HO
GENT

Faculteit Bedrijf en Organisatie

# Automatische transformatie van ingescande tabellen naar gestructureerde digitale data

Milad Nazari

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

Promotor:
Martijn Saelens
Co-promotor:
Bram Vandewalle

Instelling: Into Care by Predictive NV

Academiejaar: 2019-2020

Derde examenperiode

Faculteit Bedrijf en Organisatie

# Automatische transformatie van ingescande tabellen naar gestructureerde digitale data

Milad Nazari

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

Promotor:
Martijn Saelens
Co-promotor:
Bram Vandewalle

Instelling: Into Care by Predictive NV

Academiejaar: 2019-2020

Derde examenperiode

# Woord vooraf

Ik zou graag meneer Vandewalle willen bedanken voor enerzijds deze bachelorproefonderwerp en anderzijds voor de inhoudelijke ondersteuning en hulp die hij aangeboden en gegeven heeft. Hiernaast wil ik eveneens meneer Saelens bedanken voor de feedback en opvolging van mijn bachelorproef.

# Samenvatting

# Inhoudsopgave

# Lijst van figuren

# Lijst van tabellen

# Woordenlijst

**CNN**

Een Convolutionele Neuraal Netwerk (CNN) is een klasse van diepe neurale netwerken waarbij gebruik gemaakt wordt van een variatie van multilayer perceptrons ontworpen om minimale eisen voorbewerking.. 28, 29, 30

**F1-score**

De F1-score is een numerieke waarde die men verkrijgt door de product van de Recall en de Precisie te delen door de som van de Recall en de Precisie, en deze waarde nog met 2 te vermenigvuldigen.. 34

**GNN**

Een Graaf Neuraal Netwerk (GNN) is een neurale netwerkarchitectuur die relaties en interacties tussen knopen kan modelleren en deze voorstellen als numerieke waarden.. 29, 30

**Hidden Markov Model**

Hidden Markov Model (HMM) is een model uit de statistiek waarin het te modelleren systeem een markov-proces is met onbekende parameters. Het doel is de verborgen parameters te bepalen op basis van de waarneembare parameters. De op deze manier verkregen parameters kunnen vervolgens worden gebruikt voor toepassingen als patroonherkenning.. 28

**IoU**

De Intersection Over Union (IoU) is een concept die veelal gebruikt wordt bij objectdetectie. De IoU kan men berekenen door de intersectieoppervlakte te delen over de unieoppervlakte van twee regio's: de ground truth regio en de voorspelde regio. Met de IoU wordt steeds een threshold gekoppeld. Indien men voor een

threshold van 0,5 kiest en de IoU voor een voorspelling gelijk is of groter is dan de threshold van 0,5, dan heeft men een echt positief (true positive), anders heeft men te maken met een foutpositief. Een foutnegatief verkrijgt men wanneer een ground truth aanwezig is maar geen voorspelling gemaakt kon worden.. 34

**JSON**
JavaScript Object Notation (JSON) is een gestandaardiseerd dataformaat.. 32

**OCR**
Optical Character Recognition (OCR), optische tekenherkenning, is de transformatie van afbeeldingtekst in bewerkbare, digitale tekst.. 8, 17, 18, 20, 35

**Precisie**
De precisie is een numerieke waarde die bepaald wordt het aantal echt positieven te delen door de som van het aantal echt positieven en foutnegatieven.. 15, 34

**Recall**
De recall is een numerieke waarde die bepaald wordt door het aantal echt positieven te delen door de som van het aantal echt positieven en foutpositieven.. 15, 34

**REST**
Representational State Transfer (REST) is een software-architectuur voor Internet-diensten.. 32, 36

**ROI**
Een Region Of Interest (ROI) is een subset van een dataset, geïdentificeerd voor een specifiek doel.. 28

**SVM**
Een Support Vector Machine (SVM) is een binaire classificeerder; ze wijst aan de hand van een aantal kenmerken objecten toe aan een van twee klassen. Daarvoor moet ze eerst een numeriek model van deze objecten maken als punten in een vectorruimte.. 28

**tupel**
In de wiskunde en de informatica is een tupel (ook tuple) een eindige rij van objecten. In een tupel is de volgorde van belang; als de objecten in een andere volgorde staan is het een ander tupel. Ook hoeven de objecten niet van hetzelfde datatype te zijn.. 23

# 1. Inleiding

In deze sectie wordt de context en achtergrond rond deze bachelorproef meegedeeld. Alsook wordt de probleemstelling, de onderzoeksvragen en onderzoeksdoelstellingen uitgelegd. Daarbovenop wordt de opzet van de bachelorproef verduidelijkt.

## 1.1 Probleemstelling

Alhoewel meer en meer processen wereldwijd volledig digitaal plaatsvinden, worden toch nog een grote deel van procedures en data opslag uitgevoerd op niet-digitale manieren. Zo krijgen de meeste mensen hun factures nog steeds per brief. Volgens de Federale Overheidsdienst Economie (2019) blijft het verzenden of ontvangen van facturen op papier een zeer gangbare praktijk. Zo verstuurde 90 % van de bedrijven er en 97 % ontving er in 2017. Daarbovenop worden kassatickets nog steeds afgedrukt op papier, en notities nemen op papier blijft de populaire keuze hoewel er tal van notitie-apps bestaan. Deze voorbeelden tonen aan dat essentiële data nog massaal op een niet-digitale, en dus niet-automatisch verwerkbare media bewaard wordt, namelijk op papier.

Tot enkele jaren geleden was dit probleem niet zo beduidend maar nu meer digitale platformen voor dataverwerking gebruikt worden, is het omzetten van data op papier naar digitale data, m.a.w. het digitalisatieproces steeds belangrijker geworden.

Hierdoor werden tal van digitalisatiesofwareproducten ontwikkeld, zoals Abby FineReader en Adobe Acrobat Pro DC. Hoewel deze software producten veel features hebben, zoals OCR, tabelherkenning, formulierherkenning, etc, zijn ze betalend en closed source. Wat als gevolg heeft dat ze voor bedrijven een merkbare kost met zich meebrengen, naast een privacy- en veiligheidsrisico aangezien het om closed source software gaat.

Sommige bedrijven enkele van hun digitalisatie oplossingen open source gemaakt, zoals Google met diens bekende OCR-software, Tesseract OCR, die door iedereen gebruikt kan worden om tekst in foto's om te zetten in tekstdata. Hoewel OCR op zich zeer belangrijk is voor digitalisatie, is het niet voldoende voor volledige digitalisatie. Zo kan men de relatie tussen verschillende documententiteiten, die normaal gezien grafisch wordt verduidelijkt, enkel met OCR digitaal niet overbrengen. In documenten worden relaties tussen woorden meestal a.d.h.v. een tabel verduidelijkt. Door gebruik te maken van OCR, verkrijgt men wel de tekst binnen een tabel, maar men verliest essentiële informatie rond de woorden, namelijk tot welke rij en kolom ze behoorden. Het valt tenslotte niet onder de verantwoordelijkheid van OCR-engines om naast tekstherkenning, ook nog tabeltransformatie uit te voeren.

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Product | Kw 1 | Kw 2 | Eindtotaal |
| 2 | Chocolade | € 744,60 | € 162,56 | € 907,16 |
| 3 | Gummibarchen | € 5.079,60 | € 1.249,20 | € 6.328,80 |
| 4 | Scottish Longbreads | € 1.267,50 | € 1.062,50 | € 2.330,00 |
| 5 | Sir Rodney's Scones | € 1.418,00 | € 756,00 | € 2.174,00 |
| 6 | Tarte au sucre | € 4.728,00 | € 4.547,92 | € 9.275,92 |
| 7 | Chocoladekoekjes | € 943,89 | € 349,60 | € 1.293,49 |
| 8 | Totaal | € 14.181,59 | € 8.127,78 | € 22.309,37 |

Figuur 1.1: Voorbeeld van een tabelafbeelding. Bron: Microsoft (2020)

