centrale actoren opgespoord worden.

Nadat iedere actant en interactie volgens bovenstaande manier werd omschreven, noteerden we deze actanten en interacties in een Excelbestand. Dit voorbereidende werk in het Excelbestand maakte het gemakkelijk om in een volgende stap de data in te geven in Gephi. Gephi is een computerprogramma dat toelaat om data te visualiseren als een netwerk (met nodes en edges, respectievelijk actanten en interacties). Om deze data te visualiseren maakten we gebruik van het *ForceAtlas-algoritme* (Jacomy, Venturini, Heymann, & Bastian, 2014) wat erop neerkomt dat actoren in clusters worden geplaatst op basis van de relatieve intensiteit van hun interacties (of niet) met (alle) andere actoren. Hoe dichter actanten bij elkaar staan, hoe meer deze direct (of indirect met relatief veel dezelfde nodes) met elkaar interageren. Omgekeerd staan actanten verder van elkaar indien deze niet direct met elkaar interageren. Op basis van deze intensiteit van relaties ontstaat een netwerk met clusters bestaande uit actanten en interacties die relatief intenser zijn dan andere (en dus meer directe interacties hebben met elkaar). Hierdoor ontstaan (mogelijks) verschillende, relatief van elkaar te onderscheiden clusters doordat sommige (clusters van) actanten meer direct met elkaar interageren dan sommige andere (clusters van) actanten. Gephi bevat ook een aantal instellingen (bv. de richting of het gewicht (i.e. de frequentie) van de interactie) en parameters (bv. aantrekkingskracht tussen nodes, zwaartekracht die nodes meer in het centrum brengt...) die naar believen gewijzigd kunnen worden. Een andere mogelijkheid die Gephi biedt, is om de grootte van de nodes representatief te maken voor het aantal relaties dat een actant heeft. Of met andere woorden, hoe groter een node is, hoe meer relaties het heeft. De volgende instellingen en parameters werden in dit onderzoek gebruikt in functie van de analyse. Voor de edges kozen we om deze allen hetzelfde gewicht (gewicht = 1) te geven aangezien we de frequentie niet hebben geteld tijdens het verzamelen van de data. Ook speelt de richting van de interactie geen rol voor deze analyse. Zo is een interactie tussen tablet-leerling en leerling-tablet hetzelfde. Zoals eerder gezegd kozen we voor het Force Atlas algoritme om de data te visualiseren. De

parameters werden hierbij aangepast totdat een heldere kaart getoond werd (Zie Bijlage 3 voor de precieze waardes en uitleg van de parameters). Zo kozen we bijvoorbeeld om de knooppunten weer te geven naar hun gewicht zodat duidelijk is welke actoren een centrum vormen. Het gewicht van deze knooppunten wordt bepaald door het aantal interacties die de actant aangaat met andere actanten. Hoe meer interacties de actant heeft, hoe groter het knooppunt is en hoe meer autoriteit de actant heeft (Venturini, Jacomy & Pereira, 2015).

2.3.3 Analyse

Het visualiseren van de data met behulp van een topologische kaart vormt het vertrekpunt voor de analyse. Het is immers zoals eerder aangehaald via de topologische visualisatie mogelijk om de praktijk in zijn relationele werking te analyseren. De analyse is hierbij grotendeels gebaseerd op het werk van Decuypere en Simons (2014a) waarin de focus van de analyse als volgt wordt omschreven: *“the focus is on who and what plays a role, and the relations involved in this who and what”* (p. 121). Om de rol van dit wie en wat te onderzoeken binnen een netwerk wordt, in een eerste stap, de compositie van het netwerk geanalyseerd. Hiervoor baseren we ons op het onderzoek van Decuypere en Simons (2014a) waarin vijf topologische dimensies aan bod komen. Deze dimensies zijn: regio's, centra, dichtheid, interface en infrastructuur (Decuypere & Simons, 2014a).

Ten eerste, **regio's** verwijzen naar clusters (Decuypere & Simons, 2014a). Dit zijn gebieden op de kaart waar een relatief te onderscheiden groep van actanten sterker interageren met elkaar. Iedere cluster toont hierbij een specifiek deel van de volledige praktijk waardoor deze clusters ook omschreven kunnen worden als deelpraktijken. De regio's kunnen sterk gedemarqueerd zijn indien regio's amper overlappen. Dit betekent dat iedere regio een specifieke deelpraktijk vormt. Omgekeerd kan het ook dat regio's relatief veel overlappen wat erop wijst dat een aantal actanten in meerdere regio's belangrijk worden gemaakt. Ten tweede, **centra** verwijzen naar actanten die met relatief veel andere actanten in relatie staan in een bepaalde regio (lokaal) of in het volledig weergegeven netwerk (globaal) (Venturini, Jacomy & Pereira, 2015). Deze centra staan in een regio of het volledig netwerk altijd vrij centraal aangezien deze met meer andere actoren verbonden zijn. Ten derde, **dichtheid** verwijst naar de 'hoeveelheid' interacties in een bepaalde cluster. Relatief veel interacties betekent dat veel actanten in die cluster met elkaar interageren. Een hoge dichtheid verwijst hierbij naar een deelpraktijk waarin veel actanten en interacties in het spel zijn om deze deelpraktijk te maken. Ten vierde, **interfaces** verwijzen naar actanten of groepen van actanten die 'toegang' hebben tot meerdere regio's. Dit betekent dat deze actanten of groepen van actanten in meerdere deelpraktijken een rol spelen en zich bijgevolg op de grens van deze regio's tonen. Of preciezer, nét door de rol van deze actanten ontstaan regio's. Indien het om actanten gaat, spreken we over grensactanten en indien het om een groep van actanten gaat spreken we over een grenszone. Ten slotte verwijst **infrastructuur**

naar de soort actanten waaruit een netwerk bestaat. Hierbij onderscheiden we actanten in vier: hardware, mensen, “online” software en “offline” software. Hardware verwijst, simplistisch gesteld, naar alles wat (fysisch) kan worden vastgenomen (bv. digitaal bord). Software verwijst, opnieuw simplistisch gesteld, naar dat wat niet (fysisch) vastgenomen kan worden (bv. PowerPoint) en bestaat zoals in het eerste deel werd aangegeven altijd uit algoritmes. Het criterium dat “online” software onderscheidt van “offline” software is de (noodzaak aan een) verbondenheid met internet. De opsplitsing tussen hardware en software is gebaseerd op de inzichten in het eerste deel. De software is een veelomvattende categorie waardoor gekozen werd om dit op te splitsen op basis van de verbinding met internet. Hoewel dit onderscheid in de bekeken literatuur over algoritmes niet aan bod kwam, is het plausibel om te stellen dat daarin een onderscheid gemaakt kan worden. Bijvoorbeeld online software die gericht is op het verzamelen van gegevens, het aanpassen van zoekresultaten (cf. *The Filter Bubble*) of de mogelijkheid heeft om online de software up te daten.

In een tweede en laatste stap verdiepen we de analyse, op basis van het werk van Decuypere en Simons (2014b), door te kijken naar de operaties en mechanismen die deze regio’s en praktijk maken. Hierbij omschrijven we *operaties* die beschrijven wat er gebeurt in een regio van actanten en hun interacties. Dit betekent dat operaties bepalend zijn voor de werking van een cluster of regio. Of nog, de regio’s zijn het effect van interacties tussen actanten. Of nog anders gezegd, de regio’s beschrijven hoe bepaalde interacties tussen actanten een groep vormen. Deze operaties omschrijven we als werkwoorden die eindigen op ‘maken’. Een voorbeeld van zo een operatie is iets hoorbaar maken. De clusters met hun operaties staan niet los van elkaar maar spelen op elkaar in. Dit betekent dat de praktijk ontstaat als effect van de operaties tussen clusters. De effecten van deze operaties omschrijven we als *mechanismen*. Analoog met operaties, betekent dit dat de mechanismen bepalend zijn voor de werking van een praktijk. Of nog, operaties gaan over de ‘werking’ van een cluster en mechanismen over het ‘samenwerken’ van deze clusters. De mechanismen omschrijven met andere woorden hoe een bepaalde praktijk gemaakt wordt als gevolg van op elkaar inspelende operaties. De mechanismen omschrijven we als verzelfstandigde werkwoorden die eindigen op ‘-ing’. Een voorbeeld van een mechanisme is visualisering. Deze operaties en mechanismen liggen aan de basis voor het maken van klas. Een klas is, met andere woorden, het effect van deze operaties en mechanismen.

2.3.4 Kwaliteit

Alvorens de resultaten te bespreken, wordt nog kort ingegaan op de methodologische kwaliteit van dit onderzoek. De methodologische kwaliteit van het onderzoek gaat hier in de eerste plaats hoe accuraat (i.e. dicht) de actoren gevolgd worden. Dit ligt in lijn met etnografisch onderzoek waarbij het ook gaat om fenomenen of praktijken rijk te beschrijven (Decuypere & Simons, 2014b). Deze rijke

beschrijvingen en accuraat volgen manifesteren zich in beschrijvingen (i.e. registraties) die zo nauwkeurig (en zo veel) mogelijk de actanten en relaties achterhaalt en beschrijft. Dit betekent dat wat er precies gebeurt, telkens zo objectief mogelijk moeten worden opgeschreven. Dat wat opgeschreven wordt is immers geen interpretatie maar een registratie waarmee verwezen wordt naar het registreren van wat men precies ziet van interactie. Om aan deze kwaliteitseisen te voldoen, werd in dit onderzoek gebruik gemaakt van een observatieformulier (zie Bijlage 1: Observatieformulier) dat uitnodigt om korte registraties te noteren. Hierbij lag de focus tijdens het observeren op de interacties die duidelijk zichtbaar waren (bv. leerkracht *swipet* op het digitaal bord). Al deze interacties werden telkens op een zo precies mogelijke manier omschreven. Aangezien het onmogelijk is om als enige observator alle interacties te registreren, werden na de observatie een aantal vanzelfsprekende interacties aangevuld (bv. verbinding tussen desktop PC en toetsenbord). Om toch in de mate van het mogelijke zoveel mogelijk interacties correct te registreren, werd gebruik gemaakt van afkortingen tijdens het noteren om zo alles snel en correct te kunnen registreren.

