



HoGent

Faculteit Bedrijf en Organisatie

Productherkenning in de fysieke winkel

Jeremie Van de Walle

Scriptie voorgedragen tot het bekomen van de graad van
professionele bachelor in de toegepaste informatica

Promotor:
Ludwig Stroobant
Co-promotor:
Philip Smet

Instelling: —

Academiejaar: 2017-2018

Tweede examenperiode

Faculteit Bedrijf en Organisatie

Productherkenning in de fysieke winkel

Jeremie Van de Walle

Scriptie voorgedragen tot het bekomen van de graad van
professionele bachelor in de toegepaste informatica

Promotor:
Ludwig Stroobant
Co-promotor:
Philip Smet

Instelling: —

Academiejaar: 2017-2018

Tweede examenperiode

Woord vooraf

Samenvatting

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus.

Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Inhoudsopgave

1	Inleiding	13
1.1	Probleemstelling	13
1.2	Onderzoeksvraag	14
1.3	Opzet van deze bachelorproef	14
2	Stand van zaken	15
2.1	Machine Learning	15
2.1.1	Supervised Learning	16
2.1.2	Unsupervised Learning	16
2.1.3	Reinforcement Learning	16
2.2	Deep Learning	16
2.3	Convolutional Neural Network	17
2.3.1	Convolutional Layer	17

2.3.2	Nonlinair Layer	18
2.3.3	Pooling Layer	18
2.3.4	Fully Connected Layer	19
3	Methodologie	21
3.1	Opbouw van het onderzoek	21
3.2	Frameworks	22
3.2.1	Tensorflow	22
3.2.2	Turi Create	22
3.2.3	Microsoft Custom Vision	23
4	Conclusie	25
A	Onderzoeksvoorstel	27
	Bibliografie	29

Lijst van figuren

2.1	Typische structuur van een Convolutional Neural Network (Shafeen, 2016)	18
2.2	Originele afbeelding gefilterd naar een feature map (Shafeen, 2016)	18

Lijst van tabellen

1. Inleiding

1.1 Probleemstelling

Steeds meer handelszaken verdwijnen door online webwinkels, dit merkt u niet alleen in het dagelijkse leven, ook cijfers benadrukken dit. Tussen 2010 en 2015 sloten maar liefst 10.000 fysieke winkels hun deuren door de opkomst van e-commerce (Knack, 2015). Sterker nog, in 2017 behaalt de Belgische onlinesector een record met net iets meer of 10 miljard euro online-uitgaven. Uit hetzelfde onderzoek van BeCommerce wordt duidelijk dat 8,4 miljoen Belgen minstens één aankoop deden online (BeCommerce, 2018).

Wellicht kent u ook de voordelen van online shoppen. Er is een duidelijk overzicht van de beschikbare artikelen en er kan gemakkelijk gezocht worden naar het nodige product. Vaak toont de webshop de beschikbare voorraad, een uitgebreide uitleg en hetzelfde artikel in andere kleuren of alternatieven hierop. Bovendien wordt in vele gevallen de bestelling de dag nadien aan huis geleverd. Wanneer een bestelling niet voldoet aan de verwachtingen van de consument, dan kan het met gemak teruggestuurd worden.

Maar ook de webwinkel haalt zelf voordelen uit het online winkelen. Er bestaan heel wat verkooptechnieken die online gemakkelijk uit te voeren zijn. Een van de methoden is het aanbieden van producten die vaak samen met het geselecteerde artikel gekocht wordt. Een elektronica internetwinkel kan bijvoorbeeld bijpassende inkt aanbieden bij een gekozen printer. Een andere aanpak is het tonen van alternatieven van het gekozen product die vaak duurder zijn. Als voorbeeld worden er gelijkaardige sweaters aangeboden wanneer een sweater op een mode webshop bekeken wordt. Hiernaast zijn nog tal van manieren om het koopgedrag van klanten te beïnvloeden.

Bovenstaande technieken toepassen in een fysieke handelszaak is niet eenvoudig. Alleen

verkopers kunnen verwante of bijhorende artikels aanbieden. Echter wordt niet elke bezoeker geholpen door een personeelslid. Terugkerende klanten zijn vertrouwd met de winkelomgeving en weten het nodige product te vinden, daarom is het inschakelen van een bediende overbodig. Anderzijds is er ook cliënteel die liever op zelfstandige basis winkelt. Hierdoor is het vaak ingewikkelder om in te spelen om het aankoopgedrag van consumenten.