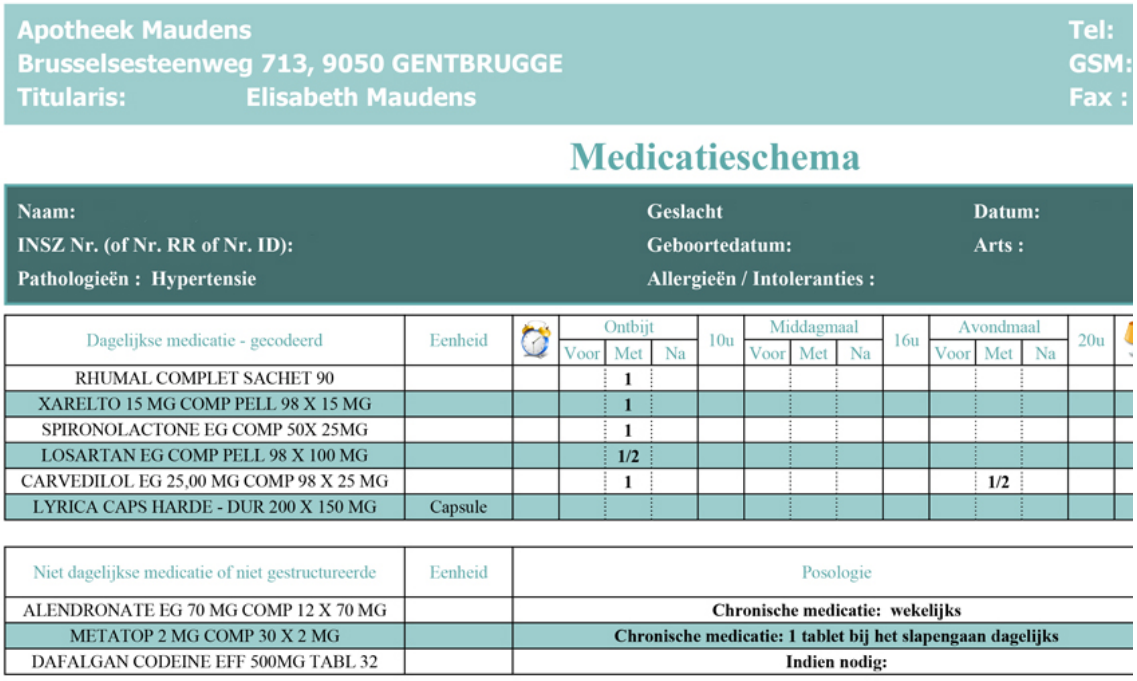
Indien men bij tabelafbeelding 1.1 enkel OCR voor digitalisatie zou gebruiken, dan verkrijgt men wel de tekst, zoals de tekststukken zoals "Kw 1", "Kw 2", "€744,60", "€ 162,56", en meer, maar men behoudt niet de relatie tussen de tekststukken. Hierdoor zal men enkel met OCR niet te weten komen of de verkoopbedrag van € 744,60 bij de eerste kwartaal behoort, of bij de tweede, wat essentiële informatie is voor verdere financïele analyse.

Tot heden bestaat er geen open source oplossing die tabellen in foto's transformeert naar digitale tabellen, m.a.w. naar digitale structuren waarbij de tekst, evenals de relatie tussen de verschillende teksten getransformeerd wordt. Daarom werd er voor deze bachelorproef besloten om een proof-of-concept van een tabeltransformatiesoftware te creëren die bij een foto automatisch tabellen detecteert en deze tabellen digitaliseert.

Een belangrijke professionele toepassing van digitale tabeltransformatie is het digitaliseren van ingescande medicatieschema's, door technologiebdrijven zoals Into.care die zich bezig houden met digitale gezondheidszorg. Medicatieschema's worden in de gezonheidszorg gebruikt om medicatiedata voor patiënten te bewaren en weer te geven. Volgens de definitie van Apothekersnetwerk (Apothekersnetwerk, 2013) is het medicatieschema een geheel van gestandaardiseerde informatie over de actieve medicatie van een patient, met inbegrip van de identiteit van de geneesmiddelen, hun dosering, indicatie, relevante gebruiksaanwijzingen en bijkomende informatie waar nodig. Het omvat zowel voorgeschreven als

niet-voorgeschreven geneesmiddelen en voedingssupplementen.

Deze oplijsting van de actieve medicatie van de patient is niet enkel een essentieel hulpmiddel voor de patient bij de correct inname van medicatie maar ook voor medische professionelen om bv. over- of onderdosering, dubbelmedicatie, en andere geneesmiddelgebonden problemen te voorkomen. Ook wordt het gebruikt bij de communicatie tussen zorgverstrekkers. Het medicatieschema wordt eveneens door verpleegsters geraadpleegd voor het klaarzetten van de medicatie.



Apotheek Maudens — Tel:
Brusselsesteenweg 713, 9050 GENTBRUGGE — GSM:
Titularis: Elisabeth Maudens — Fax :

**Medicatieschema**

Naam: — Geslacht — Datum:
INSZ Nr. (of Nr. RR of Nr. ID): — Geboortedatum: — Arts :
Pathologieën : Hypertensie — Allergieën / Intoleranties :

| Dagelijkse medicatie - gecodeerd | Eenheid | Ontbijt Voor | Ontbijt Met | Ontbijt Na | 10u | Middagmaal Voor | Middagmaal Met | Middagmaal Na | 16u | Avondmaal Voor | Avondmaal Met | Avondmaal Na | 20u |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RHUMAL COMPLET SACHET 90 | | | 1 | | | | | | | | | | |
| XARELTO 15 MG COMP PELL 98 X 15 MG | | | 1 | | | | | | | | | | |
| SPIRONOLACTONE EG COMP 50X 25MG | | | 1 | | | | | | | | | | |
| LOSARTAN EG COMP PELL 98 X 100 MG | | | 1/2 | | | | | | | | | | |
| CARVEDILOL EG 25,00 MG COMP 98 X 25 MG | | | 1 | | | | | | | | 1/2 | | |
| LYRICA CAPS HARDE - DUR 200 X 150 MG | Capsule | | | | | | | | | | | | |

| Niet dagelijkse medicatie of niet gestructureerde | Eenheid | Posologie |
|---|---|---|
| ALENDRONATE EG 70 MG COMP 12 X 70 MG | | Chronische medicatie: wekelijks |
| METATOP 2 MG COMP 30 X 2 MG | | Chronische medicatie: 1 tablet bij het slapengaan dagelijks |
| DAFALGAN CODEINE EFF 500MG TABL 32 | | Indien nodig: |

Figuur 1.2: Voorbeeld medicatieschema. Bron: Tsjoen (2020)

Zoals men in figuur 1.2 kan zien, wordt dit schema grafisch in tabulaire vorm gepresenteerd. Echter is de lay-out hiervan niet gestandaardiseerd; afhankelijk van de apotheker of andere zorgverstrekker worden andere kolomnamen, kolomverdeling, rand- en verdelingstijl, celgrootte en andere tabelelementen aangewend. Dit bemoeilijkt ernstig het ontwikkelen van een transformatiesysteem die ingescande medicatieschema's omzet in instanties van een uniform digitale datastructuur in bv. XML- of JSON-formaat voor digitale verwerking van de medicatiedata in gezondheidszorgplatformen.

Een open source tabeltransformatiesoftware zal automatisch medicatieschema's kunnen omzetten in een uniform digitale datastructuur. Hierdoor zal er geen manuele werk uitgevoerd moeten worden, wat tijd- en kostenreductie als positieve gevolgd heeft. Daarbovenop, omdat het open source zal zijn, zal men verzekerd zijn dat Into.care niet zal te maken hebben met softwarelicentiekosten of privacyschending.

Hoewel het digitaliseren van medicatieschema's een belangrijke toepassing is, zijn er tal van andere potentiële toepassingen, aangezien tabellen zo vaak gebruikt worden. Zo zou

men tabeltransformatie eveneens kunnen gebruiken voor het inscannen van kassatickets, het analyseren van een sudokuspel, het digitaal weergeven van een - op een whiteboard gemarkeerde - matrix voor online leerplatformen, het verwerken van een foto van een voedingswaardetabel op de verpakking van voedsel, en meer. Het is duidelijk dat een open source tabeltransformatiesoftware een beduidende universeel meerwaarde zal aanbieden.

## 1.2 Onderzoeksvraag

Men kan zich bij tabeltransformatie, en dus bij dit onderzoek, enkele vragen stellen.

- Uit welke processen bestaat tabeltransformatie? In welke volgorde deze plaats?
- Hoe kan men de performantie van tabeltransformatiesoftware best evalueren?
- Is preprocessing van de afbeelding nodig om de nauwkeurigheid van de resultaten te bewaren? Indien ja, uit welke stappen bestaat deze preprocessing?
- Analoog, is postprocessing van de verkregen tabel noodzakelijk? Indien ja, uit welke stappen bestaat deze postprocessing?
- Op welke manieren kan men de resultaten verbeteren, indien men in bezit is van domeinkennis? Zo zou men bijvoorbeeld kennis van de gezondheidszorg kunnnen gebruiken om medicatieschema's nauwkeuriger te digtaliseren.

## 1.3 Onderzoeksdoelstelling

Aangezien het doel van deze studie het creëren van een end-to-end tabeltransformatie-tool is, zal er niet alleen gestreefd worden subprocessen zoals OCR of preprocessing geïsoleerd te bestuderen maar evenwel de subprocessen te implementeren in code. Eveneens is het de bedoeling dat de componenten met elkaar op een geïntegreerde manier zullen kunnen functioneren.