Naast de kwaliteit van het verzamelen van data, dient ook nog vermeld te worden dat de resultaten het gevolg zijn van een iteratief proces tussen de visualisatie van de data en de interpretatie ervan (Venturini, Jacomy & Pereira, 2015). Hierbij wordt de interpretatie van de data telkens teruggekoppeld en afgetoetst aan de visualisatie van de data.

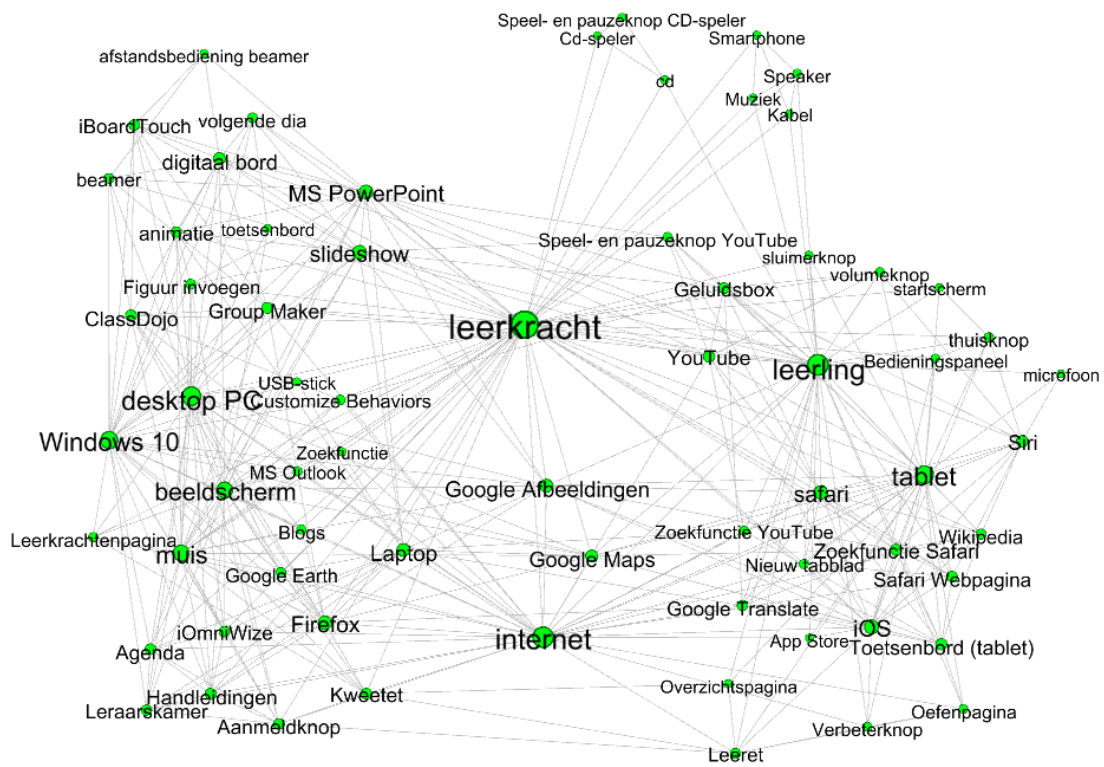
2.4 Resultaten

Dit hoofdstuk rapporteert over de gevonden resultaten van het onderzoek waarbij het visuele resultaat tekstueel ondersteund wordt. Het betreft één klaspraktijk. Er wordt gestart met de sociodigitale assemblage te tonen (zie Figuur 3 bestaande uit 68 knooppunten en 392 edges). Hierbij wordt kort ingeleid wat we zien en hoe dit gelezen kan worden. Daarna wordt er gerapporteerd over de analyse aan de hand van de vijf dimensies zoals omschreven in de paragraaf analyse. Tot slot rapporteren we over de gevonden mechanismen als effect van het samenwerken van operaties.

2.4.1 Sociodigitale assemblage

Om de assemblage (zie Figuur 3) van een klas te lezen kunnen we starten bij één actant en diens relaties volgen. Kijken we bijvoorbeeld naar de actant digitaal bord (zie Figuur 4) dan zien we dat dit in relatie staat met de leerkracht. De leerkracht gebruikt het namelijk om iets op te projecteren. Om iets te kunnen projecteren moet het digitaal bord ook in relatie staan met een beamer. Als software om iets te projecteren werd gebruik gemaakt van MS PowerPoint en de daarbij horende functies slideshow en animatie. Wat dit weergeeft is dat een actant nooit op zichzelf kan staan en steeds gedragen wordt door een netwerk van andere actanten. Zo gebruikt de leerkracht bijvoorbeeld het digitaal bord omdat dit een relatie aangaat met PowerPoint. Dit betekent dat de actanten elkaar nodig hebben om te doen wat ze doen. Of nog, de iedere actant wordt telkens gedragen door een netwerk van actanten.

Daarnaast betekent dit ook dat wat een actant doet afhankelijk is van de relatie met de andere actant(en). Deze relatie omschreven we telkens met een werkwoord als een interactie. Bijgevolg verwijst ieder werkwoord naar een specifieke interactie tussen twee actanten. Concreet betekent dit dat wat het digitaal bord doet in relatie tot de beamer, anders is dan wat het digitaal bord doet in relatie tot de leerkracht. Zo is de interactie tussen het digitaal bord en de beamer projecteren en de interactie tussen het digitaal bord en de leerkracht 'swipen'. Hieruit volgt dat een actant niet anders kan dan omschreven worden vanuit de relaties die het aangaat met andere actanten.



Figuur 3. Sociodigitaal assemblage



Figuur 4. Actor Netwerk van digitaal bord

2.4.2 De compositie van het netwerk

Om de resultaten van de sociodigitale assemblage van de klaspraktijk toe te lichten komen de volgende vijf dimensies aan bod: regio's, centra, densiteit, interfaces en tot slot infrastructuur. Deze dimensies bieden een kijkwijze en taal om de compositie van het netwerk te bespreken.

2.4.2.1 Regio's

In Figuur 3 zien we dat sommige actanten dicht bij elkaar gegroepeerd staan dan andere waardoor clusters of regio's ontstaan die te onderscheiden zijn van elkaar. Zo is de afstand tussen bijvoorbeeld Wikipedia (rechts gepositioneerd in Figuur 3) korter ten opzichte van Tablet (rechts gepositioneerd in Figuur 3) dan van Desktop PC (links gepositioneerd in Figuur 3). In Figuur 5 hebben we deze clusters ongeveer gekleurd¹⁵ op basis van de *modularity class*-functie van Gephi (Zie Bijlage 4: Resultaat Modularity Class) (Decuyper & Simons, 2014b). Deze functie toont hoe het globale netwerk is ingedeeld in gemeenschappen. De indeling in gemeenschappen is gebaseerd op een algoritme ontwikkeld door Blondel, Guillaume, Lambiotte en Lefebvre (2008)¹⁶. In wat volgt worden de actanten en interacties in iedere cluster besproken en welke operatie plaatsvindt in deze clusters.

Ten eerste, de operatie die herkenbaar is in de groene cluster benoemen we als **hoorbaar maken**. Voorbeelden van actanten in deze cluster zijn: cd, Muziek, Speaker... Voorbeelden van interacties tussen deze actanten zijn luisteren, afspelen, indrukken... Deze actanten en interacties verwijzen naar

¹⁵ De leerkracht werd niet meegekleurd vermits deze tot alle clusters behoort en dit daardoor visueel onduidelijk zou worden.

¹⁶ Het algoritme werkt in twee fases die iteratief zijn waarbij, simplistisch gezegd, in een eerste fase ieder knooppunt systematisch vergeleken wordt met 'twee buren'. Vervolgens wordt dit knooppunt geplaatst bij de 'buur' waarbij de modulariteitsscore het hoogst blijft. De tweede fase verloopt gelijkaardig maar werkt niet meer met de knooppunten maar met de gevonden gemeenschappen (dit zijn de modules waar de knooppunten van de eerste fase in geplaatst werden).

een luisteractiviteit als gevolg van de operatie hoorbaar maken. Hoewel iedere regio relatief afgescheiden is van de andere regio's, is de regio van het hoorbaar maken het archetype van een gedemarqueerde regio. Zo wordt bijvoorbeeld de cd in de cd-speler gestoken en afgespeeld die dag door de leerkracht en luisteren de leerlingen naar de cd waarbij de cd, cd-speler... enkel dat doet. Ten tweede, de paarse cluster vormt zich als een ster van interacties rond de leerling. De operatie in deze paarse regio kunnen we benoemen als **hanteerbaar maken**. De leerling drukt de sluimerknop in om de tablet aan te leggen of de speel- en pauzeknop van YouTube wordt gebruikt om YouTube klaar te zetten op het juiste moment. Ten derde, de blauwe cluster bestaat uit interacties waarbij informatie werd opgezocht via een tablet. Hierbij diende het toetsenbord om zoektermen in te typen in de zoekfunctie van Safari of om afbeeldingen op te zoeken via Google Afbeeldingen waarbij gebruik werd gemaakt van de webbrowser Safari die verbonden was met het internet (i.e. internetverbinding). De operatie in deze regio kunnen we benoemen als **raadpleegbaar maken**. Ten vierde, in de gele cluster staan actanten zoals Handleidingen, Leerkrachtenpagina, Firefox... Voorbeelden van interacties hiertussen zijn synchroniseren, invullen, aanmelden... De actanten en interacties verwijzen naar administratieve en voorbereidende taken. In deze regio kunnen we de operatie benoemen als **beheersbaar maken**. Bijvoorbeeld, de leerkracht vult de agenda in met behulp van de handleidingen die online beschikbaar zijn (mits de juiste aanmeldgegevens). Bovendien kan opgemerkt worden dat deze actanten en interacties hoofdzakelijk verwijzen naar datgene dat buiten de lesuren valt. Ten slotte, de operatie in de rode cluster kunnen we benoemen als **presenteerbaar maken**. Zo is bijvoorbeeld MS PowerPoint verbonden met de beamer om een presentatie te projecteren waarin animaties gebruikt worden of projecteert de leerkracht het resultaat van Group Maker.

