



Bachelor Thesis, GreenOrange BV.

INHOUD

Procesautomatisatie door middel van object detectie, een machine-learning ontwikkelingsmethodologie.

Stef Huttinga

S1587528

7-8-2019

Voorwoord

Voor u ligt de scriptie 'Procesautomatisatie door middel van object detectie, een machine-learning ontwikkelingsmethodologie'. Deze scriptie is geschreven ter afronding van mijn bacheloropleiding Technische Bedrijfskunde aan de Universiteit Twente en in opdracht van de organisatie GreenOrange. Daarnaast dient de scriptie als bewijsstuk voor mijn werkzaamheden en functioneren gedurende mijn afstudeerperiode bij de organisatie. In de periode van september 2019 tot en met juli 2020 heb ik onderzoek mogen doen naar de procesautomatisatie, met behulp van machine-learning componenten, voor een service binnen de organisatie.

Afgelopen jaar is een bewogen jaar geweest, waarin ik binnen mijn privésfeer veel heftige situaties heb meegemaakt die mijn afstuderen hebben verzwaard, vermoeilijkt en daardoor helaas ook vertraagd. Ondanks de lastige periodes dit jaar heb ik ook veel mooie momenten gehad en mogen delen met mijn vrienden en familie.

Via deze weg wil ik de medewerkers van GreenOrange bedanken voor de prettige werksfeer op kantoor en de interesse in mijn onderzoek. In het bijzonder wil ik graag Marc Lohuis (begeleider GreenOrange) bedanken voor zijn tijd, begeleiding en bovenal zijn begrip. Je bood mij de ruimte, tijd en vertrouwen die ik, zeker dit jaar, nodig had. Naast Marc, wil ik graag Adina Aldea (begeleider UniversiteitTwente) bedanken voor haar waardevolle inhoudelijke begeleiding, tijd een en bovenal aanpassingsvermogen aan mijn planning. Jij bood mij de rust en tijd die ik nodig had om dit onderzoek tot een succes te maken.

Ik hoop dat mijn onderzoek en bijgeleverde toepassingen door GreenOrange omarmd worden en de implementatie van mijn aanbevelingen van grote toegevoegde waarde zullen zijn binnen de organisatie.

Stef Huttinga

Enschede, 06-08-19

Managementsamenvatting

In deze scriptie wordt antwoord gegeven op de hoofdvraag: “Hoe kan een methode, op basis van machine-learning componenten, het ontwikkelingsproces voor softwareautomatisering binnen Green Orange verbeteren?”. Hierdoor staat de ontwikkeling van een methodiek, op basis van machine-learning componenten, centraal. Daarnaast levert het onderzoek een Proof of Concept van de ontwikkelde methodologie in de vorm van een machine-learning softwareapplicatie. In de probleemidentificatie is naar voren gekomen, dat binnen de organisatie een stappenplan ontbreekt ten behoeve van de ontwikkeling van machine-learning software. De belangrijkste aanleiding hiervan is dat het bedrijf machine-learning software wil gebruiken in de automatisering van een bepaald proces (de WePublish-service).

Het onderzoek berust op DesignScience, praktijkonderzoek en maakt gebruik van een kwalitatieve onderzoeksmethode om van de bestaande literatuur de methode- en de applicatie te ontwikkelen. Om antwoord te geven op de hoofdvraag is allereerst de huidige situatie visueel in kaart gebracht en zijn KPI's gescoord. Vervolgens is ook de gewenste situatie visueel in kaart gebracht. Vanuit de kennis, geëxtraheerd uit de literatuur, en de beoogde gewenste situatie is een methodiek ontworpen. Tot slot is de ontworpen methodiek toegepast en gevalideerd op de WePublish-service. De ontwikkeling van de methodiek en de softwareapplicatie zijn tevens de belangrijkste resultaten van dit onderzoek en vormen het antwoord op de hoofdvraag.

De ontwikkelde methode limiteert zich tot de ontwikkeling van object detectie software. De Proof of Concept limiteert zich tot de WePublish-service. Het is niet onderzocht of de methode in vergelijkbare vraagstukken toepasbaar is.

Vervolgonderzoek kan zich bijvoorbeeld richten op de uitbreiding van de huidige methodiek, de ontwikkeling van een alternatieve methodiek of door een validatieonderzoek, toegepast op de ontwikkelde methodiek.

Praktische aanbevelingen voor GreenOrange, ter afronding van dit onderzoek:

- Verzamel op een geautomatiseerde wijze meer gelabelde data voor iedere retailer, zodat nieuwe modellen meer accuraat worden.
- Stel nieuwe doelen voor de ontwikkeling van een minimaal levensvatbaar prototype dat binnen het FMP gaat draaien.
- Stel een autoriteit op het gebied van machine-learning software aan, zodat vragen en problemen snel verholpen worden.
- Neem de machine-learning expertise die GreenOrange nu bezit op in de marketing van het bedrijf voor meer omzet.

Inhoudsopgave

VOORWOORD	2
MANAGEMENTSAMENVATTING	3
INHOUDSOPGAVE	4
ACHTERGRONDINFORMATIE ONDERZOEK EN GREEN ORANGE	7
OVER GREEN ORANGE.....	7
GREEN ORANGE HOLDING	7
1. INTRODUCTIE	9
1.1 PROBLEEMAANLEIDING.....	9
1.1.1. <i>WePublish-service</i>	9
1.1.2. <i>Automatisering van taken medewerker1</i>	11
1.1.2. <i>Waarom automatiseren?</i>	12
1.2 PROBLEEMIDENTIFICATIE (INTRODUCTIE)	13
1.2.1. <i>Probleemkluwen</i>	13
1.2.2. <i>Relevantie van het probleem</i>	16
1.2.3. <i>Doelstelling</i>	18
1.2.4. <i>Hoofd en deelvragen</i>	18
1.2.5. <i>Afbakening onderzoek</i>	21
1.2.6. <i>Norm en realiteit</i>	22
1.3. ONDERZOEK ONTWERP EN VERSLAGOPBOUW	23
1.3.1. <i>Onderzoek ontwerp</i>	23
1.3.2. <i>Verslagopbouw</i>	27
2. THEORETISCH KADER	29
2.1. MACHINE LEARNING	29
2.1.1. <i>Vakgebieden</i>	29
2.1.2. <i>Neurale netwerken</i>	30
2.1.3. <i>Computer vision</i>	34
2.1.4. <i>Limitaties en gevaren van neurale netwerken</i>	37
2.2. METHODEN TER ONTWIKKELING VAN METHODEN	38
2.3. METHODE OMTRENT SOFTWAREONTWIKKELING	39
2.4. BUSINESS PROCESS MODELLING NOTATION.....	41
2.4.1. <i>Modelleermethoden</i>	41
2.4.2. <i>Flowcharts - BPMN</i>	42
2.5. SAMENVATTING EN CONCLUSIES	43
3. HUIDIGE SITUATIE	45
3.1. HET VOLLEDIGE WEPUBLISH PROCES.....	45
3.1.1. Sales – group	47
3.1.2. FMP preparations - group	50
3.1.3. <i>FMP editor – group</i>	52
3.2. NULMETING	54
3.2.1. <i>KPI's</i>	54
3.2.2. <i>Scoren van nulmeting</i>	56
3.3. SAMENVATTING EN CONCLUSIES	60
4. GEWENSTE SITUATIE	62
4.1. IDEALE FMP EDITOR PROCES	63
4.1.1. <i>Training en integratie van het NN</i>	64
4.2. MOGELIJKE WEGEN + VOORDELEN/NADELEN	65
4.2.1. <i>1NN</i>	65
4.2.2. <i>2NN</i>	65
4.3. MINIMAAL LEVENSVATBAAR PRODUCT (MVP)	68

4.3.1 Afwegingen.....	68
4.3.2. Conclusie MVP.....	69
4.3.2 Omschrijving MVP.....	69
4.4. VERWACHTE VERANDERINGEN BIJ IMPLEMENTATIE MVP.....	70
4.5. SAMENVATTING.....	70
5. ONTWERP- EN ONTWIKKELINGSFASE.....	71
5.1. PROBLEEM IDENTIFICATIE (METHODIEK).....	71
5.1.1. <i>Bewerkingen</i>	71
5.2. MODEL ARCHITECTUUR.....	75
5.2.1. <i>Bewerkingen</i>	75
5.3. DATA VERZAMELING.....	77
5.3.1. <i>Bewerkingen</i>	78
5.4. TRAINING.....	79
5.4.1. <i>Bewerkingen</i>	79
5.5. EVALUATIE.....	81
5.5.1. <i>Bewerkingen</i>	81
5.6. IMPLEMENTATIE.....	83
5.6.1. <i>Bewerkingen</i>	83
5.7. SAMENVATTING EN CONCLUSIE.....	84
6. DEMONSTRATIEFASE – PROOF OF CONCEPT.....	85
STAP 1 – PROBLEEM IDENTIFICATIE.....	85
STAP 2 – MODEL ARCHITECTUUR.....	85
STAP 3 – DATAVERZAMELING.....	87
STAP 4 – TRAINING.....	87
STAP 5 – EVALUATIE.....	88
STAP 6 – IMPLEMENTATIE.....	90
CONCLUSIE.....	90
7. EVALUATIE.....	91
7.1. ANALYSE ENQUÊTE.....	91
7.2. CONCLUSIE UIT ENQUÊTE.....	93
8. DISCUSSIE, CONCLUSIE EN AANBEVELINGEN.....	94
8.1 DISCUSSIE EN CONCLUSIE.....	94
8.1.1. <i>Discussie</i>	94
8.1.2. <i>Conclusie</i>	95
8.2. RELEVANTIE VAN HET ONDERZOEK.....	95
8.2.1. <i>Theoretisch bijdrage</i>	96
8.2.2. <i>Praktische bijdrage</i>	96
8.3. AANBEVELINGEN EN SUGGESTIES.....	97
8.3.1. <i>Aanbevelingen</i>	97
8.3.2. <i>Suggesties:</i>	99
8.4. VERVOLGONDERZOEK.....	99
8.5. LIMITATIES VAN HET ONDERZOEK.....	100
8.5.1. <i>BPMN modellen</i>	100
8.5.2. <i>Nulmeting</i>	100
8.5.3. <i>Ontwikkelde methodiek</i>	100
8.5.4. <i>Proof of Concept</i>	100
9. BIBLIOGRAFIE.....	101
BIJLAGE.....	103
BIJLAGE A: BPMN NOTATIE.....	103
BIJLAGE B: TERMEN BINNEN BPMN.....	106
BIJLAGE C: XML VOORBEELD.....	109

Achtergrondinformatie onderzoek en Green Orange

In deze sectie wordt de relevante achtergrondinformatie omtrent Green Orange in kaart gebracht. Ten eerste, de algemene informatie over Green Orange, gevolgd door informatie over de cultuur binnen het bedrijf. Tot slot komen de samenwerkingen binnen de Green Orange Holding aan bod.

Over Green Orange

Green Orange is een digitaal marketingbureau, gespecialiseerd in het ondersteunen, adviseren en implementeren van digitale vraagstukken voor B2B klanten. Door middel van partnerschappen probeert Green Orange synergie tussen mensen, methodologie en technologie te creëren, om op die manier partners op de lange termijn wendbaar en innovatief te houden. Door het integreren van IT-oplossingen, gecombineerd met data-gedreven strategisch advies weet Green Orange haar partners volledig te ontzorgen op het gebied van marketing, naamsbekendheid, softwareontwikkelingen, connectiviteit, integraties, compatibiliteit, business intelligence, rapportages en cloudcomputing. Het hoofdkantoor van Green Orange bevindt zich in Oldenzaal. Naast het hoofdkantoor beschikt het bedrijf over een vestiging in Utrecht en een E-office in Londen.

Cultuur

Green Orange kent rond de 35 medewerkers en zijn op een aantal uitschieters na allemaal rond de 30 jaar oud. Ze zijn voornamelijk HBO opgeleid en kennen een achtergrond in IT of Media Communicatie. Veel medewerkers zijn na een tevreden stageperiode bij Green Orange en afronding van hun opleiding aangesteld als vaste medewerker.

Green Orange kent een unieke werksfeer zonder directe leidinggevenden. Daarnaast genieten medewerkers veel vrijheid in het inrichten van hun werkdagen. Een ochtend vrij, of een middag sporten is eigen verantwoordelijkheid, zolang je de klant niet teleurstelt. Het wordt dan ook van elkaar verwacht dat hier op een professionele manier ingespeeld wordt op de vrijheid die je als medewerker krijgt.

Green Orange Holding

Green Orange is het moederbedrijf van vier spin-offs die momenteel onafhankelijk opereren binnen dezelfde holding. De holding bestaat uit: WePublish, Brandcube, Datatrics. We::code en Green Orange.

Deze spin-offs worden geleid door ondernemende ex-medewerkers van Green Orange en hebben zich als bedrijf gespecialiseerd in afgebakende componenten van digitale vraagstukken. Vanwege de goede persoonlijke band tussen de oprichters van de bedrijven en de financiële samenwerkingsvoordelen zijn de bedrijven onderdeel geworden van dezelfde holding en maken ze vaak gebruik van elkaar expertise. Waar Green Orange vroeger alle onderdelen van het beantwoorden aan de digitale vraagstukken intern uitvoerde besteed zij momenteel veel componenten van deze vraagstukken uit aan de betreffende expert.

We::code is een programmeerbedrijf gericht op het automatiseren van digitale klantvraagstukken. Ze staan open voor ieder project, een 100% uitbesteding van een grote klus of een uurtje factuurtje maakt niet uit. Middels hun agile werkwijze brengen ze processen systematisch in kaart en bedenken op creatieve wijze manieren om dit proces zo gericht mogelijk te automatiseren.

Brandcube is gespecialiseerd in de ontwikkeling van websites en webapplicaties, zowel de front end als de back end. Het bedrijf beschikt over vormgevers en programmeurs, maar werken bij gecompliceerde projecten vaak samen met gespecialiseerde programmeurbedrijven, zoals We::code. In dit geval creëert Brandcube een design en zorgt We::code ervoor dat het werkt.

Datatics is, zoals de naam suggereert, gespecialiseerd in data. Het bedrijf verzamelt zo veel mogelijk data over particuliere websurfers. Op basis van verschillende algoritme creëren ze een 360-graden klantprofielen, die gebruikt kunnen worden voor marketingactie gericht op klantprofielen die zich in een specifiek marktsegment bevinden.

WePublish is in het leven geroepen, omdat een service van Green Orange dusdanig sterk gegroeid is dat het op het gebied van naamsbekendheid en risicomanagement verstandig werd om de service haar een eigen identiteit te geven: WePublish. Het bedrijf zit bij Green Orange in hetzelfde pand, wordt bestuurd door dezelfde bestuurders als Green Orange en kent, net zoals bij de oprichting, maar één service die aangeboden wordt aan haar klanten.

De service die WePublish aanbiedt bestaat uit een platform waar retailers reclamefolders kunnen uploaden en linken. Voor de retailers (klanten) en de particulier (de eindgebruiker) een website. Bij 'linken' van een reclamefolder worden hyperlinks, per product, toegevoegd die verwijzen naar de webshop van de betreffende retailers, zodat de eindgebruiker eenvoudig meer informatie op kan vragen en het product kan aanschaffen. We spreken in dit geval van een gelinkte folder.

1. Introductie

Het centrale thema in dit onderzoek is de ontwikkeling van automatiseringssoftware voor het identificeren en selecteren van producten uit folders die beheert worden door Green Orange. De ontwikkeling van deze software is momenteel namelijk gestagneerd.

Om te analyseren waar het probleem ligt binnen de ontwikkeling van de nieuwe softwaretool wordt in dit hoofdstuk gekeken naar de aanleiding van de vraag van Green Orange naar de nieuwe softwaretool (1.1.). De ontwikkeling van de nieuwe softwaretool stagneert door verschillende obstakels. In dit onderzoek worden deze obstakels beschreven als problemen. De onderliggende causale verbanden van deze problemen worden gevisualiseerd in een probleemkluwen (1.2.). Vervolgens wordt de praktische relevantie vastgesteld (1.3.). Tot slot wordt de ondersteunende methodologie in combinatie met de probleemaanpak en verslagstructuur omschreven (1.4.).

In dit hoofdstuk, en in de rest van het onderzoek, zal om verwarring te voorkomen gerefereerd worden naar Green Orange wanneer gesproken wordt over het betreffende bedrijf en de bedrijfsprocessen, ondanks dat dit in realiteit vaak WePublish is. De naam WePublish zal enkel gebruikt worden bij verwijzingen naar het 'WePublish-proces' of de 'WePublish-service' dat een bedrijfsproces vormt binnen Green Orange.

1.1 Probleemaanleiding

Green Orange kent één service. In de eerste sub-paragraaf wordt het proces van de WePublish-service in kaart gebracht. Vervolgens zal de directe aanleiding van het onderzoek geschetst worden en volgt in de laatste sub-paragraaf van de probleemaanleiding de achterliggende motivatie van Green Orange om dit onderzoek uit te laten voeren.

In de probleemanalyse wordt verder ingegaan op de problemen die in deze paragraaf aangestipt worden.

1.1.1. WePublish-service

In deze paragraaf wordt de WePublish-service toegelicht. In hoofdstuk 2 worden de technische aspecten uit deze service verder uitgelicht.

Start samenwerking

Een klant kan gebruik maken van de WePublish-service door een jaarlijks opzegbaar abonnement aan te schaffen, waarbij de klant maandelijks een factuur ontvangt. Bij de start van een samenwerking verkrijgt de klant toegang tot het platform van het WePublish-proces. Hiermee geniet de klant de volgende mogelijkheden:

- Zelf uploaden van folders
- Zelf linken van folders
- Zelf analyseren van klikgedrag op- en binnen folders

Basic – De samenwerking waarin de klant enkel toegang tot het platform als service afneemt wordt een 'basic', of 'self-service' samenwerking genoemd. Deze kosten zijn iedere maand constant.

Premium – De samenwerking waarin een klant extra betaald voor het uitbesteden van één of meer van de hierboven genoemde mogelijkheden binnen het platform van de WePublish-service wordt

een 'premium', of 'full-service' samenwerking genoemd. *(Beide type klanten genieten dus dezelfde mogelijkheden, maar in tegenstelling tot een basic klant besteed een premiumklant ook taken uit.)* In theorie heeft iedere klant de mogelijkheid om haar folders te linken en te analyseren, dan wel zelfstandig, dan wel door uitbesteding aan Green Orange.

In de praktijk zien we echter dat basic klanten zelden gebruik maken van het linken en analyseren van folders en daarmee enkel gebruik maken van de upload mogelijkheid. Naar verwachting hebben de retailers geen uren gealloceerd voor medewerkers om de werkzaamheden onder 'mogelijkheden' uit te voeren, waardoor het dús niet gedaan wordt.

Data en inzichten

Bij het aanbieden van de WePublish-service wordt veel data verzameld omtrent het klikgedrag, en daarmee interesses, van bepaalde particulieren. Van deze individuele particulieren wordt, in samenwerking met Datatrics, in verloop van tijd een klantprofiel opgesteld, zodat een grootschalige analyse kan leiden tot nieuwe inzichten omtrent de interesses binnen bepaalde klantsegmenten ten opzichte van bepaalde producten binnen bepaalde marktsegmenten.

Data wordt op voornamelijk vier verschillende niveaus verzameld: marktsegment-, retailer-, folder- en productniveau. Bij het online zetten van een folder wordt bijvoorbeeld data verzameld omtrent de interesse in de hele folder. Wanneer een folder gelinkt is kan gedetailleerde, meer waardevolle, data verzameld worden omtrent specifieke producten. Al deze data wordt, afhankelijk van het detailniveau, gelinkt aan het klantprofiel en het klantsegment om nieuwe inzichten te genereren. De inzichtgeneratie op gebied van klantprofielen en klantsegmenten wordt in opdracht van Green Orange uitgevoerd door Datatrics.

Toegevoegde waarde voor klanten die gebruik maken van de WePublish-service

Door de verzamelde data omtrent het klantprofiel en verschillende klantsegmenten te linken aan het klikgedrag op het folder platform is Green Orange in staat om adviezen uit te brengen omtrent de verbetering van de foldercontent van klanten. Daarnaast is Green Orange in staat om particulieraafhankelijke foldercontent te genereren, afhankelijk van de interesses van de particulier. Tot slot, door samenwerking met Green Orange worden nieuwe inzichten gegenereerd die de retailer van meer omzet zullen voorzien. Dit betaalt de investering ruimschoots terug. Klanten dienen extra te betalen voor boven genoemde inzichten.

Taken van medewerkers

In de vorige secties staat op globale wijze het businessmodel van de WePublish-service beschreven. Deze sectie spitst zich op de werkzaamheden van de operationele medewerkers binnen deze service, omdat de taken van deze medewerkers voor de eventuele automatisering het meest relevant zijn om in kaart te brengen. Er wordt daarom ingezoomd op de processen die zij uitvoeren.

Binnen de WePublish-service zijn twee medewerkers verantwoordelijk voor de operationele taken binnen het businessmodel. Medewerker1 is aangesteld om folders te linken, medewerker 2 analyseert data en verwerkt deze tot PowerPoint slides. Deze slides worden door de accountmanagers binnen 'Sales' overgenomen en gebruikt tijdens klantevaluaties.

Medewerker1 linkt iedere folder van premiumklanten en upload deze folders naar het platform. Dit proces is erg arbeidsintensief, maar bevat geen, voor een mens, complexe taken. In Figuur 1 is een pagina uit de Aldi-folder getoond vanuit de editor die medewerker1 gebruikt om interacties toe te voegen aan de pagina. In het figuur geven de rode 'boxen' geven het 'interactievlak' weer. Ieder

interactievlaak kan gemaakt en bewerkt worden in het ‘interactie toevoegen’ menu aan de linker zijde. Aan de rechter zijde is het dropdown menu van het ‘interactie type’ getoond. Duidelijk is dat een interactie niet alleen een hyperlink hoeft te zijn, ondanks dat dit onderzoek zich voornamelijk focust op het hyperlink interactietype.

Medewerker2 voert regelmatig analyses op het klikgedrag van particulieren uit en maakt de inzichten per retailer inzichtelijk. Daarnaast onderneemt medewerker2, op aanvraag van de klant, ook meer complexere analyses die gedetailleerde inzichten opleveren.



Figuur 1: Interactievlakken, editor-perspectief van medewerker1

1.1.2. Automatisering van taken medewerker1

Achtergrond omtrent automatiseringstraject

Green Orange heeft de wens om de taken van medewerker1 te automatiseren. De motivatie achter deze wens is verder toegelicht in de volgende sub-paragraaf.

In 2018 zijn twee medewerkers begonnen met een onderzoek waarin zij in kaart hebben gebracht wat nodig is om automatiseringssoftware te schrijven en te implementeren, ter vervanging van de taken van medewerker1. In dit bestand staan voornamelijk obstakels in het proces van ontwikkelen en implementeren omschreven. Naast het onderzoek hebben zij ook gewerkt aan een ‘proof of concept’ dat gebruik maakt van GoogleVision.

Ondanks de tijd en energie die Green Orange geïnvesteerd heeft in het ontwikkelen van de automatiseringssoftware heeft dit nog niet geleid tot de implementatie van werkende software in het businessmodel van de WePublish-service. In de probleem identificatie wordt verder ingegaan op de achterliggende reden achter de stagnatie van het ontwikkelingsproces.

1.1.2. Waarom automatiseren?

In deze sectie wordt ingegaan op de initiële motivatie van Green Orange om aan het machine-learning softwareontwikkelingsproject (MLSP) te beginnen.

Samenvattend wordt gesteld dat optimalisatie van het WePublish-proces, in de vorm van automatisering, noodzakelijk is om de strategische visie van Green Orange op het gebied van de WePublish-service te laten slagen.

Strategie WePublish-service

Green Orange heeft binnen de WePublish-service twee hoofddoelen voor de komende 5 jaar, 1.) nieuwe klanten binden (groeien) en 2.) de waarde van de samenwerkingen met hun huidige klanten vergroten (tevredenheid vergroten). Om dit te bereiken willen zij zich ontwikkelen tot meer dan enkel een online pdf-platform.

In de strategische visie van de WePublish-service is het online pdf-platform niet langer de kern business, maar een middel ter ondersteuning van een nieuw te ontwikkelen service: consultancy. Deze te ontwikkelen service biedt strategisch advies, gespecialiseerd in marketingdoelstellingen, aan haar klanten. Bij de implementatie van deze strategische visie zal de WePublish-service veranderen van een online platform naar een consultant.

Meer data generatie

Om deze strategie in werking te zetten heeft Green Orange meer data over haar klanten nodig. Zoals beschreven in de vorige paragraaf, wordt niet iedere folder op het platform gelinkt, waardoor in ieder marktsegment veel data onbenut blijft. Deze data kan in potentie binnen hetzelfde marktsegmenten gebruikt worden om twee partijen binnen hetzelfde marktsegment met elkaar te vergelijken, denk aan een Aldi en een Lidl. Ook kan Green Orange de betrouwbaarheid van haar adviezen vergroten en dieper toespitsen wanneer zij meer data weet te verzamelen over haar klanten en de marktsegmenten waarin deze zich bevinden.

Obstakel bij het genereren van meer data

Er is één hoofdreden waarom niet ieder folder gelinkt wordt: het linken van folders is erg kostbaar, i.v.m. de hoge arbeidsintensiteit die gepaard gaat met dit proces. Green Orange is vanwege de hoge kosten niet in staat om alle folders, zonder financiële vergoeding van premium klanten, te linken ondanks de data die zij hiermee zouden verzamelen.

Mogelijke oplossingen

Om het probleem van deze hoge kosten weg te nemen heeft Green Orange drie initiatieven ondernomen. Zo hebben zij (1.) goedkope werkrachten, zoals stagairs en flexkrachten, deze taak uit laten voeren. (2.) Ook hebben ze een pilot gehad waarin zij het linken van folders uitbesteden aan een Thais bedrijf, waar voor een laag bedrag veel uren ingekocht kunnen worden. (3.) Tot slot heeft Green Orange gekeken naar automatisering van (een deel) van het WePublish-proces.

De eerste twee stappen waren vanwege het aantal folders dat wekelijks geüpload wordt nog steeds erg kostbaar. Daarnaast waren beide initiatieven niet duurzaam, aangezien bij groei van de omzet, de kosten lineair zouden oplopen en veel potentiële winst verloren zou gaan. Vanwege deze redenen heeft Green Orange besloten om te focussen op optie 3, automatisatie.

Hieronder volgt een korte uiteenzetting van het proces dat Green Orange op het gebied van optie 3, automatisatie, heeft doorlopen.

De eerste stap was om te kijken naar een mogelijkheid om deze techniek extern aan te schaffen bij een gespecialiseerd bedrijf. Green Orange heeft ondervonden dat de bedrijven die ze benaderd hebben met kostbare abonnementen op jaarbasis werken. Naast deze kostbare abonnementen zou de nieuwe kernbusiness van de WePublish-service volledig afhankelijk worden van deze abonnementen, en daarmee van een extern bedrijf. Geen wenselijke situatie, aldus verschillende medewerkers van Green Orange.

De tweede stap was om te kijken naar de mogelijkheid om (een deel) van het WePublish-proces zelf te automatiseren. Zoals benoemd in de probleemaanleiding, is de stagnering in de ontwikkeling van deze nieuwe automatiseringssoftware de aanleiding voor dit onderzoek. Dit is dan ook de situatie waar Green Orange zich nu in bevindt.

1.2 Probleemidentificatie (introdunctie)

In de vorige paragraaf is de huidige situatie rondom de WePublish-service in kaart gebracht en is toegelicht op welke manier Green Orange de service verder wil ontwikkelen, namelijk d.m.v. automatiseringssoftware die zelf ontwikkeld is. Afsluitend, is de argumentatie achter deze keuze verder toegelicht.

Deze paragraaf richt zich op het in kaart brengen van het kernprobleem achter de stagnering van de softwareontwikkeling door gebruik te maken van een probleemkluwen. Ten eerste zal de probleemkluwen toegelicht worden aan de hand van een visuele weergave in Figuur 2. Vervolgens zullen de verschillende probleemsecties uit de probleemkluwen verder worden uitgewerkt. In de uitwerking van de probleemanalyse wordt verwezen naar de nummering van problemen zoals deze voorkomen in de probleemkluwen.

1.2.1. Probleemkluwen

Een probleemkluwen geeft inzicht in de onderlinge causale verbanden tussen de verschillende problemen in de kluwen (Heerkens, H. 2012, Geen Probleem). Deze causale verbanden zijn in onderstaande probleemkluwen aangegeven met pijlen.

Het kernprobleem van de probleemkluwen is een probleem dat zelf geen gevolg meer heeft én wat beïnvloedbaar is (Heerkens, 2012). Op het moment dat meerdere kernproblemen een rol spelen in de probleemkluwen, wordt door middel van besluitcriteria en overleg met het betreffende bedrijf het belangrijkste kernprobleem gekozen om op te lossen.

In de probleemkluwen, die in Figuur 2 getoond is komen vijf kernproblemen naar voren die zelf geen gevolg meer kennen én beïnvloedbaar zijn. Dit betekent dat het belangrijkste kernprobleem nader bepaald dient te worden. De stem van Green Orange is in dit besluitproces de belangrijkste, aangezien zij de uitkomst van het onderzoek willen implementeren. Hieronder worden de probleemsecties uit de probleemkluwen doorlopen.

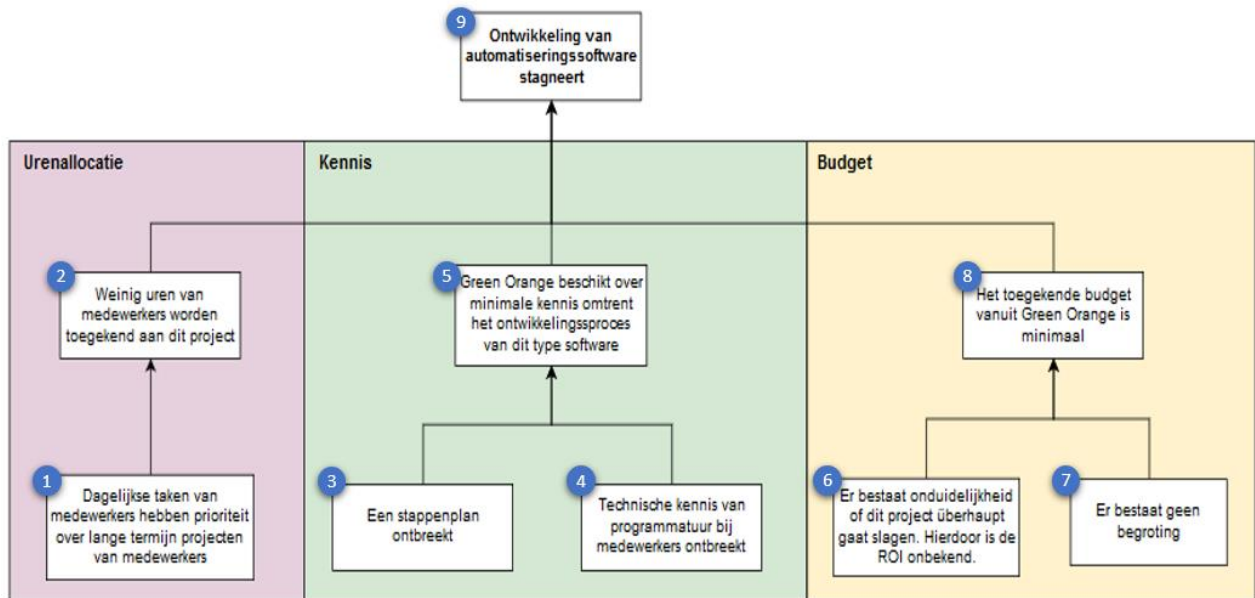
Urenallocatie. Green Orange staat achter de huidige werkwijze binnen het team en de bijbehorende urenallocatie. Ze willen dit momenteel niet aanpassen.

Kennis. Green Orange geeft aan dat kennis in haar optiek momenteel het grootste probleem is, aangezien de kennis van een dergelijk proces invloed heeft op ieder ander besluit. Zo zal

de beschikbare kennis invloed hebben op de urenallocatie van medewerkers en het toegekende budget.

Budget. De onduidelijkheid m.b.t. de Return on Investment (ROI) is volgens Green Orange voornamelijk afkomstig uit het gebrek aan proceskennis. Bij het opstellen van een stappenplan wordt de haalbaarheid bijvoorbeeld ook meer zichtbaar. Volgens J. Chen (2019) is de ROI een prestatie maatstaf om de efficiëntie van een investering te evalueren.

In Figuur 2 worden de componenten uit de probleemkluwen aangeduid met cijfers. In de hierop volgende sub-paragrafen worden deze cijfers (1. t/m 5.) genoemd.



Figuur 2: Probleemkluwen, Green Orange.

Urenallocatie

Binnen Green Orange beschikken twee medewerkers, respectievelijk medewerker 1 en 2 (dit zijn andere medewerkers dan waar eerder over gesproken werd) over de grootste inhoudelijke kennis van het machine-learning softwareontwikkelingsproject (MLSP). Andere medewerkers zijn hier minimaal bij betrokken. De twee medewerkers dragen het MLSP op eigen initiatief.

Medewerker 1 en 2 maken echter minimale voortgang op het MLSP aangezien zij weinig uren toe kunnen wijden aan dit project, ondanks dat zij dit graag anders zouden zien (2.), waardoor het MLSP stagneert.

De oorzaak van dit probleem is één ledig: de afspraken omtrent werkprioritering binnen Green Orange. De werkprioritering stelt dat werk voor klanten altijd voorrang heeft op andere (lange termijn) projecten (1.).

Momenteel wordt het gebrek in prioriteit en daarmee het gebrek in tijd herhaaldelijk, tijdelijk opgelost door een student op het MLSP te zetten in de vorm van een projectopdracht vanuit de UT, een afstudeeropdracht of dienstverband. De kennis, expertise en ervaring van studenten op dit complexe onderwerp is echter gelimiteerd. Vaak is dit vaak de eerste keer is dat de studenten met het programmeren van machine-learning software in aanraking komen, en de tijd die zij aan dit project kunnen besteden is minimaal, vaak maar +- 2 maanden. Het aanstellen van een student op tijdelijk dienstverband heeft tot nu toe geen duurzame oplossing gebleken.

Kennis over machine-learning

De inhoudelijke kennis over het ontwikkelingsproces van de machine-learning software die medewerker 1 en 2 bezitten blijkt niet voldoende (5.), waardoor het MLSP stagneert.

De oorzaak van dit probleem is tweeledig. Ten eerste, medewerker 1 en 2 beschikken niet over een handleiding of stappenplan dat zij kunnen volgen (3.). Ondanks het gebrek in ervaring kan een goed uitgewerkt stappenplan de twee medewerkers dusdanig ondersteunen dat zij in staat zullen zijn het project tot een goed einde te leiden.

Ten tweede, beide medewerkers beschikken niet over de inhoudelijke kennis om de programmatuur voor het MLSP zelf te ontwikkelen of te coördineren (4.). Er wordt naar eigen zeggen 'geknutseld' aan de programmatuur, maar diepgaande kennis over het onderwerp ontbreekt.

Een gevolg van deze ontbrekende inhoudelijke kennis is dat aansturing van eventuele studentmedewerkers belemmerd wordt, waardoor deze studenten vaak op zichzelf verder moeten werken. Daarnaast zijn eventuele investeringen moeilijk te beargumenteren, omdat een helikopterblik op het proces ontbreekt. Hierover in de volgende sub-paragraaf meer.

Om dit gebrek aan inhoudelijke kennis op te lossen wordt het programmeren van de machine-learning software uitbesteed aan de tijdelijk aangestelde studenten, zoals beschreven in de vorige sub-paragraaf. Daarnaast doen de twee medewerkers er alles aan om meer kennis te vergaren via andere bedrijven, artikelen, filmpjes en guides in de tijd die ze wél aan het MLSP kunnen besteden.

Budget

Volgens de twee medewerkers is de budgettering voor het MLSP ten opzichte van andere projecten binnen Green Orange minimaal (8.).

De oorzaak van dit probleem is opgeknipt in twee problemen, die een sterke correlatie hebben met het gebrek aan een overkoepelend plan (3.). Zoals aangestipt in de vorige sub-paragraaf is het lastig om met een ontbrekend helikopterblik op het proces een betrouwbare budgettering op te stellen (7.).

Daarnaast is het onmogelijk om de risico's van het project goed in te schatten vanwege het gebrek aan kennis over het project en bijbehorende proces (6.).