Dit betekent dat de prototype niet enkel zal bestaan uit tabelanalysesoftware, maar alsook uit een GUI, een backend server, een preprocessing pipeline, en meer.

## 1.4 Opzet van deze bachelorproef

De rest van deze bachelorproef is als volgt opgebouwd:

In Hoofdstuk 2 wordt een overzicht gegeven van de stand van zaken binnen het onderzoeksdomein, op basis van een literatuurstudie.

Verder wordt in Hoofdstuk 3 de methodologie toegelicht en worden de gebruikte onderzoekstechnieken besproken om een antwoord te kunnen formuleren op de onderzoeksvragen.

In Hoofdstuk 4 wordt vervolgens de architectuur van de proof of concept uitgelegd. Eveneens worden de verschillende algoritmen in detail besproken.

Verder worden in Hoofdstuk 5 de met de proof of concept verkregen resultaten besproken en vergeleken.

In Hoofdstuk 6 worden enkele optimalisatiemogelijkheden om de nauwkeurigheid van het systeem te verhoden, besproken.

En tenslotten in Hoofdstuk 7, wordt de conclusie gegeven en een antwoord geformuleerd op de onderzoeksvragen. Daarbij wordt ook een aanzet gegeven voor toekomstig onderzoek binnen dit domein.

# 2. Stand van zaken

In dit hoofdstuk wordt de stand van zaken besproken wat tabeltransformatie van afbeeldingen betreft. Er wordt besproken wat tabulair data is, waarom tabellen belangrijk zijn in de huidige informatiewereld, wat er bedoeld wordt met tabeldetectie en structuuranalyse, waar de uitdagingen hierbij zich bevinden en tenslotte wordt er in detail de verschillende technieken besproken die ontwikkeld werden om tabellen te kunnen detecteren en analyseren, met hun voor- en nadelen.

## 2.1 Tabulair data

### 2.1.1 Definitie

Zoals Zanibbi e.a. (2003) het aangeeft, is een tabel een vorm van visualiatie dat men gebruikt om ermee data op te zoeken en te vergelijken. Meer specifiek geeft, volgens Zanibbi e.a. (2003), een tabel indexeringschema's weer voor relaties. Een relatie heeft een verzameling van $\eta$ tupels, die de domeinen of dimensies van de relatie genoemd worden.

De dimensies kunnen d.m.v. verschillende combinaties van rijen en kolommen opgesteld worden, waardoor verschillende tabelopstellingen exact dezelfde informatie op verschillenden manieren kunnne weergeven. Dit kan gedemonstreerd worden a.d.h.v. de volgende twee figuren.

| Year | Term | Mark | | | | | |
|---|---|---|---|---|---|---|---|
| | | Assignments | | | Examinations | | Grade |
| | | Ass1 | Ass2 | Ass3 | Midterm | Final | |
| 1991 | Winter | 85 | 80 | 75 | 60 | 75 | 75 |
| | Spring | 80 | 65 | 75 | 60 | 70 | 70 |
| | Fall | 80 | 85 | 75 | 55 | 80 | 75 |
| 1992 | Winter | 85 | 80 | 70 | 70 | 75 | 75 |
| | Spring | 80 | 80 | 70 | 70 | 75 | 75 |
| | Fall | 75 | 70 | 65 | 60 | 80 | 70 |

Figuur 2.1: Een tabel van evaluaties. Het geeft dezelfde informatie weer als tabelfiguur 2.2. Bron: Long, 2010

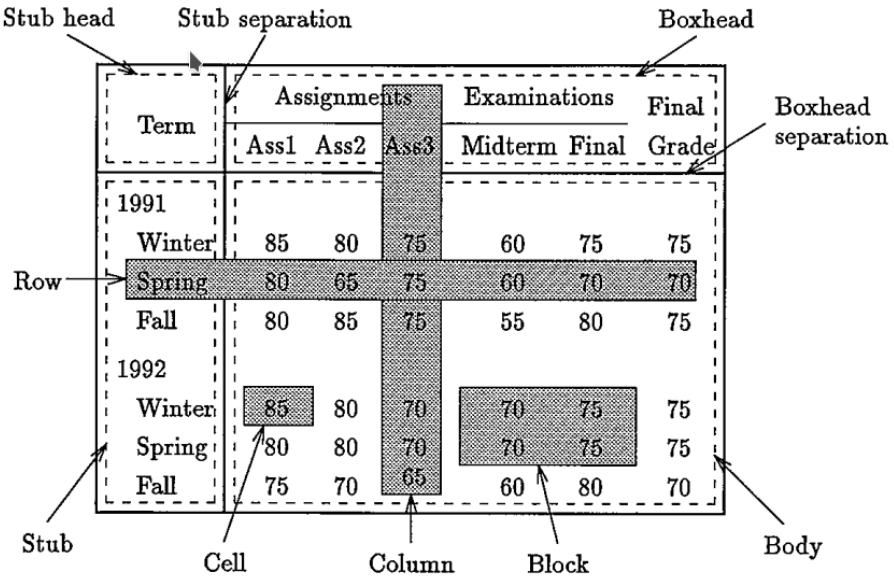
| Term | Assignments | 1991 | 1992 |
|---|---|---|---|
| Winter | Ass1 | 85 | 85 |
| | Ass2 | 80 | 80 |
| | Ass3 | 75 | 70 |
| | Midterm | 60 | 70 |
| | Final | 75 | 75 |
| | Grade | 75 | 75 |
| Spring | Ass1 | 80 | 80 |
| | Ass2 | 65 | 80 |
| | Ass3 | 75 | 70 |
| | Midterm | 60 | 70 |
| | Final | 70 | 75 |
| | Grade | 70 | 75 |
| Fall | Ass1 | 80 | 75 |
| | Ass2 | 85 | 70 |
| | Ass3 | 75 | 65 |
| | Midterm | 55 | 60 |
| | Final | 80 | 80 |
| | Grade | 75 | 70 |

Figuur 2.2: Een tabel van evaluaties. Het geeft dezelfde informatie weer als tabelfiguur 2.1. Bron: Long, 2010

Hoewel beide tabellen identiek zijn wat informatieinhoud betreft, kan duidelijk gemerkt worden dat tabelfiguur 2.1 de evaluaties duidelijker weergeeft. Meestal wordt een combinatie van rijen en kolommen zodanig gekozen zodat de data van de tabel zo eenvoudig en snel mogelijk gelezen en geïnterpreteerd kan worden. Ook kunnen verschillende lettertypes, kleuren en lettergroottes gebruikt worden om de leesbaarheid te vergroten.

### 2.1.2 Anatomie

Volgens Wang (1996) is een tabel, door *stub scheiding* en *boxhead scheiding*, verdeeld in vier hoofdregio's die in onderstaande figuur 2.3 merkbaar zijn. De regio linksbeneden die de rijhoofdingen bevat en de regio rechtsboven die de kolomhoofdingen bevat, worden respectievelijk de *stub* en de *boxhead* genoemd. De regio linksboven, die de categorieën in de *stub* inhouden is gekend als de *stub head* en de *body*, tenslotte, is de regio rechts van de *sub* en onder de *boxhead* die de tabeldata-elementen bevat. De snijpunt van een rij en een kolom wordt een *cel* genoemd; en een rechthoekig verzameling van *cellen* is gekend als een *blok*.




Figuur 2.3: De anatomie van de structurele rij-kolomvoorstelling van een tabel. Bron: Wang, 1996

Zoals men in figuur 2.3 kan zien, kunnen multidimensionele relaties in een twee dimensionele tabel gepresenteerd worden door meer dan één categorie te associeren met de *boxhead* en/of met de *stub*. Zo worden hier de rijhoofdingen niet enkel met één hoofdcategorie "Term" maar eveneens met meerdere subcategorieën, "1991" en "1992" geassocieerd. Analoog zijn de kolomhoofdingen gekoppeld aan drie categorieën, namelijk "Assignments", "Examinations" en "Finals".

### 2.1.3 Functie

Als vermeld door Shahzad e.a. (2019), worden tabellen veelal gebruikt voor het gestructureerd vertonen van essentiële informatie in documenten. Ze worden gebruikt in boeken, artikelen, onderzoekspapers, en verschillende andere soorten media. In sectoren zoals de financiële en de administratieve sectoren wordt data veelal in tabelvorm geformuleerd omdat tabellen, volgens Coüasnon en Lemaitre (2014), veel informatie voorstellen op

een beknopte manier waardoor het begrijpbaar blijft voor de lezer; ze laten ook zo toe de belangrijke delen te benadrukken.