Tabel 1: Operaties, interacties en actanten van het sociodigitaal netwerk

Kleur	Operaties	Interacties (actanten)
Rood	Presenteerbaar maken	<ul style="list-style-type: none"> • <i>Swipen</i> (leerkracht – volgende dia) • Projecteren (beamer – digitaal bord) • Toevoegen (animatie – Figuur invoegen) • Groepjes maken (Group Maker – digitaal bord) • ...
Geel	Beheersbaar maken	<ul style="list-style-type: none"> • Invullen (agenda – leerkracht) • Aanmelden (Kweetet – Aanmeldknop) • Synchroniseren (agenda – handleidingen) • ...
Blauw	Raadpleegbaar maken	<ul style="list-style-type: none"> • Opzoeken (Zoekfunctie Safari – Wikipedia) • Vertalen (Toetsenbord (tablet) – Google Translate) • Informeren (Wikipedia – Leerling) • ...
Paars	Hanteerbaar maken	<ul style="list-style-type: none"> • Aanleggen (sluimerknop – leerling) • Geluidsvolume aanpassen (volumeknop – tablet)

		<ul style="list-style-type: none"> • Klaarzetten (Speel- en pauzeknop YouTube – YouTube) • ...
Groen	Hoorbaar maken	<ul style="list-style-type: none"> • Luisteren (muziek – leerling) • Afspelen (cd – cd-speler) • ...

2.4.2.2 Centra

Centra verwijzen naar actanten die met relatief veel andere actanten in relatie staan. Dit kan zowel lokaal (in een cluster of regio) als globaal (in het volledig netwerk) zijn. Deze centra zijn gevisualiseerd in Figuur 5 met de functie *Node Ranking Degree* in Gephi. Dit zorgt ervoor dat de grootte van een knooppunt afhankelijk is van het aantal interacties die dat knooppunt heeft.

In Figuur 5 is het knooppunt **Leerkracht** (centraal in Figuur 5) groter dan het knooppunt cd (bovenaan in Figuur 5). Dit betekent dat de leerkracht meer interacties aangaat en dit met meer andere, verschillende actanten dan de cd. De leerkracht is tevens het grootste knooppunt. Dat wil zeggen dat de leerkracht de actant is die het meeste interacties aangaat in vergelijking met alle andere actanten. Dit verklaart waarom de leerkracht een globaal centrum is. De leerkracht staat immers in het centrum van de topologische kaart doordat zij met bijna alle andere actanten in relatie staat. Op ongeveer gelijke hoogte met de leerkracht staan de actanten **Google Afbeeldingen, Google Maps en internet**. In Figuur 3 staan deze, net zoals de leerkracht, centraal in het netwerk waardoor deze ook globale centra vormen. In tegenstelling tot de leerkracht, staan deze drie globale centra iets lager in Figuur 3 en zijn ze ook kleiner aangezien deze minder interacties hebben. Specifiek hebben ze minder interacties met de regio presenteerbaar maken (rode regio) en zelfs bijna geen interacties met de regio's hoorbaar maken (groene regio) en hanteerbaar maken (paarse regio). Dat precies deze vier centra centraal staan betekent dat deze vier centra autoriteit in de klaspraktijk verwerven doordat verschillende andere actanten er interacties mee aangaan.

Een ander knooppunt dat groter is dan de meeste andere knooppunten is de actant leerling. Dit betekent opnieuw dat de actant, in dit geval de leerling, met veel andere actanten in relatie staat. Toch zien we in Figuur 3 dat de leerling niet globaal in het centrum staat zoals bijvoorbeeld de leerkracht, maar dat deze in een bepaalde regio in het centrum staat. De **leerling** is met andere woorden een lokaal centrum in de regio hanteerbaar maken. Dit komt omdat de leerling met alle actanten in deze cluster interageert. De regio hanteerbaar maken heeft nog een tweede lokaal centrum, namelijk de actant **tablet**. Hiervoor geldt dezelfde reden namelijk de tablet staat in relatie met alle andere actanten in deze regio. Deze twee lokale centra situeren zich bovendien aan de grens van twee regio's (zie interfaces). Vervolgens, in de regio raadpleegbaar maken is **iOS** een lokaal centrum. Deze actant is

groter dan alle andere actanten in deze regio en staat centraal in deze regio als gevolg van de interacties die het aangaat met alle andere actanten in de regio. In de regio van beheersbaar maken zijn zes lokale centra gevisualiseerd. Vier lokale centra bevinden zich centraal in de regio. Deze zijn **Firefox, Laptop, Muis en Beeldscherm**. Daarbovenop bevinden zich twee lokale centra aan de grens tussen twee regio's, namelijk **Desktop PC en Windows 10**. Tot slot, in de regio van presenteerbaar maken zijn **MS PowerPoint en Slideshow** de lokale centra.

Samenvattend gesteld, dankzij de visualisatie van het netwerk kan worden gekeken naar de globale en lokale centra. De visualisatie toont vier globale centra, namelijk: Leerkracht, Google Afbeeldingen, Google Maps en Internet. Naast deze globale centra zijn er 11 lokale centra, deze zijn: Leerling, Tablet, iOS, FireFox, Laptop, Muis, Beeldscherm, Desktop PC, Windows 10, MS PowerPoint en Slideshow.

2.4.2.3 *Densiteit*

De densiteit van een cluster representeert het aantal interacties in een bepaalde cluster. Hoe meer interacties er zijn in een cluster, hoe dichter (in termen van dichtheid) de cluster is.

In Figuur 5 is de regio van hoorbaar maken een voorbeeld van een cluster met een relatief **lage densiteit**. Er is relatief weinig interactie tussen de actanten onderling in deze cluster. De Smartphone, bijvoorbeeld, staat slechts met vier andere actanten in relatie (namelijk Speaker, Muziek, Kabel en Leerkracht). Dit betekent dat om iets hoorbaar te maken, in deze deelpraktijk, relatief weinig andere actanten nodig zijn en dat iedere actant bijgevolg eigen specifieke functie heeft en eerder op zichzelf staat. Ditzelfde geldt voor de regio hanteerbaar maken. Hoewel de densiteit in deze regio iets hoger is, is ze nog steeds relatief laag. Bijvoorbeeld, de sluimerknop wordt enkel gebruikt door de leerkracht en de leerling waarbij de interactie indrukken is om de tablet aan te leggen. De sluimerknop staat met andere woorden opnieuw relatief op zichzelf en heeft een specifieke functie in de regio hanteerbaar maken.

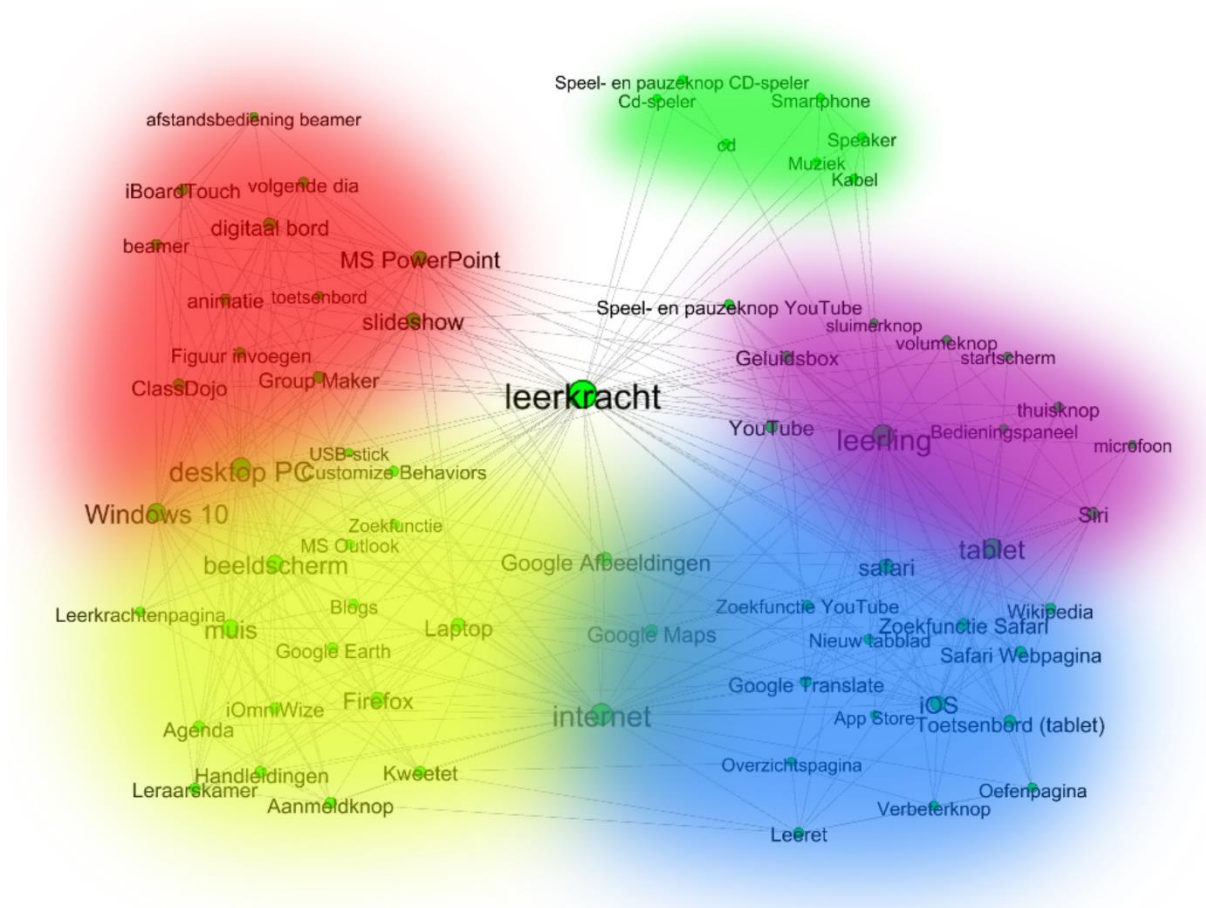
In de regio's beheersbaar maken, presenteerbaar maken en raadpleegbaar maken staan de actanten bijna allemaal met elkaar in relatie waardoor deze clusters een **hoge densiteit** hebben. Dit betekent dat de actanten afhankelijker zijn van de interacties tussen meerdere verschillende actanten en meerdere functies hebben. Dit hangt bovendien samen met het aantal centra die in deze regio's aanwezig. De centra worden immers door hun interacties met andere actanten gecentreerd. FireFox, bijvoorbeeld, heeft 18 relaties met andere actanten in de regio beheersbaar maken. Dit komt omdat FireFox de webbrowser is waarmee de meeste andere actanten (Kweetet, Leraarskamer, Handleidingen, Blogs ...) in deze regio uitgevoerd worden.

