1-11-2023

Assignment 2

Thijs Veldkamp

500858543

Inhoud

Inhoud

[1.Samenvatting 2](#_Toc149777350)

[2.Inleiding 3](#_Toc149777351)

[3.Methodes 4](#_Toc149777352)

[3.1.MongoDB 4](#_Toc149777353)

[3.2. Dask 6](#_Toc149777354)

[3.2.1. Wat is Dask 6](#_Toc149777355)

[3.2.2. Toepassing van Dask op mijn code 7](#_Toc149777356)

[3.3. Neurale netwerken 8](#_Toc149777357)

[3.3.1. Wat is een neuraal netwerk 8](#_Toc149777358)

[3.3.2. Convolutional neural networks 9](#_Toc149777359)

[3.3.3. Recurrent neural networks 9](#_Toc149777360)

[3.3.4 Toepassing van neurale netwerken in mijn code 10](#_Toc149777361)

[3.4. Visualisatie dashboard 15](#_Toc149777362)

[3.4.1. Overzicht 15](#_Toc149777363)

[3.4.2. Zoom 16](#_Toc149777364)

[3.4.3. Filteren 17](#_Toc149777365)

[3.4.4. Details op aanvraag 17](#_Toc149777366)

[4 Conclusie 18](#_Toc149777367)

# 1.Samenvatting

Deze samenvatting beschrijft een vervolgopdracht van Big Data Assignment 1, waarbij ik verder werk met de hotel review dataset. Ik leg uit hoe ik een verbinding heb kunnen maken met een MongoDB om gegevens eenvoudig naar de database te uploaden en in Python te laden. Daarnaast wordt inzicht gegeven in wat MongoDB is en hoe het functioneert. Bovendien worden functies beschreven die het gebruik van MongoDB in nieuwe Python-bestanden vergemakkelijken.

In het volgende hoofdstuk wordt Dask geïntroduceerd en toegelicht, inclusief voorbeelden van code die laten zien hoe Dask kan worden toegepast en welke voordelen het biedt ten opzichte van machine learning met scikit-learn. Hierbij worden snelheid en nauwkeurigheid met elkaar vergeleken.

Vervolgens wordt de conceptie van neurale netwerken uitgelegd, waarbij verschillende soorten neurale netwerken en hun toepassingen aan bod komen. Ik presenteer tevens codevoorbeelden om deze concepten in Python toe te passen. Hierbij wordt onderzocht hoe het aantal lagen, epochs en de batchgrootte van invloed zijn op de snelheid en nauwkeurigheid van het trainen.

In het laatste hoofdstuk wordt er een dashboard ontwikkeld dat gegevens op verschillende manieren visualiseert, met als doel interessante inzichten te verschaffen over de data, zoals het totale aantal recensies per hotel en de herkomst van de reviewers per land.

Ten slotte volgt een conclusie.

# 2.Inleiding

In dit verslag ga ik verder met de dataset waar ik mee werkte in assignment 1. Voor het opslaan van de dataset moet ik dit keer een NOSQL database gebruiken. Hiervoor gebruik ik MongoDB. Voor parallel en gedistribueerd rekenen gebruik ik DASK. Verder pas ik machine learning algoritmes en neural networks toe. Hierna vergelijk ik de resultaten met die van de eerste opdracht. Tot slot maak ik een dashboard om de dataset te visualiseren.

**Beschrijving van de opdracht.**

De aanleiding van deze opdracht was om alle kennis van BDSE van blok 2 van het semester te combineren in 1 opdracht. Voor deze opdracht moeten de volgende doelstellingen worden behaald.