### 2.1.4 Creatie en representatie

Doorheen de tijd werden verschillende software applicaties ontwikkeld om digitaal tabulair data aan te maken, te beheren en voor te stellen. Een veelgebruikte software voor tabelcompositie is Microsoft Excel. Het is, zoals Wang (1996) het vermeldt, een complexe rekenbladprogramma waarbij tabulair data in een werkblad, in een twee dimensionele rooster die a.d.h.v. rij en kolomindexes geadresseerd kan worden, geplaatst wordt.

Een andere bekend software voor het creëren van tabellen is LaTeX. Het is een systeem voor het zetten van documenten. Wang (1996) geeft aan dat tabellen in LaTeX gespecifieerd kunnen worden met de “tabular“- en de “array“-omgeving. De eerste omgeving wordt meestal gebruikt voor tekstuele tabeldata, de tweede voor wiskundige uitdrukkingen.

Voor de voorstelling van tabellen op het internet, m.a.w. op internetbrowsers, wordt de opmaaktaal HTML gebruikt. Door middel van de “table“-, “tr“-, “th“- en “td“-tags kunnen tabellen gemaakt en voorgesteld worden.

## 2.2 Tabeltransformatie

Verschillende technieken voor tabeltransformatie werden reeds ontwikkeld. Echter blijft een algemeen toepasbare oplossing een moeilijk uitdaging, en dit voor diverse redenen.

- Tabellen bezitten uiteenlopende layouts en designs, zonder enige standaardisatie (Kasar e.a., 2014)
- Verschillende tabellayouts hebben verschillende features (Kasar e.a., 2014)
- De typisch kleine inter-klasse variantie tussen tabellen, figuren en grafieken vermoeilijkt de detectie van tabellen; de kleine variantie is verantwoordelijk voor de hoge hoeveelheid valse positieven bij tabeldetectie (Embley e.a., 2006)

Kasar e.a. (2014) beschreeft tabeltransformatie als een proces bestaand uit voornamelijk twee subprocessen: tabeldetectie en tabelstructuuranalyse.

Met tabeldetectie worden eerst regio’s in een bepaalde document geïdentificeerd die overeenkomen met tabellen. Vervolgens wordt tabelstructuuranalyse toegepast om relationele informatie te extraheren van de geïdentificeerde tabelregio’s om de logische structuur van de tabellen te achterhalen, zoals bijvoorbeeld de rijhoofdingen, kolommhoofdingen, cellen en meer.

### 2.2.1 Tabeldetectie

Tabeldetectietechnieken kan men, kijkend naar de stand van zaken, opdelen in twee klassen: klassieke, op regelgebaseerde algoritmen enerzijds en de recentere, datagedreven algoritmen die gebruik maken van machinaal leertechnieken.

#### Regelgebaseerde technieken

Watanabe e.a. (1991) waren de auteurs van één van de vroegste werken om tabellen te identificeren. De basis voor de tabelidentificatie hier is de identificatie van individuele blokken, ingesloten door horizontale en verticale lijnsegmenten. Eerst worden lijnsegmenten gedetecteerd en hiermee wordt vervolgens de positie van hoekpunten, gevormd door deze lijnen, bepaald. Hierna wordt a.d.h.v. de positie van deze hoekpunten individuele blokken geïdentificeerd. De relatie tussen de verschillende blokken wordt uiteindelijk in globale en individuele boomstructuren gebruikt om te beslissen of het over een tabel gaat of niet.

Het jaar daarop stelden Laurentini en Viada (1992) een methode voor waarbij tekstregio's op een bottom-up manier gedetecteerd worden. De gedetecteerde karakters worden samengebracht tot woorden en deze woorden worden op hun beurt aan elkaar samengevoegd tot tekstblokken. Ook worden de scheidingslijnen gedeteceerd. Voor elke tekstblok wordt diens positie vergeleken met de scheidingslijnen, om te bepalen of het tot een bepaalde tabel behoort.

TINTIN werd door Pyreddy en Croft (1997) voorgesteld, om tabellen te detecteren. Hun algoritme steunt voor de analyse op de extra PDF-metadata van de PDF-documenten.

Enkele jaren later werd het systeem T-Recs, door Kieninger en Dengel (2001), voorgesteld. Het systeem vormt rechthoeken (bounding boxes) voor woorden in het tabel en op een bottom-up manier worden deze bounding boxes gegroepeerd volgens hun logische eenheden.

#### Datagedreven technieken

Datagedreven technieken vereisen veel data om nauwkeurig te kunnen werken. Indien het om gesuperviseerde machinale leertechnieken gaat, is er daarbij nog ground truth labeling van de tabeldatasets nodig, wat een tijdrovend proces is. Wangt e.a. (2001) ontwikkelden een galabelde dataset generator die op basis van één gelabelde tabel, met bijhorende metadata, automatisch, mits kleinde aanpassingen, datasets van gelabelde tabelafbeeldingen genereert en dus de ground truth labeling proces automatiseert. Deze tool zou zeer handig geweest zijn voor modeltraining van datagedreven algoritmen en voor algoritme-evaluaties. Wangt e.a. werden voor gecontaceerd om deze tool beschikbaar te maken maar helaas is de software niet meer ter beschikking.

Objectdetectie bij afbeeldingen d.m.v. machinale leertechnieken is sinds enkele tiental jaren een populair onderzoeksonderwerp geworden. De traditionele pipeline voor

objectdetectie bestaat uit een feature extraheerder (feature extractor), gevolgd door een classificatiesysteem.

Cesarini e.a. (2002) waren één van de eersten die geprobeerd hebben machinale leertechnieken te gebruiken voor tabeldetectie. Hierbij wordt, door de besproken methode Tabfinder, de document omgezet in een Modified-XY-boomvoorstelling en wordt er gezocht naar blokken omgeven door horizontale of verticale lijnen; dit gebeurt recursief. Indien deze verticale of horizontale lijnen gevonden zijn, dan bevat het document mogelijks een tabel. Om de veronderstelling te verifiëren, wordt er in diepere niveaus van de boom gezocht naar lijnen die loodrecht staan op de reeds gedetecteerde lijnen. Indien deze lijnen daarbovenop gevonden zijn, dan kan er met zekerheid vastgesteld worden dat het om een tabel in het document gaat. Na de boomanalyse worden vervolgens subtabellen behorend tot dezeflde tabel samengevoegd.

In een paper introduceerden Mandal e.a. (2006) een simpele maar efficiënte algoritme om tabellen te identificeren. De algoritme steunt namelijk op de observatie dat de hoeveelheid witruimte tussen elementen van verschillende kolommen significant groter is dan de witruimte tussen woorden in paragrafen.

Silva (2009) heeft eveneens een data-gedreven model voorgesteld, gebruikmakend van Hidden Markov Modellen en PDF-documenten. De tekst van de PDF-documenten wordt eerst omgezet in ASCII-karakters, en hierna verwerken de Hidden Markov Modellen de waarschijnlijkheidsdistributies van de samenhang van de verschillende opeenvolgende ASCII-karakters. De modellen houden voor elke horizontale lijn ook bij of het deel uitmaakt van een tabel, of niet.

Een SVM-classificatiesysteem, steunend op enkele manueel geselecteerde dimensiefeatures van de horizontale en verticale scheidingslijnen, werd door Kasar e.a. (2013) gepresenteerd. Om de donkere, dunne en lijnachtige structuren, die als scheidingslijnen van een tabel beschouwd worden, goed te kunnen detecteren, wordt de inputafbeelding eerst verzacht met een Gaussiaanse filter. Vervolgens worden op de input enkele top-hat-transformaties toegepast. Uiteindelijk wordt voor elke groep van kruising van horizontale en verticale lijnen een SVM-classifier met 26 lijnfeatures gebruikt om te bepalen of de regio deel uitmaakt van een tabel.

Fan en Kim (2015) gebruikten zowel een niet-gesuperviseerde leermodel als tekstuele informatie van een bepaalde zone voor de detectie van tabellen. De gebruikte niet-gesuperviseerde leermodel bestaat uit een ensemble van generatieve en distriminatieve modellen.

In hetzelfde jaar presenteerden Tran e.a. (2015) een methode die zich baseert op de ruimtelijke ordening van uitgehaalde tekstblokken en op ROI's. In tegenstelling tot verschillende andere traditionele algoritmen, is hun voorstel direct bruikbaar op afbeeldingen van gescande documenten.

Eén van de eerste pogingen om deep learning toe te passen, werd gerealiseerd door Hao e.a. (2016). Kandidaattabellen worden geselecteerd op basis van scheidingslijnfeatures. Hierna worden deze kandidaattabellen verwerkt door een CNN. Uiteindelijk vindt de klassificatie

“tabel“ of “geen tabel“ plaats.