2.4.2.4 Interfaces

Interfaces zijn actanten die een grens maken – als *interface* – tussen twee of meer regio's. Deze actanten bevinden zich dus aan de grens van twee of meer regio's en hebben daardoor 'toegang' tot meerdere regio's. Dit kan een grensactant zijn (indien het om één actant gaat) of een grenszone (indien het om een groep grensactanten in een zone gaat). Het is precies door deze grensactanten en grenszones dat er zich een demarcatie vormt tussen regio's. In Figuur 5 zijn er acht grensactanten.

De **leerkracht** is hierin uniek aangezien deze de enigste is die geen grenszone vormt. Daarnaast is de leerkracht de enigste grensactant die aan de vijf regio's grenst. Dit betekent dat de leerkracht een belangrijke actant is om deze klaspraktijk te maken aangezien deze zich tussen de verschillende regio's 'verplaatst'. Daarnaast betekent dit ook dat het veranderen van regio onder andere via de leerkracht gebeurt. De leerkracht is de enigste actant die op zichzelf staat en een grens vormt tussen vijf regio's.

De overige zeven grensactanten situeren zich in drie grenszones. Een eerste grenszone bestaat uit de **desktop PC** en **Windows 10** tussen de regio van presenteerbaar maken en beheersbaar maken. De desktop PC en Windows 10 maken het zo mogelijk om zich te verplaatsen van de ene regio naar de andere. Of nog, de desktop PC en Windows 10 vormen een passagepunt tussen de regio van presenteerbaar maken en de regio van beheersbaar maken. Bovendien zijn beide grensactanten in beide regio's nodig om deze deelpraktijken te maken. Een tweede grenszone bestaat uit **Google Afbeeldingen**, **Google Maps** en **Internet**. Deze drie grensactanten vormen een zone tussen beheersbaar maken en raadpleegbaar maken. Bovendien staat deze grenszone relatief op zichzelf wat betekent dat deze grenszone in verschillende deelpraktijken wordt gebruikt. Of nog, de drie grensactanten zijn in deze praktijk noodzakelijk om verschillende deelpraktijken te maken. Bovendien is wat de grensactanten doen afhankelijk van de regio waarin ze een rol spelen. Google Maps en Google afbeeldingen, bijvoorbeeld, worden in de ene regio gebruikt om een toekomstige presentatie beheersbaar te maken en dienen in de andere regio als informatiebron die wordt geraadpleegd. De laatste grenszone situeert zich tussen hanteerbaar maken en raadpleegbaar maken en bestaat uit de actanten **Leerling** en **Tablet**. De leerling en de tablet zijn beide noodzakelijk om iets hanteerbaar te maken en iets raadpleegbaar te maken. Opnieuw, zonder deze grenszone zouden deze twee deelpraktijken niet mogelijk zijn. Het is immers precies door deze twee grensactanten dat de regio van hanteerbaar maken zich onderscheidt van de regio van raadpleegbaar maken.



Figuur 5. Sociodigitaal assemblage: clusters en regio's

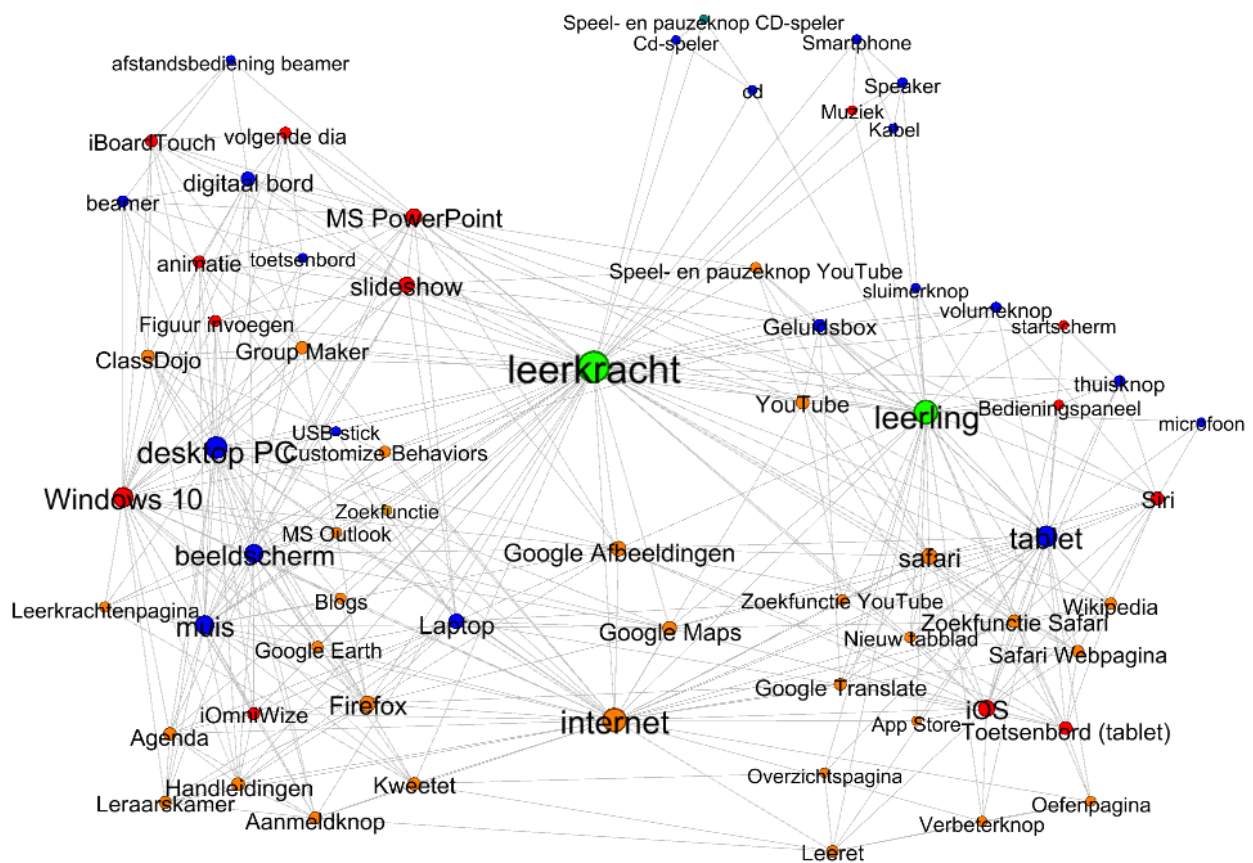
2.4.2.5 Infrastructuur

Tenslotte, in Figuur 6 tonen we de *infrastructuur* van het netwerk. Hierin staan de actanten gecategoriseerd in hardware (blauw), mensen (groen) en software. De software splitsen we op in “online” software (oranje) en “offline” software (rood) waarbij de noodzaak aan het al dan niet verbonden zijn met internet het criterium vormt. Software bevat immers altijd algoritmes zoals in het eerste hoofdstuk werd aangetoond.

De eerste categorie, **hardware**, bevat 21 actanten zoals onder andere Toetsenbord, Laptop, Cd-speler etc. De actanten die tot deze categorie behoren, spreiden zich over iedere regio in het netwerk. De hardware is met andere woorden niet gebonden aan een bepaalde regio. Opvallend is wel dat de regio van hoorbaar maken bijna uitsluitend bestaat uit hardware¹⁷. De tweede categorie, **mensen**, bestaat uit de leerkracht en de leerling (die voor iedereen en niemand in het bijzonder staat). Hoewel er slechts twee actanten zijn, toont het toch iets over de infrastructuur. De leerkracht staat immers centraler in de topologische kaart dan de leerling (niettemin staande dat de leerling ook een relatief centrale positie

¹⁷ Dit dient wel genuanceerd te worden aangezien, bijvoorbeeld, een laptop en cd niet enkel en alleen maar hardware zijn maar ook software bevatten. Deze nuance moet gelegd worden aangezien dit een gevolg is van het netwerk ‘af te snijden’ op een bepaald moment (de data op de cd kan immers ook als actant opgevat worden).

inneemt). Dit betekent dat de leerkracht als spil dient om de praktijk te maken aangezien deze een actant is die in iedere regio voorkomt. Daarnaast positioneren deze actanten zich ongeveer tussen de regio's beheersbaar en raadpleegbaar maken (waarin de categorie "online" software een voorname rol speelt) en de regio's presenterbaar, hoorbaar en hanteerbaar maken (waarin de categorie "online" software zo goed als geen rol speelt). De derde categorie, "**online**" software, is bijna uitsluitend terug te vinden in de regio's beheersbaar maken en raadpleegbaar maken. Bovendien bestaan deze twee regio's ook hoofdzakelijk uit "online" software-actanten (31 actanten) zoals Agenda, Kweetet, Safari, Wikipedia... Of nog, in tegenstelling tot bijvoorbeeld hardware-actanten die zich spreiden over de verschillende regio's, zijn de "online" software-actanten regionaal gebonden. De redenering kan ook omgedraaid worden, namelijk, om iets presenterbaar te maken zijn in dit geval amper "online" software-actanten nodig. De laatste categorie, "**offline**" software, bevat 14 actanten zoals bijvoorbeeld Windows 10, MS PowerPoint, Bedieningspaneel, Toetsenbord (tablet)... Deze categorie van actanten spreidt zich over het volledige netwerk wat betekent dat ze in iedere regio nodig zijn om die deelpraktijk te maken.



Figuur 6. Sociodigitaal assemblage: infrastructuur

2.4.3 Werking van de klaspraktijk

In de vorige paragraaf werden de vijf operaties van de klaspraktijk gerapporteerd (zie Tabel 1). Deze operaties zijn bepalend voor de werking van een regio. In deze laatste paragraaf van de resultaten,

worden de mechanismen in de klaspraktijk gerapporteerd. De mechanismen zijn de effecten van het 'samenwerken' van verschillende operaties waaruit de klaspraktijk ontstaat. Dit betekent dat de mechanismen beschrijven welke werking mogelijk wordt gemaakt in de klaspraktijk.