1.2 Onderzoeksvraag

Dit onderzoek zal drie machine learning frameworks vergelijken die oplopen in complexiteit in implementatie. Het zal vergelijken of een complex framework even nauwkeurig is als een makkelijk implementeerbaar framework. Daarnaast gaat dit onderzoek na of productherkenning door middel van machine learning een oplossing kan bieden op het beperkt aanbieden van productinformatie in een fysieke winkel. Een praktische applicatie zal verdere productinformatie geven voor een bepaald product, en hierop wordt gecontroleerd of het koopgedrag van de consument beïnvloed wordt. Maar ook of de consument deze applicatie zelf als meerwaarde ziet. Concreet bestaat het onderzoek uit drie deelonderzoeksvragen:

- Wat is het nauwkeurigste machine learning framework voor productherkenning?
- Wordt productherkenning als hulpmiddel beschouwd?
- Beïnvloed productherkenning het koopgedrag van de consument?

1.3 Opzet van deze bachelorproef

2. Stand van zaken

2.1 Machine Learning

Machine learning is een vakgebied binnen artificiële intelligentie dat het mogelijk maakt om presentaties van een systeem te verbeteren op basis van voorbeeldgegevens of ervaringen uit het verleden. Het biedt computers de mogelijkheid om te leren zonder expliciet geprogrammeerd te zijn (De Vreese, 2017). Sommige taken of problemen kunnen niet voorafgaand geprogrammeerd worden. Vaak zijn ze te complex en onpraktisch om rechtstreeks door mensen gecodeerd te worden. Evenwel wordt het moeilijk om in een veranderende omgeving een programma te ontwikkelen die optimaal werkt omdat bepaalde kenmerken van de omgeving niet bekend zijn tijdens het ontwerp (Nilsson, 1998).

Gebrek aan kennis wordt bij machine learning goedge maakt door data. Op basis van een groot aantal voorbeelden kan een model gedefinieerd worden om voorspellingen te doen of om kennis te verzamelen. Een voorbeeldtoepassing hiervan is een spamfilter. Om reden dat spam veranderd met de tijd en verschilt van persoon tot persoon is het ingewikkeld om een specifiek algoritme te ontwikkelen die spam e-mails onderscheidt van geldige e-mails. De bedoeling is dat een computer automatisch een algoritme uitwerkt om spam te herkennen op basis van duizenden gekende spam e-mails en geldige e-mails (Alpaydin, 2012). Verder wordt machine learning gebruikt in allerlei toepassingen in verschillende sectoren. Zo gebruikt Netflix en Spotify een aanbevelingsalgoritme om hun klanten te voorzien van films respectievelijk muziek die ze waarschijnlijk leuk vinden. Daarnaast gebruikt Paypal dergelijke algoritmes om fraude te detecteren (Feldman, 2018).

Door het veelzijdig gebruik van machine learning algoritmes met verschillende doelstellingen kan het onderverdeeld worden in drie hoofdcategorieën: Supervised learning, Unsupervised learning, Reinforcement learning.

2.1.1 Supervised Learning

Supervised machine learning bouwt een model steunend op een reeks invoergegevens waarvan de uitvoer gekend is. Bij de uitvoer wordt er onderscheidt gemaakt tussen regressie, waar het resultaat een numerieke waarde is en classificatie waar kwalitatieve waarde is zoals een klasse of een label. Een regressietaak is bijvoorbeeld het voorspellen van de wisselkoers. Een classificatie opdracht moet dan weer onderscheidt maken tussen bier en wijn op basis van het kleur en alcoholgehalte. Het supervised algoritme gebruikt de bekende invoergegevens om voorspellingen te genereren voor de uitvoer van nieuwe gegevens. Het algoritme kan steeds verder getraind worden door het toevoegen van voorbeeldgegevens tot het een acceptabel nauwkeurigheidsniveau heeft bereikt (Mueller & Massaron, 2016).