Rashid e.a. (2017) gebruikten een bottom-up algoritme waarbij voor elk woord een feature-vector werd aangemaakt. Elke feature-vector bevat features zoals de dimensies van het woord, de afstand tot de andere woorden in de nabijheid, de hoeveelheid witruimte, etc. Deze feature-vectors werden uiteindelijk gebruikt om een AutoMLP-klassificatiessysteem te trainen.

Een andere deep learning techniek, gebaseerd op de Faster R-CNN-architectuur werd voorgesteld door Gilani e.a. (2017). Normaliter worden pixelwaarden als input gebruikt, voor een convolutionele neurale netwerk. Bij dit onderzoek is dat niet het geval. In plaats van pixelwaarden wordt de witruimte tussen tekstblokken als input verwerkt, aangezien bij tabellen witruimte tussen tekstblokken een bepaalde patroon bevat die door het model gedetecteerd wordt.

Faster R-CNN is een poopulaire modelkeuze geworden door de hoge nauwkeurigheid dat het aanbiedt bij objectdetectie in verschillende domeinen. Het bestaat, zoals meegedeeld door Shahzad e.a. (2019), uit twee processen die sequentieel uitgevoerd worden. Het eerste proces is Region Proposal Network (RPN) dat kandidaaatregio’s voor de detectienetwerk identificeert. Het tweede proces is de klassificatiesysteem die a.d.h.v. de detectienetwerk voor elke kandidaatregio beslit of het als een tabel beschouwd kan worden of niet. De RPN-module van Faster R-CNN biedt hogere performantie dan de selectieve zoekproces (selective search) van Fast R-CNN, de voorganger van Faster R-CNN.

Siddiqui e.a. (2018) introduceerden een incrementele architectuurverbetering, door Faster R-CNN te combineren met een vervormbare (deformable) CNN. De RestNet-101-model werd gebruikt voor transfer learning, aangezien er niet voldoende gelabelde data beschikbaar was. De reden waarom gebruik gemaakt werd van een vervormbare CNN i.p.v. een klassieke CNN is dat de klassieke CNN een vaste receptieve veld heeft (fixed receptive field), wat niet gewenst is door de verschillende dimensies en transformaties zoals verdraaing, vergroting, verschuiving en meer dat tabellen kunnen hebben. Een vervormbare CNN kan zijn receptieve veld aanpassen aan de inputdata, hierdoor is het te gebruiken op elk soort tabel, ongeacht de layout ervan.

### 2.2.2 Tabelstructuuranalyse

Om de structuur van een tabel te kunnen analyseren, gebruikten Nazemi e.a. (2016) celsegmentatie. De horizontale en verticale scheidingslijnen van de tabel worden geïdentificeerd, en op basis van deze lijnen en hun kruisingen worden de tabelcellen gevormd. Voor tabellen zonder scheidingslijnen worden rijen gevormd door tekstblokken die dezelfde Y-waarde bezitten, en kolommen worden gevormd door tekstblokken waartussen weinig witruimte zit, m.a.w. door tekstblokken die relatief dicht bij elkaar staan.

Qasim e.a. (2019) meenden dat CNN, die normaliter een populaire keuze is, niet de ideale neurale netwerk architectuur is voor structuuranalyse van tabellen, op een datagedreven manier. Ze stelden een architectuur voor die gebaseerd die is op GNN’s. Hoewel een

tabelafbeelding verwerkt kan worden door een CNN, kan een tabel zelf beter verwerkt worden door een GNN aangezien de cellen van de tabel en de relatie tussen cellen eenvoudig gepresenteerd en gemodelleerd kunnen worden door een graaf. Een CNN wordt gebruikt voor visuele feature extractie, gevolgd door drie GNN's: één om te bepalen of twee tekstblokken tot dezelfde cel behoren, een tweede om na te gaan of twee tekstblokken deel uitmaken van dezelfde rij en tenslotte een derde GNN om te bepalen of twee tekstblokken tot dezelfde kolom behoren.

### 2.2.3 End-to-end-systemen

Tot hier werden onderzoeken besproken die ofwel het probleem van tabeldetectie ofwel het probleem van tabelstructuuranalyse aanpakten. Echter werden recent nieuwe technieken geïntroduceerd die voor beide subprocessen een oplossing aanbieden.

Zo hebben Green en Krishnamoorthy (1996) aan een layoutmodelgebaseerde top-down techniek gewerkt. Eerst worden horizontale lijnen, verticale lijnen en witruimte gebruikt om de tabelregio te detecteren. Vervolgens worden individuele cellen binnen de uitgehaalde tabel gelabeld, d.m.v. de layoutmodel; zo wordt relationele informatie van de cellen verkregen.

Even later hebben Oro en Ruffolo (2009) hun oplossing, PDF-TREX genoemd, ontwikkeld. Het is een systeem die op PDF-metadata steunt.

Een state-of-the-art deep learning algoritme, gebaseerd op afbeeldingen van ingescande documenten werd door Schreiber e.a. (2017) voorgesteld. Hierbij werd gebruik gemaakt van een Faster R-CNN-model, pretrained op de gekende Pascal-VOC-dataset, voor de detectie. De ZFNet (Zeiler & Fergus, 2014) en VGG-16 (Simonyan & Zisserman, 2014) modellen werden hierbij gebruikt voor transfer learning. Voor de tabelstructuuranalyse werd gebruik gemaakt van semantische segmentatie, m.b.v. een FCN-X-architectuur (Shelhamer e.a., 2016).

Tenslotte introduceerden Prasad e.a. (2020) recent een end-to-end-systeem voor tabeltransformatie. Hierbij werd gebruik gemaakt van één CNN model, namelijk een R-CNN-HRNet-model. Dit is een deep learning model specifiek ontworpen voor instantiesegmentatie, en niet voor de meer populaire toepassing, objectdetectie. Met dit model wordt instantiesegmentatie, en meer specifiek, tabelsegmentatie uitgevoerd op elke afbeelding waarin mogelijks tabellen zich bevinden om elke tabelinstantie te identificeren. Niet enkel wordt met dit model elke tabelinstantie geïdentificeerd, maar eveneens wordt elke tabelinstantie geclassificeerd als *bordered* of *borderless*. Een *bordered* tabel wordt gedefinieerd als een tabel waarin elke cel omgeven is door een rand. De cellen van een *borderless* tabel bezitten juist geen rand. Ook individuele tabelcellen worden gesegmenteerd. Nadat tabel- en celidentificatie uitgevoerd is, vindt de structuuranalyse plaats. Indien het om een *borderless* tabel gaat, wordt voor elke geïdentificeerde tabelcel een rij- en kolompositie toegekend. Indien de tabel van de type *bordered* is, worden de horizontale en verticale scheidingslijnen van de tabel gebruikt om de tekstblokken in tabelcellen te segmenteren; de geïdentificeerde tabelcellen worden hierbij genegeerd.

# 3. Methodologie

In dit hoofdstuk worden eerst het doel en de systeemvereisten van de proof-of-concept besproken. Vervolgens wordt de selectie van technologieën voor de proof-of-concept behandeld, inclusief de keuze van de tabelstransformatie-algoritme(n). Uiteindelijk wordt de evaluatiessyteem verduidelijkt die gebruikt wordt om de performantie van de proof-of-concept te beoordelen.

## 3.1 Systeemvereisten

### 3.1.1 Goals

Om een proof-of-concept succesvol te kunnen realiseren, moeten de volgende einddoelen hiervoor bereikt worden:

- Een end-to-end-systeem dient gecreëerd te worden. Dit betekent dat de proof-of-concept niet enkel een tabeldetectiecomponent moet bevatten, maar eveneens een tabelstructuuranalysecomponent, een GUI en andere nodige elementen om de tabeltransformatie zoveel mogelijk te automatiseren.

- De software moet modulair geïmplementeerd worden. Code geschreven voor tabeldetectie, bijvoorbeeld, mag niet afhankelijk zijn van code geschreven voor tabelstructuuranalyse en vice versa. Dit maakt het mogelijk om later één deel van de tabeltransformatiesoftware te verbeteren of opnieuw te herschrijven, zonder hierdoor een impact te hebben op de rest van de software.

- Enkel open source software en libraries, zonder commercële restricties, mogen gebruikt worden.

- De proof-of-concept moet kunnen functioneren op verschillende eenvoudige tabellayouts. D.w.z. dat de tabeltransformatie niet mag afhangen van de fysieke dimensies van de tabel zelf of van de cellen. Noch mag het afhankelijk zijn van het aantal rijen of kolommen. Verder moet de software contextonafhankelijk zijn; een voedingswaardetabel bijvoorbeeld moet even nauwkeurig getransformeerd kunnen worden als een medicatieschema.