Een gevolg van de minimale budgettering is dat het voor de twee medewerkers vrijwel onmogelijk wordt om externe kennis in te kopen of bepaalde hypothesen te toetsen. Naar verwachting kan Green Orange meer budget vrijmaken bij een complete onderbouwing van het hele project. Momenteel zijn de aanvragen veelal ad-hoc geplaatst.

Afweging van het kernprobleem

Normaalgesproken volgt uit de probleemkluwen een kernprobleem dat geselecteerd wordt om op te lossen. Uit dit op te lossen kernprobleem volgt de doelstelling van het onderzoek. Het onderzoek lost bij het oplossen van het kernprobleem de initiële probleemstelling op (9.).

In de situatie van dit onderzoek loopt het anders, Green Orange heeft namelijk vooraf besloten dat er een methode ontwikkeld dient te worden. Daarmee staat het doel van het onderzoek vast, zonder dat deze afgeleid kan worden uit het geselecteerde kernprobleem. Deze sub-paragraaf toetst op kwalitatieve wijze in hoeverre de doelstelling aansluit bij één van de kernproblemen en in hoeverre de doelstelling bijdraagt aan het oplossen van de initiële probleemstelling (9.).

Hieronder worden de kernproblemen uit de probleemkluwen één voor één naast de doelstelling van het onderzoek gelegd. De schaal waarin aansluiting tussen de doelstelling en de kernproblemen

worden beoordeeld loopt van laag tot hoog: 'sluit niet aan', 'sluit minimaal aan', 'sluit gemiddeld aan', 'sluit goed aan', 'sluit zeer goed aan'.

- (1.) De methode zal weinig veranderen aan de prioriteringsafspraken die Green Orange intern handhaaft.
Het ontwikkelen van de methode kan tot gevolg hebben dat medewerker 1 en medewerker 2, in combinatie met de tijdelijk aangestelde werkkrachten, efficiënter en effectiever te werk kunnen gaan. Hierdoor is het mogelijk dat bij dit geselecteerde gezelschap medewerkers meer uren vrijkomen.
De doelstelling sluit minimaal aan bij dit kernprobleem.
- (3.) De methode is in feite een stappenplan waardoor een helikopterblik op het probleem wordt gecreëerd.
De doelstelling sluit zeer goed aan bij dit kernprobleem.
- (4.) De methode zal de medewerkers niet in directe zin van technische kennis voorzien, maar zal de medewerkers wél begeleiden in waar zij deze technische kennis kunnen vinden. Ook zal de methode de medewerkers begeleiden in het keuzeprocess om technische kennis zélf te vergaren, óf een bepaald onderdeel van het proces uit te besteden.
De doelstelling sluit goed aan bij dit kernprobleem.
- (6.) De methode verschaft Green Orange van een helikopterblik op het te doorlopen ontwikkelingsproces. De methode heeft echter niet als hoofddoel om de kosten van iedere stap van het proces in kaart te brengen en zal zich niet focussen op het opstellen van een risico analyse.
De doelstelling sluit gemiddeld aan bij dit kernprobleem.
- (7.) Zoals benoemd bij (6.) zal de methode niet als hoofddoel hebben om de kosten van iedere stap van het proces in kaart te brengen. Wél zal er een globale begroting worden bijgeleverd, in combinatie met aanbevelingen om bepaalde onderdelen van het proces uit te besteden.
De doelstelling sluit gemiddeld aan bij dit kernprobleem.

Uit bovenstaande analyse valt te concluderen dat het onderzoek een relevante bijdrage zal leveren aan het oplossen van de initiële probleemstelling (9.), doordat het onderzoek kernprobleem 3 volledig zal verhelpen. Daarnaast zal het onderzoek, met de focus op kernprobleem 3, voor het oplossen van kernprobleem 4, 6 en 7 in de toekomst, een fundamentele basis hebben gelegd.

1.2.2. Relevantie van het probleem

In de 'Afweging van het kernprobleem' wordt geconcludeerd dat kernprobleem 3 het meest relevant is om op te lossen. In deze paragraaf wordt probleem 9, in combinatie met de geformuleerde doelstelling, in directe- en indirecte relevantie uitgesplitst. Binnen deze twee uitsplitsingen wordt gekeken naar de positieve- en negatieve ontwikkelingen. Aan het einde is ook een kopje 'overige' toegevoegd.

Met directe relevantie wordt bedoeld op de acute 'win', of 'lose' die voor Green Orange gecreëerd wordt wanneer probleem 9 wordt aangepakt.

Met indirecte relevantie wordt bedoeld op nieuwe ontwikkelingen die op de lange termijn, bij het oplossen van probleem 9, ook gecreëerd worden. Deze mogelijkheden zijn niet het hoofddoel van dit onderzoek en zullen aan het einde van het verslag terugkomen in het hoofdstuk 'aanbevelingen'.

Aan het einde van deze paragraaf is duidelijk waarom het oplossen van probleem 9. überhaupt relevant is voor Green Orange. Hiervoor wordt voor een groot deel teruggegrepen op de formuleringen in de sub-paragraaf: 'waarom automatiseren?'.

Directe relevantie

Positieve veranderingen

- Green Orange is in staat om de WePublish service voor het grootste deel automatisch uit te voeren. Hierdoor kan meer werk door hetzelfde aantal medewerkers gedaan worden.
- Green Orange is in staat om folders van alle partners te linken, zonder de data die hiermee gegenereerd wordt aan de betreffende partners te leveren. Hierdoor is Green Orange in staat om meer verschillende inzichten te genereren en bestaande inzichten nauwkeuriger aan te leveren aan premium partners.
 - o Voorbeeld: Green Orange is nu in staat om meer- en nauwkeurigere vergelijkingsanalyses uit te voeren tussen haarpartners (zowel premium, als basic) in dezelfde marktsector. Deze inzichten blijven bij Green Orange, tenzij een partner betaald voor deze inzichten.
- Doordat Green Orange in staat is om vergelijkingsanalyses uit te voeren is Green Orange in staat om voorspellende uitspraken te doen over eventuele toekomstige partner (een toekomstige partner is nog niet geclassificeerd tot premium- of basic partner) die zich in een reeds in kaart gebracht marktsegment bevindt. Hierdoor beschikt Green Orange over een krachtige propositie tegenover de potentiële partner. Green Orange zal beter in staat zijn om haar omzet te laten toenemen, door middel van het binden van nieuwe partners.
- Samengevat, Green Orange is in staat om uit te groeien tot digitaal consultant, in plaats van digitaal marketing kantoor. Deze stap sluit goed aan bij het strategische doel van Green Orange.

Negatieve veranderingen

- Green Orange loopt het risico dat zij medewerkers opnieuw moet alloceren wanneer de omzet niet groeit. Hetzelfde werk zal verzet kunnen worden door minder medewerkers.
- Green Orange loopt het risico dat medewerkers niet in staat zijn om met de nieuwe software te leren werken. Hierdoor is Green Orange genoodzaakt om de betreffende medewerker binnen een ander project te alloceren.
- Green Orange loopt het risico dat de tijds- en materiaal investering uiteindelijk uitlopen op een ongeslaagd project dat niet geïmplementeerd kan worden.
- Green Orange, creëert een grote afhankelijkheid van de nieuwe machine-learning software.
 - o Voorbeeld: De Down-time van de service kan een negatieve invloed hebben wanneer een partner verwacht dat haar folder direct gelinkt wordt.
 - o Voorbeeld: Concurrenten zouden met de vergaarde kennis van Green Orange vergelijkbare software kunnen bouwen wanneer Green Orange geen stappen onderneemt om haar IP te beschermen.
- Green Orange loopt het risico dat partners in de toekomst minder gebruik van folders zullen maken, waardoor de nieuwe ontwikkelingen op het gebied van de machine-learningsoftware niet toegepast kan worden.
 - o Voorbeeld: De markt maakt een transitie naar enkel pop-up reclame, waardoor folders van retailers niet meer worden ontwikkeld.

Indirecte relevantie

Positieve veranderingen

- Green Orange is in staat om vergelijkbare processen, zoals de WePublish service, binnen het bedrijf te automatiseren. Het automatiseren van deze processen kent in essentie geen limiet.
 - o Momenteel kan geen concreet voorbeeld gegeven worden.
- Green Orange is in staat om de ontwikkelde methode te gebruiken om een nieuwe service op de markt te brengen waarmee zij vergelijkbare automatiseringsprojecten van partners op zich kan nemen. In theorie kan dit uitgroeien tot één van de meest waardevolle services die Green Orange beschikt.

Negatie veranderingen

- Geen onderwerpen gevonden.

Overige, neutrale opmerking

- Green Orange kan zich in de toekomst af te vragen wat haar hoofd business is en welke services zij aan haar klanten wilt blijven aanbieden. Een aftakking voor deze machine-learning tak kan een afweging zijn.
- Green Orange kan op den duur uitzoeken in hoeverre zij door ontwikkeling van deze automatiseringssoftware gaat concurreren binnen haar eigen holding en of dit wenselijk is.

1.2.3. Doelstelling

In deze paragraaf wordt het doel van dit onderzoek toegelicht. De doelstelling is in extensief overleg met de begeleider van Green Orange vastgesteld.

Het doel van het onderzoek: *het ontwikkelen van een methode, op basis van machine-learning componenten, waarin het ontwikkelingsproces van automatiseringssoftware staat beschreven.*

De methode zal inzicht geven in de verschillende processen die doorlopen dienen te worden om tot een gewenste automatisering te komen. Om dit onderzoek naast de theorie te ondersteunen zal de methode getoetst worden op het initiële automatiseringsprobleem van Green Orange; het automatiseren van de WePublish-service. Dit proces staat verder beschreven in 2. Huidige situatie en zal door middel van een casestudie getoetst worden in 6. Demonstratiefase – Proof of concept. De te ontwikkelen methode dient aan te sluiten bij de bestaande bedrijfsvoering van Green Orange en dient praktische ondersteuning te bieden aan medewerkers die in de toekomst een vergelijkbaar automatiseringsproces doorlopen, zoals omschreven in de casestudie. De methode dient zowel bij processen binnen Green Orange, als bij bedrijfsprocessen buiten Green Orange gebruikt te kunnen worden.

De methode wordt vanuit Green Orange als succesvol beschouwt wanneer deze toepasbaar is op de casestudie en in het vervolg gebruikt kan worden bij een nieuwe service die Green Orange wenst te lanceren; het leiden van vergelijkbare automatiseringsprocessen bij hun klanten.

1.2.4. Hoofd en deelvragen

Dit kopje formuleert op basis van de probleemidentificatie en de beschreven onderzoeksdoelstelling een hoofdvraag. Vervolgens wordt de formulering van de bijbehorende deelvragen, opgedeeld in kennis- en designvragen, gegeven. Aan de hand van een korte omschrijving wordt aangegeven op welke wijze de deelvragen bijdragen aan de beantwoording van de hoofdvraag en op welke wijze de deelvragen geïnterpreteerd worden.

De onderzoeksvragen zijn ontwerp- en probleemoplossend van aard, omdat nieuwe oplossingen voor bestaande problemen gevonden dienen te worden en de vragen zich richten op het oplossen van problemen in de nabije toekomst.

Hoofdvraag: “Hoe kan een methode, op basis van machine-learning componenten, het ontwikkelingsproces voor softwareautomatisering binnen Green Orange verbeteren?”.

De hoofdvraag wordt door middel van een kwalitatief onderzoek beantwoord. Dit houdt in dat er voornamelijk gebruik wordt gemaakt van literatuuronderzoek en nieuwe theorievorming. (Hierover in 1.3. Onderzoeksopzet en verslagopbouw meer).

Deelvragen

De deelvragen van dit onderzoek zijn opgedeeld in twee categorieën: kennisvragen en designvragen. Het beantwoorden van kennisvragen verschaft een fundament aan kennis om het onderzoek te starten. De kennisvragen worden in het onderzoek uitgewerkt in 2. Theoretisch kader. Het beantwoorden van de designvragen draagt direct bij aan het beantwoorden van de hoofdvraag waardoor iedere designvraag haar eigen hoofdstuk kent.

Tabel 1 geeft een overzicht van de hoofdstukken uit het onderzoek, in combinatie met bijbehorende deelvragen en de verdeling, per DSRM-stap (verder uitgewerkt in 1.4.3. Onderzoeksmethodiek – DSRM).

a. Kennisvragen

De kennis die vergaart wordt bij het beantwoorden van kennisvragen is essentieel voor het beantwoorden van de designvragen, die op hun beurt weer bijdragen aan het beantwoorden van de hoofdvraag. Iedere kennisvraag wordt met behulp van literatuuronderzoek beantwoord. Het theoretisch kader geeft antwoord op de overkoepelende kennisvraag: “Welke kennis omtrent de beschreven doelstelling bestaat al?”.

De volgende kennisvragen worden in het theoretisch kader behandeld:

- **Kennisvraag 1:** “Welke componenten van machine learning zijn van toepassing op het omschreven probleem?”.

Zoals benoemd in de inleiding van 1.4.2. Afbakening onderzoek is bij aanvang van het onderzoek beperkte kennis omtrent machine-learning beschikbaar. Om een duidelijk antwoord op de hoofdvraag te geven is het essentieel om het vakgebied binnen machine-learning, dat van toepassing is op de omschreven doelstelling van het onderzoek, vast te stellen. In de beantwoording van deze vraag wordt dan ook het vakgebied en eventuele subsets die van toepassing zijn op de omschreven doelstelling vastgesteld. Tevens wordt een uitgebreide uitleg gegeven over de werking van machine-learning (inclusief korte wiskundige notaties), zodat gebruiker van de methodiek beter in staat zijn om problemen tijdens de ontwikkeling van software zelf op te lossen.

- **Kennisvraag 2:** “Welke methoden worden in de ontwikkeling van software gebruikt?”.

De ontwikkeling van automatiseringssoftware, gebaseerd op machine-learning componenten, is een softwareontwikkelingsproces. Speciaal ontwikkelde methoden worden gebruikt bij de ontwikkeling van software. In deze vraag wordt onderzocht welk type methoden dit zijn en op welke manier deze methoden toegepast kunnen worden op de beantwoording van de hoofdvraag.

- **Kennisvraag 3:** “Welke methoden zijn beschikbaar voor het ontwikkelen van een methode?”. In de beantwoording van deze vraag wordt gekeken naar verschillende bestaande methodieken die als handleiding kunnen fungeren bij de ontwikkeling van een methode. Er wordt gezocht naar een check-list, of stappenplan dat afgelopen kan worden bij het beantwoorden van de hoofdvraag.
- **Kennisvraag 4:** “Welke methoden zijn beschikbaar voor het visueel in kaart brengen van een bedrijfsproces?”. In de beantwoording van deze vraag wordt gekeken naar verschillende bestaande methoden die als handleiding kunnen fungeren bij het in kaart brengen van het WePublish- (bedrijfs)proces. Er wordt gezocht naar een visuele methode, die eenvoudig te begrijpen en gebruiken is door de stakeholders en de stakeholder van een helikopterbeeld van het bedrijfsproces verschaft.

b. Designvragen

De designvragen behandelen ieder afzonderlijk een onderdeel dat bijdraagt aan het beantwoorden van de hoofdvraag. Gezamenlijk vormen ze de basis waaruit het antwoord op de hoofdvraag afgeleid kan worden. De designvragen van dit onderzoek bestaan uit de volgende vragen:

- **Designvraag 1:** “Hoe ziet het huidige proces achter de WePublish-service eruit?”. Momenteel is er geen duidelijke visualisatie van de verschillende processtappen die plaatsvinden in de WePublish-service. Deze visualisatie is nodig om specifieke stappen te definiëren die eventueel vervangen kunnen worden door de te ontwikkelen automatiseringssoftware. Bij de eventuele invoering van automatiseringssoftware vind een verandering in het proces plaats. Om een indicatie te geven over of de eventuele verandering positief of negatief is dient een vergelijkbare meting uitgevoerd te worden, één op de huidige situatie en één na de implementatie van de verandering. In de beantwoording van deze vraag wordt een BPMN-model ontwikkeld dat een visuele weergave van de huidige processen van de WePublish-service toont. Daarnaast behandelt de beantwoording van de vraag de ontwikkeling van het Pythonscript (projectresultaat) dat gebruikt wordt bij de nulmeting en in de toekomst gebruikt kan worden bij vergelijkbare metingen. Het BPMN-model wordt met behulp van de vastgestelde notatie uit de literatuur omschreven en door middel van interviews met medewerkers opgesteld. Het Pythonscript wordt zelfstandig ontwikkeld en toegepast op een datadump uit de Interactie Database (3. Huidige situatie) ten behoeve van de nulmeting.
- **Designvraag 2:** “Hoe ziet het ideale proces achter de WePublish-service eruit?”. Nadat de huidige processen achter de WePublish-service in kaart zijn gebracht wordt bij de beantwoording van deze vraag een BPMN-model ontwikkeld dat een visuele weergave van de nieuwe processen van de WePublish-service in kaart toont waarop de verschillen ten opzichte van de huidige situatie duidelijk weergegeven zijn. Daarnaast wordt een minimaal levensvatbaar product gedefinieerd dat als leidraad wordt gebruikt bij de ontwikkeling van de proof of concept (projectresultaat). Het BPMN-model wordt op basis van het eerder ontwikkelde BPMN-model opgesteld en aangepast met behulp van de kennis die verkregen is bij de beantwoording van de eerste kennisvraag. De proof of concept wordt in Python geschreven en maakt gebruik van de vergaarde kennis van de eerste kennisvraag.

- **Designvraag 3:** “Hoe ziet een methode, op basis van machine-learning componenten, waarin het ontwikkelingsproces voor software automatisering staat beschreven eruit?”.
In de beantwoording van deze vraag wordt teruggesproken naar de kennis die vergaart is bij de beantwoording van de kennisvragen en de inzichten die verkregen zijn bij de beantwoording van de designvragen om antwoord te geven op de hoofdvraag.
- **Designvraag 4:** “Hoe kan de ontwikkelde methode worden toegepast bij Green Orange?”.
In de beantwoording van deze vraag wordt de ontwikkelde methode toegepast op de omschreven WePublish-service om een proof of concept te creëren.

1.2.5. Afbakening onderzoek

Deze sectie focust zich op het inperken van de omvang van het onderzoek. Hiervoor worden eerst de projectresultaten (deliverables) opgesomd, om vervolgens de scope van het onderzoek te definiëren. Zowel de gekozen projectresultaten, als de scope worden beperkt door drie factoren: gelimiteerde tijd voor het onderzoek (12 weken), minimale inhoudelijke begeleiding- en feedback vanuit GreenOrange en beperkte kennis omtrent machine learning bij aanvang van dit onderzoek.

Projectresultaat

Het projectresultaat is hetgeen dat aan het einde van het onderzoek moet worden opgeleverd aan de opdrachtgever, Green Orange. In dit kopje worden drie projectresultaten vastgesteld. Voor het vaststellen van het eerste projectresultaat wordt teruggesproken naar de eerder geformuleerde doelstelling en het bijbehorende kernprobleem. Dit is tevens het belangrijk projectresultaat.

- 1.) Het eerste projectresultaat bestaat uit het onderzoeksverslag inclusief methodiek, op basis van machine-learning componenten, waarin het ontwikkelingsproces van automatiseringssoftware staat beschreven.

Dit projectresultaat wordt op volgende drie componenten afgebakend:

- a. Praktische werkbaarheid – De praktische werkbaarheid voor medewerkers van Green Orange komt centraal te liggen. Dit betekent bijvoorbeeld dat complexe onderdelen niet tot in detail zullen worden beschreven en voor de werknemers van Green Orange irrelevante onderdelen niet in de methode zullen worden opgenomen.
- b. Terminologie – De methode zal gebruik maken van terminologie die in het theoretische kader wordt toegelicht. Van de lezer/gebruiker wordt minimale voorkennis verwacht op het gebied van machine-learning en ICT.
- c. Prototype – Het onderzoek zal ook tijd besteden aan het ontwikkelen van een proof of concept van de omschreven automatiseringssoftware, zodat de validiteit van de methode aangetoond kan worden. De proof of concept wordt als projectresultaat vastgesteld.

- 2.) Het tweede projectresultaat bestaat uit de ontwikkeling van een proof of concept van de omschreven automatiseringssoftware, zodat validiteit van de methode aangetoond wordt.

In 1.1 Probleemaanleiding wordt duidelijk dat het ontwikkelen van een methode voortkomt uit de vraag om de WePublish-service voor een deel te automatiseren. Om voortgang binnen de WePublish-service te meten wordt in 2. Huidige situatie een nulmeting gedaan. Om in de toekomst een vergelijkbare meting te doen wordt een Pythonscript ontwikkeld dat de WePublish-service scoort op verschillende prestaties middels KPI's (*binnen 3. Theoretisch kader meer over KPI's*).

3.) Het derde projectresultaat bestaat uit de ontwikkeling van een Pythonscript dat de WePublish-service op verschillende momenten op dezelfde wijze kan analyseren. Tot slot verschaft het onderzoek ook inzicht in het huidige proces en het toekomstige proces, maar worden deze hier niet opgesomd, omdat het geen fysieke deliverables zijn.

Scope

In dit kopje wordt ten eerste de scope van het onderzoek en haar doelstelling vastgesteld, vervolgens wordt de scope van het tweede en derde projectresultaat omschreven.

Het hoofddoel van het onderzoek, zoals beschreven in 1.4.1. Doelstelling, is de ontwikkeling van een methode, op basis van machine-learning componenten, waarin het ontwikkelingsproces van automatiseringssoftware staat beschreven. De methode wordt in opdracht van GreenOrane ontwikkeld. Zowel GreenOrange, als de Universiteit Twente biedt begeleiding in de ontwikkeling van de methode. De methode maakt gebruik van de Agile ontwikkelingsmethodiek die binnen softwareontwikkelingsprojecten gebruikt wordt. De methode kent zes stappen, die afhankelijk van keuzes, elkaar iteratief opvolgen. De voorgestelde methodiek wordt gelimiteerd tot ComputerVision, één van de onderzoeksrichtingen binnen het machine-learning vakgebied. Binnen ComputerVision focust de methodiek zich enkel op de ObjectDetection toepassing. De methodiek kan toegepast worden op ObjectDetection automatiseringsvraagstukken binnen ieder bedrijf en onderzoek.

De proof of concept maakt gebruik van een bestaande NN-modelarchitectuur (InceptionV2), ontwikkeld door het onderzoeksteam van Google en is getraind op handmatig gelabelde data specifiek van toepassing op de situatie van GreenOrange: de WePublish-service. De outputkwaliteit van de proof of concept wordt gelimiteerd door haar trainingsdata, bestaande uit 200 images gelabeld op enkel drie productcategorieën. De proof of concept kan gebruikt worden als basis voor ObjectDetection automatiseringsvraagstukken en wordt als validatie gezien van de ontwikkelde methode.

Het Pythonscript maakt gebruik van verschillende packages die de bewerking van data vereenvoudigen: Pandas, Numpy en Exl. Het Pythonscript wordt gelimiteerd door de gekozen KPI's die berekend worden door het script en de beschikbare inputdata die veel onbruikbare velden bevat. Het Pythonscript kan gebruikt worden bij de nulmeting, zoals in 3.2. Nulmeting, en toekomstige metingen die uitgevoerd worden op de WePublish-service.

1.2.6. Norm en realiteit

In deze sectie wordt verder ingegaan op het belangrijkste projectresultaat: oplevering van de methode. Om te kunnen stellen of de ontwikkelde methodiek het kernprobleem heeft opgelost, is het noodzakelijk om een norm en realiteit vast te stellen. De realiteit omschrijft de huidige situatie ten opzichte van de doelstelling (=norm).

Realiteit - Bij aanvang van dit onderzoek werd machine-learning software binnen het bedrijf op ongestructureerde wijze en incorrecte technieken ontworpen. Dit resulteert in ongewenste output en veel verloren uren.

Norm - In de probleemkluwen van het onderzoek is het kernprobleem vastgesteld: het bedrijf mist een stappenplan voor de ontwikkeling van machine-learning software. Het stappenplan dient begeleidend te zijn en toepasbaar op de WePublish-service. Het kernprobleem is opgenomen in de doelstelling en in de hoofdvraag van het onderzoek. De doelstelling en de beantwoording van de hoofdvraag vormt de norm van dit onderzoek.

Binnen 1.2.3. Doelstelling staat omschreven wanneer het doel (en daarmee de norm), volgens het bedrijf, behaald is.

1.3. Onderzoek ontwerp en verslagopbouw

In deze sectie staat de formulering van de hoofdvraag centraal. Hiervoor zal eerst de doelstelling van het onderzoek, in combinatie met haar afbakening, in kaart worden gebracht. Vervolgens wordt de probleemaanpak, met onder andere de onderzoeksmethodiek verwoord. Tot slot wordt de verslagopbouw geschetst waarin de geformuleerde deelvragen gelinkt worden aan de onderdelen van de onderzoeksmethodiek.

1.3.1. Onderzoek ontwerp

In de paragraaf wordt ten eerste een methodologie omschreven die gekozen is, ter begeleiding van dit onderzoek en haar opbouw. Vervolgens wordt verder ingegaan op het type wetenschappelijk onderzoek en de gekozen data-analyse technieken en de dataverzamelingmethodieken die gebruikt zijn. Vervolgens wordt een

Begeleidende methodologie: DSRM

In deze sectie wordt toegelicht waarom de Design Science Research Methodology (DSRM) gebruikt wordt voor het begeleiden van mijn onderzoek, en wat de voordelen van deze methodiek zijn ten opzichte van de meer gebruikelijke methodiek: de Algemene Bedrijfskunde Probleemaanpak (ABP). Binnen de opleiding Technische Bedrijfskunde staat de ABP centraal en wordt als standaard methodiek ingezet bij het uitvoeren van Bachelor afstudeeropdrachtonderzoeken.

De ABP, is een systematische aanpak om bedrijfskundige problemen tot een oplossing te brengen (Heerkens, 2012). De algemene vorm waarin de methodiek is opgesteld kan erg nuttig zijn, bijvoorbeeld bij het systematisch in kaart brengen van een complex probleem en het opstellen van bijbehorende oplossingen. De stappen in de methodiek bieden echter minimale ondersteuning in de ontwikkeling van een methode.

Vanwege de ontbrekende ondersteuning in de ontwikkeling van een methode is de ABP niet geschikt bevonden en is op aanraden van A. Aldea de DSRM onderzocht. De DSRM biedt een raamwerk voor het succesvol uitvoeren van Design Science (DS) onderzoek (K. Peffers, 2007). DS onderzoek wordt door Hevner, March en Park, in 2004, omschreven als het ontwikkelen en evalueren van IT artefacten die bedoeld zijn om organisatieproblemen op te lossen (A.R. Hevner, 2004). In het kort omvatten deze artefacten ieder te ontwerpen object met een oplossing voor een omvatbaar onderzoeksprobleem. Het ontwikkelen van een methodiek die het ontwerpen van nieuwe automatiseringssoftware omschrijft is waar de DSRM voor ontworpen is.

Het doel van dit onderzoek is de ontwikkeling van een methode, op basis van machine-learning componenten, waarin het ontwikkelingsproces van automatiseringssoftware staat beschreven. In DSRM termen, de ontwikkeling van een nieuwe IT artefact (de methode) die omschrijft hoe nieuwe IT artefacten (machine-learning software) ontworpen kunnen worden. In deze zin is de stagnering van de ontwikkeling van nieuwe IT-artefacten het organisatieprobleem waar GreenOrange mee kampt.

Om bovenstaande redenen is gekozen voor de DSRM-methodologie om dit onderzoek vorm te geven. In de laatste sectie van deze paragraaf, in de verslagopbouw, worden de stappen van de DSRM-methodologie gekoppeld aan de onderzoeksvragen en de hoofdstukken.

Onderzoeksmethodiek

In deze sectie wordt toegelicht welke wetenschappelijke methodieken zijn gebruikt bij de beantwoording van de onderzoeksvragen.

Er is gekozen om een toegepast, gemixte-methoden onderzoek uit te voeren om de opgestelde onderzoeksvragen uit de vorige paragraaf te beantwoorden. Uit het onderzoek komen conclusies en aanbevelingen naar voren die direct toepasbaar zijn in de praktijk. Ook zal de ontwikkelde methode direct in de praktijk te gebruiken zijn en mogelijk tot de ontwikkeling van nieuwe producten of services leiden.

Het kwalitatieve aspect komt naar voren in het beschrijvende karakter van het onderzoek, waarin de onderzoeksvragen beantwoord worden op basis van interpretaties en ervaringen en weergegeven worden in woordelijke beschrijvingen. Ook is de realiteit in dit onderzoek interpretatief en dynamisch, voornamelijk door sterke afhankelijkheid van omschrijvingen uit gevonden literatuur en gesprekken met medewerkers.

Het kwantitatieve aspecten komt naar voren bij het afnemen van gestructureerde, kwantitatieve enquêtes. De gekwantificeerde enquêtes die de ontwikkelde methodiek analyseren worden in dit onderzoek gebruikt om een kwalitatieve uitspraak te doen over de mening van de respondenten op verschillende aspecten van de methodiek. Daarnaast wordt de huidige situatie kwantificeerbaar uitgedrukt in de vorm van een nulmeting.

Data verzameling en data analyse

Zoals hierboven omschreven wordt in het onderzoek gebruikgemaakt van zowel kwalitatieve- als kwantitatieve data. In deze sectie wordt iedere dataverzamelmethode toegelicht en kort uitgelegd op welke manier de data van de dataverzamelmethode verzamelt wordt en op welke wijze de data geanalyseerd gaat worden.

Literatuuronderzoek (kwalitatief): Ter beantwoording van de kennisvragen, opgesteld in 1.2.4. Hoofd en deelvragen, wordt per kennisvraag een literatuuronderzoek uitgevoerd. De zoekopdrachten, in combinatie met de gekozen inclusie- en exclusiecriteria zijn te vinden in Bijlage D:

Literatuuronderzoek. De resultaten van het literatuuronderzoek worden geïnterpreteerd en gecontextualiseerd om vervolgens met gebruik van bronnen uit de literatuur een kwalitatief antwoord op iedere kennisvraag te formuleren.

Interviews (kwalitatief): Voor de beantwoording van de designvragen: “Hoe ziet het huidige proces achter de WePublish-service eruit?” en “Hoe ziet het ideale proces achter de WePublish-service eruit?” wordt voornamelijk gebruik gemaakt van open interviews. De data wordt met behulp van ongestructureerde gesprekken met medewerkers binnen de WePublish-service verkregen. Er is gekozen voor ongestructureerde gesprekken, omdat de informatie gedurende de werkzaamheden tussendoor is vergaard en omdat bepaalde medewerkers aan hebben gegeven geen behoefte hebben aan een gestructureerd interview. De medewerkers wordt tussen de werkzaamheden door gevraagd om het proces van de WePublish-service uit te leggen. De verzamelde data wordt geïnterpreteerd en in combinatie met het veldonderzoek gecontextualiseerd om een totaalbeeld te vormen van zowel de huidige WePublish-service, als de ideale WePublish-service in kaart gebracht.

Veldonderzoek (kwalitatief): Naast de interviews is voor de beantwoording van de designvragen omtrent het huidige proces en het ideale proces ook gebruik gemaakt van veldonderzoek. Voor het veldonderzoek is toegang verleend tot de processen binnen de WePublish-service en een testcasus genereerd, zodat de stappen van de processen persoonlijk doorlopen kunnen worden. De vergaarde data in de vorm van feitelijk te doorlopen stappen, en hun complicaties, worden naast de open

interviews gelegd om een gecontextualiseerd, visueel, totaalbeeld te vormen van zowel de huidige WePublish-service als de ideale WePublish-service.

Enquêtes (kwantitatief en kwalitatief): Bij de evaluatie van de ontwikkelde methode is gebruik gemaakt van gestructureerde enquêtes waarin op kwantitatieve wijze cijfers (tussen 1 en 5) worden vergaard voor verschillende onderdelen van de evaluatie. De enquêtes worden bij alle softwareontwikkelaars van GreenOrange afgenomen, dat het totaal aantal participanten op vijf brengt. Er worden geen extra selectiecriteria gebruikt. De kwantitatieve data wordt geanalyseerd op positieve- en negatieve uitschieters, om deze wegingen te interpreteren en te contextualiseren in combinatie met alle beschikbare kennis rondom GreenOrange en de ontwikkelde methode. Het resultaat is een kwalitatieve omschrijving van de resultaten uit de enquête.

Cijfer analyseren (kwantitatief): Data omtrent interacties binnen de WePublish-service (publicatie_ID, interactie_ID en moment van creatie) tussen 01-01-2019 en 01-03-2019 wordt verzameld. De data wordt gebruikt om een bepaalde uitspraken te doen over de prestaties van de service in de hierboven omschreven tijdsperiode, die geïnterpreteerd wordt als de huidige situatie.

Onderzoekskwaliteit

In deze sectie wordt de onderzoekskwaliteit middels omschrijving van de betrouwbaarheid, validiteit en de limitaties van de gekozen onderzoeksmethodiek en dataverwerkingsmethoden omschreven.

Betrouwbaarheid

In deze sectie wordt eerst ingegaan op de algemene betrouwbaarheid van het onderzoek, om vervolgens de verschillende dataverzamelmethodeken op grond van betrouwbaarheid, op kwalitatieve wijze te bespreken. De betrouwbaarheid van het onderzoek heeft te maken met de controleerbaarheid en de repliceerbaarheid van het gehele onderzoek. Om de betrouwbaarheid te waarborgen is het van belang dat er expliciete informatie wordt verschaft over de gebruikte methoden.

De gebruikte methodologieën voor de uitvoering van dit onderzoek zijn hierboven uitvoerig omschreven. Deze methodologieën worden in de verslagopbouw gekoppeld aan de opgestelde onderzoeksvragen en de implementatie binnen dit onderzoek. Hierdoor wordt het onderzoek stap voor stap toegelicht en wordt de herhaalbaarheid van het onderzoek vergroot. Een expliciet gedocumenteerd en gedetailleerd audit-trail is niet bijgehouden, wél is tijdens iedere stap duidelijk en uitgebreid uitgelegd op welke wijze de resultaten zijn verkregen, wat een andere onderzoeker in staat stelt hetzelfde onderzoek nogmaals uit te voeren (met vergelijkbare resultaten).

Het literatuuronderzoek kan met behulp van Bijlage D: Literatuuronderzoek erg betrouwbaar opnieuw worden uitgevoerd. De externe betrouwbaarheid kan echter altijd beïnvloed worden door de interpretatie van bronnen: er valt niet te garanderen dat door andere een onderzoeker een andere interpretatie wordt gedaan. Desondanks wordt een verkeerde interpretatie door zorgvuldig verwijzen van bronnen en het veelvuldig gebruik van wiskundige notaties binnen de gevonden literaire bronnen geminimaliseerd.

De interviews en het veldonderzoek zijn minimaal omschreven in dit onderzoek. Naar verwachting zal dit geen invloed hebben op de betrouwbaarheid van de resultaten uit deze dataverzamelmethodeken, aangezien er weinig variatie is in de mogelijke antwoorden van de geïnterviewde medewerkers en geen variatie is in de uitvoering van het veldonderzoek.

De vragen uit de enquêtes zijn, per evaluatie-onderdeel, arbitrair gekozen uit een set van door Venkatesh opgestelde vragen voor evaluaties van systemen (V. Venkatesh, 2003). Het is mogelijk dat bij herhaling van het onderzoek gekozen wordt voor andere vragen. Het resultaat van de evaluatie zal echter een zeer vergelijkbaar en daarmee een betrouwbaar antwoord zijn.

De geanalyseerde cijfers voor onder andere de nulmeting zijn door middel van opgestelde software exact opnieuw uit te voeren en daarmee uitermate betrouwbaar.

Validiteit

In deze sectie wordt ingegaan op de validiteit van het onderzoek om vervolgens de verschillende dataverzamelingsmethodieken op kwalitatieve wijze op grond van validiteit op kwalitatieve wijze te bespreken. De validiteit heeft betrekking op de vraag of de resultaten uit het onderzoek overeenkomen met de werkelijkheid (Swaen, 2017).

In dit onderzoek wordt de combinatie van verschillende databronnen en theorieën vanuit het literatuuronderzoek dataverzamelingsgeldigheid gewaarborgd.

De open interview zijn zonder weten van de respondent uitgevoerd in een voor de respondenten vertrouwde omgeving, de werkvloer, om de ecologische validiteit van de interviews te verbeteren. Bij de open interviews is aangenomen dat de medewerkers niet doelbewust verkeerde informatie geven.