2.1.2 Unsupervised Learning

Wanneer er enkel invoerdata beschikbaar is, is er sprake van unsupervised learning. Het algoritme gaat opzoek naar verborgen patronen of onbekende structuren in de gegevens. Het heeft als doel om bruikbare informatie uit een set van invoergegevens te halen. Voor mensen is dit type algoritme nuttig om inzicht te krijgen in een groep verzamelde gegevens. De meest voorkomende techniek is clustering, het verdeelt de gegeven data in gelijkaardige groepen. Het groeperen van klanten op basis van het koopgedrag is een veelgebruikt voorbeeld van clustering (Mathworks, 2018).

2.1.3 Reinforcement Learning

Wanneer reinforcement learning van toepassing is, dan leert een machine uit zijn voorheen genomen acties. Elke actie is verbonden met een resultaat dat positief of negatief kan zijn. Neemt een machine een verkeerde actie, dan levert dit een kost op. Een goede actie levert een beloning op. Het algoritme leert continu op een iteratieve manier van de omgeving en zijn ondernomen beslissingen met als doel het maximaliseren van de beloning op lange termijn. Reinforcement learning wordt gebruikt om computers zelf aan te leren om videogames te spelen (Sutton & Barto, 2017).

2.2 Deep Learning

Deep learning is een onderdeel van machine learning die gebruikt maakt van algoritmes die gebaseerd zijn op de structuur van het menselijk brein. Klassieke machine learning technieken zijn beperkt in hun capaciteit om gegevens te verwerken. Tussen de invoer en het resultaat zit doorgaands een enkele verwerkingslaag die gebruikt worden voor voorspellingen of classificatie. Diep learning verwerkt de invoer op meerdere niveaus met toenemende complexiteit en abstractie. Elke verborgen laag in de hiërarchie past een transformatie toe op de uitvoer van de voorgaande laag tot het een aanvaardbaar maat van nauwkeurigheid heeft bereikt. Bij elke iteratie wordt het model dat de computer ontwikkeld

dus complexer maar ook nauwkeuriger. Het grote voordeel hierbij is dat het algoritme zichzelf aanleert om laag per laag aspecten van de input te herkennen (LeCun, Bengio, & Hinton, 2015).

Als voorbeeld wordt een analogie uit het dagelijkse leven aangehaald. Een peuter leert wat een kat is door naar dieren te wijzen en het woord kat uit te spreken. De ouder corrigeert de peuter wanneer hij naar een foutief dier wijst. Na verloop van tijd wordt de peuter zich meer en meer bewust van de kenmerken die een kat heeft. Hij bouwt een hiërarchie op van verschillende lagen waarin elk niveau van abstractie gecreeërd wordt met de kennis die opgenomen is uit de voorgaande laag. Deep learning algoritmes doorgaan hetzelfde proces. Het ontvangt een gegevensset waarvan geweten is of het al dan niet een kat is. Daarna wordt de elke input getransformeerd naar een volgend niveau om zo patronen te herkennen. In de eerste laag wordt bijvoorbeeld gezocht naar de aan- of afwezigheid van randen op bepaalde locaties in het beeld. In de tweede laag kunnen als voorbeeld vormen gedetecteerd worden zoals oren en poten. Zo wordt de input zodanig getransformeerd totdat het model nauwkeurigheid genoeg is om voorspellingen te doen (Rouse, 2018).

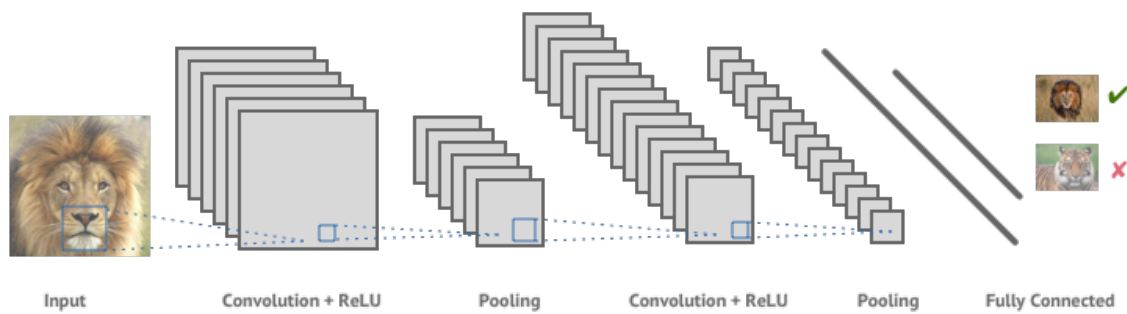
Het doel van deep learning is de werking van het menselijk brein zo goed mogelijk te reproduceren. Er zijn al heel wat applicaties die gebruikmaken van deep learning algoritmen. Afbeelding- en spraakherkenning zijn de meest voorkomende toepassingen. Maar ook het automatisch inkleuren van zwartwit foto's en zelfrijdende auto's maken gebruik van d