- Een tabeltransformatie moet kunnen uitgevoerd worden door een REST-server op te roepen. De tabeltransformatieresultaat die de server bij deze terug zal geven, moet een intuïtief, goed opgesteld JSON-object zijn.

### 3.1.2 Non-goals

Naast einddoelen die de scope van de proof-of-concept vormen zijn er eveneens non-goals, vereisten die expliciet buiten de scope liggen:

- Het is niet de bedoeling om een volledige softwarepakket te ontwikkelen, met error handling, unit tests, integratie tests, authenticatie en meer.

- De proof-of-concept moet niet complexe tabellen kunnen verwerken. Hiermee worden voornamelijk tabellen bedoelt met meerdere niveau's van rijen en kolommen of tabellen met subtabellen.

- Preprocessing van de inputafbeeldingen wordt niet uitgevoerd. Er wordt verwacht dat de documenten correct zijn ingescand.

- Tabellen met handgeschreven tekst worden niet in beschouwing genomen.

## 3.2 Selectie technologieën

Op basis van de literatuurstudie (hoofdstuk 2) en de systeemvereisten (paragraaf 3.1) kan de selectie van algoritmes en technologieën plaatsvinden.

### 3.2.1 Tabeldetectie en Tabelstructuuranalyse

De verschillende algoritmes voor tabeldetectie en tabelstructuuranalyse, behandeld in de literatuurstudie (hoofdstuk 2), hebben elk hun voor- en nadelen. Deze voor- en nadelen worden in de volgende tabel 3.1 weergegeven.

| | Werkt zonder lijnen | Bruikbaar op afbeelding | Diverse layouts |
|---|---|---|---|
| *Tabeldetectie* | | | |
| Watanabe e.a. | | ✓ | ✓ |
| Laurentini en Viada | | ✓ | ✓ |
| Pyreddy en Croft | ✓ | | ✓ |
| Kieninger en Dengel | ✓ | ✓ | |
| Cesarini e.a. | | ✓ | ✓ |
| Mandal e.a. | ✓ | ✓ | |
| Silva | | | ✓ |
| Kasar e.a. | | ✓ | ✓ |
| Fan en Kim | ✓ | ✓ | ✓ |
| Tran e.a. | ✓ | ✓ | ✓ |
| Hao e.a. | | | |
| Rashid e.a. | ✓ | ✓ | |
| Gilani e.a. | ✓ | ✓ | ✓ |
| Siddiqui e.a. | ✓ | ✓ | ✓ |
| *Tabelstructuuranalyse* | | | |
| Nazemi e.a. | | ✓ | ✓ |
| Qasim e.a. | ✓ | ✓ | ✓ |
| *End-to-end-systemen* | | | |
| Green en Krishnamoorthy | | ✓ | |
| Oro en Ruffolo | ✓ | | ✓ |
| Schreiber e.a. | ✓ | ✓ | ✓ |
| Prasad e.a. | ✓ | ✓ | ✓ |

Tabel 3.1: Voor- en nadelen van tabeltransformatiealgoritmes

Wat tabeldetectie betreft, bieden de algoritmes (Tran e.a., 2015), (Gilani e.a., 2017) en (Siddiqui e.a., 2018) alle voordelen aan. Alle drie technieken werken zonder de aanwezigheid van horizontale of verticale tabelscheidingslijnen. Ze kunnen direct gebruikt worden op afbeeldingen, en zijn dus niet afhankelijk van PDF-metadata bijvoorbeeld. Tenslotte kunnen ze tabellen van verschillende layouts detecteren.

Van de technieken voor tabelstructuuranalyse heeft (Qasim e.a., 2019) alle voordelen.

Tenslotte bieden de end-to-end-technieken, (Schreiber e.a., 2017) en (Prasad e.a., 2020), eveneens alle voordelen.

Om het aantal kandidaatalgoritmes voor de proot-of-concept verder te verminderen, worden hun performanties met elkaar vergeleken. Omdat voor de tabelstructuuranalyse geen algoritmes te vinden zijn wiens performantiemeting met dezeflde methodologie uitgevoerd is, worden deze performanties niet in rekening gehouden. In onderstaande tabel 3.2 kan men de performantie van de verschillende technieken voor tabeldetectie terugvinden.

| | Recall | Precisie | F1-score |
|---|---|---|---|
| Tran e.a. | 0,9636 | 0,9521 | 0,9578 |
| Gilani e.a. | 0,9067 | 0,8230 | 0,8629 |
| Siddiqui e.a. | 0,996 | 0,996 | 0,996 |
| Schreiber e.a. | 0,9615 | 0,9740 | 0,9677 |
| Prasad e.a. | 1,0 | 1,0 | 1,0 |

Tabel 3.2: Tabeldetectieperformanties van de verschillende algoritmes

Deze performantiemetingen werden uitgevoerd op de ICDAR 2013 Table Competition dataset. Dit is één van de meest bekende datasets voor tabeldetectie en tabelstructuuranalyse. Het bevat in totaal 238 ingescande documenten. Voor de berekening van de F1-score wordt bij deze dataset een IoU threshold van 0,5 gebruikt.

Aangezien de tabeldetectiealgoritme van Prasad e.a. (2020) de meest performante is, wordt deze geselectioneerd als de tabeldetectiecomponent van de proof-of-concept. Voor de tabelstructuuranalyse kan men nog kiezen tussen (Qasim e.a., 2019), (Schreiber e.a., 2017) en opnieuw (Prasad e.a., 2020). Hoewel (Qasim e.a., 2019) en (Schreiber e.a., 2017) gepaste keuzes zijn, wordt er beslist om voor de structuuranalyse eveneens (Prasad e.a., 2020) te selectioneren. Dit komt voornamelijk omdat Prasad e.a. (2020) niet enkel hun volledige modeltrainingdataset maar ook de code-implementatie van hun algoritme openbaar hebben gemaakt, met een open source licentie.

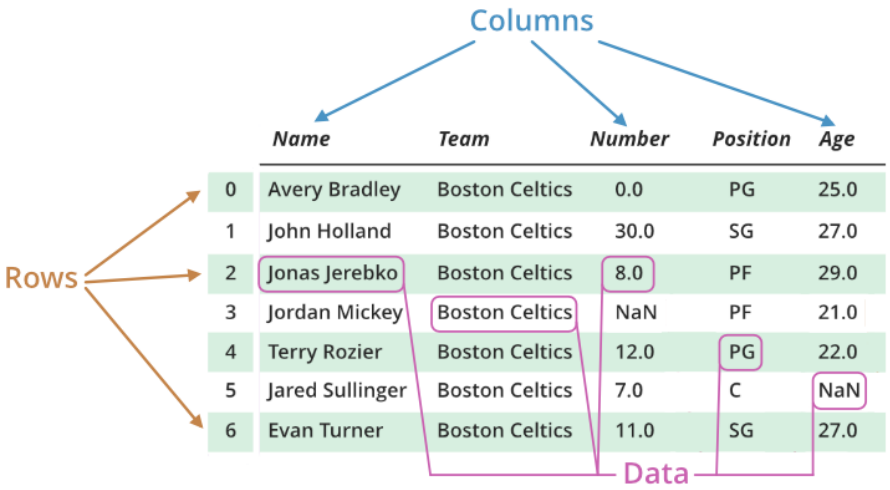
### 3.2.2 Programmeertaal

Voor de programmeertaal van de proof-of-concept wordt Python (Van Rossum & Drake, 2009) gekozen, en dit voor verschillende redenen:

- Python is een multifunctioneel programmeertaal die toelaat om eenvoudig en snel softwareprototypes te creëren.
- Het heeft een grote bibliotheek van libaries voor onder meer statistiek en data-analyse, zoals Numpy (Oliphant, 2006) en Pandas (McKinney, 2010).
- De open source code van Prasad e.a. (2020) is reeds in Python geschreven. Door met Python verder te werken, kan code herbruikt worden en zal een een volledig nieuwe reïmplementatie van de software dus niet nodig worden.

### 3.2.3 Interne tabelmodel

Om de getransformeerd tabel te kunnen verwerken en verbeteren, moet de software van de proof-of-concept deze d.m.v. een datastructuur bijhouden. Hoewel voor de datastructuur JSON of XML gekozen kan worden, wordt er beslist om intern de getransformeerd tabel te presenteren en te verwerken d.m.v. een Pandas Dataframe. Een Pandas Dataframe is een twee dimensionele tabulaire datastructuur van de Pandas library. Deze datastructuur bestaat voornamelijk uit drie hoofdcomponenten, die in onderstaande figuur worden weergegeven: rijen, kolommen en data.