In de visualisatie en het samenspel van de clusters en hun operaties zijn er drie mechanismen op te sporen die de werking van de klaspraktijk mogelijk maken. Het eerste mechanisme is **visualisering**. Dit ontstaat uit het samenspel van de operaties beheersbaar maken, raadpleegbaar maken en presenterbaar maken. Dit betekent dat zowel in het beheersbaar maken als in het raadpleegbaar maken als in het presenterbaar maken de actanten (bv. Google Afbeeldingen, Google Maps, Laptop, Safari Webpagina ...) dienen om iets te visualiseren. Het mechanisme verwijst telkens naar het zichtbaar maken van iets. Dit kan gaan om afbeeldingen, het resultaat van de Group Maker, MS Powerpoint... In dit geval gaat het telkens om iets dat digitaal zichtbaar gemaakt wordt. Het tweede mechanisme is **verbalisering**. Dit mechanisme ontstaat uit het samenspel van alle operaties in de klaspraktijk. Hiermee wordt er verwezen naar de mondelinge toelichting bij zowel tekst als afbeeldingen. Een archetypische actant hiervan is de leerkracht, maar uiteraard is het mechanisme niet tot één enkele actant te reduceren. Het is immers vanuit de operaties dat het mechanisme ontstaat. Het derde mechanisme is **verbinding**¹⁸. Dit ontstaat uit de operaties beheersbaar maken en raadpleegbaar maken. Dit betekent dat om iets beheersbaar te maken en iets raadpleegbaar te maken dat het telkens ging om een zich verbinden tot iets. Het mechanisme verwijst naar een connectie die gemaakt wordt met 'het internet' om de klaspraktijk te maken. De verbinding ontstaat niet enkel uit de actant internet, maar het gaat ook om de verbinding tussen bijvoorbeeld Leeret en Kweetet, iOmniWize en de agenda etc.

In bijna iedere regio (uitgezonderd de regio van hoorbaar maken) zijn de drie mechanismen, namelijk visualisering, verbalisering en verbinding, op te sporen. Dit betekent dat de werking van een klaspraktijk ontstaat uit iets tonen (visualisering), iets zeggen (verbalisering) en zich verbinden (verbinding). Of nog, de klaspraktijk bestaat uit beeld, klank en connectie.

2.5 Conclusie

De digitalisering van de wereld weerspiegelt zich vandaag in het onderwijs en in klassen die aan het digitaliseren zijn (bv. Gordon, 2000; John & Wheeler, 2008). Dit betekent dat er een andere vorm van sturing is in klassen. Vandaar dat dit verkennend onderzoek een antwoord zocht op de onderzoeksvraag: *welke vormen van sturing ontstaan er in een klaspraktijk onder invloed van de digitalisering?* Om zicht te krijgen op deze vorm van sturing wordt de klaspraktijk sociomaterieel benaderd (gebaseerd op Actor Netwerk Theorie van Latour (2005)). De klaspraktijk bestaat volgens

¹⁸ Dit mechanisme verwijst niet per se naar de actant internet.

deze benadering zowel uit mensen (het sociale) als dingen (het materiële) waarbij die dingen, in dit onderzoek, enkel het digitale omvatten. Daarnaast wordt de klaspraktijk gevisualiseerd aan de hand van een topologische kaart. Hierdoor kon een visuele netwerkanalyse (Venturini, Jacomy & Pereira, 2015) uitgevoerd worden waarbij we de compositie van het netwerk bespraken aan de hand van vijf dimensies (regio's, centra, densiteit, interfaces en infrastructuur) (Decuypere & Simons, 2014a) om vervolgens de operaties en mechanismen toe te lichten die deze praktijk maken (Decuypere & Simons, 2014b).

De resultaten tonen aan dat er vijf regio's zijn waarin we telkens een operatie kunnen benoemen. De gevonden operaties zijn: hoorbaar maken, hanteerbaar maken, raadpleegbaar maken, beheersbaar maken en presenteerbaar maken. Deze operaties spelen op elkaar in waardoor er mechanismen ontstaan. In de resultaten komen de volgende mechanismen aan bod: visualisering, verbalisering en verbinding. Dit betekent dat de klaspraktijk vorm krijgt door beeld, tekst en connectie. Concreet betekent dit, bijvoorbeeld, dat het mechanisme visualisering ontstaat uit de operaties beheersbaar maken, raadpleegbaar maken en presenteerbaar maken. Wat er precies in die regio's gebeurt (Firefox openen, Blogs lezen, Google Afbeeldingen bekijken...) kan omschreven worden als een visualiseringsmechanisme. Dit visualiseringsmechanisme wijst op het tonen van iets (Google Afbeeldingen, Google Maps...) waarbij dit mechanisme in andere praktijken ook een rol kan spelen (en dus herkenbaar kan zijn in die andere praktijken) maar hoogstwaarschijnlijk toch niet exact dezelfde actanten en interacties zal bevatten. In dit mechanisme is bovendien de rol van algoritmische actanten ("software") niet te onderkennen aangezien deze het grootste deel de regio's beslaan. Dat wil met andere woorden zeggen dat algoritmes een sturende rol spelen hierin doordat deze een centrale rol spelen (bv. Google Afbeeldingen) en sterk vertegenwoordigd (cf. interface) zijn om iets te visualiseren. Algemeen gesteld, de operaties en mechanismen liggen aan de basis van het maken van klas. Dit betekent dat de klaspraktijk het effect is van deze operaties en mechanismen. De operaties en mechanismen beschrijven met andere woorden vormen van sturing (i.e. doen doen) die aanwezig zijn in een klaspraktijk zoals toegelicht met het mechanisme visualisering.

Algemeen besluit

Samenvatting

Uit de vaststelling dat we steeds meer leven in een digitale wereld, was de opzet van deze masterproef tweeledig.

Het **eerste deel** heeft als doel om een fundamenteel concept van die digitale wereld beter te begrijpen, namelijk: algoritmes. Dit concept geniet de laatste decennia aandacht in de internationale onderzoeksliteratuur van onder andere de interdisciplinaire wetenschap van Science and Technology

Studies (STS). Deze onderzoeksliteratuur is in Vlaamse contreien en in pedagogische middens nog relatief onbekend. Vandaar dat het eerste deel de onderzoeksliteratuur over algoritmes in kaart wou brengen aan de hand van een literatuuronderzoek waarbij de richtinggevende onderzoeksvraag was: 'Wat is de werking van een algoritme?'.

In het eerste hoofdstuk werden er definiëringen van een algoritme opgespoord. Hierin blijkt dat algoritmes een algemeen en veelomvattend concept zijn. De algemene idee is dat algoritmes een procedure zijn om een bepaalde soort input om te zetten naar een bepaalde soort output. Toch varieert deze invulling van een algoritme aangezien een aantal onderzoekers de klemtoon leggen op het gebruik ervan in computers door bijvoorbeeld te stellen dat algoritmes in code of pseudocode geschreven zijn. Het wordt vervolgens nog complexer doordat het algoritme in de geraadpleegde onderzoeksliteratuur opgevat wordt als socio-technisch. Dit betekent dat het algoritme niet louter een technische aangelegenheid is maar bovenal ook een sociale aangelegenheid is aangezien algoritmes noch in het luchtledige gecreëerd worden noch in het luchtledige werken.

In het tweede hoofdstuk worden de kenmerken van algoritmes besproken die terugkomen in de onderzoeksliteratuur. Kenmerkend is dat onder de noemer algoritmes doorgaans verwezen wordt naar een algoritmisch systeem. Het gaat met andere woorden bijna nooit om één algoritme maar om een systeem van algoritmes. Dit betekent dan ook dat algoritmes in kluwen van code geschreven staan en niet eenduidig af te leiden zijn uit die code. Een ander kenmerk is dat algoritmes op de achtergrond dienen te werken zodat de gebruiker zich niet bewust is van hoe het algoritme werkt. Indien de gebruiker bewust is van het algoritme, verliezen de ontwikkelaars van het algoritme bijvoorbeeld een handelsmerk (bv. het PageRank-algoritme ligt aan de basis van het succes van Google) of wordt het mogelijk om het algoritme te *gamen* in iemands voor- of nadeel.

Het derde hoofdstuk rapporteert over de macht van het algoritme. Allereerst werd aangegeven dat deze macht van algoritmes wijst op een shift van een hegemonie naar een post-hegemonie of van een *surveillance* model naar een *capture* model. Vervolgens wordt de macht van algoritmes gesitueerd op drie verschillende 'plaatsen': in het algoritme zelf, door het netwerk van het algoritme of in het discours van het algoritme. Ten slotte worden een aantal algoritmische technieken toegelicht die uitgebreid besproken worden in de onderzoeksliteratuur. De algoritmische technieken beschrijven een grond tussen concreet geïmplementeerde algoritmes en de algemene theorie van een algoritme. De aangehaalde technieken zijn: de *Bayes Classifier*, clusteranalyse en *indexing en ranking*.

In het laatste hoofdstuk van dit deel wordt expliciet ingegaan op de onderzoeksmethodes die gebruikt worden in de onderzoeksliteratuur. De methodes zijn parallel met de situering van macht, namelijk: fenomenologisch, etnografisch en discursief.