De validiteit in de evaluatie enquête wordt gewaarborgd door gesloten vragen te stellen die vergelijkbaar, ten opzichte van andere respondenten, gescoord worden. Bovendien worden de vragen individueel en unaniem ingevuld, waardoor de respondent tijdens het invullen niet beïnvloed kunnen worden door anderen. De respondenten hebben de vrijheid gekregen om een naar eigen keuze geschikt moment binnen 48 uur de enquête in te vullen.

De cijferanalyse kan door middel van een statistische analyse gecontroleerd worden en daarmee gevalideerd worden.

Limitaties

In de 1.2.5. Afbakening onderzoek is het onderzoek ingeperkt. In deze sectie wordt ingegaan op de limitaties van de onderzoeksmethodologieën die in dit onderzoek gebruikt worden. Allereerst worden de limitaties van een kwalitatief onderzoek, in combinatie met de limitaties van een toegepast onderzoek belicht. Vervolgens wordt ingegaan op de limitaties van het DSRM en wordt afgesloten met de limitaties van het kwantitatieve aspect van het onderzoek.

De beantwoording van de hoofdvraag uit dit onderzoek is voor een groot deel afhankelijk van het kwalitatieve onderzoek, met name het literatuuronderzoek. Alle verzamelde data wordt door de onderzoeker geïnterpreteerd en verwerkt in dit onderzoek. Deze is subjectief, omdat de onderzoeker de data anders kan interpreteren en daarmee het onderzoek kan beïnvloeden. Naast de interpretatie van de data uit het literatuuronderzoek is de data ook gelimiteerd tot de gekozen zoektermen, inclusiecriteria en exclusiecriteria. Concluderend wordt gesteld dat het ontwerp van de methode en de conclusies en aanbevelingen voor een groot deel onderzoeker alleen gevormd is. Relatief weinig controle en afstemming vanuit het bedrijf was mogelijk door beperkte kennis van het onderwerp binnen het bedrijf.

De toegepaste component van een onderzoek maakt een onderzoek vaak lastig generaliseerbaar, omdat het onderzoek een specifiek probleem bij het bedrijf aanpakt. In dit onderzoek is echter bij de ontwikkeling van de methode rekening gehouden met de wens van GreenOrange om de methode toe te passen op andere object detection problemen en is daarmee gericht op schaalbaarheid. Het

onderzoek is sterk gericht op het oplossen van de vraag van het bedrijf en minder gericht op het creëren van nieuwe wetenschappelijke inzichten.

Vanwege het ontwikkelende karakter van het onderzoek, met behulp van de DSRM, dient de opdrachtgever het te ontwikkelen artefact uitvoerig te omschrijven. Het DSRM claimt niet dé DS methodologie te zijn, maar een. De methode is bedoeld als handleiding en veroordeelt afwijking van de ontwikkelde stappen niet. Het DSRM is ontwikkeld om onderzoek te ondersteunen, maar kan in theorie ook voor praktische artefact ontwikkelingen gebruikt worden.

In het geval van dit onderzoek beschikt de opdrachtgever over gelimiteerde kennis over het onderwerp, waardoor de omschrijving van de deliverable minimaal was en tevens sterk voor interpretatie van de onderzoeker vatbaar. Het onderzoek mist hierdoor gedetailleerde sturing in de ontwikkeling van de methodiek. Daarnaast zal de ontwikkelde methodiek niet allesomvattend zijn en wordt van de gebruiker verwacht om de methode te vertalen naar het nieuwe probleem.

De steekproef voor de nulmeting, van 01-01-19 tot 03-01-19, hoeft niet representatief te zijn voor alle maanden van het jaar. Daarnaast is de data uit de database niet specifiek, er is bijvoorbeeld niet duidelijk waarom een bepaalde interactie lang, of kort duurt.

1.3.2. Verslagopbouw

In deze paragraaf wordt ingegaan op de opbouw van de verschillende hoofdstukken uit dit onderzoek. Hiervoor wordt als eerste iedere fase van het DSRM uitgelegd en geïnterpreteerd. De verschillende stappen binnen de DSRM worden in Figuur 3 weergegeven. Vervolgens zal minimaal één hoofdstuk worden toegekend aan iedere fase uit het DSRM. De verdeling van DSRM fases ten opzichte van de hoofdstukken is weergegeven in Tabel 1.

De verschillen fasen uit het DSRM zijn door onder andere Hevner als volgt uit gelegd (A.R. Hevner, 2004):

1. Binnen de probleem identificatiefase is het belangrijk dat het onderzoeksprobleem gedefinieerd wordt en de relevantie van de oplossing aangetoond wordt.

De probleem identificatie fase is deels uitgevoerd. In de voorgaande paragrafen is het probleem, in combinatie met de aanleiding van het probleem, geïdentificeerd en de relevantie toegelicht. Als uitbreiding op de probleem identificatie zal 3. Huidige situatie meer specifiek ingaan op de technische componenten die een rol spelen in de WePublish-service. Het is van belang om de huidige software en hardware integraties gedetailleerd en overzichtelijk in kaart te brengen, om op basis hiervan met behulp van een nieuwe methode toe te werken naar de gewenste situatie.

2. In deze fase worden de oplossingsdoelstellingen vastgesteld. Dit zijn de vereisten waar het doel van het onderzoek, het creëren van een artefact, aan moet voldoen.

De oplossingsdoelstelling formuleringsfase start volgens de DSRM-methode met een onderzoek naar bestaande kennis over mogelijke oplossingen voor het geformuleerde probleem. In dit onderzoek zal het theoretische kader, dat verder in deze paragraaf beschreven is, dit onderdeel van de oplossingsdoelstelling vormen. Daarnaast wordt tijdens deze fase ook de gewenste situatie geschetst en worden mogelijke oplossingen vergeleken. In 4. Gewenste situatie zal de gewenste situatie omtrent de WePublish-service verwoord worden.

3. In de ontwerp- en ontwikkelingsfase wordt het artefact ontworpen met inachtneming van de gestelde eisen in fase 2.

Tijdens de ontwerp- en ontwikkelingsfase wordt de methode, zoals beschreven in de doelstelling en de hoofdvraag, ontwikkeld. H5. Ontwerp- en ontwikkelingsfase omvat de volledige methode. De

gewenste situatie, zoals omschreven in fase 2, zal als leidraad gelden voor de ontwikkeling van de methode.

4. In de demonstratiefase wordt de bruikbaarheid van het ontwikkelde artefact aangetoond door deze toe te passen op een geschikte activiteit, zoals een casestudy of een simulatie.

De methode die in fase 3 ontwikkeld wordt zal in de demonstratiefase getoetst worden op een casestudy in combinatie met een user test. De demonstratiefase zal in 6. Demonstratiefase – Proof of concept worden samengevoegd met de ontwikkelde methode.

5. Tijdens de evaluatiefase wordt geobserveerd en meetbaar gemaakt hoe goed het ontwikkelde artefact de oplossing voor het probleem ondersteund.

Hoofdstuk 7. Evaluatie zal zich toespitsen op de evaluatiefase.

6. De communicatiefase richt zich op het verspreiden van de opgedane kennis naar onderzoekers en andere belanghebbenden, zoals andere professionele beroepsbeoefenaars.

De communicatiefase wordt in dit onderzoek uitgevoerd door middel van het scriptieverslag, het colloquium, de persoonlijke gesprekken met medewerkers en de eindpresentatie bij GreenOrange.



Figuur 3: DSRM proces.

Tabel 1: Overzicht van hoofdstukken, in combinatie met bijbehorende deelvragen, per DSRM-stap.

DSRM	Hoofdstuk	Deelvragen
Probleem identificatie	1. Introductie 2. Theoretisch kader 3. Het huidige proces	- Welke kennis omtrent de beschreven doelstelling bestaat al? Hoe ziet het huidige proces achter de WePublish-service eruit?
Oplossingsdoelstellingen	4. Gewenste proces	Hoe ziet het ideale proces achter de WePublish-service eruit?
Ontwerp en ontwikkeling	5. Ontwikkelde methode	Hoe ziet een methode, op basis van machine-learning componenten, waarin het ontwikkelingsproces voor software automatisering staat beschreven eruit?
Demonstratie		Hoe kan de ontwikkelde methode worden toegepast bij Green Orange?
Evaluatie	6. Evaluatie	
Communicatie	7. Conclusie en aanbevelingen	

2. Theoretisch kader

In dit hoofdstuk wordt een theoretisch kader gegeven rondom de kennisvragen die opgesteld zijn in de 1.2.4. Hoofd en deelvragen. Allereerst wordt ingegaan op de verschillende componenten van machine learning en vastgesteld welke componenten voor het omschreven probleem van toepassing zijn (3.1.). In paragraaf 3.2. volgt een overzicht van methoden die ondersteuning bieden bij het opstellen van een methode. Vervolgens worden een aantal methoden en technieken voor de ontwikkeling van software omschreven (3.3.). Tot slot worden in paragraaf 3.4 de belangrijkste bevindingen samengevat en worden antwoorden gekoppeld aan de kennisvragen.

2.1. Machine learning

In deze paragraaf wordt antwoord gegeven op de eerste kennisvraag door allereerst aandacht te geven aan verschillende vakgebieden die vaak door elkaar gebruikt worden: artificiële intelligentie (AI), machine learning (ML) en Deep learning (DL). Ook worden veel voorkomende termen, zoals neuraal netwerk (NN), computer vision (CV), image recognition (IR), object classification (OC), en object detection (OD) gedefinieerd.

Na opheldering van de verschillende vakgebieden en bijbehorende termen wordt dieper ingegaan op het vakgebied DL en het verband dat DL heeft met ObjectDetection. Tot slot worden bovenstaande begrippen gebruikt om Convolutionele Neurale Netwerken (CNN) toe te lichten.

2.1.1. Vakgebieden

In deze sectie wordt onderscheid gemaakt tussen drie vakgebieden. Uit onderzoek van Medium, NVIDIA, TowardsDataScience en DataScienceCentral zijn verschillende definities geëxtraheerd en samengevoegd tot drie onderscheiden vakgebieden ((Bahmani, 2018), (Copeland, 2016), (Garbade, 2018) en (Venkatesan, 2018)).

AI is een breed begrip dat geïnterpreteerd kan worden als het inlijven van menselijke intelligentie bij machines op iedere mogelijke manier. Binnen AI wordt onderscheid gemaakt tussen twee concepten: general AI en narrow AI. General AI wordt gezien als een machine die precies voelt en denkt als een mens. Vooralsnog is deze vorm van AI niet mogelijk. Narrow AI wordt gezien als een machine die een specifieke, complexe, taak op menselijk niveau, of beter, kan uitvoeren.

ML is een subset van narrow AI. ML stelt computers in staat om de input en output uit een dataset te verwerken en hier zelf uit te leren, door algoritmes aan te passen, naarmate meer informatie verwerkt wordt. ML stelt computers in staat om algoritmes op te stellen, waarmee een nieuw datapunt voorspelt kan worden. Het algoritme dat gebruikt wordt bij ML is te interpreteren als een NN van één laag neuronen. In 2.1.2. Neurale netwerken meer hierover.

DL is een subset van ML dat gebruikt maakt een NN met meerdere lagen. Door deze lagen wordt het algoritme dat een DL kan leren significant complexer. Hierdoor is een DL netwerk (over het algemeen) in staat om betere resultaten te genereren dan een ML algoritme bij dezelfde taken en is het in staat complexere taken succesvol uit te voeren. Binnen DL wordt onderscheid gemaakt tussen twee concepten: supervised learning en unsupervised learning. Met supervised learning zoekt de computer naar een relatie tussen de input en (bekende) output, waar unsupervised learning zoekt naar een relatie in data, zonder gedefinieerde output. In dit onderzoek wordt slechts gesproken over supervised learning.

2.1.2. Neurale netwerken

In deze sectie wordt antwoord gegeven op de eerste kennisvraag: Welke componenten van machine learning zijn van toepassing op het omschreven probleem? Zoals omschreven in de vorige sectie berust ML op het gebruik van NN'en. Voor de beantwoording van de vraag wordt dan ook allereerst onderzocht en uitgelegd welke verschillende termen nodig zijn om de werking van een NN uit te leggen. Ook zal de werking van een NN wiskundig worden toegelicht, zodat gebruikers van de methode inzicht krijgen in de bewerkingen die in een NN plaatsvinden. De sectie wordt afgesloten met één voorbeeld dat op drie verschillende NN'en, met verschillende complexiteit, wordt toegepast.

Wanneer in dit onderzoek gesproken wordt over een NN, dan wordt gesproken over een kunstmatig NN dat is afgeleid uit de manier waarop hersenen werken. Hersenen zijn biologische NN'en. Een kunstmatig NN is gebaseerd op een verzameling van verbonden eenheden genaamd kunstmatige neuronen. Iedere verbinding tussen neuronen kan een signaal overbrengen naar een andere neuron. De neuron die een signaal ontvangt kan het signaal verwerken en vervolgens een nieuw signaal versturen naar verbonden neuronen die zich verder in de keten bevinden. Pitts was de eerste, in 1943, die de communicatie tussen neuronen vertaalde naar wiskundige vergelijkingen (Pitts, 1943), waarna Rosenblatt in 1957 het eerste neurale netwerk introduceerde (Rosenblatt, 1957). Nielson en Gad bouwen voort op het pionierende werk van Pitts en Rosenblatt en stelt dat aan ieder signaal dat van neuron naar neuron verstuurd wordt een weight (gewicht) hangt en dat iedere ontvangende neuron beschikt over een bias (een constante) (Nielsen, 2013), (Gad, 2018). De weight bepaalt het belang van het ontvangen signaal en de bias wordt gebruikt om het model vrijheid te geven om zich zo goed mogelijk aan te passen naar de data. De bias kan het best geïnterpreteerd worden als een maatstaf die aangeeft hoe makkelijk het is om de betreffende neuron te activeren. Wanneer een NN nog niet getraind is begint het met enkel random weights en biases. De inputdata wordt door deze random parameters verwerkt tot voorspelde outputdata. De voorspelde outputdata wordt vergeleken met de werkelijke outputdata.

Voorbeeld: wanneer invoer = 2, dan dient de output = 8 te zijn. 8 is in dit geval de werkelijke outputdata. Door de random parameters is de voorspelde output echter niet 8, maar 3. Er is sprake van een foutmarge.

Training van een NN

Back-propagation, gradiënt descent en foutmarge (loss function)

In dit kopje wordt uitgelegd op welke manier een NN 'leert' (traint). Hiervoor wordt ingegaan op de hierboven beschreven foutmarge die ontstaat door de random waarden van de weights and biases. Allereerst wordt ingegaan op het back-propagation algoritme dat gebruik maakt van gradiënt descent en de loss function (foutmarge) om het algoritme van het NN te verbeteren, zodat het NN de input naar een gewenste output vertaalt.

In 1982 is Werbos één van de eerste die het back-propagation algoritme toepast op het trainen van NN'en (Werbos, 1982). Na verschillende onderzoeken die inzicht verschaffen in de optimale werking van backpropagation in NN'en is Le Cun et al. in 1989 de eerste die het back-propagation algoritme weet toe te passen binnen CNN's, waarover in 2.1.4. Computer vision meer (Y. Le Cun et al., 1989).

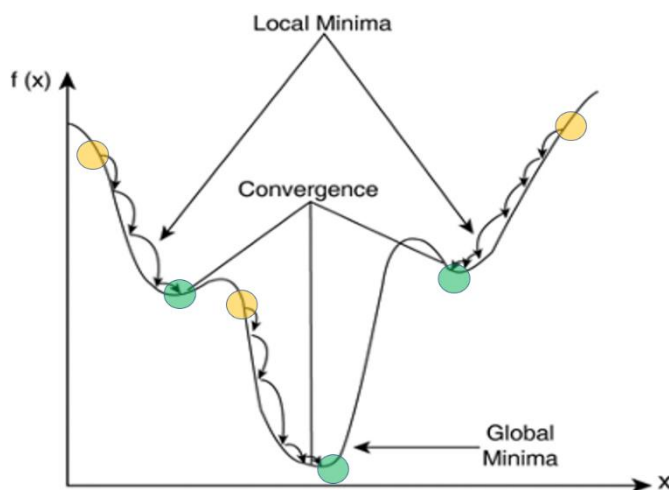
Onderstaande uitleg omtrent back-propagation wordt onderbouwd door Werbos, Le Cun, Nielson en Gad (Werbos, 1982), (Y. Le Cun et al., 1989), (Nielsen, 2013), (Gad, 2018). Het back-propagation algoritme ken drie stappen. 1.) Ten eerste wordt de foutmarge tussen de voorspelde outputdata en

de werkelijke outputdata, per weight en bias. 2.) Vervolgens wordt de afgeleide van de foutmarge bepaald: de gradiënt, per weight en bias. Al deze gradiënten geven inzicht in hoeverre iedere weight en bias bijdraagt aan de foutmarge van het totale netwerk. 3.) Tot slot wordt iedere weight en bias in de juiste richting 'gedraaid'.

Het aanpassen van de weights en biases wordt gradiënt descent genoemd. In Figuur 4 wordt de foutmarge van een functie van een simplistisch NN in 2D weergegeven: op de x-as iedere mogelijk combinatie van parameters en op de y-as de foutmarge van de voorspelde output ten opzichte van de werkelijke output. Met behulp van back-propagation wordt met de input van iedere trainingsdata een stapje gezet naar een lokaal minimum, of het globale minimum: in Figuur 4 te zien als één pijlstep.

Doordat een NN met random parameters begint bestaat de kans dat de parameters convergeren naar een lokaal minimum dat niet in de buurt ligt van het globale minimum. Om dit te voorkomen dient een NN vaker getraind te worden. Van de getrainde netwerken heeft het NN met het laagste foutmarge het laagste lokale minimum, of globale minimum gevonden. In Figuur 4 begint een training bij een gele markering en eindigt bij een groene markering.

Wanneer het globale minimum (of een vergelijkbare waarde in de vorm van een lokaal minimum) is bereikt, is het NN getraind en is het klaar om nieuwe inputdata te verwerken en voorspellingen over de outputdata te doen.



Figuur 4: Gradiënt descent visualisatie

Wiskundige notatie en voorbeelden

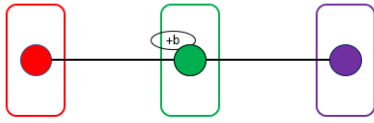
In deze sectie wordt de wiskundige notatie bij het hierboven beschreven proces gevoegd en wordt met behulp van drie NN'en toegewerkt naar een beter begrip van ML en haar subset DL. In onderstaande uitleg wordt gebruik gemaakt van onderzoeken en omschrijvingen van Le Cun, Nielson en Bishop (Y. Le Cun et al., 1989), (Nielsen, 2013), (Bishop, 2006).

Het besluitproces van een NN kan, zoals omschreven in Vergelijking 1, wiskundig worden genoteerd. Hier is w het gewicht van neuron x , waarin $w > 0$. De b is de bias, die gedefinieerd wordt als $b = -$ threshold.

Vergelijking 1: Vereiste voor vuren neuron

$$\text{Output} = \begin{cases} 0 & \text{if } w * x + b \geq 0 \\ 1 & \text{if } w * x + b < 0 \end{cases}$$

Situatie 1.



Figuur 5: Situatie 1: NN met 1 laag, met 1 neuron

Stel, we willen het NN uit Figuur 5 trainen om graden Celsius (x) te converteren naar graden Fahrenheit (y). In Vergelijking 2 staat de formule die de output van het NN omschrijft. In Tabel 2 staan 4 meetwaarden die gebruikt kunnen worden om het NN te trainen, zodat we het kunnen gebruiken om te voorspellen welke waarde correspondeert met 35 en 180.

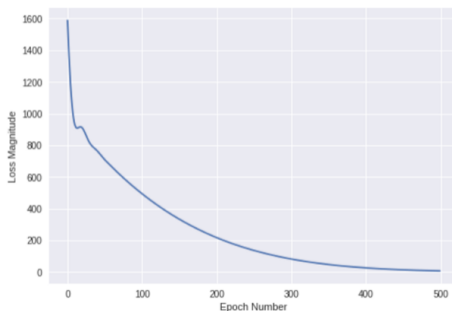
Vergelijking 2: Weights en Biases, situatie 1

$$y = w * x + b$$

Tabel 2: Meetwaarden, Celsius naar Fahrenheit

Input (x) in Celsius	0.0	8.0	15.0	22.0	35	180
Output (y) in Fahrenheit	32	46.4	59	71.6	?	?

Met behulp van back-propagation en gradiënt descent wordt de waarde van w en b uit Vergelijking 2 aangepast, totdat het foutmarge acceptabel is. In Figuur 6 is een dergelijk trainingsproces geplot. Hierin is de y -as een waarde voor de foutmarge en staat epoch voor het aantal iteraties dat is uitgevoerd op de 4 meetwaarden. In Vergelijking 3 wordt de voorspelde output vergeleken met de correcte output.



Figuur 6: Trainingsproces, Celsius naar Fahrenheit

Vergelijking 3: Situatie1, voorspelde output vs correcte output

Zoals te zien in Figuur 6 is na 500 iteraties de foutmarge ≈ 0 .

$$\begin{aligned} w &= 1.824 \\ b &= 29.892 \end{aligned}$$

Bij het invoeren van $x = 35$ en $x = 180$ geeft het NN als output

$$\begin{aligned} x = 35, \quad y &= 35 * 1.824 + 29.892 = 93.618 \\ x = 180 \quad y &= 180 * 1.824 + 29.892 = 358.212 \end{aligned}$$

De formule om Celsius om te rekenen naar Fahrenheit is gegeven:

$$F = 1.8C + 32$$

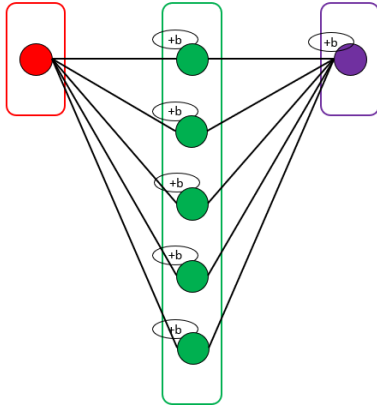
Invoer van dezelfde waardes geeft:

$$C = 35, F = 35 * 1.8 + 32 = 95$$

$$C = 180, F = 180 * 1.8 + 32 = 356$$

We kunnen concluderen dat ons minimalistische NN een formule heeft gevonden die met een kleine afwijking voorspellende uitspraken kan doen over de conversie van Celsius naar Fahrenheit.

Situatie 2.



Figuur 7: Situatie2, NN met 1 laag, met 5 neuronen

Stel dat we hetzelfde probleem, graden Celsius naar graden Fahrenheit converteren, op willen lossen met het NN uit Figuur 7.

In deze situatie zijn 5 neuronen aangeduid met x_i , waarin $i \in (1,2,3,4,5)$ De output van iedere neuron valt op dezelfde wiskundige wijze te omschrijven als in Vergelijking 2. In deze situatie wordt echter een sommatie gebruikt voor nette notatie.

Vergelijking 4: Weights en Biases, meerdere input

$$\text{Output} = \begin{cases} 0 & \text{if } \sum_i w_i x_i + b \leq 0 \\ 1 & \text{if } \sum_i w_i x_i + b > 0 \end{cases}$$

Bij training worden aan 10 weights en 6 biases gedraaid. In totaal komt dit neer op 16 parameters. In Tabel 3 zijn de weights en biases na een training weergegeven. Zoals te zien is aan de weights en biases is niks meer af te leiden. Daarom worden NN'en ook wel een black box genoemd.

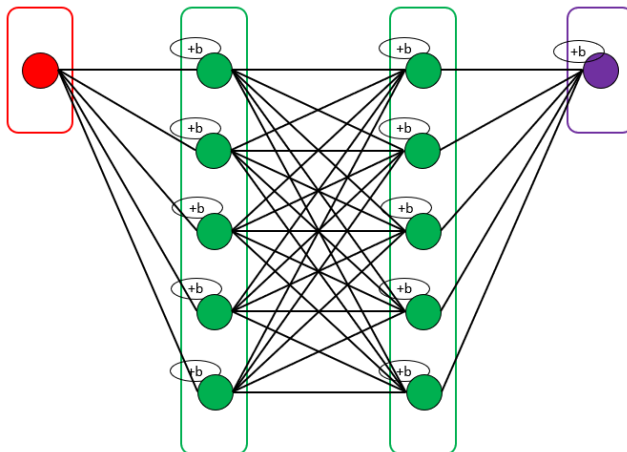
Bij invoer van 35 en 180 is de output respectievelijk 94.88 (exacte waarde = 95) en 355.58 (exacte waarde = 356). We zien dat deze uitvoer nauwkeuriger is dan ons vorige NN. Dit komt omdat het NN uit situatie 2 veel gevoeliger kan bijstellen.

Tabel 3: Parameters situatie2.

Weights (1)	Biases (1)
-0.44988054	3.3572698
0.2192589	-3.3885067
-0.8051902	-5.140557
0.5252715	5.0017242
-0.42604864	-5.1918674
Weights (2)	Bias (2)

1.117872	4.683527
-0.50522655	
-1.0478513	
1.9218719	
-1.3106703	

Situatie 3.



Figuur 8: Situatie3, NN met 2 lagen en 5 neuronen per laag

Stel dat we met hetzelfde voorbeeld nóg een stap verder gaan. In Figuur 8 is een NN weergegeven met twee lagen, waardoor we nu van een DL NN spreken.

De output van iedere neuron valt hier op dezelfde manier te omschrijven als in Vergelijking 4. Bij training wordt echter aan significant meer parameters gedraaid. Namelijk: $5 + 5 * 5 + 5 = 35$ weights en $5 + 5 + 1 = 11$ biases. In totaal komt dit neer op $35 + 11 = 46$ parameters ten opzichte van 16 parameters in situatie2.

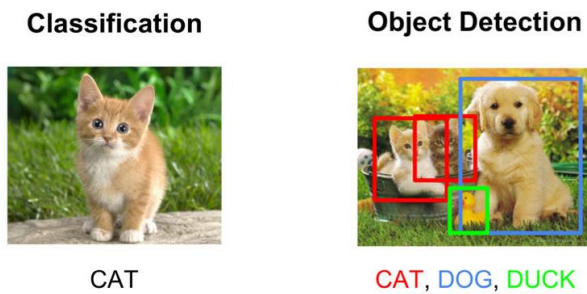
Bij invoer van 35 en 180 is de output respectievelijk 94.88 en 355.58. We zien dat een dieper NN niet altijd leidt tot een nauwkeuriger uitkomst.

2.1.3. Computer vision

In de vorige sectie is duidelijk geworden hoe een NN werkt. In deze sectie wordt gefocust op de manier waarop een NN een image (afbeelding) verwerkt. Hiervoor wordt eerst inzicht gegeven in het vakgebied dat zich hierop focust. Vervolgens wordt ingegaan op drie typische problemen binnen CV. Tot slot wordt de link gelegd naar NN dat speciaal ontwikkeld is voor de verwerking van images. Lecun heeft naast zijn pionierende werk in 1989, in 2015 een review over DL geschreven. De sectie wordt ondersteund door de review van LeCun en de boeken van Nielsen en Gad (Y.L. LeCun, 1998), (Nielsen, 2013), (Gad, 2018).

CV is een interdisciplinair wetenschappelijk vakgebied dat zich focust op het mogelijk maken van computers om betekenisvolle beschrijvingen bij digitale afbeeldingen of video's te plaatsen, zoals wij mensen dat kunnen. Zoals Ballard in de Preface van zijn boek omschrijft kan CV gezien worden als een subset van AI (D. H. Ballard, 1982). CV is echter ook een subset van DL.

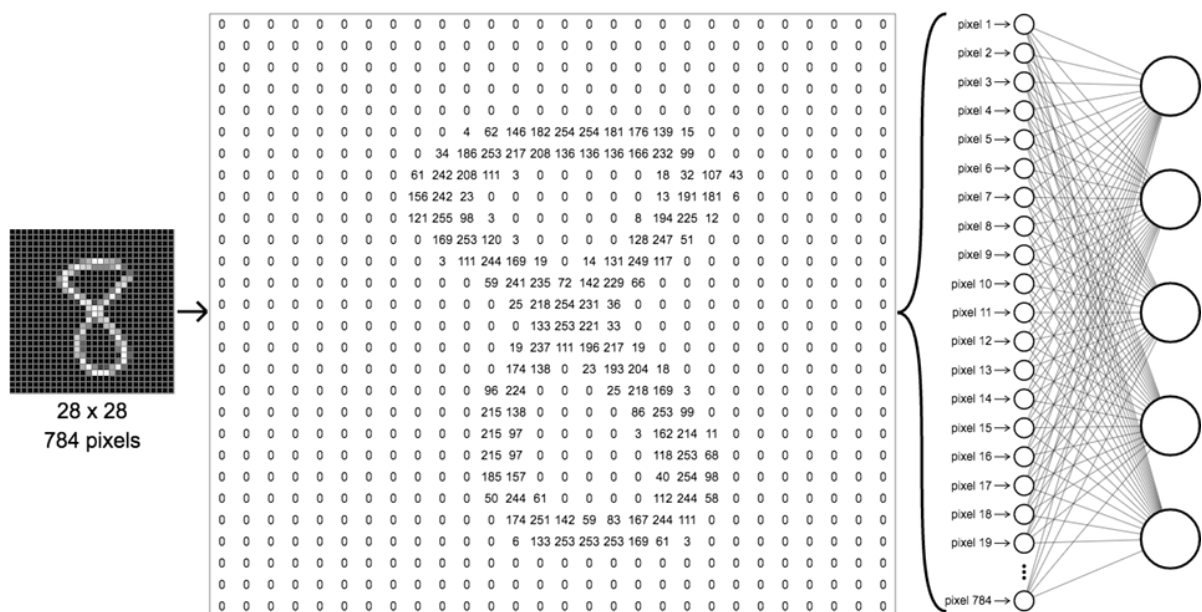
Typische problemen binnen CV zijn: object classification (object classificatie), object detection (object detectering) en object identification (object identificatie). Object classification herkent één klasse uit een image (afbeelding), object detection herkent meerdere klassen binnen één image (Figuur 9). Object identification focust zich op het herkennen van een specifieke instantie binnen een klasse, bijvoorbeeld alleen de kat van de burens. Producten herkennen en labelen in een image (pdf folder) is een object detection probleem. Om deze problemen beter te begrijpen wordt in de volgende paragraaf ingegaan op de manier waarop een computer images interpreteert.



Figuur 9: Object Classification vs Object Detection

Images in een NN

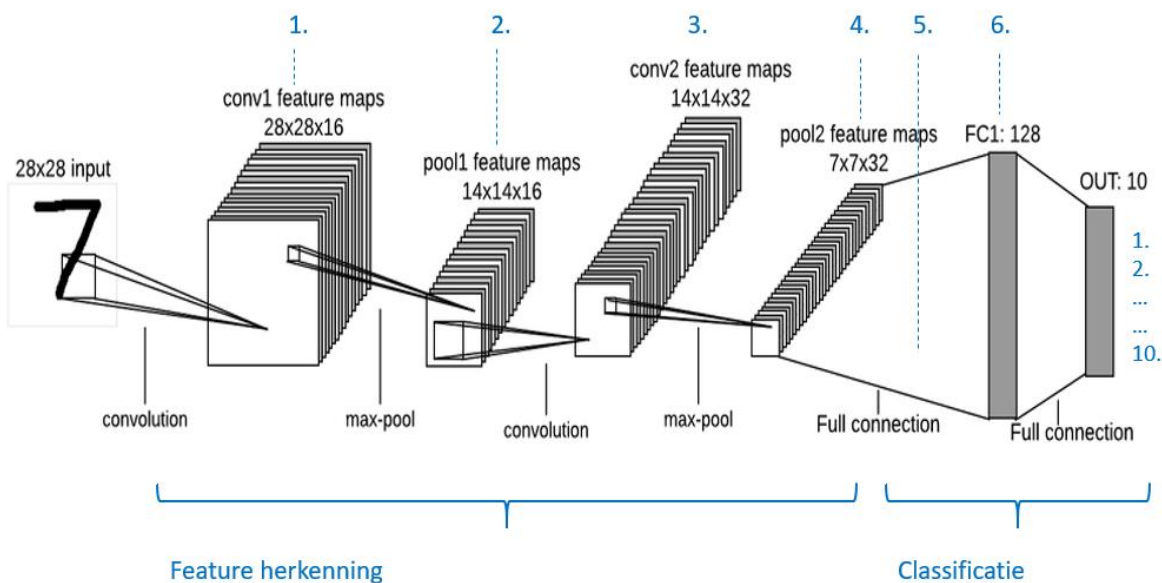
In Figuur 10 is visueel weergegeven op welke manier een computer een images interpreteert. De input is een 28x28 pixels zwart/wit afbeelding van het getal "8". De computer ziet deze afbeelding als een matrix van pixelwaarden tussen 0 en 256, waarin 0 gelijk is aan zwart en 256 gelijk is aan wit. In totaal bestaat de input van het NN uit (28x28x1) 764 pixelwaarden die in een 764x1 matrix worden ingevoerd. Terugkomend op de uitleg in de vorige paragraaf wordt aan iedere pixelwaarde een weight gehangen en op de volgende laag neuronen afgevuurd. Uiteindelijk vormt de combinatie van geactiveerde neuronen en niet geactiveerde neuronen in de laatste neuronen laag de output van het NN (in dit geval een getal tussen de 0 en de 9).



Figuur 10: NN interpretatie van image

Een CNN

Patroon herkenningproblemen, zo ook object detection problemen, vormden vanaf de vroege jaren 90' een grote uitdaging voor wetenschappers. Veelal werd de herkenning van features (kenmerken) deterministisch geprogrammeerd. Vervolgens werd de herkenning van die features als input voor een eenvoudig NN gebruikt om een voorspellende output te genereren. De patroonherkenningssoftware voor object detection in images scoorde die tijd laag in accuraatheid en was sterk afhankelijk van de creativiteit van de programmeur om met een goede set features te komen. Met de eerste introductie van back-propagation, toegepast op CNN's, werden ongekend accurate resultaten gescoord, op het gebied van object detection en kwam het onderzoeksveld in een stroomversnelling die vandaag de dag nog gaande is (Y.L. LeCun, 1998).



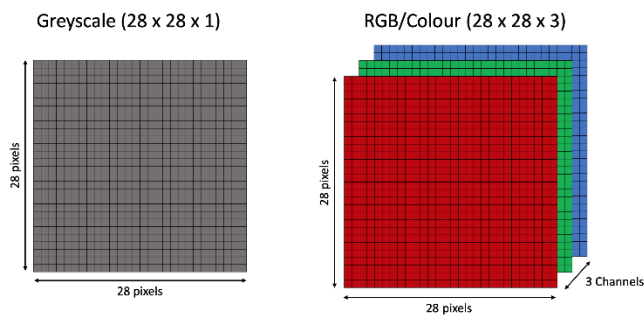
Figuur 11: CNN verwerking van image

Een CNN is een speciaal type DL NN dat bestaat uit twee onderdelen: 1.) feature herkenning en 2.) classificatie van de set features die in een image gevonden is (Figuur 11). In het feature herkenning gedeelte komen drie bewerkingen aan bod: a.) convolutie, b.) normalisering en c.) pooling. Eén opeenvolging van deze drie bewerkingen noemen we een convolutie laag. De output van het feature herkenning gedeelte is een set featuremaps. Deze featuremaps worden op vergelijkbare wijze verwerkt door het classificatie gedeelte als in Figuur 10.

1. De input wordt in dit voorbeeld gecontroleerd op 16 verschillende features. De scoring van iedere feature wordt opgeslagen in een featuremap (een hittemap van een bepaalde vorm). Er ontstaan 16 images (28x28x16 in plaats van 28x28x1) die weergeven hoe sterk een bepaalde feature in de inputimage voorkomen. Bij iedere convolutiebewerking wordt een normaliseringsfunctie toegepast om de wiskunde binnen het netwerk beheersbaar te houden. Bijvoorbeeld de ReLu functie maakt van iedere negatieve waarde in een featuremap een 0.
2. De (max-)pooling bewerking verkleint de feature images door alleen relevante pixels te selecteren. In dit geval de hoogste pixelwaardes binnen een klein groepje pixels.