Figuur 3.1: Anatomie van een Pandas Dataframe. Bron: GeeksforGeeks (2020)

Niet enkel modelleert een Pandas Dataframe, met zijn twee dimensionele structuur, op een gepaste manier een tabel maar het bezit eveneens andere voordelen. Zo kan de opgeslagen data heterogeen zijn, d.w.z. dat de datatypes van de data-elementen niet identiek hoeven te zijn. Verder bezit het meerdere functionaliteiten om data eenvoudig te manipuleren. Tenslotte kan een Pandas dataframe zeer efficiënt met grote hoeveelheden data werken.

### 3.2.4 OCR

Voor de OCR-component zijn er niet veel keuzes. Tesseract (Kay, 2007) is momenteel de enige OCR-software die open source is en nauwkeurig tekst in afbeeldingen detecteert en transformeert. Bovendien kan de Python library, Pytesseract (Lee, 2009), gebruikt worden om Tesseract-functionaliteiten met Python te gebruiken. Zo kan men Tesseract functionaliteiten d.m.v. Pytesseract in Python code oproepen en kunnen de OCR-resultaten als een Pandas Dataframe verkregen worden.

### 3.2.5 Back end server

Voor de REST-server wordt Flask (Grinberg, 2018) gekozen. Flask is een eenvoudige micro web framework, bedoeld voor Python-softwareontwikkeling, met REST-ondersteuning.

### 3.2.6 Front end

Er bestaan verschillende front end frameworks om snel en eenvoudig GUI's te ontwerpen die via de browser gebruikt zullen kunnen worden, zoals Angular (Jain e.a., 2014) en React (Fedosejev, 2016). Voor deze proof-of-concept wordt React gebruikt, in combinatie met de UI framework Ant Design (Financial, 2020). De UI framework Ant Design biedt herbruikbare UI-componenten, onder meer voor tabulair data, die de ontwikkeling van de proof-of-concept vereenvoudigen en versnellen.

# 4. Proof of concept

# 5. Resultaten

# 6. Optimalisatiemogelijkheden

# 7. Conclusie

# 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.

## A.1 Introductie

Het medicatieschema is een geheel van gestandaardiseerde informatie over de actieve medicatie van een patiënt, met inbegrip van de identiteit van de geneesmiddelen, hun dosering, indicatie, relevante gebruiksaanwijzingen en bijkomende informatie waar nodig. Het omvat zowel voorgeschreven als niet-voorgeschreven geneesmiddelen en voedingssupplementen (Apothekersnetwerk, 2013).

Deze oplijsting van de actieve medicatie van de patiënt is niet enkel een essentieel hulpmiddel voor de patiënt bij de correct inname van medicatie maar ook voor medische professionelen om bv. over- of onderdosering, dubbelmedicatie, en andere geneesmiddelgebonden problemen te voorkomen. Ook wordt het gebruikt bij de communicatie tussen zorgverstrekkers. Het medicatieschema wordt eveneens door verpleegsters geraadpleegd voor het klaarzetten van de medicatie.

Dit schema wordt grafisch steeds in tabulaire vorm gepresenteerd. Echter is de lay-out hiervan niet gestandaardiseerd; afhankelijk van de apotheker of andere zorgverstrekker worden andere kolomnamen, kolomverdeling, rand- en verdelingstijl, celgrootte en andere tabelelementen aangewend. Dit bemoeilijkt ernstig het ontwikkelen van een transformatiesysteem die ingescande medicatieschema's omzet in instanties van een uniform digitale

datastructuur in bv. XML- of JSON-formaat voor digitale verwerking van de medicatiedata in gezondheidszorgplatformen.

Hierdoor is er een nood aan een digitalisatiesysteem die medicatieschema's van verschillende vormen en met verschillende lay-outs nauwkeurig omzet in corresponderende instanties van een uniforme datastructuurschema. Voor deze bachelorproef wordt gebruik gemaakt van het datastructuurschema van Into Care by Pridictiv NV. De doelstelling van dit onderzoek is het bestuderen van de mogelijkheden om een dergelijk systeem tot stand te brengen en het implementeren van een proof-of-concept van een optimale oplossing. De volgende onderzoeksvragen kunnen gesteld worden bij dit onderzoek:

- Wat zijn de structuren en de relaties tussen de entiteiten in tabulaire data?
- Wat zijn de uitdagingen en complicaties bij tabelherkenning en -analyse? Kan er meer complexiteit ondervonden worden bij medicatieschematabellen?
- Hoe kan de correctheid en nauwkeurigheid van de transformatie van een tabel geëvalueerd worden?
- Welke oplossingen bestaan er reeds voor tabelherkenning en/of tabelanalyse?
- Wat is de optimale oplossing voor medicatieschema's? Hoe kan deze bepaald worden?
- Hoe kan domeinkennis gebruikt worden om de oplossing te optimaliseren?

## A.2 State-of-the-art

Verschillende oplossingen voor tabeldetectie zijn reeds beschikbaar:

- Vervormbare convolutionele neurale netwerken (Siddiqui e.a., 2018)
- Verticale en horizontale lijnendetectie (Gatos e.a., 2005)
- Naïve Bayes en documentstructuur (Li e.a., 2006)

Ook voor tabelanalyse zijn enkele oplossingen voorgesteld:

- Cellsegmentatie (Nazemi e.a., 2016)
- Fast CNN (Oliveira & Viana, 2017)
- Faster R-CNN (Schreiber e.a., 2017)
- Graafgebaseerde neurale netwerken (GNN's) (Qasim e.a., 2019)

## A.3 Methodologie

Het uitvoeren van het onderzoek zal beginnen met het ontwerpen van een scoresysteem, ook wel een benchmarksysteem genoemd, waarbij de nauwkeurigheid, precisie, performantie en andere factoren van de tabelherkenningsoplossingen in rekening gebracht zullen worden.

Hiervoor zullen reeds bestaande geannoteerde, geanonimiseerde medicatieschemadatasets gebruikt worden.

Hierna zullen de verschillende oplossingen geïmplementeerd en tevens geëvalueerd worden a.d.h.v. de benchmarksysteem. De optimale oplossing zal op deze manier bepaald worden.

Verder zullen potentiële optimalisatieopportuniteiten bestudeerd worden, zowel algemene optimalisaties als optimatlisatimogelijkheden binnen een medisch-farmaceutisch context zoals anomaliedetectie van tijdstippen van medicatieinnamen.

## A.4 Verwachte resultaten

Enerzijds bestaan er in tabellen relaties tussen kolommen en cellen, en relaties tussen cellen onderling die voorgesteld kunnen worden door grafen en anderzijds vertonen de verschillende lay-outs van tabellen een patroon die door het menselijke brein maar dus ook door diepe neurale netwerken zeer snel herkend kan worden. Er wordt daarom verwacht dat een graafgebaseerde Deep Learning-oplossing de best resultaten zal opleveren.

## A.5 Verwachte conclusies

Aangezien zowel state-of-the-art algoritmen als reeds bestaande softwareimplementatie-oplossingen beschikbaar zijn, wordt er verwacht dat een performante proof-of-concept van een digitalisatiesysteem voor medicatiesystemen successvol gecreëerd zal worden. Eveneens wordt er verwacht dat domeinkennis de nauwkeurigheid van het systeem zal verhogen.

# Bibliografie


Apothekersnetwerk, V. (2013, juli 27). *Standpunt medicatieschema*. https://vlaamsapothekersnetwerk.be/index.php/informatie/nieuws/8-berichten-van/54-van-standpunt-medicatieschema

Cesarini, F., Marinai, S., Sarti, L. & Soda, G. (2002, augustus 11). Trainable table location in document images, In *Object recognition supported by user interaction for service robots*, IEEE Comput. Soc. https://doi.org/10.1109/icpr.2002.1047838

Coüasnon, B. & Lemaitre, A. (2014). Recognition of Tables and Forms.

Embley, D. W., Hurst, M., Lopresti, D. & Nagy, G. (2006). Table-processing paradigms: a research survey. *International Journal of Document Analysis and Recognition (IJDAR)*, *8*(2-3), 66–86. https://doi.org/https://doi.org/10.1007/s10032-006-0017-x

Fan, M. & Kim, D. S. (2015). Detecting Table Region in PDF Documents Using Distant Supervision.

Federale Overheidsdienst Economie, M. e. E., K.M.O. (2019). *Barometer van de informatiemaatschappij (2019)* (onderzoeksrap.). Federale Overheidsdienst Economie, K.M.O., Middenstand en Energie.

Fedosejev, A. (2016, januari). *React.js Essentials*.