Het **tweede deel** rapporteert over een exploratief onderzoek waarbij een specifieke methode werd gebruikt om de sturing van algoritmes in een klaspraktijk op te sporen. De methode is gebaseerd op Decuyper en Simons (2014a, 2014b) waarin de Actor Netwerk Theorie als benadering wordt gekozen en waarin gebruik gemaakt wordt van een visualisatie (namelijk topologische kaart) om de data te analyseren. Het visualiseren van de data liet toe om de compositie van het netwerk te analyseren volgens vijf dimensies. De eerste dimensie is regio's, dit zijn de clusters van actanten die ontstaan in een klaspraktijk als gevolg van hun interacties met elkaar. De regio's verwijzen op die manier naar bepaalde actanten en interacties die deelpraktijken van de klaspraktijk vormen. De tweede dimensie is centra. Globale centra zoals Leerkracht of Google Afbeeldingen gaan met veel verschillende andere actanten interacties aan en staan globaal in het midden van het netwerk. Lokale centra zoals Leerling of iOS verwijzen naar actanten die in een bepaalde regio in het centrum staan. De derde dimensie is de dichtheid en toont hoe dicht een bepaalde regio is. Een regio met een lage dichtheid is bijvoorbeeld de regio van hoorbaar maken. Deze regio bestaat uit actanten die onderling weinig interactie aangaan met elkaar en dus een duidelijke functie hebben. De vierde dimensie is interfaces. Interfaces bestaan uit actanten zoals Leerkracht of Desktop PC die grenzen tussen twee of meerdere regio's. Dat is het gevolg van het feit dat de actanten in meerdere regio's nodig zijn om de regio te maken. De vijfde dimensie toont de infrastructuur van de klaspraktijk. De infrastructuur bestaat uit hardware, mensen en software waarbij dit laatste opgesplitst wordt in 'online' en 'offline' software. De hardware en offline software spreiden zich over heel de topologische kaart en de online software hoofdzakelijk in de regio van beheersbaar maken en raadpleegbaar maken. Naast de vijf dimensies, analyseerden we ook de operaties en mechanismen die de regio's maken in het netwerk. De operaties en mechanismen beschrijven vormen van sturing in de klaspraktijk. De vijf gevonden operaties zijn presenteerbaar maken, beheersbaar maken, raadpleegbaar maken, hanteerbaar maken en hoorbaar maken. Iedere regio is het effect van de operatie die in deze regio gebeurt. De drie gevonden mechanismen zijn visualisering, verbalisering en verbinding. De mechanismen zijn het gevolg van de operaties die inspelen op elkaar. De drie mechanismen verwijzen ernaar dat deze klaspraktijk bestaat uit beeld, klank en verbinding.

Discussie

Het empirisch onderzoek rapporteert hoofdzakelijk over het digitale in plaats van over algoritmes. Dit is vanuit de vaststelling dat algoritmes op zich niet aanwezig waren in het netwerk. Ze maken wel een onderdeel uit van een aantal actanten maar deze zijn niet zuiver een algoritme. Google Afbeeldingen bijvoorbeeld is een actant die ongetwijfeld typisch is voor een algoritme, maar het is niet enkel en alleen maar een algoritme, het is immers meer dan dat. De algoritmische techniek zelf wordt op deze manier niet geobserveerd. Nochtans blijkt in de resultaten van het eerste deel dat algoritmes als actant

opgevat kunnen worden in een netwerk (bv. Introna, 2016; Neyland & Möllers, 2017). Het algoritme zelf als actant beschrijven om die visueel weer te geven bleek in de opzet van dit onderzoek niet realiseerbaar. Dit sluit niet uit dat het analyseren van de macht van algoritmes in een netwerk met digitale actanten niet mogelijk is, zoals het tweede hoofdstuk bevestigt. De macht analyseren vraagt immers om een analyse van onderlinge relaties (Foucault, 2004). Het analyseren van relaties stond centraal in het tweede hoofdstuk waarbij die relaties visueel werden weergegeven. Het visualiseren van algoritmes op deze manier is, naar mijn weten, nog niet eerder gebeurd.

Een andere boeiende vaststelling is de sturing door algoritmes in dit onderzoek. In het tweede deel van de masterproef werd telkens de werking van de algoritmes en de algoritmische technieken summier toegelicht. Dit beperkte zich tot een algemene beschrijving van de werking waarbij aangesloten wordt bij Rieder (2017) die het belang van onderzoek naar deze algoritmische technieken benadrukt zoals in het eerste hoofdstuk aan bod kwam. Het is dan ook interessant om even stil te staan bij hoe dit onderzoek gestuurd werd door algoritmes. Een volledig overzicht geven van de volledige sturing door het algoritme in het volledige onderzoeksproces valt hier buiten het bestek. Er wordt gekozen om drie algoritmes en/of algoritmische technieken toegelicht die dit onderzoek in sterke mate hebben gestuurd. Een eerste algoritmische techniek is de clusteranalyse waarvan de werking reeds eerder uitgelegd werd. Doordat deze algoritmische techniek clusters vormt, was het in dit onderzoek mogelijk om bijvoorbeeld regio's en daarin operaties te benoemen. De algoritmische techniek doet dit onderzoek een bepaalde richting uitgaan, namelijk het opsporen van regio's, operaties, centra... Zonder deze algoritmische techniek zou dit onderzoek niet mogelijk geweest zijn. Een ander algoritme zit in de functie *Modularity Class* waarmee gemeenschappen van actanten opgespoord worden in een netwerk. Deze functie lag aan de basis voor het inkleuren van de regio's. Door gebruik te maken van het algoritme, worden gemeenschappen getoond die anders misschien niet als gemeenschap herkend zouden worden (bv. de regio van hanteerbaar maken). Opnieuw stuurt het algoritme het onderzoek in een bepaalde richting. Een derde algoritme is PageRank. Dit algoritme is niet specifiek gebonden aan dit onderzoek, maar aan alle(?) onderzoek. PageRank is namelijk de naam van het Google-algoritme. De zoekresultaten bestonden namelijk telkens uit onderzoeken die aansloten bij mijn initiële speurtocht naar algoritmes vanuit STS. PageRank vormde zo een *filter bubble* waardoor de literatuur in het eerste hoofdstuk mogelijks een eenzijdig beeld weergeeft.

Reflectie

De masterproef is uiteraard niet feilloos en bevat een aantal zwaktes. Het eerste deel rapporteert over een selecte groep van onderzoeken die zich allen in hetzelfde vakgebied situeren. Door reeds in het begin van de masterproef algoritmes in dat vakgebied te situeren, is het mogelijk dat ander relevant werk door de mazen van het net geglipt zijn. Dat impliceert dus dat er een eenzijdig beeld uit de

literatuur over algoritmes weergegeven wordt. Een andere zwakte is dat de resultaten in het tweede deel hoofdzakelijk gebaseerd zijn op de visualisatie van het netwerk. In de visualisatie heeft iedere interactie tussen twee actanten hetzelfde gewicht gekregen waardoor een vertekend beeld gegeven wordt van de feitelijke klaspraktijk. Een aantal interacties komen immers veel vaker voor (bv. *swipen*) dan andere (bv. gaan halen). Tot slot, is het niet evident om de actanten te bepalen tijdens de registratie aangezien alles en iedereen een actant kan zijn. Bijvoorbeeld is de website Kweetet een actant of de tabbladen die daarin beschikbaar zijn? Hetzelfde geldt, bijvoorbeeld, voor een slideshow. Is de slideshow de actant of is het de dia of is het de afbeelding op de dia? Deze keuze weerspiegelt zich in de visualisatie van het netwerk en heeft gevolgen voor de analyse van het netwerk.

Tot slot zijn er veel mogelijke pistes voor vervolgonderzoek. Een aantal interessante pistes voor vervolgonderzoeken die in deze masterproef zijn opgedoken worden in deze laatste paragraaf beschreven. Een eerste piste voor vervolgonderzoek lijkt mij om meer eenduidigheid te creëren over wat precies bedoeld wordt met een algoritme. In de literatuur circuleren immers verschillende opvattingen over algoritmes en de meeste onderzoeken spreken vaak enkel in algemene termen van een algoritme. Maar zoals uit de resultaten van het eerste hoofdstuk blijkt, is het concept 'algoritme' een heel breed concept. Bijgevolg zou het vruchtbaar zijn om onderzoek naar de verschillende soorten algoritmes te doen (bv. *divide and conquer*-algoritmes, *recursive* algoritmes, *brute force* algoritmes...). Om dit soort typologie te ontwikkelen kan inspiratie gehaald worden in de computerwetenschappen. Een tweede interessante piste voor vervolgonderzoek is een discoursanalyse van het algoritme zelf of een analyse van het discours dat mogelijk wordt door het algoritme. Hier wordt immers een paar keer naar verwezen in de literatuur maar daar is nog weinig onderzoek naar gedaan. Een derde piste voor vervolgonderzoek is methodologisch van aard waarbij nagegaan wordt wat de precieze implicaties zijn van het aantal keer een interactie voorkomt in de visualisatie en de daaruit volgende interpretatie. Ten slotte, de vierde piste voor vervolgonderzoek is om de sturing door het digitale na te gaan in bijvoorbeeld een observatie, interview of discoursanalyse. Hierin gaat de sturing dan niet om operaties en mechanismen, maar om hoe bijvoorbeeld de leerkracht of leerling zich beweegt door en positioneert in de klas (bijvoorbeeld voornamelijk aan de laptop zitten in plaats van vooraan staan aan het bord), welke handelingen de leerkracht uitvoert (bijvoorbeeld *swipen* op het bord om een woord te tonen in plaats van het zelf vooraan op te schrijven), hoe de leerkracht zijn taak opvat (bijvoorbeeld als begeleider van leerplatforms in plaats van als onderwijzer van een vak) etc. De status quo die teruggevonden werd in het onderzoeksveld duidt op veel boeiende wegen die verder bewandeld kunnen worden voor vervolgonderzoek naar algoritmes.