2.3 Convolutional Neural Network

Een convolutional neural network, afgekort CNN is een supervised deep learning algoritme dat vooral gebruikt wordt voor beeld- en videoherkenning. Ook voor het classificeren van afbeeldingen wordt een CNN gebruikt. Het netwerk krijgt hierbij een afbeelding als input en geeft de waarschijnlijkheid dat de afbeelding tot een klasse behoort terug als resultaat. De input kan hierbij aanzien worden als een drie dimensionale array van pixels. Een afbeelding met een grootte van 20 bij 20 pixels heeft een array van $20 \times 20 \times 3$ pixels. De diepte 3 is omwille van de drie kleurkanalen rood, groen en blauw. Voor elk kanaal is er een twee dimensionale array waarbij elke pixel een waarde heeft tussen 0 en 255. Bij een zwart-wit beeld is de diepte vanzelfsprekend gelijk aan 1. In dit netwerk bestaan de verborgen lagen uit een aantal convolutional, nonlineair, pooling en fully connected layers (Rohrer, 2016). Figuur 2.1 geeft de grafische voorstelling hoe een CNN is opgebouwd weer.

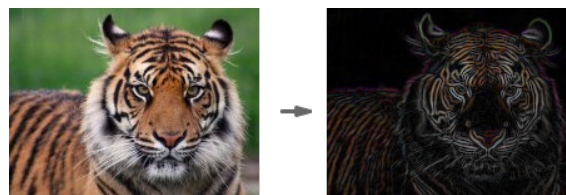
2.3.1 Convolutional Layer

De eerst verborgen laag in een CNN is een convolutional layer. In deze laag wordt elk deel van de afbeelding vermenigvuldigd door bepaalde filters, een filter schuift of convoleert als het ware over de afbeelding. De bedoeling van een filter is om een feature of kenmerk te detecteren in een afbeelding. Dit kan gaan van eenvoudige herkenbare vormen zoals



Figuur 2.1: Typische structuur van een Convolutional Neural Network (Shafeen, 2016)

randen en bepaalde curves tot complexere structuren zoals ogen en oren. Het resultaat van de vermenigvuldiging geeft een quotering in hoeverre de feature aanwezig is. Nadat de filter over alle locaties van de invoer heeft geschoven vormen alle resultaten één nieuwe matrix. In de convolutional layer wordt één afbeelding omgezet naar een groep gefilterde afbeeldingen, afhankelijk van het aantal filters. Deze output wordt de input voor de volgende verborgen laag (Vernwal, 2017). Figuur 2.2 toont een mooi voorbeeld van een feature map waarbij de filter randen detecteerde op een afbeelding van een tijger.



Figuur 2.2: Originele afbeelding gefilterd naar een feature map (Shafeen, 2016)

2.3.2 Nonlinair Layer

Deze laag ontvangt als input een verzameling van gefilterde afbeeldingen of ‘feature maps’. Aangezien de filter of functie uit de vorige laag lineair is, kan het resultaat op een deel van de afbeelding ook negatief zijn. De nonlinair layer zet de lineaire convolutielaag om naar een niet-lineaire laag door een gelijkrichterfunctie, ReLu (Rectified Linear Unit) toe te passen. Deze functie zet alle negatieve waarden om naar de nulwaarde (Hijazi, Kumar, & Rowen, 2015).

2.3.3 Pooling Layer

Pooling, of ook wel subsampling genoemd wordt gebruikt om de dimensie of grootte van de feature maps te verkleinen. In deze laag wordt het vaakst max-pooling toegepast. Een filter met een bepaalde dimensie overloopt de volledige feature map en haalt de maximum waarde daaruit. Net zoals in de nonlinair layer wordt de functie toegepast op elke feature map in de groep van feature maps die de convolutional layer als output gaf. Deze laag wordt

gebruikt om de gebieden die het minst overeenstemmen met de filters uit de convolutionele laag te verwijderen om zo de berekeningen in het netwerk te verminderen (Jordan, 2017).