3. De tweede convolutiebewerking controleert iedere verkleinde feature image, in dit voorbeeld, op twee andere features. Het totaal aantal featuremaps wordt hiermee verdubbeld naar 32.
4. De pooling bewerking verkleint de feature images, zoals omschreven.
5. Alle featuremaps worden omgezet naar een $(7 \times 7 \times 32) = 1568 \times 1$ matrix en geïnterpreteerd door de computer, zoals beschreven in Figuur 10.
6. Iedere pixelwaarde in deze matrix krijgt, zoals omschreven in de drie situaties, een weight waarnaar een output waarde wordt voorspeld.

Wanneer een input image bestaat uit kleur blijven de bewerking hetzelfde. Het enige verschil is dat input nu uit een diepte van drie lagen bestaat en daarmee alle bewerkingen ook uit drie lagen dienen te bestaan (Figuur 12). Het verwerken van een RGB image, ten opzichte van een zwart/wit image kost dan ook significant meer rekenkracht.



Figuur 12: Zwart/Wit vs Kleur (RGB)

2.1.4. Limitaties en gevaren van neurale netwerken

Neurale netwerken, en met name DeepLearning netwerken, zijn krachtig gereedschap die toepasbaar zijn binnen veel verschillen toepassingen. Eén van deze toepassingen, binnen object detection neurale netwerken, is het netwerk ontwikkeld in dit onderzoek. In deze sectie worden echter niet de mogelijkheden van neurale netwerken op een rijtje gezet, maar de limitaties en gevaren die op de loer liggen. De sectie maakte onderscheid tussen vier limitaties en gevaren: 1.) hoge gevoeligheid (inclusief manipuleerbaarheid), 2.) intelligentie, 3.) afhankelijkheid en 4.) transparantie.

1. A. Jain stelt dat naast een neuraal netwerk in staat is om in theorie, met genoeg data, iedere functie kan representeren is een neuraal netwerk ook erg gevoelig (Jain, sd). Zo kan na analyse van de getrainde functie van een neuraal netwerk een antagonist van deze functie geconstrueerd worden. Een antagonist is een kunstmatig gegenereerde input die zo gegenereerd is dat het een bepaalde output van een model teweeg brengt. In het geval van object detection kan een antagonist, zelfs met stimulans van één pixel, een foutieve classificatie teweeg brengen (Su, 2017). Zoals aangetoond door onder andere Eyholt is dit een gevaarlijke gevoeligheid van neurale netwerken bij bijvoorbeeld de toepassing van object detection in autonome voertuigen (K. Eyholt, 2018).
2. Zoals gesteld door Nielson en Gad neurale beschikken neurale netwerken niet over gezond verstand. Waldrop heeft, net zoals R. Amit een artikel geschreven waarin hij meerdere limitaties van neurale netwerken belicht (Waldrop, 2019) (Amit, 2018)). Beiden onderbouwen de claim dat neurale netwerken niet in staat zijn om eenvoudige conclusies

kunnen deduceren, maar enkel een statistisch geconstrueerd algoritme toepassen op de nieuwe inputdata.

3. Vanwege de noodzaak voor veel data wordt een grote afhankelijkheid gecreëerd en wordt automatisch uitgesloten dat een neuraal netwerk ooit zelf (in de vorm van intelligentie) met een oplossing zal komen. Een neuraal netwerk zal vanuit statistische algoritmes een oplossing genereren, waar als dat mensen causaal kunnen redeneren. *Een voorbeeld van causale redernatie: Mensen hebben kennis van natuurwetenschap gebruikt om een raket succesvol op de maan te zetten, zonder dat hier voorbeelden van waren.*
4. Tot slot het nadeel van transparantie. Vanaf het moment dat een neuraal netwerk getraind is, is het in veel situaties niet mogelijk vast te stellen op welke manier keuzes maakt worden. Dit maakt het een ethische kwestie die veel praktische toepassingen limiteert. *Een voorbeeld van een ethische kwestie vanwege een praktische toepassing: een netwerk is getraind om verdachte reizigers op een vliegveld te herkennen. Het is niet vast te stellen of het netwerk mensen selecteert op basis van geslacht, etnische achtergrond of aan gedrag. Deze selectie vormt een ethische discussie met de vraag: "Is het ethisch verantwoord om mensen aan de hand van een onbekend algoritme te selecteren op een (vervelend verhoor)?"*

2.2. Methoden ter ontwikkeling van methoden

In deze paragraaf wordt antwoord gegeven op de tweede kennisvraag door verschillende mogelijkheden voor de ontwikkeling van een methodiek in kaart te brengen. Uit de literatuur wordt een lijst met benaderingen voor de ontwikkeling van methoden geëxtraheerd. Vervolgens zal met theoretische onderbouwing worden vastgesteld welke benaderingen de beste toepassingen kennen binnen dit onderzoek.

De ontwikkeling van methodes valt binnen het vakgebied Method Engineering (ME, Methode Ontwikkeling). Uit onderzoek van Harmsen, in 1997, wordt geconcludeerd dat dat een universele methode die op iedere situatie toepasbaar is, praktisch niet haalbaar is (Harmsen, 1997). Harmsen ontwikkeld in zijn proefschrift een speciale variant op ME, genaamd Situational ME (situationele ME). Henderson-Seller concludeert in een review op het gebied van Situation ME, in 2010 hetzelfde als Harmsen (B. Henderson-Sellers, 2010). Henderson-Seller omschrijft ME als een vakgebied dat zich focust op het ontwerpen, bouwen en aanpassen van methodes, gereedschappen en technieken voor de ontwikkeling van systemen.

Harmsen maakt binnen Situational ME onderscheid tussen zes verschillende aanpakken voor de ontwikkeling van methoden:

1. *Use of "rigid" methods (gebruik van onveranderlijke methodes).* Een regid methode is een onveranderlijke methode die geen ruimte laat voor flexibele aanpassingen.
2. *Selection of rigid methods (selectie van onveranderlijke methodes).* Deze aanpak selecteert een onveranderlijke methode, gebaseerd op de projectsituatie.
3. *Toolkit/Multiview (gereedschapskist).* De gereedschapskist aanpak creëert flexibiliteit door verschillende methoden toe te passen binnen verschillende specifieke aspecten van het proces.
4. *Paths within one method (keuzemogelijkheden binnen één methode).* De vierde aanpak stelt de gebruiker in staat om binnen de methode te kiezen tussen verschillende paden.
5. *Selection and tuning of a method outline (selectie en afstelling van een methode principe).* De aanpak stelt de gebruiker in staat om een keuze te maken tussen methodes en vervolgens de geselecteerde methode aan te passen naar voor bruikbaarheid binnen de huidige situatie.

6. *Modular method construction (Modulaire modelconstructie)*. De laatste aanpak is de meest radicale oplossing in vergelijking tot de hierboven genoemde aanpakken. Vanuit methode fragmenten wordt een methode geconstrueerd.

Om de te ontwikkelen methode uit dit onderzoek praktisch bruikbaar te maken dient de gebruiker een aantal vaste stadia te doorlopen die voor ieder object detection probleem hetzelfde zijn, omdat dit het overzicht van de gebruiker bevordert. Binnen ieder stadia dient de gebruiker de mogelijkheid te hebben om de methode toe te passen op het specifieke probleem voor handen, omdat dit de effectiviteit van de te ontwikkelen methode verhoogt. (In 5. Ontwerp- en ontwikkelingsfase wordt iedere stadia een 'processtap' genoemd en het toepassen van de methode op een specifiek probleem behandeld in 'bewerkingen'). Daarnaast zal de methode iteratieve aspecten bevatten (zie 2.3. Methode omtrent softwareontwikkeling).

In deze sectie worden bovenstaande zes aanpakken één voor één langsgelopen, om een geschikte aanpak voor dit onderzoek te selecteren.

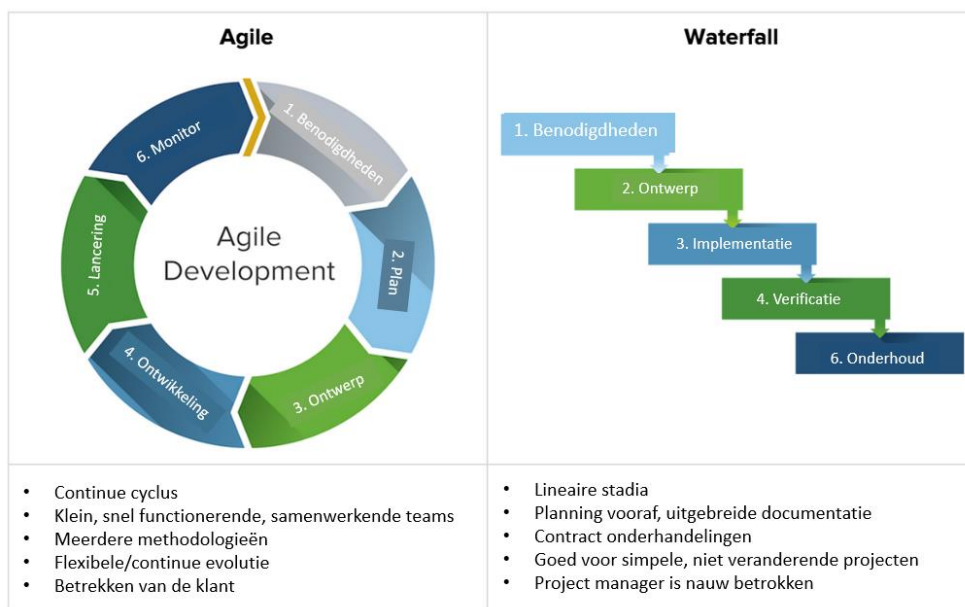
1. *Use of "rigid" methods* is voor dit onderzoek geen geschikte aanpak. Zo kan een machine-learning probleem omgezet worden in een specifiek probleem, maar kan dit probleem niet altijd op dezelfde manier (met dezelfde methode) opgelost worden.
2. *Selection of rigid methods* is tevens geen geschikte aanpak voor dit onderzoek. Zoals hierboven vermeld kan een machine-learning probleem omgezet worden in één specifiek probleem, bijvoorbeeld een object detection probleem, maar valt een object detection probleem niet altijd op dezelfde manier op te lossen. Zo is het afhankelijk van factoren welke stappen je zet bij het oplossen van een object detection probleem. (Denk bijvoorbeeld aan de verschillende datadimensies en verschillende manieren van dataverzamelingen).
3. *Toolkit/Multiview* hanteert een goede aanpak. Afhankelijk van het machine-learning probleem voor handen wordt een nieuwe methode (in de vorm van 'bewerkingen') geselecteerd. In iedere stap kunnen verschillen bewerkingen aan bod komen, zodat de gebruiker met behulp van deze bewerkingen toe kan werken naar het volgende stadia.
4. *Paths within one method* is geval van dit onderzoek complementair aan aanpak3. Om te navigeren tussen de bewerkingen zijn keuzemomenten nodig (paden in de methode) die de gebruiker in staat stellen om afhankelijk van hun specifieke situatie naar een volgende bewerking, of stadia te stappen. Tevens biedt deze aanpak ook de mogelijkheid om een iteratief aspect in de te ontwikkelen methode toe te voegen.
5. agile methode toegepast om de methode een agile ontwikkelingsmethodiek karakter te geven.
6. *Modular method construction* is ook een geschikte aanpak, maar is tijdrovend en complex. Naar verwachting levert het toepassen van deze aanpak een vergelijkbaar resultaat op, in vergelijking tot wanneer een combinatie van hierboven geschikt geachte aanpakken worden toegepast. Gekozen is om geen gebruik van deze aanpak te maken.

2.3. Methode omtrent softwareontwikkeling

In deze paragraaf wordt antwoord gegeven op de derde kennisvraag door naar de geschiedenis van de ontwikkeling van de softwareontwikkelingsmethodieken te kijken. Vandaag de dag zijn iteratieve ontwikkelingsmethoden voor het ontwikkelen van computersoftware erg populair. Om een passende ontwikkelingsmethodiek te kiezen voor de ontwikkeling van machine-learning software wordt kritisch gekeken naar de algemeen geaccepteerde iteratieve methoden.

De oorspronkelijke softwareontwikkelingsmethode is sterk vergelijkbaar met de traditionele projectplanningsmethodiek, waarin eerst een plan wordt gemaakt en daarna de uitvoering van het plan wordt gerealiseerd. De traditionele projectplanningsmethodiek en oorspronkelijke softwareontwikkelingsmethodiek werd na het artikel van Royce, 1970 voor het eerst het waterfall model genoemd vanwege de visuele weergeven van de procesplanning die op een waterval lijkt (Royce, 1970).

Zoals omschreven door Larman, in 2003, werd in de loop van de jaren 50', toen de Amerikaanse overheid grote softwareprojecten leidde, al voor het eerst een iteratieve softwareontwikkelingsmethode toegepast (C. Larman, 2003). Succesverhalen van, voornamelijk Amerikaanse, overheidsinstanties hebben door de decennia heen ervoor gezorgd dat de iteratieve ontwikkelingsmethode op den duur overgenomen werd door het bedrijfsleven. Vanaf 2001, het tekenen van het agile manifest, is de iteratieve manier van softwareontwikkeling pas de norm¹. Wanneer gesproken wordt over iteratieve softwareontwikkeling, wordt gesproken over agile softwareontwikkeling.



Figuur 13: Agile vs Waterfall

¹ Agile Manifesto: <https://agilemanifesto.org/principles.html>

In Figuur 13 zijn het Agile- en Waterfall proces, en de verschillende stappen binnen het proces, visueel weergegeven. Daarnaast zijn onder de illustraties de belangrijkste eigenschappen van beide processen opgesomd. Stap 3 van het agile proces wordt ook wel het ontwerp van een minimaal levensvatbaar product (MVP) genoemd (hierover meer in 4.3. Minimaal levensvatbaar product (MVP)).

In een review van agile methodologieën analyseert Dingsøyr, in 2012, de geschreven literatuur omtrent agile methoden vanaf 2001 en stelt dat Conboy, in 2009, de meest uitgebreide definitie van agile softwareontwikkeling heeft gegeven (T. Dingsøyr, 2012). De definitie van Conboy luidt: “Agile methoden streven ernaar om snel of inherent veranderingen te creëren, proactief of reactief omgaan met veranderingen en leren van veranderingen terwijl het bijdraagt aan waargenomen klantwaarde, door middel van zijn collectieve componenten en relaties met de omgeving” (Conboy, 2009).

Om te valideren of de vandaag de dag standaard agile methode de best passende methode is voor de ontwikkeling van ML-software wordt gekeken naar het vergelijkende onderzoek tussen traditionele methodieken en agile ontwikkeling van Papadopoulos, in 2015 (Papadopoulos, 2015). Papadopoulos concludeert uit verschillende case studies dat bij gebruik van agile ontwikkelingsmethodiek de kwaliteit verbeterd, veranderingen in opdrachtomschrijving gedurende het project kunnen veranderen en dat de werknemerstevredenheid verhoogd wordt. Ook wordt gesteld dat het implementeren van een agile werkwijze, zeker bij grote bedrijven met een lange geschiedenis in traditionele methoden, een uitdagend en langdurig proces is.

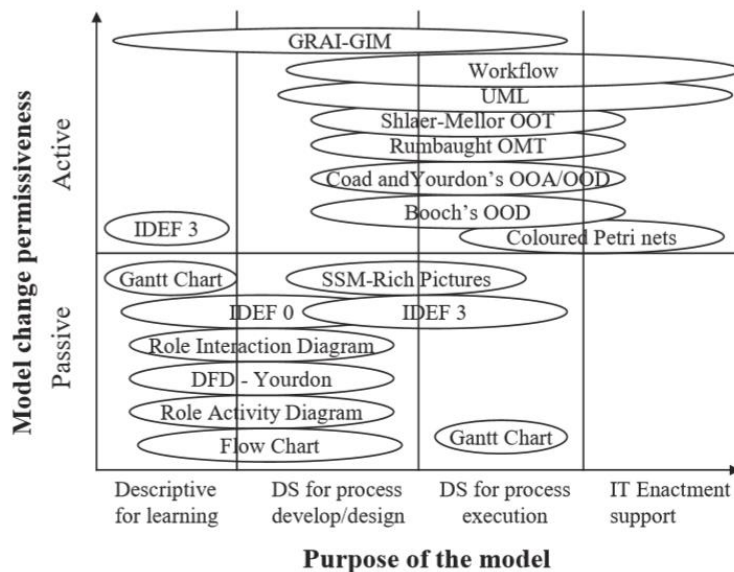
2.4. Business Process Modelling Notation

In deze paragraaf wordt antwoord gegeven op de vierde kennisvraag door naar verschillende methodieken rondom het in kaart brengen van bedrijfsprocessen te kijken. Vervolgens zal met theoretische onderbouwing vastgesteld worden welke methode het best gebruikt kan worden binnen dit onderzoek en wordt afgesloten met een korte uitleg van de gekozen methode.

2.4.1. Modelleermethoden

In deze sectie worden verschillende soort modelleermethoden in kaart gebracht. De sectie wordt afgesloten met de selectie van een modelleermethode die gebruikt kan worden in de visuele weergave van het WePublish-bedrijfsproces.

Modelleermethoden bestaan al sinds dat mensen processen op visuele wijze in kaart brengen. Het aantal methoden om deze processen in kaart te brengen is door de jaren alsmaar toegenomen. In 2003 heeft Auguilar-Savén een literatuuronderzoek gedaan naar alle bestaande proces modelleer technieken. Het doel van het onderzoek is om de belangrijkste technieken te omschrijven en een raamwerk voor te stellen waarmee de gebruiker een bedrijfsproces modelleringstechniek kan selecteren (Auguilar-Savén, 2003).



Figuur 14: Classificatie raamwerk ter selectie van bedrijfsproces modelleertechnieken (Aguilar-Savén, 2003)

Figuur 14 toont het classificatieraamwerk van Aguilar-Savén. Op de horizontale as staat het doel van het model en op de verticale as staat de toegankelijkheid van het model. Voor het in kaart brengen van het bedrijfsproces van de WePublish-service dient een beschrijvend model ontwikkeld te worden dat ondersteuning biedt bij het ontwikkelen van een nieuw proces. Ook vereist het te ontwikkelen beschrijvende model geen interactie.

De beoogde modelleertechniek valt met bovenstaande omschrijving in de tweede kolom, "DS for process develop/design", en de eerste rij, "passive". Binnen dit kwadrant worden de opties afgegaan en wordt afgesloten met de beste keuze.

Het "Role Interaction Diagram", het "Data Flow Diagram (DFD)" en het "Role Activity Diagram" zijn bedoeld om een specifiek gedeelte van een productieproces weer te geven. Hierdoor mist een helikopterbeeld van het totale bedrijfsproces. De "Integrated Definition for Function Modelling" is in theorie een geschikte kandidaat, maar kent complexe regels en is niet eenvoudig aan te leren. De "Flowchart" biedt een flexibele manier van werken en stelt de gebruiker in staat om op eenvoudige wijze bedrijfsprocessen van verschillende niveaus samen te voegen tot één overzichtelijke helikopterbeeld. Om bovenstaande redenen wordt in dit onderzoek gebruik gemaakt van flowcharts om het WePublish-bedrijfsproces op visuele wijze in kaart te brengen.

2.4.2. Flowcharts - BPMN

Een flowchart is een sequentiële grafische weergave van een proces. Het gebruikt symbolen om bepaalde processen of gebeurtenissen weer te geven. Flowcharts zijn uniek in de mate waarin ze begrijpelijk zijn voor zowel de maker als de gebruiker. Er kan echter grote verwarring ontstaan bij gebruik van flowcharts wanneer onduidelijke afspraken zijn gemaakt over het gebruik van symbolen. Zo kan gebruiker1 een ander symbool gebruiken dan maken2 voor het weergeven van vergelijkbare processtap. Hierdoor kunnen procesmodellen binnen hetzelfde bedrijf, of zelfs binnen dezelfde afdeling, lastig vergeleken worden.

Om deze verwarring en verlaagde effectiviteit van flowcharts tegen te gaan heeft het BPMI (Business Process Modeling Initiative) in 2004 een standaard notatiewijze gelanceerd: het BPMN (Business Process Modeling Notation). In 2006 is de BPMN tot standaard bedrijfsproces model notatie

uitgeroepen en is het vandaag de dag nog steeds de onbetwiste de-facto mondiale notatiewijze voor het beschrijven van bedrijfsprocessen (Nijssen G.M., 2009).

Tegenwoordig wordt er veel gebruik gemaakt van BPMN 2.0. Ten opzichte van eerdere versies heeft BPMN 2.0 veel meer mogelijkheden om relaties tussen processen, informatiesystemen en procesmodellen weer te geven. Chinosi stelt dat de nieuwe versie ook onjuistheden in de eerdere systemen weg neemt en meer nadruk legt op datastromen (M. Chinosi, 2012). Om deze reden wordt in dit onderzoek alleen BPMN 2.0 besproken.

Hoewel BPMN 2.0 over het algemeen als datavisualisatie tool wordt gebruikt is er ook de mogelijkheid om de visualisatie te koppelen aan de uitvoering van softwareprocessen. Deze koppeling wordt echter beperkt gebruikt omdat deze uitvoerformat soms een scheve weergave geeft ten opzichte van de realiteit (Geiger, 2018). Uit het onderzoek blijkt dat gebruikers van BPMN 2.0 content lijken te zijn met enkel het visualisatie aspect en hebben geen behoefte aan een koppeling in implementatie.

2.5. Samenvatting en conclusies

In deze paragraaf wordt het theoretisch kader kort samengevat, om vervolgens per kennisvraag te concluderen welke kennis geëxtraheerd kan worden voor mijn onderzoek.

Kennisvraag 1: Welke componenten van machine learning zijn van toepassing op het omschreven probleem?

ML is een vakgebied dat gebruik maakt van NN'en om voorspellende uitspraken te doen over inputdata, na het zien van veel relaties tussen inputdata en werkelijke outputdata. DL is een sub-set van ML dat gebruikt maakt van meerdere lagen van neuronen in een NN. Een CNN is een speciaal soort DL NN dat gebruikt maakt van feature herkenning om images nauwkeuriger te verwerken tot outputdata. Alle NN'en leren middels een back-propagation algoritme dat gebruik maakt van gradiënt descent om de parameters van het model naar het globale minimum, of vergelijkbaar lokaal minimum, bij te stellen.

Geconcludeerd kan worden dat een CNN, met bijbehorende componenten, uitkomst in het vraagstuk omtrent object detectie biedt.

Kennisvraag 2: Welke methoden worden in de ontwikkeling van software gebruikt?

ME is het vakgebied dat zich bezighoudt met de ontwikkeling van methodes, ter ontwikkeling van methodes. Eén allesomvattende methode binnen ME bestaat niet, waardoor Situational ME is ontwikkeld. De ontwikkeling van een CNN, dat gespecialiseerd is in de verwerking van images uit de folders van GreenOrange, is een specifieke taak waarin veelvuldig verschillende methodieken toegepast dienen te worden in combinatie met veelvuldig gebruik van keuzemomenten. Er is gekozen om gebruik te maken van aanpak3, aanpak4 en (in mindere mate) aanpak 5.

Kennisvraag 3: Welke methoden zijn beschikbaar voor het ontwikkelen van een methode?

Softwareontwikkeling gebeurt vandaag de dag middels een agile methode, waarvan de basis principes zijn vastgelegd in het Agile Manifesto. Belangrijk is een snelle, flexibele en voornamelijk klantgerichte aanpak waarin meerdere iteraties van ontwikkeling mogelijk zijn. Uit onderzoek van Papadopoulos blijkt dat agile methoden betere prestaties genereren. Geconcludeerd kan worden dat

het agile framework een waardevolle toevoeging is voor de ontwikkeling van de te ontwikkelen methodiek.

Zo dient het eerste prototype in de vorm van een MVP te worden opgesteld, zodat deze snel getoetst kan worden in de praktijk. Aan de hand van inzichten die verkregen zijn door de implementatie van het prototype kan doorgewerkt worden naar een nieuw MVP etc. Deze iteratieve werkwijze is terug te zien in Figuur 13.

Kennisvraag 4: Welke methoden zijn beschikbaar voor het visueel in kaart brengen van een bedrijfsproces?

Voor het visueel in kaart brengen van het WePublish-bedrijfsproces wordt gebruik gemaakt van flowcharts. De notatiewijze die gehanteerd wordt om dit bedrijfsproces in een flowchart weer te geven is de BPMN. Geconcludeerd kan worden dat de BPMN notatiewijze in dit onderzoek toegepast zal worden in de visuele weergave uit 3. Huidige situatie en 4. Gewenste situatie. In dit onderzoek wordt enkel gebruik gemaakt van het visuele aspect van BPMN 2.0.

3. Huidige situatie

In dit hoofdstuk wordt de huidige gang van zaken omtrent het publicatie proces van WePublish in detail beschreven. Allereerst wordt in paragraaf 2.1 het volledige publicatie proces beschreven. Vervolgens wordt in paragraaf 2.2. een nulmeting gedaan, waarbij eerst wordt gekeken welke KPI's gemeten dienen te worden tijdens deze nulmeting om vervolgens de nulmeting te scoren aan de hand van de gekozen KPI's. Ter afsluiting van het hoofdstuk volgt een samenvatting en een korte conclusie in paragraaf 2.3.

3.1. Het volledige WePublish proces

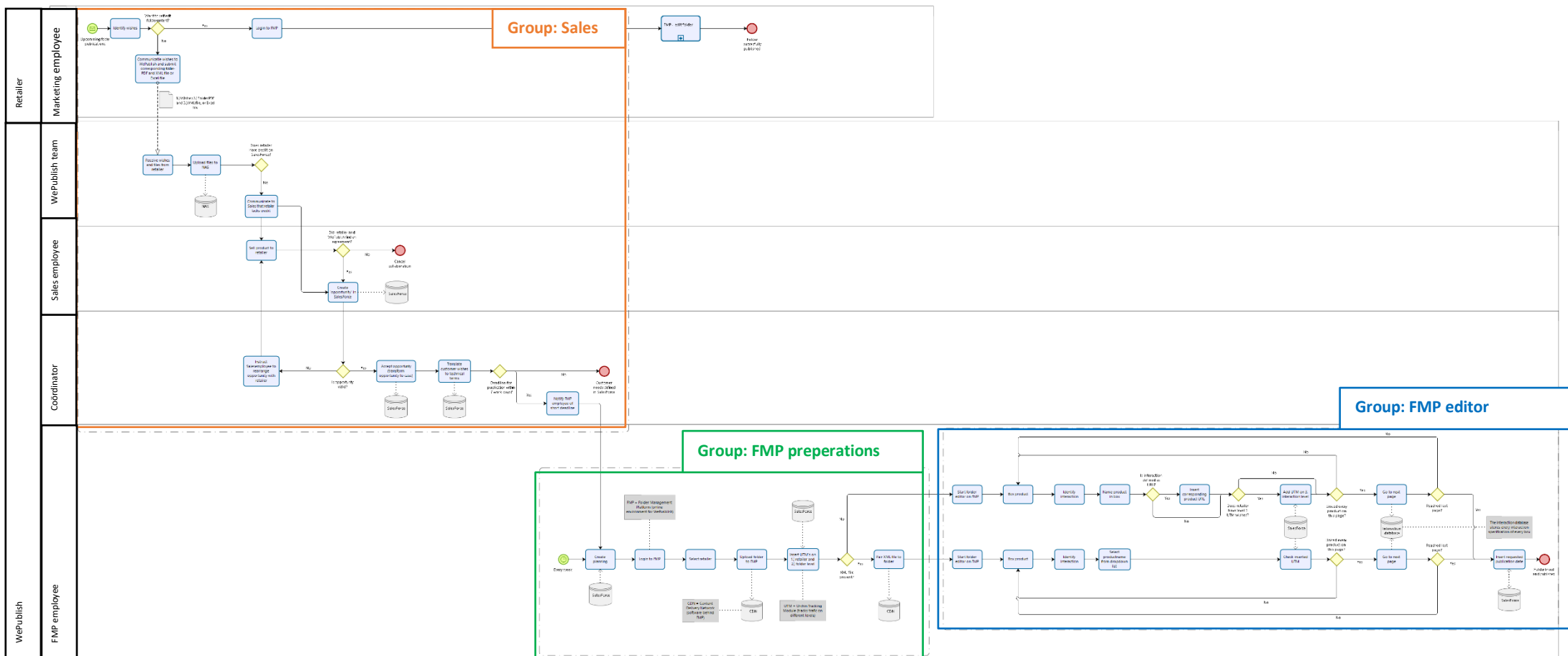
Het huidige proces is met behulp van de Business Process Modeling Notation (BPMN) weergegeven in Figuur 15, op de volgende pagina. Het figuur geeft in de twee horizontale balken, de *pools*, de verschillende bedrijven weer. De *pools* zijn opgebouwd uit *lanes*, die onderscheid maken tussen het type medewerker dat verantwoordelijk is voor het uitvoeren van een bepaalde taak. Het figuur omschrijft de stappen vanaf het moment dat een retailer, die reeds klant is bij WePublish, een publicatie van een reclamefolder in het vooruitzicht heeft.

Uitleg over de BPMN-notatie is te vinden in bijlage A en uitleg over verschillende vaktermen die voorkomen in het BPMN-model is te vinden in bijlage B.

Het huidige proces is opgedeeld in drie fasen welke overeenkomen met de verschillende stappen uit het publicatie proces. In het figuur zijn de verschillende fasen, ook wel de 'groups', aangegeven met ieder een unieke kleur. In de volgende paragrafen wordt iedere *group* vergroot weergegeven, in combinatie met een specifieke omschrijving en indien van toepassing extra achtergrondinformatie. De stappen zijn: **1.) Sales**, **2.) FMP preparations** en tot slot **3.) FMP editor**.

Volledige proces

Onderstaande illustratie verschaft inzicht in de manier waarop de drie verschillende groepen met elkaar in verbinding staan. De illustratie is los van de group-names niet leesbaar in de afdrukweergave.



Figuur 15: BPMN WePublish process

3.1.1. Sales – group

Salesgroup proces

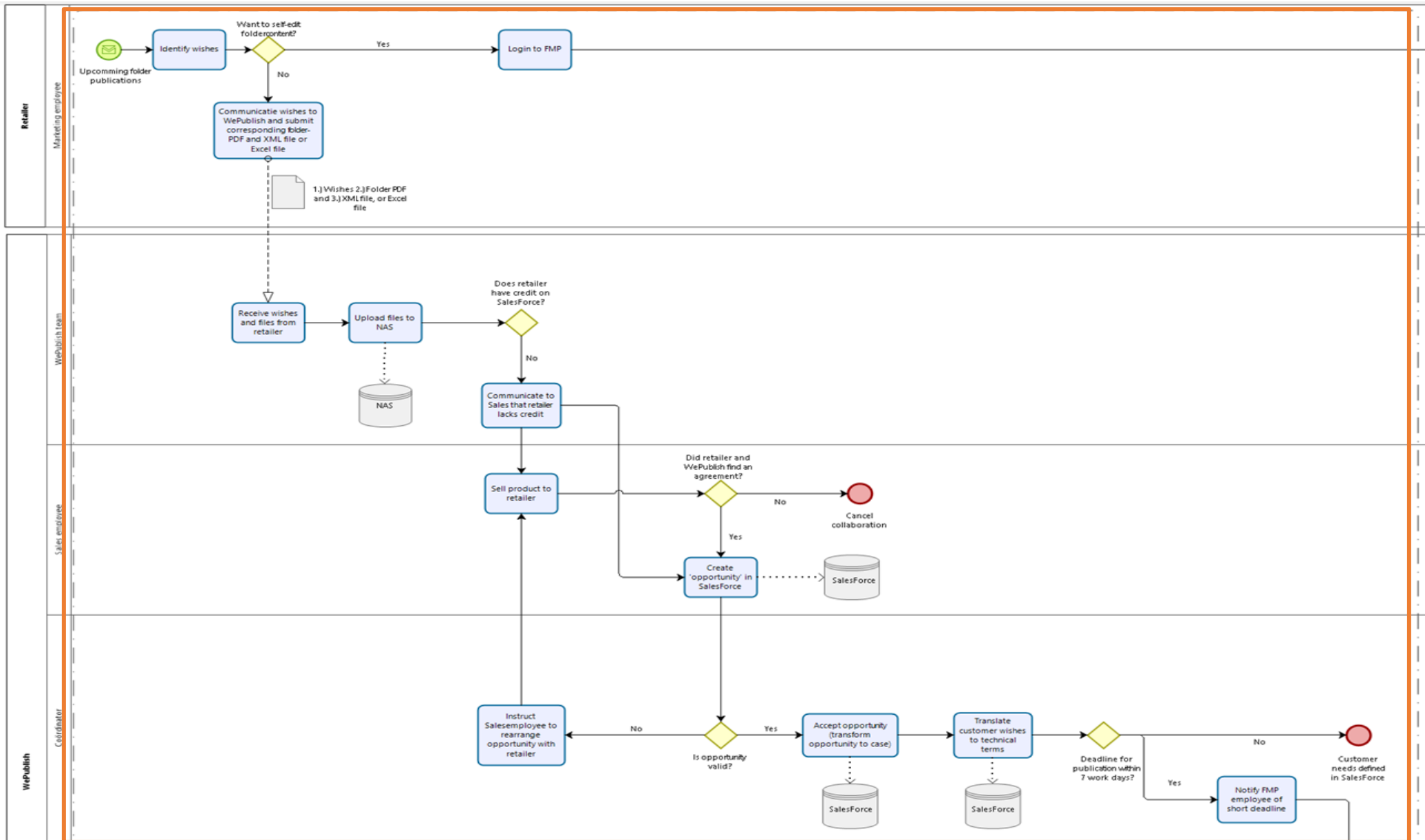


Figure 16: Salesgroup-proces

SalesForce

In SalesForce wordt onderscheid gemaakt tussen een kans ('opportunity') en een zaak ('case'). Een kans kan geaccepteerd worden, waardoor een kans in een zaak verandert. Een zaak kan verzilverd worden, daarmee wordt de zaak in het productieproces opgenomen.

Een kans is enkel een mogelijkheid. In deze fase zijn vanuit WePublish nog geen toezegging gemaakt richting de klant. Ook heeft de klant hier nog nergens voor betaald.

Een zaak is een overeenkomst. In deze fase beloofd WePublish om op een nader te bepalen moment de wensen van de klant uit te voeren. De klant is in deze fase akkoord met het te betalen bedrag, of heeft al betaald.

Een zaak wordt verzilverd wanneer de publicatiedatum en alle andere wensen omtrent deze publicatie bekend zijn bij WePublish.

Een kans en een zaak bestaan uit dezelfde invoervelden die gezamenlijk de wensen van de klant samenvatten in een tabelvorm. Figuur 17 toont de verschillende invoervelden en is ingevuld met een voorbeeld van retailer1. De Sales employee vult deze tabel in en de Coördinator controleert de invoer bij het aanmaken van een kans.

Indien noodzakelijk past de Coördinator de invoervelden aan door de omschrijvende notatiewijze van de Sales employee om te zetten in technische termen die exacter uitgevoerd kunnen worden door de FMP employee.

Krediet

Een vaste klant heeft vaak de wens om ieder jaar een factuur te ontvangen, i.p.v. bijvoorbeeld iedere maand. Voor deze manier van samenwerking wordt het begrip 'krediet' gebruikt.

Wanneer een klant vooruit betaald, bijvoorbeeld 12 edities voor het komende jaar, dan bouwt deze klant krediet op. Op dit moment worden er aan deze klant 12 lege zaken gekoppeld. In deze lege zaken staat het product/service dat de klant gekocht heeft, maar nog geen gedetailleerde omschrijving. De klant heeft iedere maand de mogelijkheid om gedetailleerde informatie aan te leveren, waardoor een zaak gevuld raakt. Denk bij deze gedetailleerde informatie aan Deze informatie bestaat uit een omschrijving en verzilverd wanneer de publicatiedatum bekend is.