Referenties

- Ashby, R. (1956). *An introduction to cybernetics*. New York: John Wiley & Sons Inc. Opgehaald van <https://archive.org/stream/introductionto00ashb#page/n7/mode/1up>
- Bauer, F. L. (2010). *Origins and Foundations of Computing*. Heidelberg: Springer. Opgehaald van <https://link.springer.com/content/pdf/10.1007%2F978-3-642-02992-9.pdf>
- Beer, D. (2009). Power through the algorithm? Participatory web cultures and the technological unconscious. *New Media & Society*, 11(6), 985-1002. doi:10.1177/14614448093336551
- Beer, D. (Ed.). (2017a). The Social Power of Algorithms [Special issue]. *Information, Communication & Society*, 20(1).
- Beer, D. (2017b). The social power of algorithms. *Information, Communication & Society*, 20(1), 1-13. doi:10.1080/1639118X.2016.1216147
- Blondel, V.D., Guillaume, J.-L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 1-12. doi:10.1088/1742-5468/2008/10/P10008
- Breslow, L., Pritchard, D. E., DeBoer, J., Stump, G., Ho, A. D., & Seaton, D. T. (2013). Studying Learning in the Worldwide Classroom: Research into edX's First MOOC. *Research & Practice in Assessment*, 8, 13-25. Opgehaald van <http://www.rpajournal.com/dev/wp-content/uploads/2013/05/SF2.pdf>
- Bucher, T. (2012). Want to be on the top? Algorithmic power and the threat of invisibility on Facebook. *New Media & Society*, 14(7), 1164-1180. doi:10.1177/1461444812440159
- Bucher, T. (2016). Neither Black Nor Box: Ways of Knowing Algorithms. In S. Kubitschko, & A. Kaun (Eds.), *Innovative Methods in Media and Communication Research* (pp. 81-98). Basingstoke: Palgrave Macmillan.
- Bucher, T. (2017). The algorithmic imaginary: exploring the ordinary affects of Facebook algorithms. *Information, Communication & Society*, 20(1), 30-44. doi:10.1080/1369118X.2016.1154086
- Buss, S.R., Kechris, A.S., Pillay, A., & Shore, R.A. (2001). The prospects for mathematical logic in the twenty-first century. *The Bulletin of Symbolic Logic*, 7(2), 169-196.

- Cegłowski, M. (2016, juni). *The moral economy of Tech*. Paper gepresenteerd op SASE conferentie, Berkeley. Geraadpleegd via http://idlewords.com/talks/sase_panel.htm
- Ceulemans, C. (2015). *Het beroepsprofiel van de leraar als 'black box': Over de werking van onderwijsstandaarden*. Antwerpen: Garant.
- Cheney-Lippold, J. (2011). New algorithmic identity: soft biopolitics and the modulation of control. *Theory, Culture and Society*, 28(6), 164-181. doi:10.1177/0263276411424420
- Chistian, B., & Griffiths, T. (2016). *Algorithms to Live By*. New York: Henry Holt and Co.
- Cormen, T.H., Leiserson, C.E., Rivest, R.L., & Stein, C. (2009). *Introduction to Algorithms*. Cambridge: MIT Press, 3^e editie.
- Crawford, K. (2016). Can an Algorithm be Agonistic? Ten scenes from Life in Calculated Publics. *Science, Technology & Human Values*, 41(1), 77-92. doi:10.1177/0162243915589635
- Crowstone, K. (1994). Electronic communication and new organizational forms: a coordination theory approach, *Working Paper Series*, 175, MIT Center for Coordination Science.
- Dasgupta, S., Papadimitriou, C.H., & Vazirani, U.V. (2006). *Algorithms*. Ohio: McGraw-Hill Education.
- de la Cruz Paragas, F., & Lin, T. T. (2016). Organizing and reframing technological determinism. *New Media & Society*, 18(8), 1528-1546. doi:10.1177/1461444814562156
- Decuyper, M., & Simons, M. (2014). An Atlas of Academic Practice in Digital Times. *Open Review of Educational Research*, 1(1), 116-143. doi:10.1080/23265507.2014.973899
- Decuyper, M., & Simons, M. (2014). On the composition of academic work in digital times. *European Educational Research Journal*, 13(1), 89-106. doi:10.2304/eej.2014.13.1.89
- Decuyper, M., & Simons, M. (2016). On the critical potential of sociomaterial approaches in education. *Teoría de la Educación. Revista Interuniversitaria*, 28(1), 25-44. doi:10.14201/teoredu201628125
- Diakopoulos, N. (2015). Algorithmic Accountability. *Digital Journalism*, 3(3), 398-415. doi:10.1080/21670811.2014.976411
- Fioravanti, C., & Velho, L. (2010). Let's follow the actors! Does Actor-Network Theory have anything to contribute to science journalism? *Journal of Science Communication*, 9(4), 1-8.

- Foote, B., & Yoder, J. (1997 september). *Big Ball of Mud*. Paper gepresenteerd op the Fourth Conference on Patterns Languages of Programs, Illinois. Geraadpleegd via <http://www.laputan.org/pub/foote/mud.pdf>
- Foucault, M. (2004). *De geboorte van de biopolitiek*. Amsterdam: Boom.
- Gillespie, T. (2014). The relevance of Algorithms. In T. Gillespie, P. Boczkowski & K. Foot (Eds.), *Media Technologies* (pp. 167-194). Cambridge: MIT Press.
- Gillespie, T. (2017). Algorithmically recognizable: Santorum's Google problem, and Google's Santorum problem. *Information, Communication & Society*, 20(1), 63-80.
doi:10.1080/1369118X.2016.1199721
- Gillespie, T. & Seaver, N. (2016). Critical Algorithm Studies: a Reading List [Visual Timeline]. Afgehaald van <https://socialmediacollective.org/reading-lists/critical-algorithm-studies/>
- Goffey, A. (2008). Algorithm In M. Fuller (Ed.), *Software studies\ a lexicon* (pp. 15-20). Cambridge: The MIT Press.
- Gordon, D. (2000). *The Digital Classroom: How Technology is changing the way we teach and learn*. Cambridge: Harvard Education Pr.
- Graham, S. (2005). Software-sorted Geographies. *Progress in Human Geography*, 29(5), 562-580.
doi:10.1191/0309132505ph568oa
- Harel, D., & Feldman, Y. (1987). *Algorithmics: The Spirit of Computing*. Berlin: Springer
- Hazewinkel, M. (1990). Algorithm. In *Encyclopaedia of Mathematics* (Vol.5, pp. 131-152).
- Henderson, H. (2009). Algorithm. In *Encyclopedia of Computer Science and Technology* (pp. 7-8).
- Introna, L. (2016). Algorithms, Governance, and Governmentality: On Governing Academic Writing. *Science, Technology, & Human Values*, 41(1), 17-49. doi:10.1177/0162243915587360
- Jacomy, M., Venturini, T., Heymann, S., & Bastian, M. (2014). ForceAtlas2, a continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software. *PLoS ONE*, 9(6).
doi:10.1371/0098679
- John, P.D., & Wheeler, S. (2008). *The Digital Classroom: Harnessing Technology for the Future of Learning and Teaching*. Michigan: Routledge.

- Kitchin, R. (2017). Thinking critically about and researching algorithms. *Information, Communication & Society*, 20(1), 14-29. doi:10.1080/1369118X.2016.1154087
- Kitchin, R., & Dodge, M. (2011). *Code/space: Software and Everyday Life*. Cambridge, Massachusetts: The MIT Press.
- Lash, S. (2007). Power after Hegemony: Cultural Studies in Mutation? *Theory, Culture & Society*, 24(3), 55-78. doi:10.1177/0263276407075956
- Latour, B. (1987). *Science in Action*. Cambridge, Massachusetts: Harvard University Press.
- Latour, B. (2005). *Reassembling the social: An introduction to Actor-Network-Theory*. Oxford: Oxford University Press.
- Lupton, D. (2016). The diverse domains of quantified selves: Self-tracking modes and dataveillance. *Economy and Society*, 45(1), 101-122. doi:10.1080/03085147.2016.1143726
- MacCormick, J. (2013). *Nine Algorithms That Changed the Future: The ingenious idea that drive today's computers*. Princeton: Princeton University Press.
- MacKenzie, A. (2006). *Cutting Code: Software and Sociality*. New York: Peter Lang.
- Moschovakis, Y.N. (2001). What is an algorithm? In B. Engquist, & W. Schmid, (Eds.), *Mathematics Unlimited*. Verlag: Springer (pp. 919-936).
- Neyland, D. (2016). Bearing Account-able Witness to the Ethical Algorithmic System. *Science, Technology, & Human Values*, 41(1), 50-76. doi:10.1177/0162243915598056
- Neyland, D., & Möllers, N. (2017). Algorithmic IF ... THEN rules and the conditions and consequences of power. *Information, Communication & Society*, 20(1), 45-62. doi:10.1080/1369118X.2016.1156141
- O'Neil, C. (2016). *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*, New York: Crown.
- Pariser, E. (2011). *The Filter Bubble: What the Internet is Hiding from You*. New York: Penguin Press.
- Pasquale, F. (2015). *The Black Box Society*. Cambridge, Massachusetts: Harvard University Press.

- Perrotta, C., & Williamson, B. (2018). The social life of Learning Analytics: cluster analysis and the 'performance' of algorithmic education. *Learning, Media and Technology*, 43(1), 3-16. doi:10.1080/17439884.2016.1182927
- Rieder, B. (2017). Scrutinizing and algorithmic technique: the Bayes classifier as interested reading of reality. *Information, Communication & Society*, 20(1), 100-117. doi:10.1080/1369118X.2016.11811995
- Ruming, K. (2009). Following the Actors: mobilising an actor-network theory methodology in geography. *Australian Geographer*, 40(4), 451-469. doi:10.1080/00049180903312653
- Salavati, S. (2016). *Use of Digital Technologies in Education: The complexity of Teachers' Everyday Practice* (Doctoral dissertation). Afgehaald van <https://www.diva-portal.org/smash/get/diva2:1039657/FULLTEXT01.pdf>
- Seaver, N. (2013). *Knowing Algorithms*. Media in Transition 8, Cambridge. Afgehaald van <https://static1.squarespace.com/static/55eb004ee4b0518639d59d9b/t/55ece1bfe4b030b2e8302e1e/1441587647177/seaverMIT8.pdf>
- Serres, M. (2014). *De wereld onder de duim: leve de internetgeneratie!*. Amsterdam: Boom.
- Sørensen, E. (2009). *The Materiality of Learning: Technology and Knowledge in Educational Practice*. Cambridge: Cambridge University Press.
- Souto-Otero, M., & Beneito-Montagut, R. (2016). From governing through data to governmentality through data: Artefacts, strategies and the digital turn. *European Educational Research Journal*, 15(1), 14-33. doi:10.1177/1474904115617768
- Strathern, M. (1996). Cutting the Network. *The Journal of the Royal Anthropological Institute*, 2(3), 517-535. doi:10.2307/3034901
- Tondeur, J., De Bruyne, E., Van Den Driessche, M., McKenney, S., & Zandvliet, D. (2015). The physical placement of classroom technology and its influences on educational practices. *Cambridge Journal of Education*, 45(4), 537-556. doi:10.1080/0305764X.2014.998624
- Tufekci, Z. (2015). Algorithmic harms beyond Facebook and Google: Emergent challenges of computational agency. *Journal on Telecommunications and High Technology Law*, 13, pp. 203-218.