2.3.4 Fully Connected Layer

Voorgaande verborgen lagen kunnen steeds na elkaar uitgevoerd worden. Aan het eind van het netwerk wordt de fully connected layer toegepast. Deze laag krijgt als input een groep op hoog niveau gefilterde afbeeldingen en bekijkt welke functies het meest overeenkomen met een bepaalde klasse. Het geeft een vector terug met de waarschijnlijkheid dat de afbeelding tot die klasse behoort, per klasse terug (ujjwalkarn, 2016).

3. Methodologie

3.1 Opbouw van het onderzoek

Dit onderzoek vergelijkt drie frameworks voor afbeeldingclassificatie. Er wordt onderzocht of eenvoudig implementeerbare frameworks even nauwkeurig resulteren als complexere frameworks. De drie frameworks worden getraind met negen producten uit de Feestwinkel, te Oudenaarde. De producten worden onderverdeeld in drie groepen: kleine producten, middelgrote producten en grote producten. Per groep wordt er voor elk framework een eigen classificatiemodel gecreëerd.

Als eerste framework is er gekozen voor een standaard machine learning bibliotheek. Onder deze noemer vallen frameworks zoals Caffe, Torch, Theano en Tensorflow. In dit onderzoek is gekozen voor Tensorflow, het scoort naast Theano het beste rekening houdend met snelheid, classificatie nauwkeurigheid en complexiteit van implementatie (Kovalev, Kovalev, & Kalinovsky, 2016). Bovendien is er heel wat informatie te vinden op het web omtrent het toepassen van Tensorflow voor verschillende doeleinden (Huyen, 2017). De andere twee frameworks die worden toegepast in dit onderzoek zijn Turi Create en Microsoft Custom Vision. Deze frameworks zijn respectievelijk eenvoudiger in het ontwikkelen van een eigen afbeeldingclassificatie model. De drie frameworks zijn gratis te gebruiken.

Nadat een eigen classificatie model getraind is met de trainingsdata per groep van producten, zal het model geëxporteerd worden naar een CoreML model. Het CoreML framework is gelanceerd door Apple op het WWDC2017. Het is een nieuwe toepassing in iOS 11 en het zorgt ervoor dat ontwikkelaars gemakkelijk machine learning models kunnen uitvoeren in hun applicaties. Zo ondersteund ook CoreML functies zoals gezichts- herkenning en tekstherkenning (Apple, 2017a).

3.2 Frameworks

3.2.1 Tensorflow

Tensorflow is open source deep learning framework door Google. Het is oorspronkelijk ontwikkeld in 2015 door het Google Brain Team om onderzoek te doen naar zowel machine learning, deep learning en neurale netwerken. In 2017 is de eerste open source versie van Tensorflow uitgebracht. Tensorflow wordt onder andere gebruikt voor tekstherkenning en stemherkenning maar ook voor afbeeldingclassificatie. De gemakkelijkste en de meest complete API van dit framework is de Python API, maar Tensorflow levert ook nog een API's in Java, C++ en Go (Solano, 2017).

Dit framework kan ook worden toegepast om een eigen afbeeldingclassificatie model te ontwikkelen. Het bouwen van een model vanaf nul kan enkele weken duren, daarom gebruikt Tensorflow een deel van een classificatie model die reeds eerder getraind is en behoorlijk presteert in het classificeren van afbeeldingen uit allerlei categorieën. Het model dat benut wordt is Inception v3 en het is in getraind met ImageNet, een grote database van afbeeldingen. Het CNN kan afbeeldingen classificeren in duizend verschillende klassen (Tensorflow, 2018).

Tensorflow maakt gebruik van transfer learning om een eigen model te generen met eigen trainingsdata. Het is minder efficiënt dan het trainen van het model vanaf nul, maar het maakt het wel mogelijk om een degelijk model te ontwikkelen met aanzienlijk veel minder trainingsdata en tijd. Tensorflow berekend voor elke afbeelding uit de training set de bottleneckwaarde en slaat deze tijdelijk op. De term 'bottleneck' wordt door Tensorflow gebruikt als een verwijzing naar de laag net voor de laatste laag die verantwoordelijk is voor de classificatie. Tensorflow verwijdert de laatste laag van het Inception v3 model, berekend de bottleneckwaarde van elke afbeelding, en traint een nieuwe laatste laag voor classificatie. Standaard voert Tensorflow vierduizend trainingstappen uit met tien willekeurige afbeeldingen uit de trainingsdata. De bottleneckwaarden van de afbeeldingen worden opgeslagen zodat de waarden niet steeds opnieuw moeten berekend worden door het model wanneer de afbeeldingen hergebruikt worden in de trainingstap (Zaccane, Karim, & Menshawy, 2017).