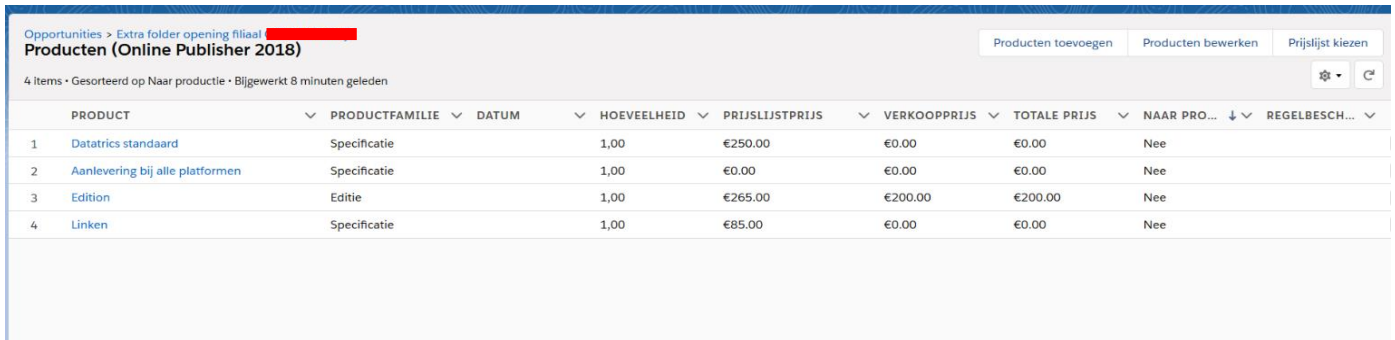
The screenshot shows a Salesforce form with the following sections:

- Beschrijvingsgegevens**
 - Aantal edities: 1
 - Onderwerp: Extra folder [redacted]
 - Specificaties: Linken & DT Box. Beschrijving, box targeting instellen op postcode gebied rondom [redacted]. Postcodes zijn aangeleverd. In landelijk [redacted] folder mag de box getoond worden, met CT naar [redacted] folder.
 - Bijzonderheden: (empty)
 - Interne opmerkingen: (empty)
 - Beschrijving: Linken & DT Box. Beschrijving, box targeting instellen op postcode gebied rondom [redacted]. Postcodes zijn aangeleverd. In landelijk [redacted] folder mag de box getoond worden, met CT naar [redacted] folder.
- OP Proces**
 - Verwachte opleverdatum: 24-5-2019
 - Aanleverdatum: 17-5-2019
 - Dag van de week (opleverdatum): Vrijdag
 - Klant map: (empty)

Figuur 17: Wensen klant (voorbeeld retailer1, extra filiaal opening)

Wensen van de klant

In het sales-proces is het belangrijk dat de klant, de retailer, haar wensen duidelijk communiceert naar WePublish. De retailer dient haar wensen te communiceren naar het hello@wepublish.com mailadres dat beheerd wordt door het vijfkoppige WePublish team. In realiteit zoekt de retailer vaak direct mail, of telefonisch contact met de medewerker van Sales of de Coördinator. Wanneer de retailer haar wensen heeft doorgegeven wordt een kans aangemaakt en worden de bestelde producten gekoppeld aan de bestelling. In Figuur 18 staat een voorbeeld van een kans die is aangemaakt voor retailer1.



PRODUCT	PRODUCTFAMILIE	DATUM	HOEVEELHEID	PRIJSLIJSTPRIJS	VERKOOPPRIJS	TOTALE PRIJS	NAAR PRO...	REGELBESCH...
1 Datatrics standaard	Specificatie		1,00	€250.00	€0.00	€0.00	Nee	
2 Aanlevering bij alle platformen	Specificatie		1,00	€0.00	€0.00	€0.00	Nee	
3 Edition	Editie		1,00	€265.00	€200.00	€200.00	Nee	
4 Linken	Specificatie		1,00	€85.00	€0.00	€0.00	Nee	

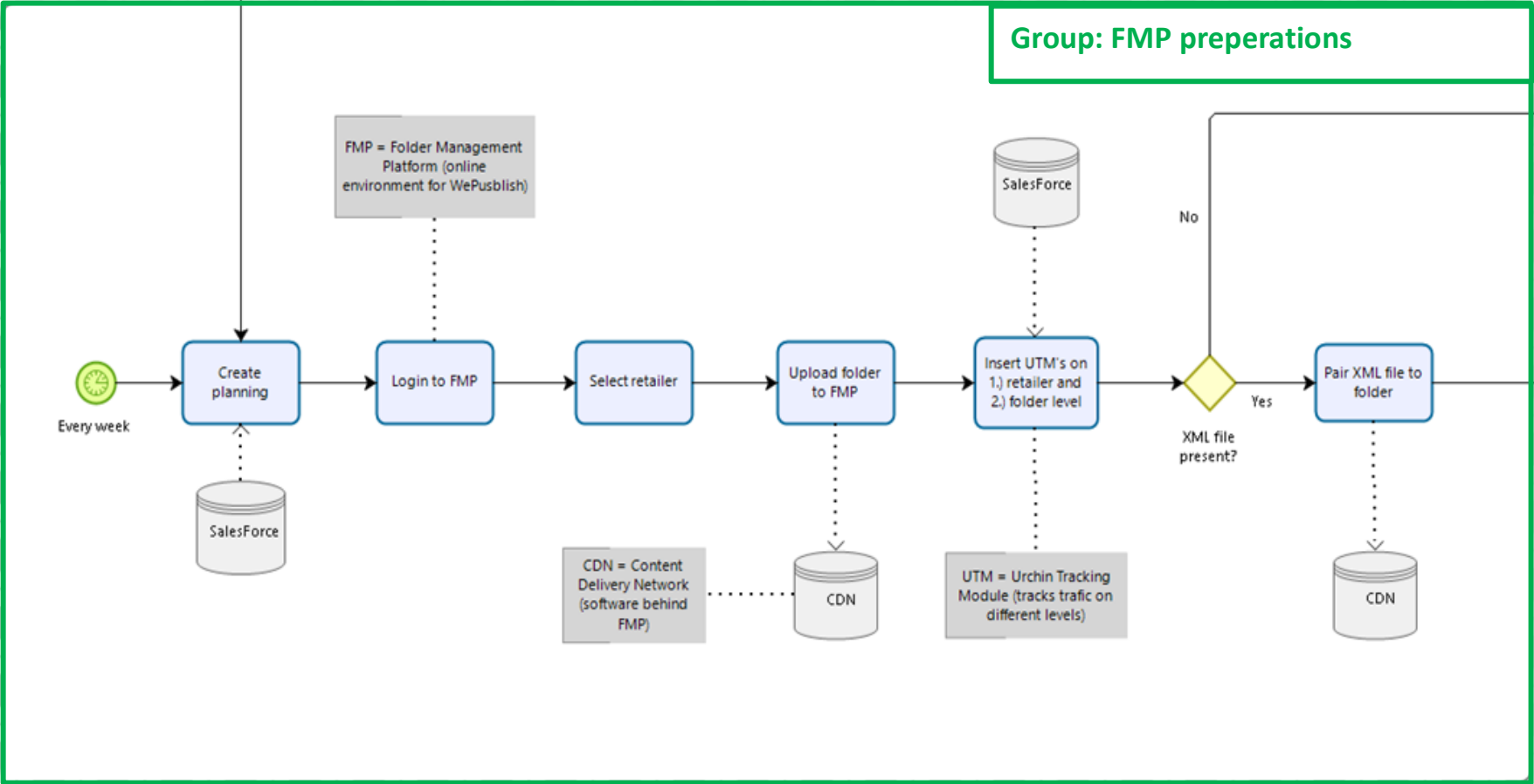
Figuur 18: Overzicht bestelde producten (voorbeeld retailer1, extra filiaal opening)

In het voorbeeld wordt de retailer gefactureerd voor:

- Linken is het proces dat zich afspeelt in 'FMP editor – group'.
- Edition is een vaste facturatie voor een nieuwe publicatie.
- Aanlevering bij alle platformen is een optie, de klant betaald voor de publicatie en kiest op welke platformen de publicatie actief dient te worden.
- Datatrics standaard is een service waarbij de data die met linken wordt verkregen wordt verwerkt door Datatrics om klantprofielen te creëren, om op deze manier gepersonaliseerde content in de publicatie te genereren.

3.1.2. FMP preparations - group

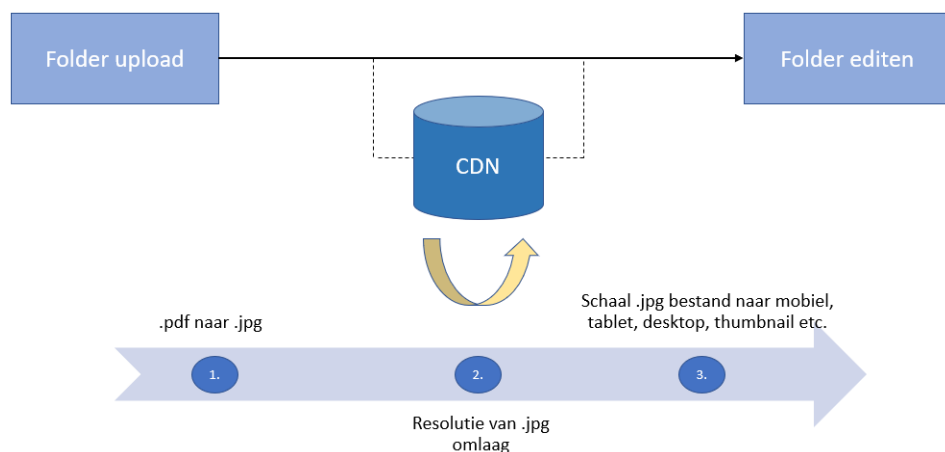
FMP preparationsgroup-proces



Figuur 19: FMP Preparationsgroup-proces

CDN

De FMP employee upload de PDF folder middels het FMP naar het Content Delivery Network (CDN). In het CDN wordt het .pdf bestand, zoals eerder beschreven verwerkt tot bruikbare componenten. In Figuur 20 is te zien welke stappen doorlopen worden op het moment dat deze is geüpload naar het CDN. Na stap 3. van het CDN is de folder beschikbaar om geëdit te worden op het FMP. Voor de beeldvorming, het proces in Figuur 20 neemt ongeveer 0.32 ms in beslag, afhankelijk van de grootte van het pdf.

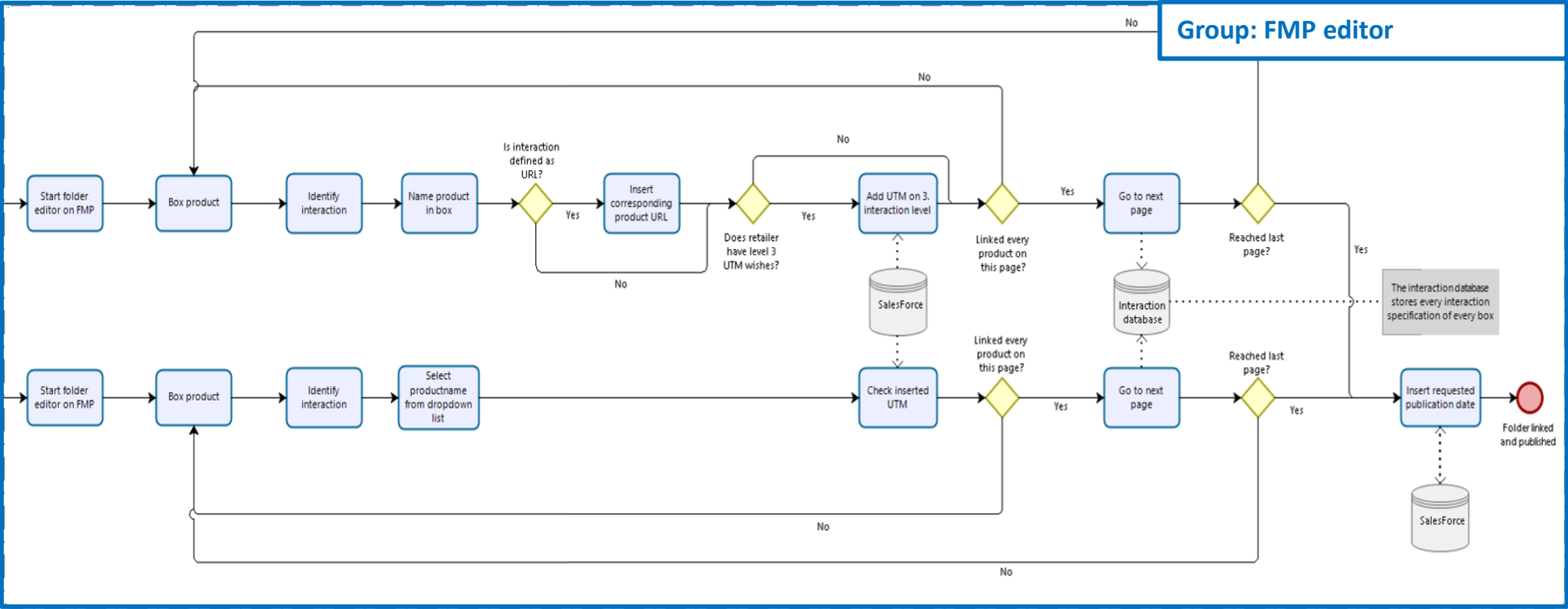


Figuur 20: CDN pdf verwerking

Daarnaast linkt de FMP employee niveau 1- en niveau 2 UTM's aan de folder waarmee hij aan de slag gaat en upload het bijbehorende XML bestand, wanneer de retailer die aangeleverd heeft.

3.1.3. FMP editor – group

FMP editorgroup - proces



Figur 20: FMP Editor-proces

Proces omschrijving

Dit deel van het procesmodel is voor dit onderzoek het meest relevant, aangezien de eventuele aanpassingen in dit deel van het proces worden geïmplementeerd. In de 'gewenste situatie' zal dit deel van het procesmodel aangepast weergegeven worden.

Het proces is opgesplitst in twee processen die sterk op elkaar lijken. Het bovenste proces wordt doorlopen wanneer géén XML bestand is aangeleverd, het onderste proces wordt doorlopen wanneer wél een XML bestand is aangeleverd. Deze sectie focust zich op het verschil tussen deze twee processen en gaat expliciet in op het verschil in de procestijd. In deze sectie wordt aangenomen dat de lezer begrijpt wat het boxen van een product en het identificeren van een interactie is, omdat deze in de probleemaanleiding aan bod komen.

In bijlage C is een voorbeeld gegeven van een XML bestand en de manier waarop het geïnterpreteerd kan worden door mensen.

Het verschil in procestijd wordt veroorzaakt door een verandering in drie activiteiten uit het BPMN procesmodel. In Tabel 4 wordt praktische uitwerking en de impact op de procestijd, per activiteit, weergegeven.

Tabel 4: XML vs no XML, FMP editor

Naam van product ('Name product in box')	
XML	Selecteer de naam van het product uit een dropdownlist.
No XML	Type de naam van het product dat in de box geselecteerd is over in dit invoerveld.
Impact	Het verschil in procestijd is minimaal.

Product-URL invoeren ('Insert corresponding product URL')	
XML	De URL wordt automatisch ingevoerd, nadat een product uit de dropdownlist is geselecteerd.
No XML	Ga naar de website van de retailer en zoek naar het product dat in de box geselecteerd is en kopieer de bijbehorende URL van het product in dit invoerveld.
Impact	De juiste URL bij een product zoeken kan 30 seconden tot een minuut duren.

UTM niveau 3 invoeren ('Add UTM on 3. interaction level')	
XML	De UTM wordt automatisch ingevoerd, nadat een product uit de dropdownlist is geselecteerd.
No XML	Kijk in Salesforce naar een eventuele specifieke wens van de retailer om een UTM te plaatsen bij een specifiek product of een productreeks.
Impact	Het verschil in procestijd is minimaal

Tot slot wordt het proces afgesloten met de publicatie van de folder op de gewenste datum.

3.2. Nulmeting

Bij het invoeren van een verandering is het belangrijk om meetbaar te maken of de verandering een positief effect heeft gehad, of een negatief effect heeft gehad. Om de mate van verandering meetbaar te maken en te analyseren wordt een nulmeting uitgevoerd.

De nulmeting wordt gescoord op verschillende KPI's (Kritieke Prestatie Indicatoren) die in overleg met het bedrijf en met kennis uit de literatuur zijn gekozen. Over het algemeen beschikt een bedrijf, of een afdeling van een bedrijf, over een samengevatte set van de belangrijkste variabelen of maatstaven die managers informeren over de mate waarop een organisatie haar doelen bereikt (Boddy, 2014).

3.2.1. KPI's

KPI's zijn een samengevatte set van de belangrijkste variabelen of maatstaven die managers informeren over de mate waarop een organisatie of proces haar doelen bereikt (Boddy, 2014).

KPI's uit literatuur

Aangenomen dat het proces dat zich afspeelt in 'FMP editor' (waarin interacties geproduceerd worden) vergelijkbaar is met een standaard productieproces vinden we uit de literatuur de volgende veelvoorkomende KPI's.

Gunasekaran (2004) noemt een aantal belangrijke prestatie indicatoren, op productieniveau, om een supply chain te meten. Een groot aantal van deze prestatie indicatoren zijn ook van toepassing bij het produceren van interacties.

Voor dit onderzoek zijn voornamelijk de volgende prestatie indicatoren van belang:

- Capaciteitsbenutting: Volgens Slack et al. (1995) is capaciteitsbenutting belangrijk, omdat het de flexibiliteit en levertijd beïnvloed. Wanneer capaciteit niet volledig benut wordt, is er sprake van verspilling.
- Order levertijd: De gehele productiecyclus die doorlopen wordt na een order geeft weer hoe snel het bedrijf kan reageren op de klant. Door de productietijd te verkorten kan een bedrijf een competitief voordeel behalen (Christopher, 1992).
- Variëteit in productie: Volgens Mapes et al. (1997) is het voor fabrieken die een brede variëteit in producten produceren waarschijnlijker dat zij minder snel nieuwe producten aan kunnen bieden. Fabrieken die veel verschillende producten produceren voegen meestal minder waarde per werknemer toe. Daarnaast gaat dit vaak ten koste van snelheid en leverbetrouwbaarheid

KPI's uit interviews met medewerkers en persoonlijke inbreng

De FMP employee en verschillende andere medewerkers die het FMP editor proces in het verleden uitgevoerd hebben, kwamen met de volgende lijst mogelijke KPI's (mijzelf inclusief):

- gemiddelde tijd, per interactie
- aantal interacties per dag
- aantal interacties per uur
- aantal ontvangen klachten, per interactie
- gemiddelde tijd, per folder
- gemiddelde tijd, per interactie, per retailer (of type retailer) --> AH, of Supermarkt)
- gemiddeld aantal folders per dag (folders van specifieke retailer, of type retailer)

- mate van spreiding (boxplot, mediaan)

Afweging KPI's

In deze sectie worden de KPI's vanuit het bedrijf aan de KPI's uit de literatuur gehangen. Met een kwalitatieve afweging wordt onderbouwt welke KPI's uiteindelijk geselecteerd worden voor metingen binnen dit onderzoek.

Capaciteitsbenutting

Om de capaciteitsbenutting te bepalen is het noodzakelijk om de maximale capaciteit tegen de huidige capaciteit weg te zetten. De huidige capaciteit kan gemeten worden met de groen gemarkeerde KPI's uit Tabel 5: KPI afwegingen. De maximale capaciteit kan het beste uitgedrukt worden in

Tabel 5: KPI afwegingen

Mogelijke KPI	Opmerkingen en afwegingen	
gemiddelde tijd per interactie	Eén getal, goed vergelijkbaar met meting op later moment. Biedt inzicht in gemiddelde tijd die medewerker kwijt is aan het plaatsen van één interactie.	
Percentage aantal interacties per dag	Waardevolle set getallen die in een plot een overzicht van productie, per dag/uur verschaft. Handige tool die inzage biedt in het productieproces en eventuele verschuivingen van werkdruk.	
Percentage aantal interacties per uur		
gemiddelde folder-interactietijd	Eén getal, goed vergelijkbaar met meting op later moment. Biedt inzicht in gemiddelde tijd die medewerker kwijt is aan één folder.	
gemiddelde interactietijd per folder	Deze set getallen zegt alleenstaand niet veel en kan een manager niet helpen besluiten te maken. Het is noodzakelijk om de folders te linken aan retailers of retailersets.	
gemiddelde interactietijd per retailer/retailerset	<p>Waardevolle set getallen die inzicht biedt in de interactietijd per retailer, of retailerset. Hiervoor dient de folder_id gelinkt te worden aan de retailernaam, en/of retailergroep.</p> <p>*Bijvoorbeeld: ID-123124 → folder van AH → supermarkt</p> <p>Analyse van verschillende retailers en retailerset is erg krachtig en kan tot specifieke inzichten leiden.</p>	
gemiddeld aantal folders per dag	Deze set getallen is sterk afhankelijk van de demand. Zonder bijvoorbeeld uitvoerige analyse van het marketingbudget investering per maand en jaarlijkse demandtrends valt er met deze set getallen weinig af te leiden.	
gemiddeld aantal folder per uur		

Order levertijd

De order levertijd van het FMP editor proces is te vertalen naar de publicatiedatum van een folder. Daarnaast kan de kwaliteit van deze publicatie gemeten worden met het aantal klachten per folder.

Tabel 6: KPI afwegingen²

Mogelijke KPI	Opmerkingen en afwegingen
Percentage publicaties op tijd	Eén getal, goed vergelijkbaar met meting op later moment. Biedt inzicht in mate waarop deadlines van klanten gehaald wordt.
Klachten per 100 folders	Eén getal, goed vergelijkbaar met meting op later moment. Biedt inzicht in mate van fouten per folder. Het getal 100 is gekozen, zodat zoveel mogelijk met hele getallen gewerkt kan worden en niet met decimalen. Het aantal klachten per folder, is namelijk erg laag.

Variëteit in productie

In een productieproces is een variëteit in aangeboden producten sterk gecorreleerd met lage flexibiliteit, leverbetrouwbaarheid en productiesnelheid. Het WePublish-proces biedt haar klanten veel een grote set producten aan. Aanbeveling om hier minder van te maken.

In het FMP editor proces, waarop het onderzoek inzoomt, wordt maar één product geanalyseerd en aangeboden. Deze KPI uit de literatuur is dus niet van verdere toepassing op dit onderzoek.

Definitieve KPI's / Conclusie

Aangezien de nulmeting zich focust op een specifiek deel uit het FMP-editor proces is gekozen voor de volgende vier KPI's:

- Gemiddelde tijd per interactie
- Gemiddelde folder-interactietijd
- Percentage publicaties op tijd
- Klachten per folder

Onderstaande KPI's verschaffen inzichten in verdere optimalisatie van het FMP editor proces en stellen de lezer in staat bedrijfskundige aanpassingen aan het proces te ondernemen. Onderstaande KPI's zijn echter niet van direct belang voor het toetsen van de ontwikkelde software.

- gemiddeld aantal interacties per dag
- gemiddeld aantal interacties per uur
- gemiddelde interactietijd per retailer/retailerset
 - o Ondanks dat dit een waardevolle KPI is kan deze met de huidige dataset niet geanalyseerd worden in de volgende sectie.

3.2.2. Scoren van nulmeting

In deze sectie wordt een kwantitatieve analyses van de huidige situatie, met behulp van data uit de interactiedatabase, uitgevoerd. Eerst wordt gekeken naar de gemiddelde interactietijd en de spreiding van de interactietijden. Vervolgens wordt de gemiddelde interactietijd van een folder bepaald. Daarnaast worden de interactietijden geanalyseerd in relatie tot de uren van een dag en de

dagen van de week. Tot slot volgt een korte toelichting over de scoring van de gewenste situatie en de vergelijking met de nulmeting.

Figuur 21 toont de eerste 7- van de 22709 rijen uit de database.

- 'Interaction_id' kolom bevat een unieke code, voor iedere interactie.
- 'Publication_id' kolom bevat een unieke code, per folder (332 folders).
- 'Created_at' kolom bevat de datum + tijd waarop de interactie is aangemaakt.

1	interaction_id	publication_id	created_at
2	00045d6b-4d87-403a-9838-3d81487bb218	b4129ab1-0b13-483d-8e14-9d0a0dedd95f	13-3-2019 16:52
3	000a0236-f33e-4baf-9fd6-c98a697d15f0	72ef207f-635e-4bab-9858-7628cd4ca519	21-1-2019 09:51
4	000eddd2-be51-47bb-986a-7c9153782758	3f8dcd3d-2360-493e-a864-9b8304deddcc	28-2-2019 15:42
5	001435a1-c41d-42c1-be8d-971c1772a427	8767b689-f7c3-4045-81a8-31f7738bcfe1	27-2-2019 11:35
6	0014c343-ef1a-447f-8989-313ed199df1b	f53fcf5f-d0ab-40f5-b7bd-649c1cc702b3	3-1-2019 09:01
7	0016f1cb-7949-45f1-a0c3-5d318e141214	1bba57e9-76b0-4e30-b884-6bc10293c2f4	22-3-2019 10:35
8	0018001c-efd4-40fd-8908-d445e08ed352	9b97e7c6-9769-4a61-beae-48afafac9dee	18-1-2019 13:45

Figuur 21: Interactiedatabase gegevens

Gemiddelde tijd per interactie

Onder interactietijd wordt het volgende verstaan: de tijd tussen het loggen twee opeenvolgende interacties. Om deze te berekenen dient de data olopend gesorteerd te worden (oudste data bovenaan) op 'created_at'. Vervolgens kan de interactietijd weggeschreven worden door het verschil in tijd tussen opeenvolgende rijen te berekenen.

Uitschieters worden niet meegenomen door de data te filteren op interactietijden < 00:00:02 en > 00:01:00. Interactietijden lager dan 2 seconden zijn waarschijnlijk klikfouten (en geen werkelijke interacties), interactietijden hoger dan 60 seconden zijn waarschijnlijk interacties die gelogd zijn terwijl een medewerker afgeleid was (denk bijvoorbeeld aan wc, koffie of wandelen).

Uit de overgebleven 6418 interacties is 00:00:18 het gemiddelde.

Gemiddelde interactietijd, per folder

Onder interactietijd, per folder wordt het volgende verstaan: de totaal benodigde tijd om één folder te verwerken. Dit is op twee manieren te berekenen: 1.) bereken het tijdsverschil tussen de eerste interactie en de laatste interactie, per folder; 2.) sommeer de interactietijd, per folder.

Optie 1 is onnauwkeurig en biedt geen mogelijkheid om uitschieters te filteren. Optie 2 is nauwkeurig en biedt de mogelijkheid om irrelevante data te filteren. Optie 2 wordt gebruikt om de gemiddelde interactietijd, per folder te berekenen.

Eerst wordt de gefilterde data weggeschreven, per folder. Vervolgens wordt per folder het aantal bruikbare interactietijden geteld. Wanneer een folder >10 bruikbare interactietijden telt wordt de folder, met bijbehorende interactietijden genegeerd. Vervolgens wordt van de gefilterde folders de som weggeschreven en wordt het gemiddelde over de sommaties van alle gefilterde folders genomen.

Uit de overgebleven 94 folders is 00:19:02 het gemiddelde. In Tabel 7: KPI's nulmeting zijn de belangrijkste KPI's gescoord.

Tabel 7: KPI's nulmeting

Belangrijkste KPI's voor nulmeting:	Scoring
Gemiddelde tijd per interactie	00:00:18
Gemiddelde folder-interactietijd	00:19:02
Percentage publicaties op tijd	100%
Klachten per 100 folders	2

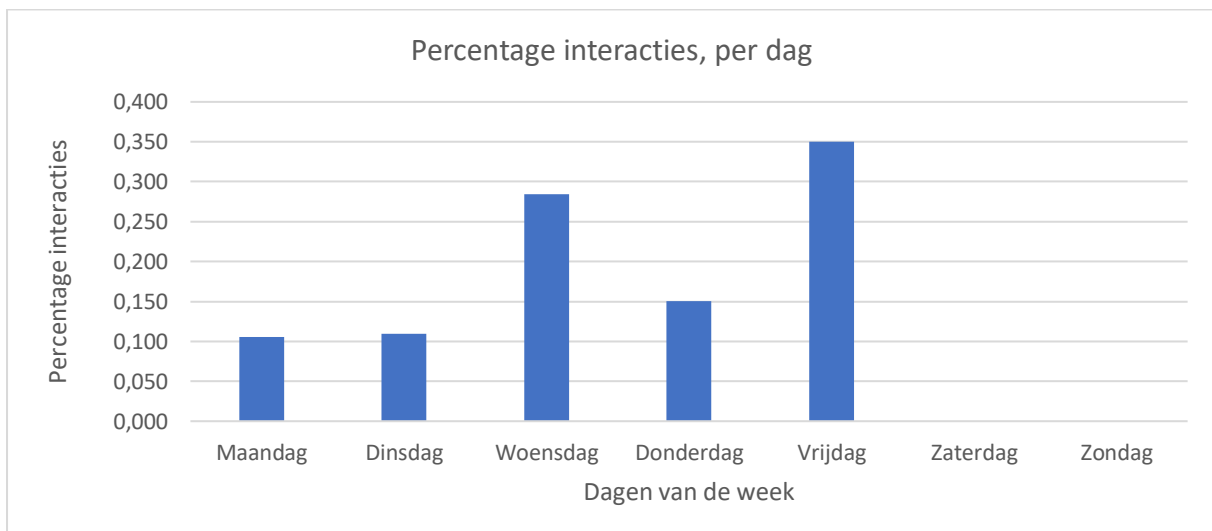
Overige inzichten:

Om inzicht te krijgen in de werkdruk, productiviteit en piekbelasting van de FMP employee zijn onderstaande grafieken weergegeven. De gefilterde interacties zijn geteld en geplot tegen de dag waarop de interactie is aangemaakt (Figuur 22).

Daarnaast is

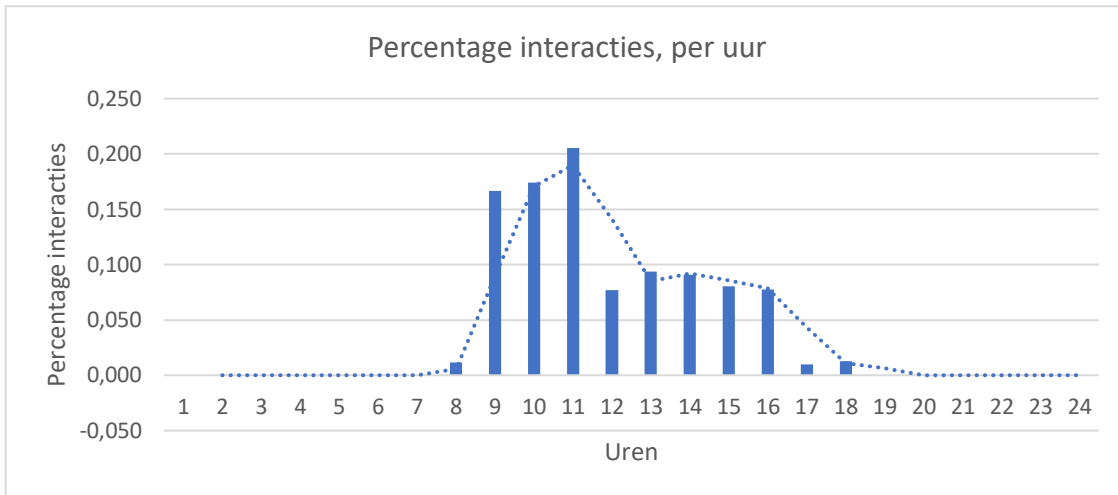
Figuur 23 geplot om inzicht te geven in de dagen die het meeste invloed hebben op het percentage interacties, per uur.

In alle drie de grafieken is gekozen om met percentages te werken in plaats van absolute aantallen, zodat een vergelijking in de toekomst makkelijker gemaakt kan worden.



Figuur 22: percentage interacties, per dag

Opvallend zijn de pieken op woensdag en vrijdag, die naar alle waarschijnlijk ontstaan door deadlines van retailers die op zondagmiddag en op woensdagavond liggen.

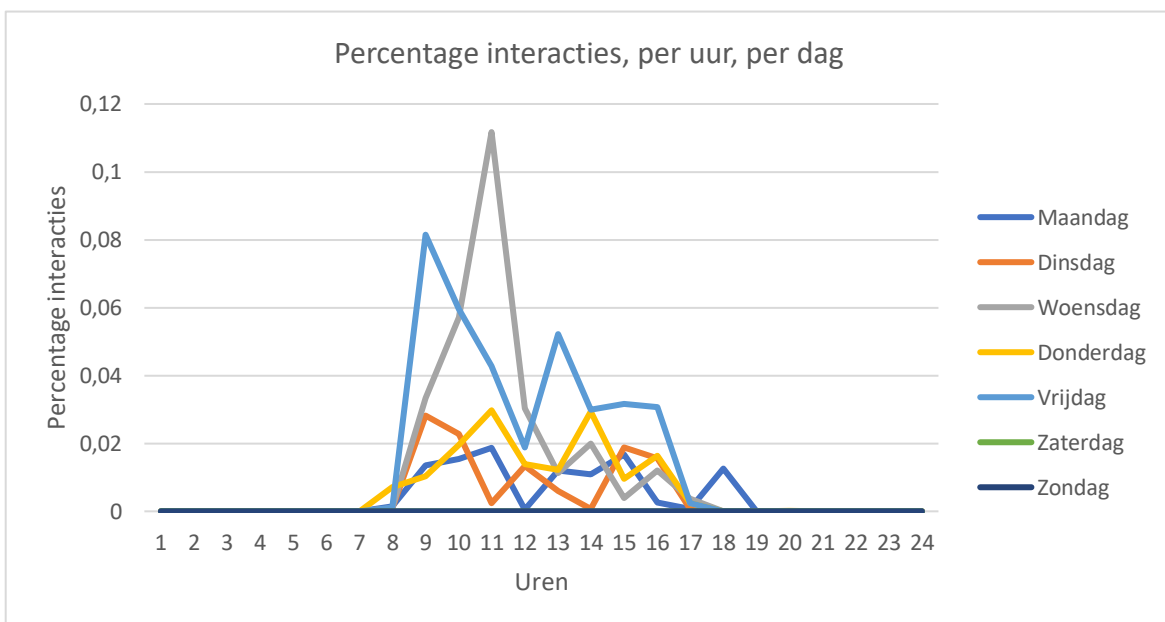


Figuur 23: Percentage interacties, per uur

Opvallend is de piek in de ochtend, ten opzichte van de middag. Het kan meerdere verklaringen hebben. Er zijn meer gesprekken en observaties nodig om hier een eenduidige oorzaak aan te hangen. Hieronder schets ik twee mogelijkheden waar de twee uitersten wellicht vandaan komen.

- 1.) Het kan het zijn dat de FMP employee 's ochtends inplant voor het aanmaken van interacties en de middagen met vergaderingen, waardoor het gemiddelde aantal interacties in de middagen sterk afneemt.
- 2.) De concentratie van de FMP employee neemt op een dag gestaag af.
- 3.) De interesse in het werk neemt op een dag gestaag af. Bijvoorbeeld omdat het werk eentonig is. Dit kan ook een correlatie hebben met 2.).

Opvallend is ook het dal op 12:00, wat eenvoudig te verklaren is met de middagpauze van 12:30 tot 13:30.



Figuur 24: Percentage interacties, per uur, per dag

Opvallend zijn de hoge pieken, voornamelijk in de ochtenden, op de woensdag en de vrijdag. Het kan meerdere verklaringen hebben. Er zijn meer gesprekken en observaties nodig om hier duidelijke oorzaken aan te hangen.

Zo zien we dat het aantal interacties op de woensdagmiddag 5x minder is dan het aantal interacties op de woensdagochtend. En zien we dat vrijdagmiddag veruit de meest productieve middag is.

Scoren van nieuwe situatie

In deze sectie wordt toegelicht op welke manier de scoring van de nieuwe situatie in z'n werk zal gaan. Deze scoring van de nieuwe situatie, en daarmee de vergelijking met de nulmeting, zal plaatsen op een moment in de toekomst dat ik niet meer werkzaam ben bij GreenOrange. Om op exact dezelfde wijze om te gaan zoals in de nulmeting is besloten om een code te schrijven die de berekeningen uit de nulmeting automatisch uit kan voeren.

De input dient te bestaan uit een .csv bestand met dezelfde drie kolommen, zoals te zien in Figuur 21. Vervolgens dient het analyse.py bestand (Python code) gerund te worden. Het .py bestand zet het .csv bestand om in werkbare matrixen en exporteert de output, zoals hieronder omschreven, naar een .xlsx bestand (Excel) voor visualisatie.

De output zal bestaan uit de volgende informatie:

- Gemiddelde tijd per interactie
- Gemiddelde folder-interactietijd
- Plot: percentage interacties, per dag
- Plot: percentage interactie, per uur
- Plot: percentage interacties, per uur, per dag

Tot slot dient de 'percentage publicaties op tijd' en 'het aantal klachten per 100 folders' opgevraagd te worden bij de medewerkers die gebruik maken van Salesforce.