- Turing, A.M. (1950). Computing Machinery and Intelligence. *Mind*, 49, 433-460.
- Valenti, C. (2000). Preparing Teachers for the High-Tech Classroom. In D. Gordon (Ed.), *The Digital Classroom: how technology is changing the way we teach and learn* (pp. 78-87). Cambridge: Harvard Education Letter.
- Vanden Buerie, L., & Simons, M. (2017). School stuff: a pedagogical regime of enunciation? *Pedagogy, Culture & Society*, 25(1), 105-119. doi:10.1080/14681366.2016.1229689
- Venturini, T., Jacomy, M., & Pereira, D. (2015). *Visual Network Analysis* [Working Paper]. 1-20. Opgehaald van http://www.tommasoventurini.it/wp/wp-content/uploads/2014/08/Venturini-Jacomy_Visual-Network-Analysis_WorkingPaper.pdf
- Williamson, B. (2014). Algorithmic skin: health-tracking technologies, personal analytics and the biopedagogies of digitized health and physical education. *Sport, Education and Society*, 20(1), 133-151. doi:10.1080/13573322.2014.962494
- Williamson, B. (2015). Governing software: networks, databases and algorithmic power in the digital governance of public education. *Learning, Media and Technology*, 40(1), 83-105. doi:10.1080/17439884.2014.924527
- Williamson, B. (2016). Digital education governance: an introduction. *European Educational Research Journal*, 15(1), 3-13. doi:10.1177/1474904115616630
- Williamson, B. (2017a). *Big Data in Education: The digital future of learning, policy and practice*. London: Sage Publications Ltd.
- Williamson, B. (2017b). Computing brains: learning algorithms and neurocomputation in the smart city. *Information, Communication & Society*, 20(1), 81-99. doi:10.1080/1369118x.2016.1181194
- Willson, M. (2017). Algorithms (and the) every day. *Information, Communication & Society*, 20(1), 137-150. doi:10.1080/1369118X.2016.1200645
- Yeung, K. (2017). 'Hypernudge': Big Data as a mode of regulation by design. *Information, Communication & Society*, 20(1), 118-136. doi:10.1080/1369118X.2016.1186713
- Ziewitz, M. (Ed). (2016a). Governing Algorithms [Special issue]. *Science, Technology, & Human Values*, 41(1).

Ziewitz, M. (2016b). Governing Algorithms: Myth, Mess, and Methods. *Science, Technology, & Human Values*, 41(1), 3-16. doi:10.1177/0162243915608948

Bijlages

Bijlage 1: Observatieformulier

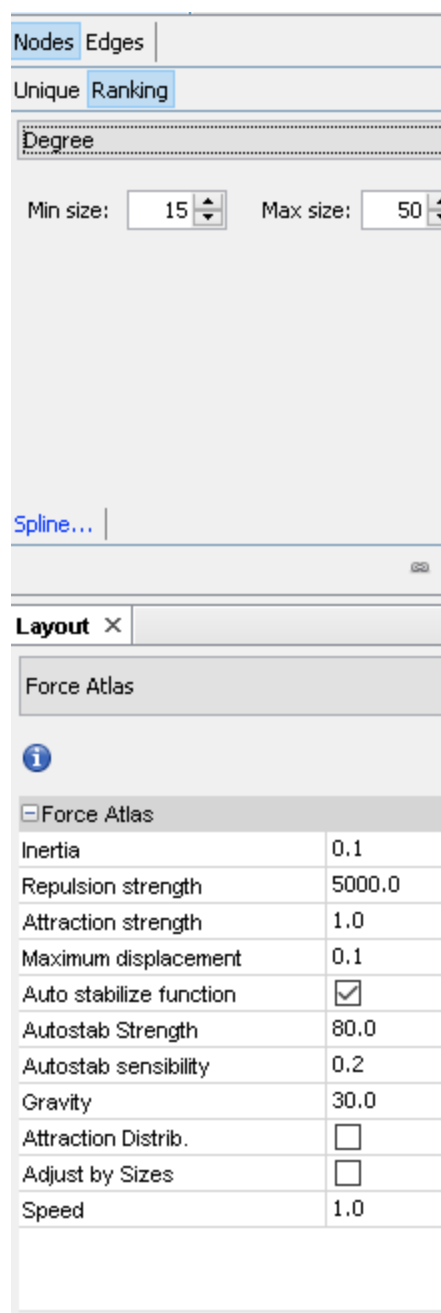
(sociale) actor	Interactie	(digitale) actor (hardware + software [programma + functies])
<i>Bv.</i> <i>Leerkracht</i>	<i>Tekenen, naar bord kijken, mondeling uitleg geven</i>	<i>SmartBoard, tool voor figuren tekenen.</i>
<i>Leerling</i>	<i>Feedback vragen</i>	<i>Knop/functie feedback vragen</i>

Bijlage 2: Voorbeelden interviewvragen

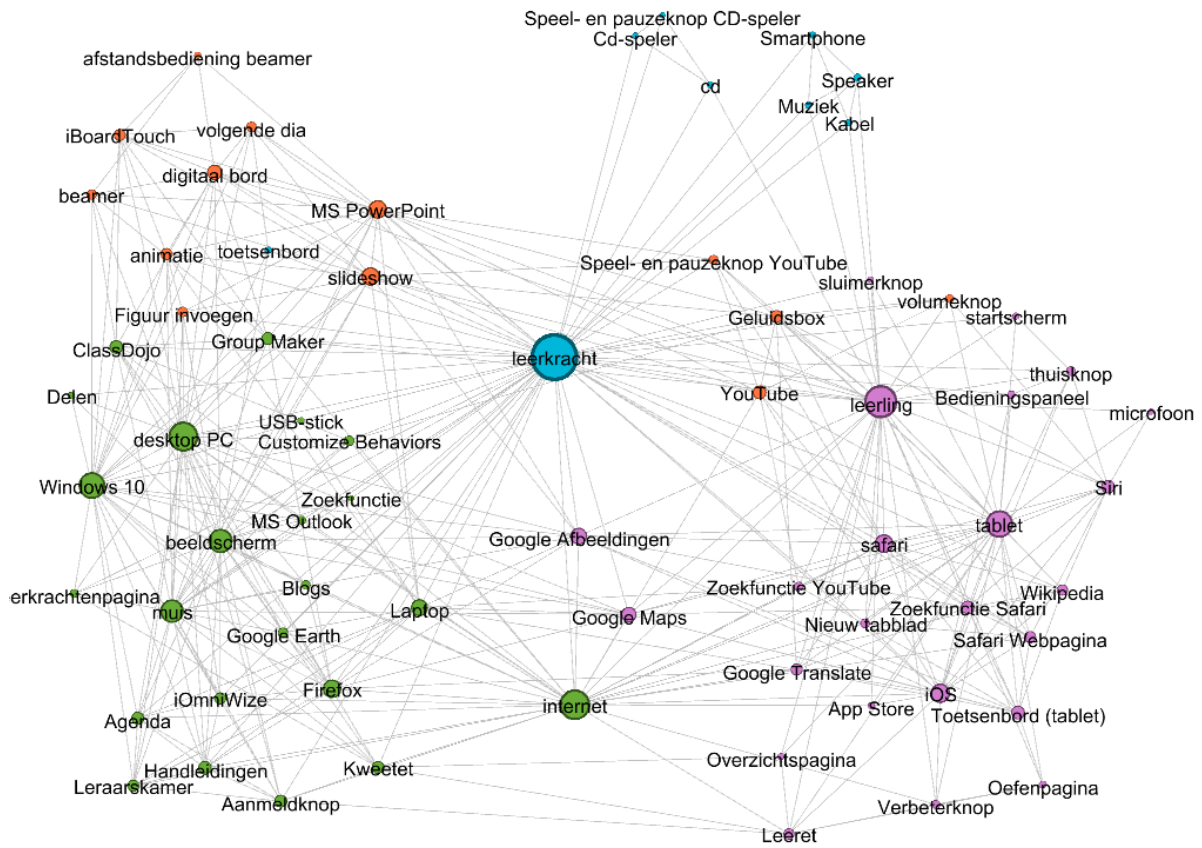
- “Wat heb je gedaan om deze les te kunnen geven?” *bordschema opgesteld*
- “Hoe heb je je bordschema gemaakt?” *Computer, muis, toetsenbord, PowerPoint, thema/format van PowerPoint, ...*
- “Waarvoor diende dat?” *lets opzoeken via Google (= actor), typen, bekijken agenda vorige lessen, ...*

Bijlage 3: Overzicht instellingen en parameters Gephi

Parameter	Uitleg
Nodes Ranking Degree	De minimum en maximum puntgrootte van een knooppunt
Force Atlas	Een algoritme dat actoren in clusters plaatst op basis van de relatieve intensiteit van hun interacties met andere actoren
Inertia	Snelheid/traagheid waarmee knooppunten zich plaatsen
Repulsion strength	Hoe sterk stoten de knooppunten elkaar af
Attraction strength	Hoe sterk trekken paren van knooppunten elkaar aan
Maximum displacement	Hoe ver kan een knooppunt maximum staan (vermijden van knooppunten die heel ver liggen)
Auto stabilize function	Stabiliseren van knooppunten die zouden blijven bewegen
Autostab Strength	Kracht van de auto stabilize functie
Autostab sensibility	Snelheid/traagheid waarmee knooppunten zich stabiliseren
Gravity	Kracht die alle knooppunten naar een centrum trekt
Attraction distribution	De aantrekkende kracht is verdeeld over de buitenkanten. Het duwt bepaalde middelpunten naar buiten en plaatst autoriteiten centraal
Adjust by sizes	Vermijdt dat knooppunten overlappen
Speed	Snelheid van convergentie (naar elkaar toegroeien) ten koste van precisie



Bijlage 4: Resultaat Modularity Class¹⁹



¹⁹ <https://github.com/gephi/gephi/wiki/Modularity>