3.2.2 Turi Create

Apple kocht in 2016 het bedrijf Turi over. Kort daarna gaf Apple het framework Turi Create vrij voor het publiek (Miller, 2017). Apple beweert dat men met hun framework geen deskundige in machine learning hoeft te zijn. Een eigen afbeeldingclassificatie model kan gebouwd worden in enkele lijnen Python. Daarnaast kan het framework gebruikt worden voor onder andere objectdetectie en tekstclassificatie (Apple, 2017b).

Turi Create functioneert volledig gelijklopend zoals Tensorflow. Het past ook transfer learning, met het verschil dat het dit framework gebruik maakt van het convolutional neural network ResNet-50 en niet Inception v3 (Toole, g.d.). Uit een vergelijkende studie van de nauwkeurigheid van verschillende CNN's scoort Inception v3 met een nauwkeurigheid

van 79%, waar ResNet-50 scoort met 76% (Canziani, Culurciello, & Paszke, 2017).

3.2.3 Microsoft Custom Vision

Microsoft lanceerde de Custom Vision API in 2017. De API is een uitbreiding op de Microsoft Azure Cognitive Services, een verzameling van verschillende machine learning API's. Ontwikkelaars kunnen op die manier eenvoudiger functies van artificiële intelligentie toepassen in hun applicaties. Zo kunnen de API's gebruikt worden voor onder andere gezichts-, en spraakherkenning. Maar ook video- en emotiedetectie. Daarnaast kunnen de services benut worden voor tekstanalyse en taalbegrip (Microsoft, 2016).

De Custom Vision API wordt gebruikt om een eigen afbeeldingclassificatie model te ontwikkelen. Deze API is in de vorm van een webapplicatie en vereist geen enkele kennis omtrent machine learning. Er dienen enkele afbeeldingen te worden online geplaatst per categorie en het model wordt in luttele minuten getraind (Lele, 2017).

De details over hoe microsoft erin slaagt om met een ettelijk aantal afbeeldingen een nauwkeurig classificatiemodel te creëren, houden ze voor zich. Al lijkt het er sterk op dat ook dit framework transfer learning toepast een voorgetraind CNN. Microsoft heeft namelijk reeds een model dat gebruikt wordt in de Computer Vision API, het model kan een afbeelding classificeren op basis van meer dan tweeduizend categorieën waarop het model is getraind. Past Microsoft op dit model transfer learning toe zoals Tensorflow en Turi Create, dan is het logisch dat een eigen classificatiemodel goed scoort met een minimum aan afbeeldingen (Krueger, 2017).

4. Conclusie

Curabitur nunc magna, posuere eget, venenatis eu, vehicula ac, velit. Aenean ornare, massa a accumsan pulvinar, quam lorem laoreet purus, eu sodales magna risus molestie lorem. Nunc erat velit, hendrerit quis, malesuada ut, aliquam vitae, wisi. Sed posuere. Suspendisse ipsum arcu, scelerisque nec, aliquam eu, molestie tincidunt, justo. Phasellus iaculis. Sed posuere lorem non ipsum. Pellentesque dapibus. Suspendisse quam libero, laoreet a, tincidunt eget, consequat at, est. Nullam ut lectus non enim consequat facilisis. Mauris leo. Quisque pede ligula, auctor vel, pellentesque vel, posuere id, turpis. Cras ipsum sem, cursus et, facilisis ut, tempus euismod, quam. Suspendisse tristique dolor eu orci. Mauris mattis. Aenean semper. Vivamus tortor magna, facilisis id, varius mattis, hendrerit in, justo. Integer purus.