De medewerkers van GreenOrange zijn op deze manier instaat om exact dezelfde analyses uit te voeren op een toekomstige set data. Er wordt vanuit gegaan dat GreenOrange zelf in staat is om de toekomstige output te analyseren en conclusies te trekken.

3.3. Samenvatting en conclusies

In hoofdstuk 2 is de huidige situatie uitgebreid omschreven. De eerste paragraaf geeft inzicht in het bedrijfsproces en geeft daarmee antwoord op de eerste deelvraag: "Hoe ziet het huidige WePublish proces eruit?".

De tweede paragraaf is gericht op het kwantificeerbaar maken van het bedrijfsproces. De 'gemiddelde tijd per interactie' en de 'gemiddelde folder-interactietijd', in combinatie met het 'percentage publicaties op tijd' en het aantal 'klachten per 100 folders' zijn gekozen als belangrijkste maatstaven om het bedrijfsproces te vergelijken met toekomstige versies van het proces.

In Tabel 8, op de volgende pagina, zijn de scoren van de KPI's weergegeven.

Tabel 8: Gemeten KPI's, nulmeting

Belangrijkste KPI's voor nulmeting:	Scoring
Gemiddelde tijd per interactie	00:00:18
Gemiddelde folder-interactietijd	00:19:02
Percentage publicaties op tijd	100%
Klachten per 100 folders	2

Aangezien in dit onderzoek de focus op de FMP group ligt worden hieronder de belangrijkste aspecten van dit procesonderdeel omschreven.

- Aangeleverde XML-bestanden kunnen het proces significant versneller. Naar verwachting tot wel 20 á 30 seconden per interactie.
- Het FMP editor proces is naar verwachting een langdradig en saai proces voor de FMP employee om uit te voeren.

Daarnaast zijn er geen eenduidige afspraken met verschillende retailers over de vorm waarin documenten en informatie aangeleverd dienen te worden (Sales group).

4. Gewenste situatie

Intro hoofdstuk

In dit hoofdstuk wordt de ideale variant van het FMP editor proces beschreven. Paragraaf 3.1 omschrijft het ideale proces aan de hand van een illustratie en benadrukt de verschillen ten opzichte van de huidige situatie. Vervolgens wordt in paragraaf 3.2 twee verschillende manieren geschetst om de gewenste situatie te behalen. In paragraaf 3.3. volgt de connectiviteit en interactie van de nieuwe software. Aangezien het benaderen van een ideale situatie vaak jaren kan duren wordt in paragraaf 3.4 een minimal viable product (MVP omschreven), waarin de easy wins worden belicht. Paragraaf 3.5 geeft inzicht in de verwachte veranderingen ten opzichte van de nulmeting en mogelijke gevaren waar rekening mee dient worden gehouden. Tot slot zal paragraaf 3.6 het hele hoofdstuk samenvatten.

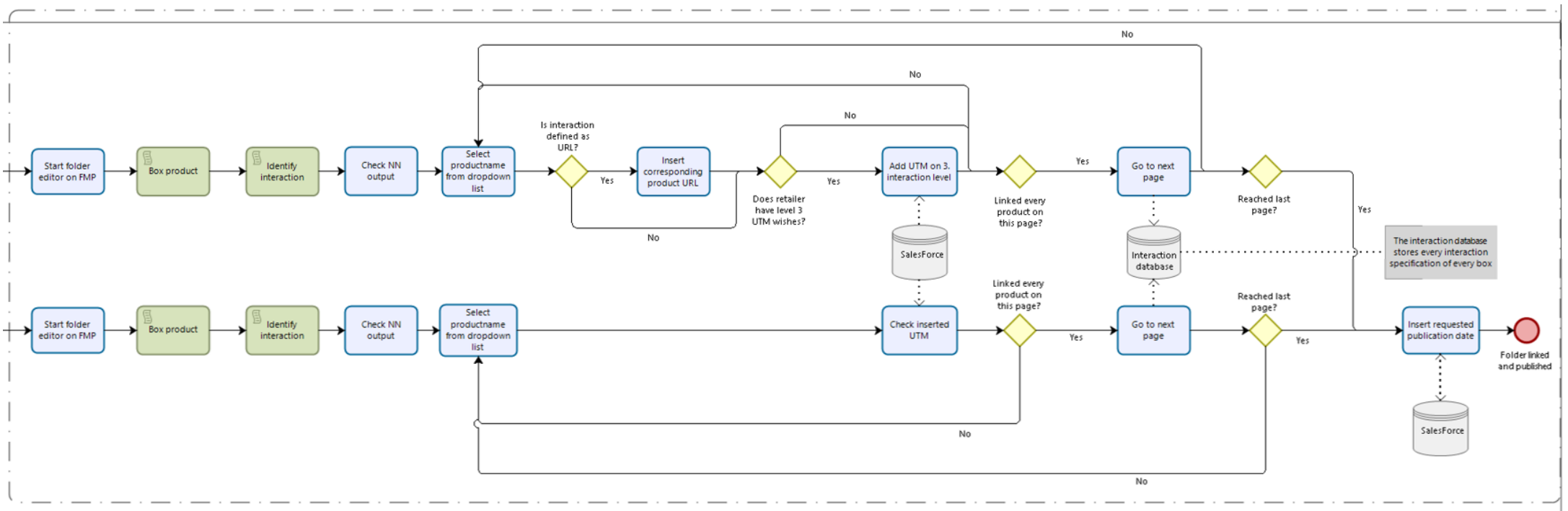
4.1. Ideale FMP editor proces

In het vorige hoofdstuk is een duidelijke splitsing van het FMP editor proces zichtbaar. In het nieuwe BPMN model, weergegeven in Figuur 25 is te zien dat een aantal handelingen uit het proces zijn veranderd, of volledig overgenomen zijn door software. Zowel 'boxproduct' als 'identify interaction' is met groen aangegeven (om op te vallen) en draagt de herkenning 'task door script'. De 'select productname from dropdownlist', uit het bovenste proces was vroeger 'name product in box'. 'Check output from NN' is een nieuwe task die toegevoegd is aan het proces van de FMP employee waarin de employee de resultaten van het NN controleert en indien nodig aanpast.

In het BPMN model is gekozen voor eenvoud, wat zich vertaalt in het achterwegen laten van een grote hoeveelheid redundantie aan 'personal tasks'. Deze zijn als default task genomen, net zoals in de huidige situatie.

In de omschrijving van het MVP wordt kort ingegaan op de manier waarop het proces het beste geïntegreerd kan worden.

FMP editor - group



Figuur 25: Gewenste FMP editor proces

4.1.1. Training en integratie van het NN

In onderstaande toelichting wordt vanuit gegaan dat het neurale netwerk (NN), of neurale netwerk (NNen) onafhankelijke zijn getraind. (In kennisvragen is uitgelegd wat een NN is en hoe het traint).

Integratie van het NN

Wanneer het NN getraind is kan een bestand worden gegenereerd waarin alle weights en biases zijn opgeslagen. Dit bestand wordt opgeslagen in een .pb bestand en wordt naar gerefereerd met 'frozen inference graph'. Zoals beschreven in 2.1.2. Neurale netwerken is de inference graph in feite niets anders dan een lange functie met bekende variabelen. Deze inference graph kan vervolgens gebruikt worden om uitspraken te doen over nieuwe data (nieuwe images). Calculaties op nieuwe data met behulp van deze inference graph kost ten opzichte van de NN-trainingen relatief weinig processorkracht. Calculaties, met behulp van de inference graph, noemen we vanaf nu labelcalculations (LC).

De meest logische integratie van de LC is om deze onderdeel te maken van het CDN. De input voor de inference graph zijn gecomprimeerde .jpg bestanden. Momenteel, in de huidige situatie, genereert het CDN al gecomprimeerde .jpg bestanden uit .pdf bestanden. Dit betekent dat enkel de LC aan het CDN toegevoegd dienen te worden voor het automatisch detecteren van producten op het FMP.

De output is een masker/hover waarin een gekleurde box om het product staat en productnaam in de vorm van een label wordt weergegeven. Het hover/masker en de .jpg worden zichtbaar wanneer de FMP-editor geopend wordt. De FMP-editor employee is instaat om de hover/masker coördinaten en de labels te overschrijven wanneer dit gewenst is. De bewerkingen worden, zoals standaard, opgeslagen in de interaction database. De kolommen die worden bijgehouden in de interaction database zijn: maximale en minimale x;y coördinaten, productlabel, en interactie informatie.

4.2. Mogelijke wegen + voordelen/nadelen

In deze sectie worden twee mogelijke manieren besproken om de gewenste situatie te bereiken.

4.2.1. 1NN

De eerste manier bestaat uit het creëren en trainen van één NN, NN1, die getraind is om boxen te plaatsen om zowel product als bijbehorende producttekst.

Aangezien de wens om producten in combinatie met bijbehorende tekst te boxen uniek is (geen open-source netwerken die hierop getraind zijn) dienen de .xml bestanden eigenhandig gecreëerd te worden.

Figuur 26 toont aan de linker zijde het .jpg bestand als input en aan de rechter zijde het .jpg bestand + hover/masker als output.



Figuur 26: Input en Output NN1

4.2.2. 2NN

Omschrijving

Een alternatieve manier bestaat uit een samenwerking van twee NN'en en een script dat de output van de 2 netwerken vergelijkt en samenvoegt. Dit geheel noemen we NN2.

Het eerste NN, NN2.1, wordt gebruikt om boxen om producten te plaatsen (in tegenstelling tot product + producttekst, door NN1). Er bestaan veel getrainde netwerken met deze capaciteit die open-source beschikbaar zijn en eenvoudig kunnen worden gedownload. In feite download je de inference graph van een getraind netwerk.

Het tweede NN, NN2.2, wordt gebruikt om boxen, om tekstboxen te plaatsen. Dit netwerk dient in staat te zijn om de leestekens uit het .jpg bestand om te zetten in een string. Ook in dit geval zijn veel netwerken met deze capaciteit open-source.

In onderstaande illustratie is hetzelfde .jpg bestand als input gebruikt als in optie1. Optie2 kent meerdere stappen voordat de gewenste output gegenereerd is.

Figuur 27 geeft links de output van NN2.1 links naar rechts, per stap aan wat de veranderingen zijn.



Figuur 27: Output NN2.1 en Output NN2.2

Vervolgens dienen twee dingen te gebeuren voordat de eindoutput van NN2 compleet is: 1.) de string van iedere box uit NN2.2 dient geanalyseerd en bewerkt te worden, 2.) het toegekende label aan de output van zowel NN2.1 als NN2.2 dient vergeleken te worden met elkaar.

- 1.) Het doel van de analyse is om één label aan een relevante string te hangen. De bewerking dient irrelevante strings, in combinatie met de box, te verwijderen en dient relevante strings te classificeren. Wanneer een box geen label krijgt toegewezen wordt de box verwijderd. Dit gebeurt van stap1, naar stap2 binnen NN2.2.

Deze analyse kan op twee manieren worden uitgevoerd:

- a. deterministisch programmeren met behulp van bijvoorbeeld een zeer uitgebreid if-else statement dat één label als output geeft, of
- b. een NN dat getraind is om relaties tussen woorden te vinden en op deze manier één label als output te geven.

Het resultaat, na deze bewerking, is weergegeven in Figuur 28.



Figuur 28: Output NN2.2 (bewerking1)

- 2.) Wanneer de labels uit NN2.1 en NN2.2 overeen komen (kan vergeleken worden met een eenvoudig if 'label1' == 'label2' statement) dienen de boxen gecombineerd te worden tot één box die zowel het product als de bijbehorende tekst omvat. Hiervoor wordt de maximale x- en y-coördinaten van de twee boxen vergeleken. De x,y-coördinaten die corresponderen met de buitenste zijdes worden gebruikt om de grenzen van de nieuwe box te definiëren. De labelnaam blijft hetzelfde. Deze bewerking is weergegeven aan de linker zijde van Figuur 29.

De uiteindelijke output van NN2 is weergegeven aan de rechter zijde van Figuur 29. Zoals rechts onderin te zien, zijn de spruiten niet helemaal correct in de box opgenomen door enkel de twee hierboven beschreven boxen met elkaar te verbinden. In het geval van de spruiten was de prijs namelijk niet opgenomen in de box van de producttekst.



Figuur 29: NN2.2 (stap2, vergelijking)

Doorontwikkeling

Een dergelijk probleem is te verhelpen door een aantal extra bewerking toe te voegen aan het hierboven omschreven proces.

- Tijdens het analyseren van strings, door NN2.2, dienen strings met prijzen gelabeld te worden met het label 'productprijs'.
- Voordat de output van NN2, zoals geïllustreerd in Figuur 29, visueel gegenereerd wordt dient een if-statement uitgevoerd te worden. Dit statement controleert of alle uiterste x- en y-coördinaten van de productprijsboxen binnen de coördinaten van de samengevoegde producttekst- en productboxen vallen. Wanneer dit niet het geval is wordt de productprijsbox samengevoegd met de dichtstbijzijnde samengevoegde box die geen productprijsbox als subset bevat.

Zo kan het ook voorkomen dat een andere relevante tekst net buiten de samengevoegde box valt, bijvoorbeeld de kg prijs van de appels. Dit kan op vergelijkbare wijze opgelost worden, als bij productprijs.

4.3. Minimaal levensvatbaar product (MVP)

Bovenstaande omschreven nieuwe situaties zijn te groot en hebben te veel componenten om in één keer te ontwikkelen. Het MVP in dit hoofdstuk omschrijft de minimale eisen om een werkend product te lanceren. Om de eisen van het MVP op te stellen zal allereerst een keuze gemaakt worden tussen de omschreven manieren van aanpak.

4.3.1 Afwegingen

De afweging tussen optie1 (de ontwikkeling van NN1) en optie2 (de ontwikkeling van NN2) zal kwalitatief worden omschreven. Hieronder worden van beide opties een aantal grote voor- en nadelen genoemd.

Optie1

Voordelen

- Aanpassingen op het NN dienen minder diep in het model gemaakt te worden.
 - o Eenvoudiger om deze optie te ontwikkelen.

Nadelen

- Creëren van gelabelde dataset kost veel tijd.
- Geen garantie dat het model gaat werken. De kans bestaat dat het NN laag scoort, omdat het geen patronen in de invoerpixels kan vinden.
- Veel rekenkracht nodig om het model te trainen

Optie2

Voordelen

- Geen training van NN nodig
 - o Minder rekenkracht nodig
 - o Geen eigenhandige gecreëerde dataset nodig
 - o Garantie dat het model werkt (is namelijk al bewezen)

Nadelen

- Meerdere componenten die nauw met elkaar samen dienen te werken
- Hoge mate van begrip van materie nodig om componenten met elkaar samen te laten werken. Aanpassingen dienen 'dieper' in het model gemaakt te worden.
- De if-statements zijn deterministisch. Wanneer een nieuwe situatie (aanpassing in de folderopbouw, door de retailer) zal opnieuw naar de code gekeken moeten worden.

4.3.2. Conclusie MVP

Voor mijn onderzoek heb ik beperkte tijd en beperkte kennis. Ook kan vanuit het bedrijf weinig inhoudelijke begeleiding en feedback worden gegeven op dit onderwerp. Daarnaast is het doel van dit onderzoek om een algemene methode te ontwikkelen die op meerdere object classificatie vraagstukken van toepassing is.

Ik heb daarom besloten om mij te focussen op optie1. Hieronder is het MVP, met als basis de omschrijving van optie1, uitgewerkt.

4.3.2 Omschrijving MVP

Het MVP bevat alleen de meest essentiële onderdelen van een product, zodat het werkt. De output van het MVP is in Figuur 30 weergegeven.

Eén retailer:

- De opmaak van iedere folder verschilt. Dit verschil is groter wanneer folders van verschillende retailers worden vergeleken. Wanneer folders van retailers uit verschillende retailerset worden vergeleken is dit verschil het grootste. Aangenomen dat een NN patronen herkent, zowel in de opmaak van de pagina, als in de manier waarop het product wordt weergegeven.

Daarom is gekozen om trainingen en testen van het NN toe te passen op folders van één retailer, in plaats van een retailerset, of de totale set retailers. Naar verwachting wordt het herkennen van patronen in de invoerpixels eenvoudiger en neemt de kans op succes toe.



Figuur 30: Output MVP

Productcategorieën

- Aangenomen dat een NN objecten classificeert aan de hand van patronen en dat een NN beter in staat is om patronen van elkaar te onderscheiden wanneer de totale set te onderscheiden patronen kleiner is. Daarom is gekozen om productcategorie labels te hanteren, in plaats van exacte productnamen als label. Naar verwachting wordt het onderscheiden van labels eenvoudiger wanneer de totale set labels minder groot is.

Daarnaast, om uniformiteit in de training- en testdata te creëren, dienen afspraken gemaakt te worden over de manier waarop een box wordt gecreëerd. Denk aan: *Welke componenten van een product vallen in een box en welke niet? Mogen boxen overlap hebben?*

De LC wordt, zoals beschreven bij het ideale proces, onderdeel van het CDN. Het schalen van de MVP en de creatie van nieuwe versies van een MVP staat verder beschreven in 'aanbevelingen en suggesties'.

4.4. Verwachte veranderingen bij implementatie MVP

In deze sectie wordt op kwantitatieve wijze beredeneerd op welke manier de implementatie van het MVP invloed heeft op de KPI's die behandeld worden in hoofdstuk 2.

De volgende stappen uit het BPMN-model zijn geautomatiseerd:

- Boxproduct
Zowel bij XML als No XML is dit proces geautomatiseerd.
- Identify interaction
Aangenomen dat het NN de url-interactie standaard plaatst is dit proces geautomatiseerd.
- Name product in box
XML: de dropdown list is minder lang (enkel de producten uit de gelabelde categorie worden weergegeven).
No XML: de naam van het product dient gekozen te worden uit een dropdown list (handeling is verandert).

Naar verwachting zal de interactietijd, zowel bij XML als No XML, verminderen. Deze daling zal echter niet groter zijn dan +/- 4 seconden.

Naar verwachting zal de interactietijd per folder verminderen. Deze daling zal meer merkbaar zijn dan de daling in interactietijd, omdat de som over de interactietijd per folder wordt getrokken.

Aangenomen dat een folder +/- 25 pagina's heeft en dat op een pagina +/- 4 producten staan, zal de interactietijd per folder verlagen met +/- $((20 \cdot 4) \cdot 4 / 60) = 5$ minuten. Dit vertaalt zich naar een daling van +/- 25%.

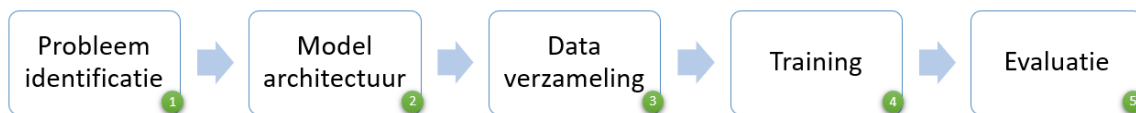
Naar verwachting zal de implementatie geen invloed hebben op zowel de foutmarge, als het percentage publicaties dat op tijd gepubliceerd wordt.

4.5. Samenvatting

De gewenste situatie is nog veel werk en wordt niet binnen een paar maanden bereikt. Daarom wordt een MVP ontwikkeld. De MVP maakt gebruik van één NN dat getraind gaat worden op één retailer en drie productcategorieën.

5. Ontwerp- en ontwikkelingsfase

In dit hoofdstuk staat het ontwerp en de ontwikkeling van het artefact, de methode met machine learning componenten, centraal. Allereerst zal de ontwikkelde methode visueel worden weergegeven met behulp van processtappen. Vervolgens zal iedere stap van de methode verder worden toegelicht in een aparte sectie. Bij iedere stap van de methode wordt systematisch 1.) de benodigde input, 2.) de benodigde bewerkingen en 3.) de vorm van de output beschreven. Per benodigde bewerking wordt een toelichting gegeven en, waar nodig, een keuzehulp aangeboden. Iedere bewerking wordt geconcludeerd met een 'rule of thumb' waarin een handvat wordt gegeven om de bewerking uit te voeren.



Figuur 31: Methode processtappen

5.1. Probleem identificatie (methodiek)

In deze stap wordt het probleem in de vorm van ML termen in kaart gebracht. De output van deze stap, het vastgestelde probleem, wordt in iedere andere stap direct, of indirect gebruikt. Tabel 9 toont als input verschillende componenten om met de methode te beginnen. Belangrijk is dat met behulp van de input een beeld ontstaat van het te automatiseren proces en dat een doel wordt gedefinieerd.

Tabel 9: Methode, stap1

Input:
Procesmodel van te automatiseren stappen uit proces
Interviews met betrokkenen
Kennis van verschillende ML termen en methoden
Gedefinieerd doel van de gebruiker

Bewerkingen:
Stel vast of de automatisering met behulp van traditioneel programmeren, of met behulp van een NN verwezenlijkt kan worden
Definieer het probleem in ML termen.
Stel de vorm en dimensie van de input en output data van het NN vast.

Output:
Gedefinieerd probleem in ML termen
<ul style="list-style-type: none"> Vorm en dimensie van de input en output data zijn onderdeel van de definitie Type ML probleem (set + subset)

5.1.1. Bewerkingen

1. Traditioneel Programmeren (TP) vs Machine Learning

Stel vast of het probleem aangepakt kan worden met TP, of dat ML toegepast dient te worden. Wanneer een probleem opgelost kan worden met zowel ML als TP en de resultaten zullen naar

verwachting vergelijkbaar zijn dan geniet TP altijd de voorkeur door de efficiënte manier waarop de output verkregen wordt (ML vergt veel energie en rekenkracht) en de verschillende nadelen en limitaties van ML die verder toegelicht zijn in 2. Theoretisch kader.

Om een goede afweging te maken tussen ML en TP wordt teruggegaan naar de definities en gebruiken van beide methoden.

- ML- Zoals omschreven in 2.1.1. Vakgebieden stelt ML computers instaat om algoritmes op te stellen, waarmee een nieuw datapunt voorspelt op basis van invoerdata. Hiervoor zijn grote hoeveelheden gelabelde data nodig.
- TP - In TP wordt een algoritme, met behulp van logische operators (if-else, OR, AND, etc.), geschreven. Met dit algoritme kan een volgend datapunt voorspelt worden op basis van ingevoerde data.

Het grote verschil tussen ML en TP, is dat ML duizenden parameters op kan nemen in het algoritme dat input naar output vertaalt. Deze grote hoeveelheid parameters maken het model, zoals omschreven in 2.1.3. Wiskundige notatie, nauwkeurig en stellen het model in staat om complexe patronen te ontdekken. In TP is het theoretisch mogelijk om ook duizenden parameters op te nemen, maar is dit onbegonnen werk. Daarnaast is een TP algoritme beperkt tot de kennis en het inzicht van de ontwikkelaar om patronen te herkennen in de aangeleverde data. Een ML algoritme kan patronen ontdekken die wij als mensen nooit voor mogelijk hadden gehouden of hadden kunnen zien/herkennen.

Rule of thumb:

Stel jezelf de vraag: Kan ik het probleem, praktisch haalbaar, oplossen met TP? Ga bij beantwoorden na of TP ondergeschikte resultaten zou opleveren, of een taak niet in volledigheid kan uitvoeren.

Ja Voer het probleem uit met TP-ontwikkelaars. *Deze stap kan je het beste uitvoeren met een adviserende TP-ontwikkelaar. De ontwikkelaar kan de praktische haalbaarheid van een TP oplossing logischerwijs redden.*

Nee Ga door naar de volgende bewerking binnen deze stap.

2. Definieer het probleem in ML termen

Het definiëren van het automatiseringsprobleem in ML termen is van groot belang van het kiezen van een passende modelarchitectuur en het verzamelen relevante data. Wanneer deze stap in het proces niet zorgvuldig wordt uitgevoerd zal het getrainde model waarschijnlijk onder verwachting presteren.

Met ML termen wordt de vaktaal van het ML vakgebied bedoelt. Het probleem definiëren in ML termen zorgt ervoor dat er efficiënt en effectief gecommuniceerd kan worden, zonder misopvattingen. Dit is noodzakelijk vanwege de grote diversiteit aan implementaties die ML te bieden heeft (en ieder jaar toe neemt).

Om het probleem te definiëren wordt onderscheid gemaakt tussen twee niveaus: 1.) onderzoeksrichting binnen het ML vakgebied (Tabel 10) en 2.) toepassingen binnen de onderzoeksrichtingen van ML. Iedere onderzoeksrichting kent haar eigen uitgebreide set aan toepassingen. Het valt buiten de scope van dit onderzoek om een complete lijst te geven van iedere mogelijke toepassing.

Deze bewerking richt zich verder op het behandelen van object detection problemen (toepassing) binnen de Computer Vision onderzoeksrichting.

Tabel 10: Onderzoeksrichtingen van ML

Type ML problemen	Voorbeelden
Natural Language Processing	Spraakherkenning Tekst interpretatie
Computer Vision	Object detection Object classification Image/video enhancement (verbetering)
Algemene voorspelling	Machine breakdown Aardbevingen Verkoopcijfers Gepersonaliseerde aanbevelingen

Rule of thumb:

Classificeer de onderzoeksrichting van jouw probleem door jezelf de vraag te stellen: Betreft mijn vraagstuk de verwerking van natuurlijke taal, computer zicht, of algemene voorspellingen? Start vervolgens een mini-onderzoek waarin je iedere keer opzoek gaat naar de subset waarin jouw probleem zich bevind. In Figuur 32 is weergegeven hoe een dergelijk mini-onderzoek er voor ons object detection probleem uitziet.



Figuur 32: Mini onderzoek ten behoeve van onderzoeksrichting classificatie

3. Dimensies van data

De dimensies van de input- en output data zijn sterk afhankelijk van het type ML probleem dat je probeert op te lossen. Voor een object detection probleem is de input data altijd in de vorm van een image (bijvoorbeeld .jpg) en een bijbehorend label in de vorm van een XML bestand (2.1.4. Computer vision).

De inputdimensie van het model is afhankelijk van de resolutie van de image (en kan per image verschillen). De relatie tussen image resolutie, accuraatheid en benodigde computerkracht zijn in Tabel 11 weergegeven (de image resolutie is hier arbitrair gekozen).

De dimensies van de output data bestaan bij object detection over het algemeen uit de verschillende klassen die gedetecteerd dienen te worden.

Tabel 11: Relatie tussen image resolutie, accuraatheid en benodigde computerkracht (afgeleid uit CNN theorie)

Image resolutie	Relaties
Hoge resolutie images (1000x1000)	Hogere accuraatheid Hoge calculaties
Lage resolutie images (256x256, 600x600)	Hoge accuraatheid Gemiddelde calculaties
Extreem lage resolutie images (28x28, 80x80)	Gemiddelde accuraatheid bij niet complexe images Relatief lage accuraatheid bij complexe images Lage calculaties

Rule of thumb:

Kies de dimensies voor de inputdata (resoluties), zodat het overeen komt met de resoluties van de images die het model na de training dient te analyseren. Wanneer een model verschillende image resoluties moet verwerken is het aan te raden om een mix van image resolutie te gebruiken voor de training. Voorbeeld: Je wilt een kat, een hond en een varken detecteren in images van 150x150.

Gebruik hierbij inputdata met resoluties van 256x256 en 128x128 (of enkel 150x150).

Kies de dimensies voor de outputdata (aantal klassen te detecteren), zodat het overeen komt met het aantal verschillende labels dat je hebt toegevoegd aan je inputdata. In hetzelfde voorbeeld bestaat de outputdimensie uit drie klassen (=kat, hond en een varken).

Over het algemeen kan je het beste lage resolutie images (256x256) gebruiken, omdat het de rekenkracht t.o.v. 1000x1000 sterk verlaagd en in accuraatheid maar een paar procent (+-2%) verliest. Enkel in gevallen waar deze 2% het verschil is tussen leven en dood, bijvoorbeeld bij de detectie van huidkankercellen bij patiënten, is het de moeite waard.

5.2. Model architectuur

In deze paragraaf wordt de model architectuur van het model vastgesteld en gedownload. De methode biedt vervolgstappen voor de niveaus 1 en 2, maar verschaft geen ondersteuning bij selectie van niveau 3 of 4. Voor deze keuze is het geïdentificeerde probleem in ML termen essentiële input, de beschikbare hardware speelt ook een rol. Bij selectie van niveau1, ga door naar stap5 van de methode, bij selectie van niveau2, ga door naar stap3 van de methode.

Tabel 12: Methode, stap2

Input:
Gedefinieerd probleem in ML termen <ul style="list-style-type: none">• <i>Type ML probleem (set + subset)</i>
Bewerkingen:
Welke hardware staat er tot de beschikking?
Onderzoek of de oplossing al bestaat in de vorm van een 'open source' NN. Classificeer het tot één van de volgende niveaus en voer de bijbehorende bewerking uit. <ul style="list-style-type: none">a) Niveau 1: Kies en download architectuurb) Niveau 2: Kies en download architectuur, verwijder de parameter waardesc) Niveau 3: Kies en download architectuur, verwijder de parameter waardes en koppel de toevoeging aan de gedownloade architectuurd) Niveau 4: Schrijf benodigde architectuur
Output:
Model architectuur: <ul style="list-style-type: none">• Niveau 1: Model architectuur, inclusief getrainde parameters → ga naar stap 5.• Niveau 2, 3, 4: Model architectuur zonder getrainde parameters

5.2.1. Bewerkingen

1. Hardware

Met hardware wordt voornamelijk verwezen naar GPU kracht van een computer. GPU's voeren de benodigde berekeningen 4 á 8 keer sneller uit dan vergelijkbare CPU's. Beschikbare hardware kan sterk uiteenlopen, zo is lokale hardware vaak zwakker dan externe hardware (cloudcomputing oplossingen van gespecialiseerde bedrijven). Cloudcomputing kan de training van een uitgebreide dataset terugbrengen van enkele dagen, naar enkele minuten. In deze bewerking dient een keuze gemaakt te worden tussen gebruik van lokale hardware en externe hardware

Rule of thumb:

Wanneer voor het eerst met ML software geprogrammeerd wordt is het verstandig om te beginnen op lokale hardware. Op deze manier wordt een beeld gevormd over de verschillende applicaties en API's die met elkaar samenwerken om de hardware op de juiste manier aan te sturen. Ook wordt op deze manier een gevoel ontwikkeld voor de benodigde rekenkracht die bij de verwerking van de data komt kijken. Wanneer vaker met ML software geprogrammeerd wordt is het

geadviseerd om de bewerkingen uit te voeren door middel van cloudcomputing. Op het gebied van cloudcomputing oplossingen is veel keuze, zowel gratis varianten, als betaalde varianten².

2. Niveau classificatie

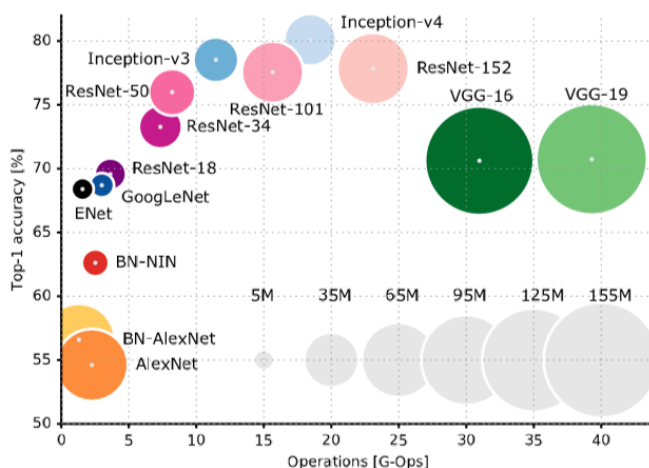
Binnen deze bewerking wordt de niveau classificatie van het geïdentificeerde probleem vastgesteld. Er wordt onderscheid gemaakt tussen in bruikbaarheid van modelarchitectuur en bruikbaarheid van voorgetrainde parameters (binnen de kozen modelarchitectuur). *Zo staat niveau1 voor het overnemen van zowel de architectuur, als de parameters en niveau4 voor het compleet zelf ontwerpen van de architectuur en zelf training van de parameters.*

Voor dit besluit is het belangrijk om een beeld te hebben van bestaande modelarchitecturen en waarop deze modellen getraind zijn. Binnen het version control system (versie controle systeem) GitHub³ waarop ontwikkelaars projecten met elkaar kunnen delen is een overzicht gegeven van bewezen modelarchitecturen in combinatie met voorgetrainde parameters op bepaalde data trainingssets⁴.

Rule of thumb:

Allereerst wordt gekeken naar de bruikbaarheid van de parameters. Dit kan je controleren door de vraag te stellen: Is de inputdata van de verschillende data trainingssets uit de GitHub repository vergelijkbaar met de input van het ML-probleem voor handen?

- Ja Selecteer de dataset die overeenkomt met het gedefinieerde ML-probleem.
- Nee Concludeer dat voorgetrainde parameters geen optie zijn en selecteer vervolgens de dataset die het meest overeenkomt met het gedefinieerde ML-probleem (Niveau 1 valt bij deze keuze concludering af). Hoogst waarschijnlijk zijn de modelarchitecturen gespecialiseerd om dit type patronen in de inputdata te herkennen.



Figuur 33: Accuraatheid tegen benodigde berekeningen (A. Canziani, 2016)

76

² https://aws.amazon.com/machine-learning/?nc2=h_m1

<https://cloud.google.com/gpu/>

<https://azure.microsoft.com/en-us/services/machine-learning-service/>

³ <https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>

⁴ https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md

76

Je hebt zojuist een keuze gemaakt voor de parameters van het model. Nu wordt een keuze gemaakt voor de modelarchitectuur. Deze keuze is een afweging tussen de accuraatheidsscore van het model, in afweging met de snelheid van het model en de beschikbare rekenkracht van de hardware. Figuur 33 kan gebruikt worden bij de afweging: op de y-as de accuraatheid van het model, op de x-as de benodigde berekeningen (andere indicatie voor trainingstijd).

Stel jezelf de vraag: Welke modelarchitectuur levert het beste resultaat voor het gedefinieerde ML-probleem? *Selecteer één van de modelarchitecturen binnen de geselecteerde dataset.*

In het geval dat externe rekenkracht wordt ingekocht (vorige bewerking), is de beschikbare rekencapaciteit in theorie oneindig. In dit geval wordt aangeraden een kosten-, baten analyse uit te voeren (bijvoorbeeld of een kleine toename in accuraatheid een relatief sterke toename in kosten waard is).

5.3. Data verzameling

In deze paragraaf wordt gefocust op de verzameling van data en, indien nodig, de bewerking van de verzamelde data. Ook hier is het geïdentificeerde probleem in ML termen essentiële input om de dimensies van de data, de type data en de eventuele bewerkingen van de data die nodig zijn vast te stellen.

Tabel 13: Methode, stap3

Input:
Gedefinieerd probleem in ML termen <ul style="list-style-type: none"> • <i>Vorm en dimensie van de input en output data zijn onderdeel van de definitie</i>
Bewerkingen:
Verzamel te bewerken data
Bewerk verzamelde data <ol style="list-style-type: none"> 1) Converteer data naar bruikbaar formaat en dimensie 2) Label de data 3) Verdeel data over een testset (20%) en een trainingset (80%) <p><i>Optioneel:</i></p> <ol style="list-style-type: none"> a) <i>Voer data augmentation uit op data</i>
Output:
Gelabelde data in trainingset
Gelabelde data in testset

5.3.1. Bewerkingen

1. Verzamel te bewerken data

Het verzamelen van rauwe data is vaak niks anders dan het aanvragen van een query in een database of het kopiëren van bestaande bestanden. Converteer de bestanden indien noodzakelijk naar het juiste formaat middels standaard tools: bijvoorbeeld van .pdf naar .jpg⁵.

2. Bewerk verzamelde data

In het geval van het object detection probleem dient ten eerste de resolutie van de inputdata gekropt (verkleint) te worden tot de gewenste resolutie. Dit kan door middel van verschillende online tools, maar ook middels het 'resizer.py' bestand dat aangeleverd is bij de methode. Ten tweede dient de data gelabeld te worden. Het labelen bij een object detection probleem is een tijdrovend proces, waarbij secuur gewerkt dient te worden. Het LabelImage.exe programma⁶ is een uitstekende tool om een label aan een bepaalde box pixels te hangen en hier automatisch een .xml bestand bij te genereren.