Financial, A. (2020). *Ant Design*. https://ant.design/

Gatos, B., Danatsas, D., Pratikakis, I. & Perantonis, S. (2005). Automatic Table Detection in Document Images. https://doi.org/10.1007/11551188_67

GeeksforGeeks. (2020). *Python | Pandas DataFrame*. https://www.geeksforgeeks.org/python-pandas-dataframe/

Gilani, A., Qasim, S. R., Malik, I. & Shafait, F. (2017, november 9). Table Detection Using Deep Learning, In *14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, IEEE. https://doi.org/10.1109/icdar.2017.131

Gobel, M., Hassan, T., Oro, E. & Orsi, G. (2013, oktober 15). ICDAR 2013 Table Competition, In *12th International Conference on Document Analysis and Recognition*, IEEE. https://doi.org/10.1109/icdar.2013.292

Green, E. & Krishnamoorthy, M. (1996, augustus 14). Model-based analysis of printed tables, In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, IEEE Comput. Soc. Press. https://doi.org/10.1109/icdar.1995.598979

Grinberg, M. (2018). *Flask web development: developing web applications with python*. O'Reilly Media, Inc.

Hao, L., Gao, L., Yi, X. & Tang, Z. (2016, april 11). A Table Detection Method for PDF Documents Based on Convolutional Neural Networks, In *12th IAPR Workshop on Document Analysis Systems (DAS)*, IEEE. https://doi.org/10.1109/das.2016.23

Jain, N., Bhansali, A. & Mehta, D. (2014). AngularJS: A modern MVC framework in JavaScript. *Journal of Global Research in Computer Science*, *5*(12), 17–23.

Kasar, T., Barlas, P., Adam, S., Chatelain, C. & Paquet, T. (2013, augustus 25). Learning to Detect Tables in Scanned Document Images Using Line Information, In *12th International Conference on Document Analysis and Recognition*, IEEE. https://doi.org/10.1109/icdar.2013.240

Kasar, T., Barlas, P., Sébastien, A., Chatelain, C. & Paquet, T. (2014). Learning to Detect Tables in Scanned Document Images using Line Information. *12th International Conference on Document Analysis and Recognition (ICDAR)*.

Kay, A. (2007). Tesseract: An Open-Source Optical Character Recognition Engine. *Linux J.*, *2007*(159), 2. https://doi.org/10.5555/1288165.1288167

Kieninger, T. & Dengel, A. (2001, september 13). Applying the T-Recs table recognition system to the business letter domain, In *Proceedings of Sixth International Conference on Document Analysis and Recognition*, IEEE Comput. Soc. https://doi.org/10.1109/icdar.2001.953843

Laurentini, A. & Viada, P. (1992, augustus 30). Identifying and understanding tabular material in compound documents, In *Proceedings., 11th IAPR International Conference on Pattern Recognition. Vol.II. Conference B: Pattern Recognition Methodology and Systems*, IEEE Comput. Soc. Press. https://doi.org/10.1109/icpr.1992.201803

Lee, M. (2009). *Python-tesseract*. https://pypi.org/project/pytesseract/

Li, J., Tang, J., Song, Q. & Xu, P. (2006). Table Detection from Plain Text Using Machine Learning and Document Structure (X. Zhou, J. Li, H. T. Shen, M. Kitsuregawa & Y. Zhang, Red.). In X. Zhou, J. Li, H. T. Shen, M. Kitsuregawa & Y. Zhang (Red.), *Frontiers of WWW Research and Development - APWeb 2006*, Berlin, Heidelberg, Springer Berlin Heidelberg.

Long, V. (2010, mei 23). *An Agent-Based ApproachtoTable Recognition and Interpretation* (proefschrift). Department of Computing, Macquarie University.

Mandal, S., Chowdhury, S. P., Das, A. K. & Chanda, B. (2006). A simple and effective table detection system from document images. *International Journal of Document Analysis and Recognition (IJDAR)*, *8*(2-3), 172–182. https://doi.org/10.1007/s10032-005-0006-5

McKinney, W. (2010). Data Structures for Statistical Computing in Python, In *Proceedings of the 9th Python in Science Conference*, SciPy. https://doi.org/10.25080/majora-92bf1922-00a

Microsoft. (2020). *De opmaak van een Excel-tabel wijzigen*. https://support.microsoft.com/nl-nl/office/de-opmaak-van-een-excel-tabel-wijzigen-6789619f-c889-495c-99c2-2f971c0e2370

Nazemi, A., Murray, I., Fernaando, C. & McMeekin, D. A. (2016). Converting Optically Scanned Regular or Irregular Tables to a Standardised Markup Format to be Accessible to Vision-Impaired. *World Journal of Education*, *6*(5), p9–19. Verkregen 2019, van https://eric.ed.gov/?id=EJ1158245

Oliphant, T. (2006). *A guide to NumPy* (Deel 1). Trelgol Publishing USA.

Oliveira, D. A. B. & Viana, M. P. (2017, oktober 22). Fast CNN-Based Document Layout Analysis, In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. https://doi.org/10.1109/ICCVW.2017.142

Oro, E. & Ruffolo, M. (2009, juli 26). PDF-TREX: An Approach for Recognizing and Extracting Tables from PDF Documents, In *10th International Conference on Document Analysis and Recognition*, IEEE. https://doi.org/10.1109/icdar.2009.12

Prasad, D., Gadpal, A., Kapadni, K., Visave, M. & Sultanpure, K. (2020, juni 14). CascadeTabNet: An approach for end to end table detection and structure recognition from image-based documents, In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, IEEE. https://doi.org/10.1109/cvprw50498.2020.00294

Pyreddy, P. & Croft, W. B. (1997, juli). TINTIN: A System for Retrieval in Text Tables, In *Proceedings of the second ACM international conference on Digital libraries - DL '97*, New York, NY, USA, ACM Press. https://doi.org/10.1145/263690.263816

Qasim, S. R., Mahmood, H. & Shafait, F. (2019). *Rethinking Table Recognition using Graph Neural Networks*.

Rashid, S. F., Akmal, A., Adnan, M., Aslam, A. A. & Dengel, A. (2017, november 9). Table Recognition in Heterogeneous Documents Using Machine Learning, In *14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, IEEE. https://doi.org/10.1109/icdar.2017.132

Schreiber, S., Agne, S., Wolf, I., Dengel, A. & Ahmed, S. (2017, november 9). DeepDeSRT: Deep Learning for Detection and Structure Recognition of Tables in Document Images, In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*. https://doi.org/10.1109/ICDAR.2017.192

Shahzad, M. A., Noor, R., Ahmad, S., Mian, A. & Shafait, F. (2019). Feature Engineering Meets Deep Learning: A Case Study on Table Detection in Documents, In *Digital Image Computing: Techniques and Applications (DICTA)*.

Shelhamer, E., Long, J. & Darrell, T. (2016). Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(4), 640–651. https://doi.org/10.1109/tpami.2016.2572683

Siddiqui, S. A., Malik, M. I., Agne, S., Dengel, A. & Ahmed, S. (2018). DeCNT: Deep Deformable CNN for Table Detection. *IEEE Access*, *6*, 74151–74161. https://doi.org/10.1109/access.2018.2880211

Silva, A. C. (2009, juli 26). Learning Rich Hidden Markov Models in Document Analysis: Table Location, In *10th International Conference on Document Analysis and Recognition*, IEEE. https://doi.org/10.1109/icdar.2009.185

Simonyan, K. & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv*, 1409.1556.

Tran, D., Aly, T., Oh, A., Kim, S. & Na, I. (2015). Table Detection from Document Image using Vertical Arrangement of Text Blocks. *International Journal of Contents*, *11*(4), 77–85.

Tsjoen, A. (2020). *Medicatieschema*. https://apotheektsjoen.be/service/medicatieschema/

Van Rossum, G. & Drake, F. L. (2009). *Python 3 Reference Manual*. Scotts Valley, CA, CreateSpace.

Wang, X. (1996). *Tabular Abstraction, Editing, and Formatting* (proefschrift). University of Waterloo.

Wangt, Y., Phillipst, I. T. & Haralick, R. (2001). Automatic table ground truth generation and a background-analysis-based table structure extraction method, In *Proceedings of Sixth International Conference on Document Analysis and Recognition*, IEEE Comput. Soc. https://doi.org/10.1109/icdar.2001.953845

Watanabe, T., Naruse, H., Luo, Q. & Sugie, N. (1991). Structure analysis of table-form documents on the basis of the recognition of vertical and horizontal line segments, In *Intl. Conf. Document Analysis andRecognition*.

Zanibbi, R., Blostein, D. & Cordy, J. (2003). A Survey of Table Recognition: Models, Observations, Transformations, and Inferences.

Zeiler, M. D. & Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. In *Computer Vision – ECCV 2014* (pp. 818–833). Springer International Publishing. https://doi.org/10.1007/978-3-319-10590-1_53