Vivamus adipiscing. Curabitur imperdiet tempus turpis. Vivamus sapien dolor, congue venenatis, euismod eget, porta rhoncus, magna. Proin condimentum pretium enim. Fusce fringilla, libero et venenatis facilisis, eros enim cursus arcu, vitae facilisis odio augue vitae orci. Aliquam varius nibh ut odio. Sed condimentum condimentum nunc. Pellentesque eget massa. Pellentesque quis mauris. Donec ut ligula ac pede pulvinar lobortis. Pellentesque euismod. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent elit. Ut laoreet ornare est. Phasellus gravida vulputate nulla. Donec sit amet arcu ut sem tempor malesuada. Praesent hendrerit augue in urna. Proin enim ante, ornare vel, consequat ut, blandit in, justo. Donec felis elit, dignissim sed, sagittis ut, ullamcorper a, nulla. Aenean pharetra vulputate odio.

Quisque enim. Proin velit neque, tristique eu, eleifend eget, vestibulum nec, lacus. Vivamus odio. Duis odio urna, vehicula in, elementum aliquam, aliquet laoreet, tellus. Sed velit. Sed vel mi ac elit aliquet interdum. Etiam sapien neque, convallis et, aliquet vel, auctor non, arcu. Aliquam suscipit aliquam lectus. Proin tincidunt magna sed wisi. Integer blandit

lacus ut lorem. Sed luctus justo sed enim.

Morbi malesuada hendrerit dui. Nunc mauris leo, dapibus sit amet, vestibulum et, commodo id, est. Pellentesque purus. Pellentesque tristique, nunc ac pulvinar adipiscing, justo eros consequat lectus, sit amet posuere lectus neque vel augue. Cras consectetur libero ac eros. Ut eget massa. Fusce sit amet enim eleifend sem dictum auctor. In eget risus luctus wisi convallis pulvinar. Vivamus sapien risus, tempor in, viverra in, aliquet pellentesque, eros. Aliquam euismod libero a sem.

Nunc velit augue, scelerisque dignissim, lobortis et, aliquam in, risus. In eu eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Curabitur vulputate elit viverra augue. Mauris fringilla, tortor sit amet malesuada mollis, sapien mi dapibus odio, ac imperdiet ligula enim eget nisl. Quisque vitae pede a pede aliquet suscipit. Phasellus tellus pede, viverra vestibulum, gravida id, laoreet in, justo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Integer commodo luctus lectus. Mauris justo. Duis varius eros. Sed quam. Cras lacus eros, rutrum eget, varius quis, convallis iaculis, velit. Mauris imperdiet, metus at tristique venenatis, purus neque pellentesque mauris, a ultrices elit lacus nec tortor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent malesuada. Nam lacus lectus, auctor sit amet, malesuada vel, elementum eget, metus. Duis neque pede, facilisis eget, egestas elementum, nonummy id, neque.

A. Onderzoeksvoorstel

Het onderwerp van deze bachelorproef is gebaseerd op een onderzoeksvoorstel dat vooraf werd beoordeeld door de promotor. Dat voorstel is opgenomen in deze bijlage.

Bibliografie

- Alpaydin, E. (2012). *Introduction to machine learning*. Cambridge University Press.
- Apple. (2017a). Core ML. Verkregen van <https://developer.apple.com/documentation/coreml>
- Apple. (2017b). Turi Create Github. Verkregen van <https://github.com/apple/turicreate>
- BeCommerce. (2018, maart). De kaap van 10 miljard online bestedingen in 2017 vlot behaald. Verkregen van <https://www.becommerce.be/nl/nieuws/d/detail/de-kaap-van-10-miljard-online-bestedingen-in-2017-vlot-behaald>
- Canziani, A., Culurciello, E., & Paszke, A. (2017). An Analysis Of Deep Neural Network Models For Practical Applications. Verkregen van <https://arxiv.org/pdf/1605.07678.pdf>
- De Vreese, S. (2017). *Introduction to Machine Learning with Spark*.
- Feldman, M. (2018, maart). 10 Real-World Examples of Machine Learning and AI. Verkregen van <https://www.redpixie.com/blog/examples-of-machine-learning>
- Hijazi, S., Kumar, R., & Rowen, C. (2015). Using Convolutional Neural Networks for Image Recognition. Verkregen van https://ip.cadence.com/uploads/901/cnn_wp-pdf
- Huyen, C. (2017). TensorFlow for Deep Learning Research. Verkregen van https://web.stanford.edu/class/cs20si/2017/lectures/slides_01.pdf
- Jordan, J. (2017, augustus). Convolutional neural networks. Verkregen van <https://www.jeremyjordan.me/convolutional-neural-networks/>
- Knack. (2015, november). Verdwijnt de lokale winkel binnenkort uit het straatbeeld? Verkregen van <http://trends.knack.be/economie/bedrijven/verdwijnt-de-lokale-winkel-binnenkort-uit-het-straatbeeld/article-normal-628617.html>
- Kovalev, V., Kovalev, S., & Kalinovsky, A. (2016, oktober). Deep Learning with Theano, Torch, Caffe, TensorFlow, and Deeplearning4J: Which One Is the Best in Speed And Accuracy. Verkregen van https://www.researchgate.net/profile/Vassili_Kovalev/publication/302955130_Deep_Learning_with_Theano_Torch_Caffe_