Tot slot, wanneer de data volledig bewerkt is dient de totale gelabelde dataset opgesplitst te worden in twee sets: trainingset en testset. De trainingset is de set waarmee het model getraind gaat worden en dient +-80% van de totale data te bevatten. De testset is de set waarmee het model gevalideerd zal worden en dient +-20% van de totale data te bevatten. In Figuur 34 is een illustratie van de

Naam	Datum	Type
image107	26-6-2019 10:54	JPG-bestand
image107	26-6-2019 13:39	XML-document
image108	26-6-2019 10:56	JPG-bestand
image108	26-6-2019 13:42	XML-document
image109	26-6-2019 10:56	JPG-bestand
image109	26-6-2019 13:43	XML-document
image110	26-6-2019 10:58	JPG-bestand
image110	26-6-2019 13:43	XML-document

Figuur 34: Illustratie trainingset

trainingsset in Windows Verkenner weergegeven.

Optioneel:

Data augmentation, een strategie die bewerkingen op de input images uitvoert (bijvoorbeeld draaien, spiegelen, of de aanpassing van kleurgradiënten) en hiermee uit de bestaande set images een grotere set images creëert⁷. Data augmentation kan worden toegepast na het labelen van de data en wordt sterk aangeraden bij kleine datasets (+-200 images). Het model heeft hierdoor meer nuttige inputdata om patronen in te herkennen, dit resulteert in een meer accurate output van het

⁵ <https://pdf2jpg.net/>

⁶ <https://github.com/tzutalin/labelimg>

⁷ <https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced>

model. Verschillende typen data augmentation en hun effectiviteit worden besproken door onder andere Wang, in 2017 (J.Wang, 2017).

5.4. Training

De trainingsfase focust zich op de instellingen en praktische uitvoering van de training. De input bestaat uit de gelabelde en bewerkte data uit de vorige stap en de geselecteerde modelarchitectuur uit stap2.

Tabel 14: Methode, stap4

Input:
Model architectuur: <ul style="list-style-type: none">• Niveau 2: Model architectuur zonder getrainde parameters
Gelabelde data in trainingset
Bewerkingen:
Alloceer/reserveer benodigde hardware
Stel hyperparameters van het model in
Run training
Exporteer de parameters naar een inference graph
Output:
Getraind model (afgestelde parameters) in gekozen architectuur

5.4.1. Bewerkingen

1. Alloceer/reserveer benodigde hardware

Bevestig dat je gebruik kan maken van beoogde hardware en dat deze hardware niet overbelast gaat worden door de uitvoering van je training (run dus geen andere dingen tegelijk).

2. Stel hyperparameters van het model in

Hyperparameters zijn de enige parameters die handmatig invloed hebben op de training, en daarmee op de afstelling van het model. De andere parameters, de weights en biases, bevinden zich in het model, de hyperparameters bevinden zich buiten het model en zijn als het ware de 'knoppen' waar de ontwikkelaar aan kan 'draaien' om het model beter te laten presteren.

Tabel 15, op de volgende pagina, verschaft een overzicht van alle beschikbare hyperparameters, aangenomen dat de gebruiker in de vorige stap niveau 1 of niveau 2 heeft gekozen, en welke invloed ze op het model hebben.

Tabel 15: Hyperparameters en hun invloed

Hyperparameter	Invloed op model
Learning rate (leer snelheid)	De snelheid waarmee naar het minimum van de functie wordt toegewerkt. <i>De leersnelheid is de stapgrootte uit Figuur 4.</i>
Aantal classes	Bepaald de dimensies van de outputdata. Bijvoorbeeld het aantal classes dat geïdentificeerd dienen te worden.
Aantal trainingsvoorbeelden	Bepaald de dimensies voor de inputdata. Bijvoorbeeld het aantal instanties van trainingsdata.
Basis van het netwerk	Bepaald de basisarchitectuur die gebruikt wordt. Bijvoorbeeld 'inception-v2'.
Image resolutie	Bepaald de resolutie waarin de inputdata, in de vorm van images, wordt omgezet.
Epoch (trainingsiteratie)	Het aantal keer dat de trainingsdata herhaald wordt ten behoeve van de training.

3. Run de training

Run het training.py bestand in cmd (command prompt) dat bij het gedownloade model wordt verschaft. Controleer de loss-value die berekent wordt bij de verwerking van inputdata en onderbreek de training bij een gewenste loss-value. Meer informatie over de loss-value is te vinden in 2.1.2. Neurale netwerken.

4. Exporteer de parameters naar een inference graph

Run het export_inference.py bestand in cmd en wacht tot parameters geladen zijn in het nieuw gegenereerde frozeninference.pb bestand. Meer informatie over inference graph is te vinden in 4. Gewenste situatie.

5.5. Evaluatie

De evaluatiestap analyseert de output van het voorgetrainde model uit niveau1, of het zelf getrainde model uit niveau2. De analyse wordt uitgevoerd door het verwerken van de testset, om vervolgens de output arbitrair te analyseren en eventueel met behulp van een accuraatheidsscore te vergelijken met andere modellen. De input is het getrainde model (inclusief model architectuur) en de gelabelde data van de testset.

Tabel 16: Methode, stap5

Input:
Getraind model in gekozen architectuur
Gelabelde data in testset
Bewerkingen:
Alloceer/reserveer benodigde hardware (nu is dit veel lager)
Run het getrainde model op de data uit testset
Analyseer resultaten (wat scoort het model op onbekende data?)
Keuzemoment: scoort het model naar behoren?
Output:
Score (indicatie van nauwkeurigheid) op onbekende data
Middelen voor implementatie

5.5.1. Bewerkingen

1. Alloceer/reserveer benodigde hardware

Bevestig dat je gebruik kan maken van beoogde hardware en dat deze hardware niet overbelast gaat worden door de uitvoering van je training (run dus geen andere dingen tegelijk).

2. Run het getrainde model op de data uit testset

Run het test.py bestand in cmd en wacht tot he getrainde model alle inputdata verwerkt heeft.

3. Analyseer resultaten

De analyse van de testset wordt op twee manieren gedaan. 1.) Ten eerste, controleer de resultaten visueel. Stel de vragen: Wat valt op aan de locaties van de bounding boxes? Wat valt op aan de toegekende labels en de bijbehorende procentuele zekerheden?

2.) Controleer vervolgens de validatie waarde die het test.py bestand genereert voor de testset van het getrainde model. In 2006 heeft onder andere Yilmaz een validatiemethode ontworpen die in één getal de prestatie van een getraind model, ten opzichte van de testset weergeeft (E. Yilmaz, 2006).

De validatiewaarde wordt uitgedrukt in mAP (mean Average Precision). De Average Precision (AP) is een eenheid die een waarde geeft voor de accuraatheid van de bounding box locaties en het

toegekende label⁸. Met de mAP wordt de gemiddelde AP van iedere verwerkte image genomen om de accuraatheid van een getraind model in één eenheid weer te geven.

Rule of thumb:

Start met de visuele analyse van de resultaten en noteer op arbitraire wijze opvallende bevindingen. Bijvoorbeeld dat het model moeite heeft met één labelklasse, of dat het model bepaalde labels alleen herkent bij een bepaalde achtergrond(kleur). Gebruik voor deze vorm van analyse gezond verstand.

Ga na of je de tweede vorm van analyse überhaupt nodig hebt in jouw probleem identificatie. Wil je meerdere modellen vergelijken? Zo ja, dan is deze vorm van analyseren voor jou interessant.

Wanneer je meerdere getrainde modellen genereert, door aan de hyperparameters te draaien bijvoorbeeld, kan je verschillende modellen met elkaar vergelijken door de accuraatheid van de verschillende modellen uit te drukken in mAP. Ook kan je bij selectie van niveau2, jouw mAP vergelijken met de mAP van voorgetrainde modellen (niveau1).

4. Keuzemoment: scoort het model naar behoren?

Stel de vraag: Scoort het model naar behoren? Ga hierbij na of je doel is bereikt en het resultaat dat je geanalyseerd heb in de vorige bewerking praktisch bruikbaar is.

- a.) Ja, ga door naar stap 6: implementatie
- b.) Nee, onderzoek met behulp van trial en error waardoor het model niet naar behoren scoort
 - 1) Run iteraties van stap 2 tot en met 5 en pas naar eigen inzicht opties aan. Probeer bijvoorbeeld verschillende combinaties tussen modelarchitectuur (stap2), dataweergave (stap3) en hyperparameters tijdens training (stap4). Evalueer iedere uitkomst van een optie in stap 5.
 - 2) Werkt het model na de meest aannemelijke opties tussen stap 2 tot en met 5 geprobeerd te hebben niet naar behoren?
 - i. Stel jezelf de vraag of stap 1 correct is uitgevoerd
 - ii. Overweeg professioneel advies van een autoriteit op het gebied van object detection en neurale netwerken
 - iii. Accepteer dat het model niet werkt en concludeer dat automatisering met behulp van een neurale netwerk in dit geval niet mogelijk is.

⁸ <https://medium.com/@timothycarlen/understanding-the-map-evaluation-metric-for-object-detection-a07fe6962cf3>
<https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52>

5.6. Implementatie

De implementatiestap is de laatste stap van de methode en gaat er vanuit dat het getrainde, of gedownloade model naar behoren functioneert. Deze stap focust zich op het verplaatsen van lokale bestanden (het model en haar parameters) naar een online omgeving (een server), zodat meerdere gebruikers de automatiseringssoftware (het getrainde model) kunnen gebruiken.

De stap is minimaal omschreven, omdat het veel gebruik maakt van TP en daarmee voor een groot deel buiten de scope van het onderzoek valt.

Tabel 17: Methode, stap6

Input:
Getraind model (afgestelde parameters) in gekozen architectuur
Middelen voor implementatie
Bewerkingen:
Plaats het getrainde model op een server
Ontwikkel online omgeving met gebruiksvriendelijk interface, zodat inputdata eenvoudig omgezet kan worden naar outputdata
Test omgeving en koppeling met getrainde model
Output:
Getraind model in gekozen architectuur binnen een gebruiksvriendelijke, online omgeving

5.6.1. Bewerkingen

1. Plaats getrainde model op server

Het .pb bestand met de inference graph kan in combinatie met het model.py bestand dat de modelarchitectuur bevat geüpload worden op een server. Vervolgens kan met behulp van een Python-pakket, zoals Flask, een mini applicatie gemaakt worden die een API (Application Programming Interface) vormt tussen de inference graph en de modelarchitectuur⁹.

2. Ontwikkel gebruiksvriendelijke online interface

Na het uploaden van de mini applicatie kan de applicatie, of webpagina, opgemaakt worden met een standaard opmaaktaal, zoals CSS (Cascading Style Sheets).

3. Test omgeving, inclusief koppeling met getraind model

Na het uploaden en opmaken van applicatie, of webpagina, is het eindresultaat bereikt en kan gecontroleerd worden of alles naar behoren werkt. Komt de uitkomst van het geüpload model, inclusief opmaak, overeen met de analyse van resultaten in H5.5.

⁹ <https://www.fullstackpython.com/flask.html>
<https://palletsprojects.com/p/flask/>
<https://cv-tricks.com/how-to/freeze-tensorflow-models/>

5.7. Samenvatting en conclusie

In deze paragraaf wordt de ontwikkelde methode, per stap, kort samengevat en worden de belangrijkste, of meest opvallende, punten in de vorm van een conclusie benoemd.

- 1.) In stap 1 wordt het probleem geïdentificeerd in de vorm van ML termen. Voordat dit gedaan wordt dient afgewogen te worden of het probleem met TP opgelost kan worden.
- 2.) In stap 2 wordt de modelarchitectuur gekozen. In deze stap wordt onderscheid gemaakt tussen verschillende niveaus en vastgesteld dat deze methode enkel uitkomst biedt bij een compleet te downloaden model inclusief getrainde parameters en een compleet te downloaden model zonder getrainde parameters.
- 3.) In stap 3 wordt uitgelegd op welke manier de gebruiker om dient te gaan met de dataset en op welke manier data gelabeld kan worden. Belangrijk is om
- 4.) In stap 4 wordt praktisch ingegaan welke bestanden gerund dienen te worden bij de training van het model en waarop gelet dient te worden.
- 5.) In stap 5 worden twee manieren gegeven waarop de uitkomst van een model geanalyseerd kan worden. De belangrijke analyse is arbitrair en visueel. De tweede is met behulp van de mAP score, die voornamelijk uitkomst biedt bij relatieve vergelijkingen.
- 6.) In stap 6 worden handvaten aangereikt ter implementatie van het getrainde model. Flask is over het algemeen de meest gebruikte package binnen Python om de inference graph en de modelarchitectuur te combineren in een webapplicatie.

6. Demonstratiefase – Proof of concept

In deze sectie wordt de ontwikkelde methode uit hoofdstuk 5, per stap, doorlopen en toegepast op het GreenOrange vraagstuk: de WePublish-service. De sectie sluit af met de resultaten die middels de proof of concept verkregen zijn.

Per stap wordt de rule of thumb van een bewerking kort herhaald, om vervolgens bij keuze de uitgevoerde handeling, of gemaakte keuze toe te lichten.

Stap 1 – Probleem identificatie

- **Bewerking 1:**

Stel jezelf de vraag: Kan ik het probleem, praktisch haalbaar, oplossen met TP? Ga bij beantwoorden na of TP ondergeschikte resultaten zou opleveren, of een taak niet in volledigheid kan uitvoeren.

Keuze: Nee, uit de literatuur omtrent object detection automatisering blijkt dat TP ondergeschikte resultaten zal leveren bij een maximale programmeringsinspanning.

- **Bewerking 2:**

Stel jezelf de vraag: Betreft mijn vraagstuk de verwerking van natuurlijke taal, computerzicht, of algemene voorspellingen? Start vervolgens een mini-onderzoek binnen literatuur (of internet=Google) waarin je op zoek gaat naar de subset waarin jouw probleem zich bevind.

Keuze: Het vraagstuk betreft computerzicht. Uit het mini-onderzoek blijkt dat binnen computerzicht wordt onderscheid gemaakt tussen de verwerking van foto's en video's: het vraagstuk betreft de bewerking van foto's.

Vervolgens wordt binnen de verwerking van foto's onderscheid gemaakt tussen de visuele bewerking van foto's, de analyse van foto's en de creatie van foto's. Het vraagstuk betreft de analyse van foto's.

Tot wordt binnen de analyse van foto's onderscheid gemaakt tussen object detection, object classification en object identification (benoemd in 2.1.3. Computer vision). Het vraagstuk betreft object detection.

Het hierboven omschreven mini-onderzoek is visueel weergegeven in Figuur 32.

- **Bewerking 3:**

Kies dezelfde input dimensie als de output dimensie. Realiseer dat lage resolutie images de training van het model versnellen, maar accuraatheid verlagen.

Keuze: Er is gekozen voor geschaalde images met een formaat van 567x850 pixels. Dit is een arbitrair gekozen resolutiewaarde die accuraatheid van de outputdata boven snelheid verkiest.

Stap 2 – Model architectuur

- **Bewerking 1:**

Er wordt aangeraden om met lokale hardware te experimenteren wanneer dit de eerste keer is dat met ML software gewerkt wordt. Wanneer vaker met ML is gewerkt wordt aangeraden om gebruik te maken van cloudcomputing.

Keuze: Dit is de eerste keer dat met ML software gewerkt wordt, daarom wordt lokale hardware gebruikt. De beschikbare GPU is de: Quadro K1100M, met een memoryClockRate

van GHz 0.7055 GHz en een totaal geheugen van 2GB. Vergeleken met de GPU's van de getrainde modellen uit de GitHub repository is dit ongeveer 85x langzamer.

- **Bewerking 2:**

Binnen deze bewerkingen dienen twee keuzes gemaakt te worden. Per keuze volgt de rule of thumb.

1. Stel jezelf de vraag of de verschillende datatrainingsets uit de GitHub repository vergelijkbaar zijn met het ML-probleem dat voor handen is.

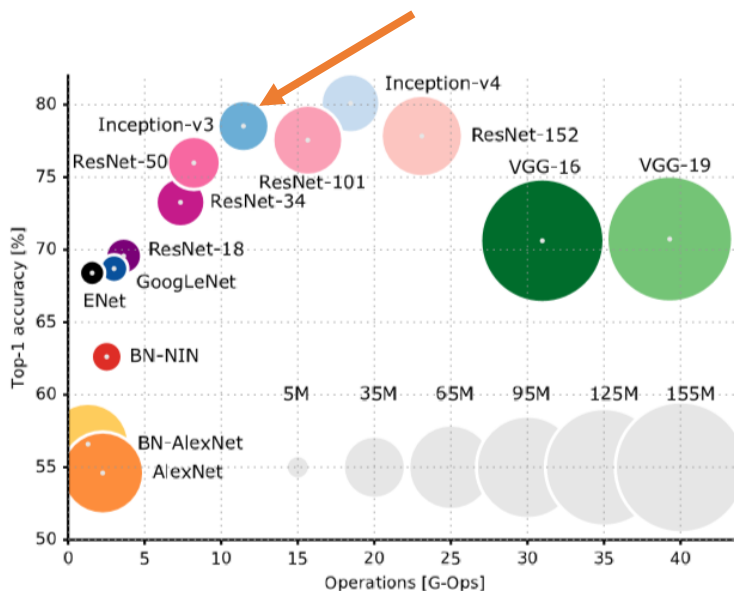
Keuze1: De datasets die in de GitHub repository zijn gebruikt zijn niet vergelijkbaar met de inputdata voor het probleem dat voor handig is. Geen enkele dataset is namelijk specifiek getraind op het herkennen van producten, inclusief bijbehorende tekst (wél op het herkennen van enkel producten). Hierdoor valt de keuze uit niveau 1 af.

COCO (Common objects in Context = veel voorkomende voorwerpen in context) komt echter het meest bij het ML-probleem dat voor handig is in de buurt. Deze dataset gebruikt als input veel voorkomende voorwerpen in context, wat sterk overeenkomt met het herkennen van producten, inclusief tekst, in een folder. Daarom wordt COCO geselecteerd.

2. Stel jezelf vervolgens de vraag: Welke modelarchitectuur levert de hoogste accuraatheid, bij een zo laag mogelijk aantal benodigde calculaties voor het gedefinieerde ML-probleem?

Keuze2: De Inception V3 modelarchitectuur, aangeduid met de oranje pijl in Figuur 35, is arbitrair gekozen. In de keuze is de voorkeur gegeven voor accuraatheid ten opzichte van de trainingssnelheid.

Het Inception-V3 in Figuur 35 komt overeen met de Inception-V2 modelarchitectuur uit de GitHub-repository. Dit komt doordat Google V2 en V3 snel achter elkaar heeft gepubliceerd. V3 wordt hierdoor in sommige community als de 2^e Versie gezien.



Figuur 35: Modelarchitectuur keuze

Stap 3 – Dataverzameling

- **Bewerking 1:**
Verzamel de data met behulp van een query of kopieer bestaande bestanden en converteer indien nodig naar het gewenste bestandsformaat.
Keuze: Bestaande PDF's uit 2018, van de retailer Aldi, zijn gedownload van de NAS. De verzamelde PDF's zijn geconverteerd naar .jpg bestanden, per pagina.
- **Bewerking 2:**
Bewerk de verzamelde data. Eerst dient met behulp van bijvoorbeeld het resized.py bestand de data geschaald te worden naar de juiste resolutie. Daarna dient de verzamelde data gelabeld te worden, om tot slot de bewerkte data te verdelen over de trainingsset (80%) en de testset (20%)
Keuze: De verzamelde data (.jpg bestanden) is met behulp van het resizer.py bestand geconverteerd naar de in stap 1 vastgestelde resolutie van 567x800 pixels. Vervolgens is de data met behulp van de aangeraden tool Labellmg.exe gelabeld en tot slot verdeeld, met de geadviseerde verhouding, over de trainingsset (80%) en de testset (20%).
De data argumentatie is niet uitvoerig toegepast, enkel de random horizontale draai is gekozen om de dataset te verdubbelen. Het resultaat van deze stap is te zien in Figuur 34.

Stap 4 – Training

- **Bewerking 1:**
Alloceer, of reserveer de benodigde hardware. In geval van lokale hardware: run geen andere zware programma's. In geval van cloudcomputing: communiceer op afgesproken wijze dat je gebruik gaat maken van de rekenkracht.
Keuze: De beschikbare GPU op mijn lokale laptop is gereserveerd, wat inhoudt dat er geen andere zware programma's worden gedraaid tijdens de training van het model.
- **Bewerking 2:**
Stel hyper parameters van het model in. Per hyper parameter is een korte omschrijving gegeven van de invloed die de instelling op de output van het model heeft. Bijna iedere parameter is al bepaald, of is vanzelfsprekend. Er kan oneindig gevarieerd worden met deze parameters wanneer uit de evaluatie (stap5) blijkt dat het resultaat ongewenst is.
Keuze:
 - Learning rate: er is gekozen voor de default setting (standaard setting) van 0.001 per datapunt. Aangezien dit de eerste keer is dat met ML-software wordt gewerkt zag ik geen reden om de leersnelheid aan te passen.
 - Aantal classes: er is gekozen voor 3 classes, aangezien het aantal classes gelijk dient te zijn aan het aantal productcategorieën dat het model dient te detecteren.
 - Aantal trainingsvoorbeelden: er is gekozen voor 200 trainingsvoorbeelden, aangezien het aantal trainingsvoorbeelden gelijk dient te zijn aan het aantal .jpg images dat gebruikt wordt voor het trainen van het model.
 - Basis van het netwerk = het Inception resnet V2 is in stap 2 gekozen

- Image resolutie = in stap 1 is de dimensie van de inputdata vastgesteld op 567x800 pixels.
 - Epochs = er is gekozen om het aantal epochs op oneindig te zetten, zodat volledige controle over de training behouden wordt en het veiligheidsprogramma (waarover in bewerking 3 en 4 meer) de training ieder moment kan onderbreken.
- **Bewerking 3 en 4:**
Run de training en exporteer het getrainde model.
Keuze: Handelingen letterlijk uitgevoerd. De training onderbroken bij een loss-value van ongeveer 2%, wat in dit voorbeeld overeenkomt met ongeveer 12.000 epochs.
Tijdens de training heeft zowel de GPU, als de CPU op 99% capaciteit gerund. Met het oog op dat de training ongeveer 10 uur zou duren is de keuze gemaakt om een extra veiligheidsprogramma op de achtergrond te draaien met als doel de laptop af te sluiten wanneer de kerntemperatuur van GPU, of CPU, boven de 110 °C komt.

Stap 5 – Evaluatie

- **Bewerking 1 en 2:**
Alloceer/ reserveer de benodigde hardware, zoals in bewerking 1 van stap4, en run het getrainde model op de testdata. Dit vergt beduidend minder rekenkracht.
Keuze: De beschikbare GPU op mijn lokale laptop is gereserveerd, wat inhoudt dat er geen andere zware programma's worden gedraaid tijdens de training van het model.
- **Bewerking 3 en 4:**
Analyseer de resultaten op arbitraire wijze visueel en analyseer resultaten met behulp van de mAP (accuraatheidsscore van een model) bij een vergelijkingsanalyse tussen twee, of meerdere modellen.
Keuze: De resultaten zijn visueel geanalyseerd door mij en door de medewerkers van GreenOrange. De visuele analyse is positief bevonden: de bounding boxes zijn nauwkeurig getrokken en de labels komen voor ongeveer 98% overeen met het werkelijke label. Daarnaast is het model over het algemeen >90% zeker van de correctheid van een geplaatst label. De visuele resultaten zijn te zien in Figuur 36. Aangezien het model niet vergeleken wordt met andere modellen wordt geen mAP score aan het model gehangen.

Het model scoort voor een proof of concept naar behoren. Er wordt doorgedaan naar stap 6.



Figuur 36: Resultaat proof of concept getraind model

Stap 6 – Implementatie

- **Bewerking 1, 2 en 3:**

Implementeer het model met behulp van TP op een server. Voor online zetten van een model wordt over het algemeen gebruik gemaakt van een applicatie die, bijvoorbeeld met Flask gerealiseerd kan worden. De opmaak kan via CSS, of een vergelijkbare taal, eenvoudig worden bijgevoegd.

Keuze: De bewerkingen zijn handelingen van TP. Deze bewerkingen zijn voor de proof of concept niet uitgevoerd, omdat het buiten de scope van het onderzoek valt om aan te tonen dat een applicatie op een server kan draaien.

Conclusie

Middels het doorlopen van de ontwikkelde methode is een proof of concept gerealiseerd die gewenst resultaat levert voor GreenOrange. De proof of concept kan gebruikt worden voor de ontwikkeling van een prototype dat op de FMP server draait.

7. Evaluatie

In dit hoofdstuk wordt de ontwikkelde methode geëvalueerd aan de hand van enquêtes met de softwareontwikkelaars van We::Code (drie ontwikkelaars) en mijn begeleider (coördinator van het ontwikkelaarsteam). De enquête is opgebouwd uit minimaal één stelling, per geïdentificeerd kwadrant, vastgesteld door Venkatesh, in 2003 (V. Venkatesh, 2003). De participanten hebben iedere stelling uit de enquête op een schaal van 1 (zeer oneens) tot 5 (zeer eens) beoordeeld. ‘Bijlage E: Evaluatie enquête, ontwikkelde methodiek’ bevat de behandelde vragen en de gescoorde antwoorden.

7.1. Analyse enquête

In deze sectie worden de opvallende kwadranten van de resultaten uit de enquêtes besproken. Allereerst wordt in Tabel 18: Analyse enquête een inzicht gegeven in de beantwoording van de enquête. Gekeken is naar de gemiddelde-, laagste- en hoogste score per vraag. Ook is de standaard deviatie per vraag berekent. De hoogste standaard deviaties zijn gemarkeerd met groen en de laagste met blauw.

Vervolgens worden opvallende scores, per kwadrant, besproken.

Tabel 18: Analyse enquête

Vragen	Analyse			
	Gemiddeld	Laagste score	Hoogste score	Standaard deviatie
1. Prestatieverwachting - Ik zou de methode bruikbaar vinden in mijn werk	2,75	2	3	0,50
2. Inspaningsverwachting - Ik zou de methode eenvoudig vinden om te gebruiken	3,25	3	4	0,50
3. Houding tegenover de methode - Gebruik maken van de methode is een goed idee	4,00	3	5	0,82
4. Sociale beïnvloeding - Mensen die belangrijk voor mij zijn vinden dat ik de methode moet gebruiken	3,25	3	4	0,50
5. Facilitaire beperkingen - Ik heb de benodigde kennis om de methode te gebruiken	3,25	2	5	1,26
6. Facilitaire beperkingen - Een specifiek persoon (of groep) is beschikbaar voor assistentie wanneer ik problemen ondervind in het gebruik van de methode	3,25	2	5	1,26
7. Zelfredzaamheid - Ik kan het werk, of een taak, voltooien wanneer niemand in de buurt vertelt hoe ik de methode moet gebruiken	2,50	2	3	0,58
8. Zelfredzaamheid - Ik kan het werk, of een taak, voltooien als ik iemand kan bellen wanneer ik vast loop in het gebruik van de methode	4,25	3	5	0,96
9. Vrees - De methodie is ietswat intimiderend voor mij	2,25	2	3	0,50
10. Gedragsintentie ter gebruik van de methode - Ik ben van plan om de methode in de komende twee maanden te gebruiken	3,50	3	5	1,00

Prestatieverwachting

De gemiddelde bruikbaarheid van de methode scoort relatief laag. Ook ligt de scoring van de participanten dicht bij elkaar. Naar verwachting komt dit, omdat de participanten niet in kunnen schatten op welke manier de methode kan helpen bij het ontwikkelen van machine learning software (er is immers geen referentiekader voor de participanten). Eén participant geeft aan dat de methode niet bruikbaar is in zijn werk. Deze participant is verantwoordelijk voor de ontwikkeling van database software en zal in alle waarschijnlijkheid geen gebruik gaan maken van de methode.

Inspanningsverwachting

De participanten verwachten dat de methode boven gemiddeld eenvoudig te gebruiken is en de scoring ligt dicht bij elkaar. Opvallend is dat de participanten aangeven dat ze verwachten dat de methode eenvoudig is te gebruiken, maar bij 'zelfredzaamheid' aangeven dat ze het relatief moeilijk vinden om een taak zelfstandig te voltooien wanneer niemand vertelt hoe de methode gebruikt dient te worden. De participanten hebben deze vraag naar verwachting beantwoord met de aanname allereerst meer kennis over het onderwerp te vergaren, waardoor de methode ook beter te gebruiken is.

Houding tegenover de methode

Het gebruik van de methode is unaniem een goed idee. De participanten zien dezelfde voordelen in de ontwikkeling van machine-learning software, zoals beschreven in 1.2.2. Relevantie van het probleem. De obstakels, zoals omschreven in 1.2 Probleemidentificatie (introductie), worden ook gedeeld door de participanten. Een logische houding is dat de participanten positief tegenover een methode staan die de obstakels wegneemt en de voordelen mogelijk maakt.

Facilitaire beperkingen

Opvallend is dat de participanten over het algemeen aangeven niet over de benodigde kennis te beschikken om de methode te gebruiken. Eén participant geeft echter aan ruim voldoende kennis te hebben. De participant zal uit eigen interesse in een eerder stadium zich meer verdiept hebben in het onderwerp machine-learning, waardoor deze meer zelfverzekerd is van zijn kunnen binnen dit vakgebied.

Opvallend is ook dat de participanten aangeven dat er geen specifiek persoon, of groep, beschikbaar is voor assistentie bij gebruik van de methode. Eén participant geeft echter aan wel te beschikken over assistentie. Eén participant heeft waarschijnlijk de aanname gemaakt dat ik met een telefoontje of mailtje beschikbaar ben voor extra informatie en uitleg.

Zelfredzaamheid

De participanten geven aan dat het moeilijk is om taken uit te voeren met de methode, wanneer niemand ze verteld wat ze moeten doen. Ook geven de participanten aan dat het werk voltooid kan worden wanneer ze iemand kunnen bellen wanneer ze vastlopen in het gebruik van de methode. Opvallend is dat beide vragen een relatief grote spreiding tonen. Dit valt te verklaren door het verschil in kennis over het onderwerp machine-learning, waardoor automatisch het vertrouwen in zelfredzaamheid daalt.

Vrees

De participanten geven aan dat de methode niet intimiderend is. Waarschijnlijk is de methode op een werkbaar detailniveau geschreven en voldoende uitleg verschaft bij uitdagende begrippen. Hierdoor schrikken participanten niet van onbekende termen en de complexiteit van de modellen. Ook bestaat de mogelijk mogelijkheid dat de participanten zich schaamden om toe te geven dat de methode intimiderend is.

Gedragsintentie ter gebruik van de methode

Opvallend is dat één participant zeker is dat hij de ontwikkelde methode in de komende twee maanden gaat gebruiken. De andere participanten geven een neutrale mening. Naar verwachting is de participant die zeker is van het gebruik de coördinator van het team. Deze participant heeft zeggenschap over de doelstellingen en focus van het team. De overige participanten kunnen niet goed voorspellen welke taken zij toebedeeld krijgen de komende periode.

7.2. Conclusie uit enquête

In deze sectie worden conclusies getrokken uit de analyse van de enquête. De conclusie is in vier stappen opgebouwd.

Iedere participant ziet meerwaarde in het gebruik van de methode, echter zijn de participanten niet zeker over de bruikbaarheid van de methode binnen hún werk en tonen niet de gedragsintentie om de komende maanden gebruik te gaan maken van de methode. Hieruit wordt afgeleid dat de participanten geen duidelijk inzicht hebben in de link tussen het gebruik van de methode binnen het bedrijf en hun rol in het gebruik van de methode. Aansluitend wordt geconcludeerd dat de participanten beter geïnformeerd wensen te worden over hun specifieke rol binnen de ontwikkeling van de machine learning software.

De methode wordt niet als intimiderend ervaren. Wél geven de participanten aan dat ze niet over voldoende kennis beschikken om zelfstandig gebruik te maken van de methode. Ook wordt aangegeven dat ze de taken waarschijnlijk niet uit kunnen voeren zonder iemand die ze zegt wat ze moeten doen, of waar ze naar toe kunnen gaan voor vragen wanneer ze vastlopen. Hieruit valt af te leiden dat de participanten meer informatie rondom het onderwerp kunnen gebruiken en behoefte hebben aan een leidend figuur met autoriteit op dit onderwerp.

8. Discussie, conclusie en aanbevelingen

In dit hoofdstuk wordt een conclusie getrokken en aanbevelingen gedaan naar aanleiding van het vastgestelde probleem en het onderzoek dat is uitgevoerd. Allereerst wordt in de discussie op iedere designvraag een antwoord omschreven en wordt de evaluatie kort samengevat. Vervolgens wordt uit de beantwoording van de designvragen een antwoord, en daarmee een conclusie, geformuleerd voor de hoofdvraag. Daarna zal achtereenvolgens gesproken worden over: de relevantie van het onderzoek, de aanbevelingen en suggesties en tot slot de reflectie.

8.1 Discussie en Conclusie

In het uitgevoerde onderzoek is antwoord gezocht op de vraag: “Hoe kan een methode, op basis van machine-learning componenten, het ontwikkelingsproces voor softwareautomatisering binnen Green Orange verbeteren?”. Voor het beantwoorden van deze vraag is een wetenschappelijk onderzoek uitgevoerd op basis van het DSRM. Het doel hiervan is: de ontwikkeling van een methode waarin het ontwikkelingsproces van automatiseringssoftware, op basis van machine-learning componenten, staat beschreven. Belangrijk binnen deze doelstelling is dat de ontwikkelde methode toepasbaar is op het automatiseringstraject van het WePublish-proces.

8.1.1. Discussie

Designvraag 1 – “Welke kennis omtrent de beschreven doelstelling bestaat al?”

Om oplossing te bieden op het vastgestelde probleem is allereerst in 2. Theoretisch kader onderzocht welke componenten van ML van toepassing zijn binnen dit onderzoek, welke methodieken er worden gebruikt in de ontwikkeling van software, welke methodieken er bestaan ter ontwikkeling van methoden en welke methoden ter ontwikkeling van visuele bedrijfsprocesweergaven. Hiermee is antwoord gegeven op de eerste designvraag: “Welke kennis omtrent de beschreven doelstelling bestaat al?”.

Designvraag 2 - “Hoe ziet het huidige proces achter de WePublish-service eruit?”

Om toe te werken naar een methodiek die binnen GreenOrange geïmplementeerd kan worden en toepasbaar is op het automatiseringstraject van de WePublish-service is in 3. Huidige situatie onderzoek gedaan naar de huidige situatie van de WePublish-service binnen GreenOrange. Hiermee is antwoord gegeven op de tweede designvraag: “Hoe ziet het huidige proces achter de WePublish-service eruit?”. Met behulp van het BPMN is het proces visueel weergegeven en zijn de stappen die met behulp van ML-componenten geautomatiseerd kunnen worden vastgesteld: box product, productname definitie en identify interaction.

Designvraag 3 - “Hoe ziet het ideale proces achter de achter de WePublish-service eruit?”

Gebaseerd op literatuuronderzoek uit 2. Theoretisch kader, in combinatie met interviews van softwareontwikkelaars binnen het bedrijf, is vastgesteld dat een Agile softwareontwikkelingsmethode waarin toegewerkt wordt naar een MVP gewenst is. Om het MVP van de ML-software te definiëren is eerst gekeken naar de ideale situatie van de WePublish-service. Hiermee is antwoord gegeven op de derde designvraag: “Hoe ziet het ideale proces achter de achter de WePublish-service eruit?”. Vervolgens zijn de meest essentiële aspecten uit de ideale service geëxtraheerd en opgenomen in een MVP. Concluderend bestaat het MVP bestaat uit de ontwikkeling

van één geïmplementeerd model dat getraind is op de dataset van één retailer, gebruikmakend van productcategorieën.

Designvraag 4 - “Hoe ziet een methode, op basis van machine-learning componenten, waarin het ontwikkelingsproces voor softwareautomatisering staat beschreven eruit?”

In hoofdstuk 5 is met literaire onderbouwing een agile softwaremethodiek ontworpen voor het ontwikkelen van automatiseringssoftware op basis van ML- componenten. In deze methodiek zijn zes stappen onderscheiden: 1.) probleem identificatie, 2.) model architectuur, 3.) data verzameling, 4.) training, 5.) evaluatie en communicatie en tot slot 6.) implementatie. Per stap is de input, de benodigde bewerkingen en de output omschreven. Hiermee is antwoord gegeven op de vierde designvraag: “Hoe ziet een methode, op basis van machine-learning componenten, waarin het ontwikkelingsproces voor softwareautomatisering staat beschreven eruit?”.

Designvraag 5 – “Hoe kan de ontwikkelde methode worden toegepast bij Green Orange?”