- TensorFlow_and_DeepLearning4J_Which_One_Is_the_Best_in_Speed_and_Accuracy/links/596f1e610f7e9bd5f763bcfa/Deep-Learning-with-Theano-Torch-Caffe-TensorFlow-and-DeepLearning4J-Which-One-Is-the-Best-in-Speed-and-Accuracy.pdf
- Krueger, F. A. (2017). "Hotdog or Not" Using Azure Custom Vision, CoreML, and Xamarin. Verkregen van <http://praeclarum.org/post/165799177168/hotdog-or-not-using-azure-custom-vision-coreml>
- LeCun, Y., Bengio, Y., & Hinton, G. (2015, mei). Deep learning. Verkregen van <https://www.cs.toronto.edu/~hinton/absps/NatureDeepReview.pdf>
- Lele, V. (2017). Training the Custom Vision API to Detect Roof Types. Verkregen van <https://www.appliedis.com/training-the-custom-vision-api-to-detect-roof-types/>
- Mathworks. (2018). Mastering Machine Learning. Verkregen van <https://nl.mathworks.com/content/dam/mathworks/tag-team/Objects/m/machine-learning-workflow-ebook.pdf>
- Microsoft. (2016). Cognitive Services. Verkregen van <https://azure.microsoft.com/en-us/services/cognitive-services/>
- Miller, C. (2017). Apple releases Turi Create framework on GitHub to 'simplify development of machine learning models'. Verkregen van <https://9to5mac.com/2017/12/08/turi-create-framework-release/>
- Mueller, J. P. & Massaron, L. (2016). 3 Types of Machine Learning. Verkregen van <http://www.dummies.com/programming/big-data/data-science/3-types-machine-learning/>
- Nilsson, J. N. (1998, maart). Introduction To Machine Learning. Verkregen van <http://robotics.stanford.edu/people/nilsson/MLBOOK.pdf>
- Rohrer, B. (2016, augustus). How do Convolutional Neural Networks work? Verkregen van http://brohrer.github.io/how_convolutional_neural_networks_work.html
- Rouse, M. (2018, januari). deep learning (deep neural network). Verkregen van <https://searchenterpriseai.techtarget.com/definition/deep-learning-deep-neural-network>
- Shafeen, T. (2016). Machines that can see: Convolutional Neural Networks. Verkregen van <https://shafeentejani.github.io/2016-12-20/convolutional-neural-nets/>
- Solano, A. (2017). Introduction To Tensorflow. Verkregen van <https://ep2017.europython.eu/media/conference/slides/introduction-to-tensorflow.pdf>
- Sutton, R. S. & Barto, A. G. (2017, november). Reinforcement Learning: An Introduction. Verkregen van <http://incompleteideas.net/book/bookdraft2017nov5.pdf>
- Tensorflow. (2018). How to Retrain an Image Classifier for New Categories. Verkregen van https://www.tensorflow.org/tutorials/image_retraining
- Toole, J. (g.d.). Building Not Hotdog with Turi Create and Core ML. Verkregen van <https://hackernoon.com/building-not-hotdog-with-turi-create-and-core-ml-in-an-afternoon-231b14738edf>
- ujjwalkarn. (2016, augustus). An Intuitive Explanation of Convolutional Neural Networks. Verkregen van <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>
- Vernwal, R. (2017, juli). How should I start with CNN. Verkregen van <https://becominghuman.ai/how-should-i-start-with-cnn-c62a3a89493b>
- Zaccone, G., Karim, R., & Menshawy, A. (2017). *Deep Learning with TensorFlow: Explore neural networks with Python*. Packt.