Een eerste stap in de automatisering van de WePublish-service is gezet door één iteratie van de ontwikkelde methode te doorlopen. Met behulp van deze toepassing van de methode is een proof of concept gecreëerd waarin wordt aangetoond dat de automatiseringssoftware op basis van ML- componenten werkt voor het betreffende bedrijfsproces en waardoor wordt aangetoond dat het volgen van de methode naar een gewenst resultaat kan leiden.

Evaluatie en communicatie

In dit hoofdstuk wordt de methodiek, met behulp van een questionnaire uit de literatuur, door vijf softwareontwikkelaars geëvalueerd. Uit de evaluatie blijkt dat de ontwikkelaars graag begeleid willen worden en meer informatie omtrent ML kunnen gebruiken.

8.1.2. Conclusie

Er kan geconcludeerd worden dat met het onderzoek antwoord is gegeven op de hoofdvraag: “Hoe kan een methode, op basis van machine-learning componenten, het ontwikkelingsproces voor softwareautomatisering binnen Green Orange verbeteren?”. GreenOrange is namelijk door middel van de ontwikkelde ontwikkelingsmethodiek in staat om automatiseringssoftware, gebaseerd op ML- componenten, te ontwikkelen en toe te passen binnen haar eigen organisatie. Met deze methodiek is GreenOrange in de toekomst in staat om vergelijkbare automatiseringssoftware, op basis van ML- componenten, te ontwikkelen voor intern of extern gebruik. GreenOrange heeft met deze methodiek een sterke technologische voorsprong op haar concurrenten, waardoor zij in staat is om meer waarde te genereren met minder middelen.

8.2. Relevantie van het onderzoek

In deze paragraaf wordt de algemene relevantie van het onderzoek besproken. Relevantie wordt opgedeeld in praktische bijdrage aan het betreffende bedrijf en vergelijkbare organisaties en theoretische bijdrage aan de wetenschap. Theoretische bijdrage wordt geïnterpreteerd als de toegevoegde waarde van het onderzoek binnen de wetenschappelijke wereld, praktische bijdrage wordt geïnterpreteerd als de toegevoegde waarde van het onderzoek binnen GreenOrange en vergelijkbare organisaties.

Een belangrijk onderscheid is toegevoegde waarde binnen de wetenschap in de vorm van een toegepast onderzoek, zoals omschreven in 1.3.1. Onderzoekskwaliteit. Dit is praktisch bijdrage van aard, maar valt onder de bijdrage aan de theorie.

8.2.1. Theoretisch bijdrage

Deze sectie omschrijft de theoretische bijdrage van het onderzoek. In onderstaande lijst worden twee verschillende bijdragen omschreven en onderbouwt. Er wordt onderscheid gemaakt tussen praktische bijdragen die generaliseerbaar zijn (er wordt gesproken over 'organisaties') en praktische bijdragen die enkel voor Green Orange van toepassing zijn (er wordt specifiek gesproken over Green Orange).

1. Zoals omschreven in 1.3.1 Limitaties is bij de ontwikkeling van de methodiek rekening met schaalbaarheid van de methode op andere problemen. Zoals omschreven 5.1. Probleem identificatie (methodiek) wordt het betreffende probleem (in de vorm van een vergelijkbaar proces waarin visuele herkenning plaatsvindt) omschreven naar een object detection vraagstuk. Door de betreffende situatie te vertalen naar een object detection vraagstuk wordt de methode generaliseerbaar en is niet van belang binnen welke organisatie het vraagstuk zich afspeelt. De voorgestelde methode is een toevoeging voor de literatuur vanwege de praktische insteek en algemene toepasbaarheid. Deze algemene toepasbaarheid beperkt zich, zoals omschreven in 5.2. Model architectuur enkel tot object detection vraagstukken. De methode beperkt zich niet tot de organisatie waarin het object detection vraagstuk zich afspeelt. In theorie is de methodiek dus toepasbaar binnen bedrijven, gezondheidszorginstellingen en binnen de wetenschap.

Voorbeeld: een podotherapeut analyseert images van de voeten van patiënten en geeft op basis van het patroon dat hij in de images herkent een diagnose en een advies. In theorie is dit een vergelijkbaar visueel herkenningsproces als de herkenning van producten binnen de WePublish-service. Het herkenningsproces van de podotherapeut kan middels 5.1. Probleem identificatie (methodiek) vertaald worden naar een object detection vraagstuk. Vervolgens kan de voorgestelde methodiek gevolgd worden. Het resultaat is een model dat images van voeten analyseert en op basis van patronen in deze images een classificatie als output genereert. Deze output is te vertalen als een diagnose, gekoppeld aan een bepaalde geadviseerde handeling.

2. In dit onderzoek zijn de principes van de Agile softwareontwikkelingsmethodiek toegepast op de softwareontwikkeling van DeepLearning componenten en gecombineerd in een methodiek die de gebruiker begeleidt bij de ontwikkeling van DeepLearning software.

8.2.2. Praktische bijdrage

Deze sectie omschrijft de praktische bijdrage van het onderzoek. In onderstaande lijst worden vijf verschillende bijdragen omschreven en onderbouwt. Er wordt onderscheid gemaakt tussen praktische bijdragen die generaliseerbaar zijn (er wordt gesproken over 'organisaties') en praktische bijdragen die enkel voor Green Orange van toepassing zijn (er wordt specifiek gesproken over Green Orange).

1. Voor deze bijdrage geldt dezelfde onderbouwing als omschreven in de theoretische bijdrage. Daarom stelt de voorgestelde methode organisaties in staat om vergelijkbare visuele herkenningsprocessen, zoals de WePublish-service, binnen de organisatie om te schrijven naar een object detection vraagstuk. Vervolgens is de organisatie in staat om het proces zelfstandig te automatiseren.
2. Ook bij deze bijdrage geldt dezelfde onderbouwing als omschreven in de theoretische bijdrage. Daarom stelt de voorgestelde methode organisaties in staat om vergelijkbare visuele herkenningsprocessen buiten de organisatie om te schrijven naar een object detection vraagstuk. Vervolgens is de organisatie in staat om de oplossing voor het uitgewerkte vraagstuk in de vorm van een service aan haar klanten aan te bieden.
3. Het ontwikkelde Python programma, waarmee prestaties van de WePublish-service automatisch gemeten en visueel weergegeven kunnen worden, verschaft Green Orange van inzicht in het effect van veranderingen op de WePublish-service en kan in de toekomst real-time verwerkt worden in een dashboard.
Daarnaast verschaft dit programma organisaties van een analysetool wanneer de te analyseren data in de juiste opmaak wordt aangeleverd.
4. Zoals omschreven in 1.2.1. Probleemkluwen beschikken medewerkers van Green Orange, verantwoordelijk voor de ontwikkeling van machine-learning software, over weinig kennis van dit vakgebied. Onder andere hierdoor is vastgesteld dat de ontwikkeling van deze software stagneert. De voorgestelde methode is in samenspraak met deze medewerkers vormgegeven en onder andere door deze medewerkers, in 7. Evaluatie, geëvalueerd. Uit de evaluatie blijkt dat medewerkers zelfredzaam zijn met de methode, wanneer een vraagbaken wordt aangesteld.
Hierdoor kan gesteld worden dat de voorgestelde methode, in combinatie met een vraagbaken binnen de organisatie, gebruikers met weinig, tot geen, ervaring in het vakgebied machine-learning, in staat stelt om object detection vraagstukken aan te pakken.
5. Het BPMN model uit 3. Huidige situatie verschaft Green Orange van een procesmatig overzicht van de WePublish-service. Dit overzicht kan in de toekomst gebruikt worden bij procesmatige veranderingen binnen de WePublish-service. Buiten Green Orange kent dit model, net zoals het model uit 4. Gewenste situatie, geen praktische bijdrage.

8.3. Aanbevelingen en suggesties

In deze paragraaf worden aanbevelingen en suggesties gedaan voor GO op het gebied van de WePublish-service, het gebruik van de ontwikkelde methodiek en de doorontwikkeling van machine learning toepassingen voor GO.

8.3.1. Aanbevelingen

1. Stel nieuwe doelen voor de ontwikkeling van een MVP_V2 en onderneem stappen uit de ontwikkelde methodiek om een MVP_V2 te ontwikkelen. Zoals omschreven in hoofdstuk 4.3. is MVP_V1, de Proof of Concept uit dit onderzoek, gelimiteerd tot één retailer (de Aldi), drie productcategorie labels en is de data voor de training en test afkomstig uit één productiejaar. Ik raad GO aan om een MVP_V2 te ontwikkelen die zich beperkt tot één retailer (de Aldi), alle productcategorie labels en gebruik maakt van data uit minimaal drie productiejaar, waarvan één productiejaar het huidige productiejaar is (2019).

2. Verzamelen van meer data: Om de verzameling van data ten behoeve van de ontwikkeling van MVP_V2 en volgende versies te vereenvoudigen raad ik GO aan om software te ontwikkelen dat .jpg en .xml bestanden genereert, zodat meer data beschikbaar is om modellen te trainen. Voor het genereren van meer bruikbare data zijn twee bronnen: toekomstig te verwerken folders en verwerkte folders.

Voor toekomst te verwerken folders: software dient bij iedere boxproduct en labelproduct handeling de x- en y coördinaten van de boxproduct en het productlabel, per pagina, weg te schrijven naar een .xml bestand. Vervolgens dient de te ontwikkelen software het .xml bestand en het bijbehorende .jpg bestand onder dezelfde naam op te slaan. De .jpg en .xml bestanden dienen per retailer gefilterd te kunnen worden. De ontwikkeling van deze software genereert, per verwerkte pagina binnen het FMP-editor proces, gelabelde data die bij toekomstige versies van een MVP gebruikt kunnen worden voor training en validatie. Voor de ontwikkeling van deze software zijn geen grote uitdagingen gevonden.

Voor verwerkte folders: van iedere boxproduct en labelproduct handeling zijn onder andere de x- en y coördinaten en het productlabel weggeschreven naar de ID. Deze informatie zou met nieuwe software verwerkt kunnen worden tot bruikbare .xml en .jpg bestanden, zoals in de vorige alinea beschreven. Uitdagingen in de ontwikkeling zijn deze software liggen in de vertaling van de x- en y coördinaten uit de ID naar bruikbare x- y coördinaten. Onder bruikbare coördinaten wordt de positioneren ten opzichte van de image resolutie bedoelt. De huidige coördinaten zijn ten opzichte van een 1000x1000 matrix gepositioneerd.

3. MVP_V3 en nieuwe versies. Bij de doorontwikkeling van de MVP's is het zaak om vast te stellen of één model kan voldoen aan de automatisering van meerdere retailers. Om dit vast te stellen dienen drie modellen getraind te worden: 1.) dataset van retailer_a, 2.) dataset van retailer_b en 3.) dataset van retailer_a en retailer_b. Vervolgens kunnen de drie modellen geëvalueerd worden middels de omschreven evaluatiebewerkingen uit hoofdstuk 5. Bij gewenst resultaat kan de hierboven omschreven vergelijkingstechniek worden uitgebreid naar meer retailers, om uiteindelijk te onderzoeken of één model volstaat voor een complete retailerset of zelfs volstaat voor de totale set van retailersets.
Mogelijk resultaten: één model voor alle folders, één model per retailerset of één model per retailer.
4. Aanbieden van object detection service aan klanten. Ik raad aan om deze service pas aan klanten aan te bieden wanneer de softwareontwikkeling op gebied van machine learning beheerst is binnen het bedrijf. In mijn ogen betekent ML als bedrijf beheersen: een model, of meerdere modellen die alle productcategorie labels van minimaal twee retailers correct kan herkennen, volledig geïmplementeerd (inclusief gebruiksvriendelijkheid) binnen het FMP-editor proces.
5. Gebruik van methode. Naar aanleiding van de evaluatie in hoofdstuk 7 raad ik aan om een autoritair op het gebied van machine learning aan te stellen die als vraagbaken dient voor de ontwikkelaars die de machine learning software gaan ontwikkelen. Ook raad ik aan om meer inhoudelijke informatie rondom machine learning aan te bieden aan de ontwikkelaars. Tot

slot raad ik aan om duidelijk te communiceren met de ontwikkelaars over wat er van ze verwacht wordt, en in welke periode dit verwacht wordt.

8.3.2. Suggesties:

1. Een suggestie is om uit te zoeken op welke manier OCR geïntegreerd kan worden binnen de modelarchitectuur, of na het labelen van data. Mijn advies, wanneer deze suggestie wordt opgepakt, is om de gelabelde data (.jpg en .xml) bestanden te gebruiken om een OCR te laten zoeken naar tekst binnen het .jpg bestand, binnen de coördinaten aangegeven in het .xml bestand. Vervolgens kan de gevonden tekst vergeleken worden met alle producten van de betreffende retailer. Bij een match kunnen allerlei UTM's worden toegevoegd aan het herkende product.
2. GreenOrange kan in een nieuwe afstudeeropdracht van een Bsc. TBK, of Bsc. BIT student een dashboard laten ontwikkelen. In dit afstudeeronderzoek kunnen de beste KPI's vastgesteld worden voor de WePublish-service. De nulmeting uit H2 en het BPMN model van de huidige situatie, uit H3, kunnen hier als basis fungeren.
3. GO kan in een nieuwe afstudeeropdracht van een Bsc. TBK student een procesoptimalisatieonderzoek laten uitvoeren op het proces van de WePublish-service. Het BPMN model van de huidige situatie (H3) kan als basis dienen voor dit onderzoek.

8.4. Vervolgonderzoek

1. Vervolgonderzoek kan zich richten op de uitbreiding van de huidige methodiek tot een meer omvattende methodiek binnen de onderzoeksrichting van Computer Vision. De voorgestelde methodiek beperkt zich tot object detection, waardoor niet ieder Computer Vision probleem binnen een organisatie opgelost. Een uitbreiding van de keuzemomenten, zoals behandeld in Figuur 32 en de bijbehorende toelichting in de rest van de stappen uit de methode leidt tot een methode die op een bredere set Computer Vision problemen toepasbaar is. Dit leidt tot een vergrote theoretische en praktische bijdrage.
Bijvoorbeeld: het toevoegen van niveau3 en niveau4 in de voorgestelde methode, in 5.2. Model architectuur.
2. Vervolgonderzoek kan zich richten op het vaststellen van de ideale KPI's voor de WePublish-service en het visueel weergeven hiervan. In dit onderzoek zijn de KPI's (die berekend worden met het Pythonprogramma) met minimale literaire onderbouwing vastgesteld. Om de validiteit van het Pythonprogramma te vergroten dient sterker onderbouwd te worden wat de beste KPI's zijn in deze service.
3. Vervolgonderzoek kan zich richten op de procesoptimalisatie binnen de overige groups van de WePublish-service, zoals beschreven in 'suggesties 3.'
Dit onderzoek richt zich op het FMP-editor proces, zoals omschreven in [3.1.3. FMP editor – group](#). De andere groups zijn voor het totaalbeeld ook uitgewerkt. Vervolgonderzoek kan zich richten op de procesoptimalisatie van het gehele WePublish-proces. Hiervoor kan het ontwikkelde BPMN model en het Pythonprogramma uit 3. Huidige situatie gebruikt worden. Dit vervolgonderzoek zal GreenOrange voorzien van een nieuwe optimalisatieslag binnen het WePublish-proces.

8.5. Limitaties van het onderzoek

In deze paragraaf worden de limitaties van het onderzoek binnen vier categorieën opgesomd: BPMN modellen, nulmeting, ontwikkelde methodiek en de Proof of Concept.

8.5.1. BPMN modellen

De BPMN modellen zijn ontwikkeld om een beeld en gevoel te krijgen bij zowel de huidige, als de nieuwe situatie. De FMP-editor group is in beide gevallen in zo groot mogelijk detail weergegeven, omdat de focus van dit onderzoek in deze group ligt. De limitatie in deze modellen is de gesimplificeerde notatie van het proces binnen de overige twee groups.

8.5.2. Nulmeting

De nulmeting is uitgevoerd op een aantal KPI's die verkregen zijn door literatuuronderzoek en interviews met medewerkers. De limitaties van deze nulmeting berusten op het feit dat het literatuuronderzoek minimaal is geweest. De keuze voor dit minimale literatuuronderzoek is de scope van dit onderzoek (beschikbare tijd en focus).

De analyse van de huidige situatie (de nulmeting) is uitgevoerd op een dataset van drie maanden. Dit maakt het onmogelijk om bepaalde trends binnen een jaar te ontdekken, of bepaalde trends over jaren te ontdekken. Ook bestond de data uit veel ruis dat door middel van, eventueel foute, aannames bewerkt is tot bruikbare data.

8.5.3. Ontwikkelde methodiek

De ontwikkelde methodiek is enkel toepasbaar op binnen de ComputerVision onderzoeksrichting van ML en limiteert zich binnen deze onderzoeksrichting op object detection. Het is niet onderzocht of de methode ook toepasbaar is op een vergelijkbare subset van ComputerVision, object classification. Hoewel enige voorkennis op het gebied van softwareontwikkeling, computertalen en ML gewenst is, is de methode toegankelijk voor leken.

De ontwikkelde methode zit vol met keuzemomenten die sterk afhankelijk zijn van de wensen van de gebruiker. De methode biedt in deze situaties niet een compleet overzicht van alle mogelijke keuzes, maar biedt handvaten om de gebruiker te helpen om de keuze te laten maken.

8.5.4. Proof of Concept

De Proof of Concept limiteert zich enkel tot het object detection probleem binnen de WePublish-service. Het is niet onderzocht of de methode bij meerdere object detection problemen toepasbaar is.

9. Bibliografie

- A. Canziani, E. C. (2016). An Analysis of Deep Neural Network Models for Practical Applications., (pp. 1-7).
- A.R. Hevner, S. M. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 75-105.
- Amit, R. (2018, April 5). *7 limitations of deep learning algorithms of AI*. Opgehaald van Amit Ray: <https://amitray.com/7-limitations-of-deep-learning-algorithms-of-ai/>
- Augular-Savén, R. (2003). Business Process Modelling: Review and Framework. *Elsevier*, 129-149.
- B. Henderson-Sellers, J. R. (2010). Situational Method Engineering: State-of-the-Art-Review. *Journal of Universal Computer Science*, 424-478.
- Bahmani, M. (2018, 11 7). *AI vs Machine Learning vs Deep Learning*. Opgehaald van Medium.com: <https://medium.com/datadriveninvestor/ai-vs-machine-learning-vs-deep-learning-ba3b3c58c32>
- Basili, C. L. (2003). Iterative and Incremental Development: A Brief History. *IEEE Computer Society*, 47-56.
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. Cambridge, Engeland: Springer.
- C. Larman, V. B. (2003). Iterative and Incremental Development: A Brief History. *IEEE Computer Society*, 47-56.
- Conboy, K. (2009). Agility from first principles: reconstructing the concept of agility in information systems development. *Information Systems Research*, 329-354.
- Copeland, M. (2016, 7 29). *What's the Difference Between Artificial Intelligence, Machine Learning, and Deep Learning?* Opgehaald van NVIDIA.com: <https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/>
- D. H. Ballard, C. B. (1982). Computer Vision. In C. B. D. H. Ballard, *Computer Vision* (pp. xiii - xiv). Englewood Cliffs, New Jersey: Prentice-Hall, Inc.
- E. Yilmaz, J. A. (2006). Estimating Average Precision with Incomplete and Imperfect Judgements. *College of Computer and Information Science*, 5-11.
- Gad, A. F. (2018). *Practical Computer Vision Applications Using Deep Learning with CNN's*. Menoufia, Egypte: Springer Science and Business Media.
- Garbade, M. (2018, 9 14). *Clearing the Confusion: AI vs Machine Learning vs Deep Learning Differences*. Opgehaald van towardsdatascience: <https://towardsdatascience.com/clearing-the-confusion-ai-vs-machine-learning-vs-deep-learning-differences-fce69b21d5eb>
- Geiger, M. (2018). BPMN 2.0: The state of support and implementation. *Future generation computer systems*, 250-262.
- Harmsen, A. (1997). *Situational Method Engineering*. Utrecht, Nederland: Moret Ernst & Young Management Consultant.
- J.Wang, L. P. (2017). 2017. *ArXiv*, 12-24.
- Jain, A. (sd). *Breaking neural networks with adversarial attacks*. Opgehaald van Towards Data Science: <https://towardsdatascience.com/breaking-neural-networks-with-adversarial-attacks-f4290a9a45aa>
- K. Eyholt, I. E. (2018). Robust Physical-World Attacks on Deep Learning Visual Classification. *CVPR*, 212-233.
- K. Peffers, M. R. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 45-78.
- M. Chinosi, A. T. (2012). An introduction to the standard. *Elsevier*, 124-134.
- Nielsen, M. (2013). *Neural Networks and Deep Learning*. -: -.
- Nijssen G.M., C. I. (2009). *Kennis Gebaseerd Werken*. Heerlen: PNA Publishing B.V.
- Papadopoulos, G. (2015). Moving from traditional to agile software development methodologies also on large, distributed projects. *Procedia*, 455-463.

- Pitts, W. M. (1943). A Logical Calculus of the Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biology*, 115-133.
- Rosenblatt, F. (1957). *The perceptron, a perceiving and recognizing automaton (Project Para)*. Buffalo, N.Y.: Cornell Aeronautical Laboratory, Inc.
- Royce, W. W. (1970). Managing the development of large software systems. *IEEE*, 1-9.
- Su, J. (2017). One pixel attack for fooling deep neural networks. *IEEE*, 12-28.
- T. Dingsøyr, S. N. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of systems and Software*, 1213-1221.
- V. Venkatesh, M. M. (2003). Acceptance of Information Technology: Toward a Unified View. *MIS Quarterly*, 425-478.
- Venkatesan, M. (2018, 5 7). *Artificial Intelligence vs. Machine Learning vs. Deep Learning*. Opgehaald van DataScienceCentral: <https://www.datasciencecentral.com/profiles/blogs/artificial-intelligence-vs-machine-learning-vs-deep-learning>
- Waldrop, M. M. (2019, Januarie 22). *What are the limits of deep learning*. Opgehaald van pnas.org: <https://www.pnas.org/content/116/4/1074>
- Werbos, P. (1982). Applications of advances in nonlinear sensitivity analysis. *System modeling and optimization*, 762-770.
- Y. Le Cun et al., B. B. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 541-551.
- Y.L. LeCun, L. B. (1998). Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 2278-2324.

Bijlage

Bijlage A: BPMN notatie

Een BPMN-model is opgebouwd uit vier verschillende onderdelen: Stroomobjecten, Verbindende objecten, zwembanen en artefacten. Hieronder worden de verschillende onderdelen van een BPMN model toegelicht. Bron: <https://www.lucidchart.com/pages/bpmn>

Gebeurtenissen

Een aanleiding die het startsein geeft voor de start, aanpassing of voltooiing van een proces. Deze gebeurtenissen vinden plaats na bijvoorbeeld een bericht, een timer, een fout of een annulering. Ze worden gevisualiseerd aan de hand van een cirkel waarin andere symbolen zijn opgenomen op basis van hun evenementstype. De reden van het plaatsvinden van het event wordt weergegeven met een symbool in het rondje.



Figuur 37: BPMN gebeurtenissen

Activiteit

Een activiteit is een taak of activiteit die een persoon of systeem uitvoert. Deze wordt weergegeven in de vorm van een rechthoek met afgeronde hoeken. Er zijn verschillende activiteiten die zich voordoen in een proces.



Figuur 38: BPMN activiteit

Poort

Een poort is een beslissingsmoment waarop een pad kan worden gewijzigd op basis van bepaalde voorwaarden of evenementen. Een poort wordt weergegeven aan de hand van een ruit met daarin een symbool. Deze symbool geeft weer welke beslissing er wordt gemaakt en waar het product in het proces heen gaat.



Figuur 39: BPMN poort

Sequentiestroom

Een sequentiestroom toont de volgorde waarin activiteiten moeten worden uitgevoerd. Hiervoor wordt een rechte lijn met een pijl aan het uiteinde gebruikt.



Figuur 40: BPMN sequentiestroom

Berichtenstroom

Een berichtenstroom geeft berichten weer die door de verschillende banen in een 'zwembad' stromen of begrenst de verschillende afdelingen binnen een organisatie.



Figuur 41: BPMN berichtenstroom

Associatie

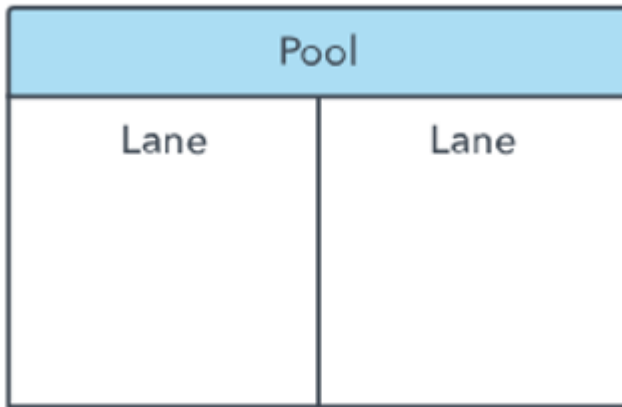
Een associatie wordt getoond aan de hand van een stippellijn. Het legt een relatie tussen een artefact en een evenement, poort of activiteit.



Figuur 42: BPMN associatie

Zwembaden en zwembanen

Een zwembad vertegenwoordigt belangrijke deelnemers in een proces. Hiermee kan er onderscheid gemaakt tussen de belanghebbenden in het bedrijfsproces. Zwembanen binnen een zwembad tonen de activiteiten van stromen voor een bepaalde rol of deelnemer binnen een afdeling of groter geheel.



Figuur 43: BPMN zwembaden en zwembanen

Artefact

Een artefact dient om extra informatie toe te voegen waaraan een ontwikkelaar van een model denkt dat het nodig is. Er zijn drie soorten artefacten. Een dataobject toont de gegevens die benodigd zijn voor een activiteit. Een groep toont een logische groepering van activiteiten maar verandert de stroom van een diagram niet. Een annotatie geeft meer informatie over een groter deel van een diagram.



Figuur 44: BPMN artefact

Bijlage B: Termen binnen BPMN

Termen BPMN:

Hieronder volgt een overzicht van de termen die binnen het BPMN model aan bod komen, onderbouwd vanuit drie verschillende online computer woordenboeken (Computer Hope, 2018-2019, Seniorweb, PC-tutorials) . De termen hebben betrekking op de (1.) data files, (2.) databases en (3.) de manieren van communicatie die binnen het model weergegeven worden.

1. Data Files

- **PDF (Portable Document Format)**, .pdf
Een bestandsformat en bestandsextensie voor digitale documenten. Het pdf. bestand stelt de gebruiker in staat om het originele voorkomen van het bestand vast te zetten. Hierdoor wordt het document op ieder apparaat op dezelfde manier weergegeven en geprint.
- **XML (Extensible Markup Language)**, xml.
Een standaard bestandsformat en extensie, gericht op het uitwisselen van data tussen computers. De gebruiker is in staat aan ieder stukje data een zelf gedefinieerde tag toe te voegen, waardoor het programma snel en eenvoudig gelezen kan worden door zelf mensen als computers.
- **Excel**, .xls or .xlsx
Een programma dat de gebruiker in staat stelt om spreadsheets te bewerken.
- **JPEG (Joint Photographic Experts Group)**, .jpg or .jpeg
Een bestandsformat en bestandsextensie voor het comprimeren van digitale afbeeldingen.

2. Databases

2.1. Interaction Database

- Een database is een verzameling van gegevens, bijvoorbeeld een adressenbestand. Data in een database kan worden geraadpleegd en worden bewerkt.
De interactie database van Green Orange is een database waar de informatie van iedere interactie wordt bijgehouden. Bij ieder .jpg bestand worden de pixellocaties van een boxproduct opgeslagen, inclusief de specificaties van de interactie.

2.2. NAS (Network Attached Storage)

- Een NAS is een opslagmedium (meestal een harde schijf) die is aangesloten op een netwerk die vanaf elke computer binnen dat netwerk kan worden aangesproken (mits de computer over de juiste rechten beschikt).
De NAS van GreenOrange wordt gebruikt om alle aangeleverde folders en bijbehorende bestanden opgeslagen.

3. Communicatie

CDN (Content Delivery Network)

- Een content delivery network is vaak een globaal netwerk dat statische bestanden in een cache (tijdelijke dataopslag) plaatst, zodat de bestanden sneller opgehaald kunnen worden. Denk aan foto's en video's om een website sneller te kunnen laden. Het CDN van Green Orange is lokaal en doet een aantal extra handelingen, voordat het de bestanden in een cache aanbiedt. Het CDN converteert de ingevoerde .pdf

bestanden naar.jpg (een afbeelding) bestanden, per pagina. Daarnaast verlaagd het CDN de resolutie van deze afbeeldingen, om snellere bewerking mogelijk te maken. De FMP employe die de folder edit, bewerkt in realiteit .jpg bestanden (de 'content') die één voor één aangeboden worden ('delivery') door het CDN.

Cloud

- Cloud is een verzamelnaam voor alle SAAS-diensten (**Software-As-A-Service**), het aanbieden van software, opslagruimte en rekencapaciteit online. Binnen Green Orange wordt de term 'cloud-server' vaak gebruikt voor zowel de Interaction Database, de NAS en het CDN.

FMP (FolderManagementPlatform)

- Het FMP van Green Orange is een online platform waar folders worden bewerkt en folderpublicaties worden gemanaged.

UTM (Urchin Tracking Module)

UTM is een stukje code dat gezien kan worden als een 'vlaggetje' dat aan een URL gehangen kan worden. Het doel van UTM-code is om inzicht te krijgen in het verkeer van en naar URL's op verschillende niveau's. Een UTM wordt veelal gebruikt in reclame campagnes in Google Analytics.

Binnen WePublish worden de UTM's voornamelijk op drie niveaus ingezet: source (s), medium (m) en campaign (c). De klant heeft de mogelijkheid om de labels, de naamgeving van de drie niveau's zelf te kiezen.

Source (s): Dit is de folderverzamelplaats waarop de folder wordt aangeboden.
Bijvoorbeeld: spotta.nl, reclamefolder.nl, allefolders.nl

Medium (m): Dit is het medium waarop de consument de folder opent.
Bijvoorbeeld: laptop, pc, tablet, smartphone

Campaign (c): Dit is de naam van de reclamefolder, of van de langdurige actie.
Bijvoorbeeld: 'AH_Pasen_1', 'AH_Pasen_2', 'Gamma_Vaderdag19'

Naast de gebruikelijke UTM niveau's kan een klant een speciale wens hebben om te kijken weten te komen hoe het verkeer naar één specifiek product, of actie zich gedraagt. Bij een dergelijke wens is de mogelijkheid om extra UTM niveau's in te voegen. Bijvoorbeeld op product-niveau.

Product (p): Dit is één product, of een reeks producten uit dezelfde actie.
Bijvoorbeeld: 'Bosch_Boor1', 'Bosch_Boor2', 'Makita_Bouwradio1', óf bijvoorbeeld alle producten van Bosch, Makita.

In is een voorbeeld gegeven van een folder genaamd 'AH_Pasen_1', die op twee verschillende folderverzamelplaatsen geplaatst is. Per datum wordt iedere opgegeven combinatie van UTM's bijgehouden. Uit onderstaande gegevens kunnen we bijvoorbeeld concluderen dat de consumenten die gebruik maken van spotta.nl significant meer gebruik maken van een tablet dan de consumenten die gebruik maken van een smartphone.

Voorbeeld 1

s = spotta.nl
m = smartphone
c = AH_Pasen_1

05-06-18 1.023

Voorbeeld 2

s = reclamefolder.nl
m = tablet
c = AH_Pasen_1

05-06-18 208

Voorbeeld 3

s = spotta.nl
m = tablet
c = AH_Pasen_1

05-06-18 9.236

Figuur 45: UTM 'AH_Pasen_1' (voorbeeld)

Bijlage C: XML voorbeeld

XML bestand:

In Figuur 46 is een kleine stukje van een XML bestand weergegeven. Aan de linker zijde staat de uitleg van de opbouw van het XML bestand, aan de rechter zijde staat de XML-code die op efficiënte wijze gelezen kan worden door een computer.

Product toevoegen	- <Export>
Product seriecode	- <Products_Joined>
Productnaam	<productcode>1211</productcode>
Productprijs	<productname>Omega One FW Flakes .42oz</productname>
Productomschrijving	<productprice>2.99</productprice>
	- <productdescription>
	<h1>Omega One FW Flakes</h1> Delicious recipe of fresh, cold water seafood blended cell wall development. Low ash and higher quality proteins mean less pollution in the tank.
	</productdescription>
Productcategorie	- <categoryids>
	Categories1 Categories2/Subcategories1/Subcategories2 Categories3/Subcategories1
	</categoryids>
Productafbeelding	<productimage>http://www.youserver.com.com/images/ebay/1207.jpg</productimage>
	</Products_Joined>

Figuur 46: XML 'Omega One' (voorbeeld)

Interpretatie van XML bestand:

In Figuur 47 is de interpretatie van het XML bestand uit Figuur 46 weergegeven.

Seriecode	1211
Productnaam	Omega <u>One</u> FW <u>Flakes</u> . 42oz
Productprijs	2,99
Productomschrijving	Omega <u>One</u> FW <u>Flakes</u> . 42oz Dit heerlijke recept van vers koud water <u>zeevoedsel</u> wordt speciaal gemengd om het gehalte aan Omega 6-vetzuren te verhogen. Omega 6-vetzuren zijn uiterst waardevol voor zoetwatersoorten voor sterke immuunsystemen en celwandontwikkeling. Eiwitten met een lage as en hogere kwaliteit betekenen minder vervuiling in de tank.
Productcategorie	Voeding & voederautomaten/Omega <u>One</u> Visvoerders
Productafbeelding	1211.jpg



Figuur 47: XML lezen 'Omega One' (voorbeeld)

Interpretatie van een XML bestand, met URL:

In is de interpretatie van een alternatief XML bestand weergegeven. In dit geval heeft de retailer ervoor gekozen een extra kopje toe te voegen in het originele XML bestand (vergelijkbaar met Figuur 46), waardoor de URL bij dit product is weergegeven.

Dit stelt GreenOrange in staat om het product automatisch semiautomatisch te verwerken in het FMP-editor proces.

Seriecode	190922232222
Productnaam	Makita_Boor1
Productprijs	128,28
Productomschrijving	Makita_Boor1 Accuboormachine voor al uw boorwerk.
Productcategorie	Gereedschap/Accutools, Gereedschap/Boormachines, Gereedschap/ <u>Electronisch</u>
Productafbeelding	190922232222.jpg



Product URL	https://www.gamma.nl/assortiment/makita-accuboormachine-df457dwe-18-volt-lithium-ion/p/B534980
-------------	---

Figuur 48: Interpretatie XML, inclusief URL

Bijlage D: Evaluatie enquête, ontwikkelde methodiek

In Tabel 19 staan de 9 geselecteerde vragen. De vragen zijn ingevuld door vier participanten, die bij iedere vraag een score van 1 (zeer oneens) naar 5 (zeer eens) hebben gegeven. Bij iedere vraag is dikgedrukt weergegeven onder welk kwadrant de vraag valt.

Tabel 19: Evaluatie enquête, ontwikkelde methodiek

Vragen	Participanten:			
	1	2	3	4
Prestatieverwachting - Ik zou de methode bruikbaar vinden in mijn werk	3	3	2	3
Inspaningsverwachting - Ik zou de methode eenvoudig vinden om te gebruiken	4	3	3	3
Houding tegenover de methode - Gebruik maken van de methode is een goed idee	4	5	3	4
Sociale beïnvloeding - Mensen die belangrijk voor mij zijn vinden dat ik de methode moet gebruiken	3	3	4	3
Facilitaire beperkingen - Ik heb de benodigde kennis om de methode te gebruiken	3	5	2	3
Facilitaire beperkingen - Een specifiek persoon (of groep) is beschikbaar voor assistentie wanneer ik problemen ondervind in het gebruik van de methode	5	3	3	2
Zelfredzaamheid - Ik kan het werk, of een taak, voltooien wanneer niemand in de buurt vertelt hoe ik de methode moet gebruiken	2	3	2	3
Zelfredzaamheid - Ik kan het werk, of een taak, voltooien als ik iemand kan bellen wanneer ik vast loop in het gebruik van de methode	4	5	3	5
Vrees - De methodie is ietswat intimiderend voor mij	2	2	2	3
Gedragsintentie ter gebruik van de methode - Ik ben van plan om de methode in de komende twee maanden te gebruiken	5	3	3	3