* Om een aantrekkelijke visuele representatie van alle gegevens in de dataset te verkrijgen, met visuele interactieve elementen om de zogenaamde Visualisatie mantra te ondersteunen. Dit moet door middel van een interactief dashboard.
* De gehele dataset wordt opgeslagen in een NoSQL-database, zoals bijvoorbeeld MongoDB. Er moet een live verbinding worden geïmplementeerd om gegevens te filteren tijdens het uitvoeren van het script. Dit betekent dat tijdens het proces van filteren en plotten nieuwe gegevens vanuit de database naar het Python-script/dashboard moeten worden overgedragen.
* Een script dat DASK implementeert in de context van Sentiment Analysis.
* Een script dat ten minste 2 neurale netwerken implementeert.
* Een vergelijking moet worden gemaakt tussen de scripts die zijn genoemd in punt 3 en 4 op basis van prestaties en de algehele nauwkeurigheid/AUC van beide methoden.

Afbeelding met tekst, clipart, Graphics, Lettertype

Automatisch gegenereerde beschrijving

# 3.Methodes

## 3.1.MongoDB

MongoDB is een populaire opensource database management systeem dat valt onder de categorie van NoSQL-database. NoSQL verwijst naar een type database die niet gebonden is aan de traditionele relationele database structuur zoals SQL-databases zijn. In plaats van tabellen en rijen te gebruiken, zoals in traditionele relationele databases, maakt MongoDB gebruik van collecties en documenten.

In tegenstelling tot SQL-databases slaat MongoDB gegevens op in een flexibele, document georiënteerde structuur. Het is populair vanwege de schaalbaarheid en flexibiliteit. Het is zeer geschikt voor toepassingen waarbij gegevens snel kunnen groeien of wanneer de structuur van de gegevens vaak veranderd. MongoDB wordt vaak gebruikt voor web toepassingen en mobiele apps waarbij flexibiliteit en snelheid belangrijk zijn.

Documenten bestaan uit sleutel-waarde paren, die de basisvorm van gegevens in MongoDB vormen. Collecties bevatten sets van documenten en functioneren als het equivalent van tabellen in relationele databases. Elk document kan verschillend zijn met een variabel aantal velden. De grootte en inhoud van elk document kunnen van elkaar verschillen.

Om te beginnen upload ik de bestanden die ik eerder heb gebruikt in de MongoDB. Het gaat hier om de originele dataset van Kaggle genaamd Hotel\_reviews.csv en de dataset die ik zelf heb gemaakt genaamd cleaned\_data.csv.

Op het bestand cleaned\_data heb ik ook labeling, stemming en lemmatizing toegepast. Ik maak eerst een connectie met de mongoDB en noem de database assignment 2. Ik maak voor de verschillende datasets een aparte collectie aan.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Connectie en upload naar MongoDB

Ik heb ook een aantal functies gemaakt die het makkelijke moeten maken om data in te laden, nieuwe data te uploaden en te bewerken.

Ik heb een functie voor het inladen van alle collecties die ik in de database opgeslagen heb. Deze heb ik laad\_data genoemd. Op deze manier kan ik de gegevens uit de collectie halen wanneer ik deze gebruik voor het trainen van mijn modellen en voor het dashboard.

Afbeelding met tekst, schermopname, Lettertype, nummer

Automatisch gegenereerde beschrijving

Functie laden data

Voor het uploaden van data naar de database heb ik de functie upload\_data gemaakt. Hiermee kan ik makkelijk een dataframe uploaden. Dit is ook handig wanneer ik bijvoorbeeld wijzigingen heb gemaakt.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Functie uploaden data

Voor het aggregeren en transformeren van de data maak ik de functie aggregate. In dit geval wordt de **$match**-operatie gebruikt om documenten te selecteren waarvan het veld 'Total\_Number\_of\_Reviews' groter is dan 5000. Dit gebruik ik later voor het visualiseren.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Functie aggregate

## 3.2. Dask

### 3.2.1. Wat is Dask

DASK is een open-source python bibliotheek die is ontworpen om parallelle en gedistribueerde computing mogelijk te maken voor taken die grotere datasets en complexe berekeningen vereisen. Het biedt een flexibele manier om data te manipuleren en berekeningen uit te voeren, vergelijkbaar met Pandas, maar met ondersteuning voor parallelle verwerking op meerdere CPU’s of zelfs gedistribueerde clusters.

Om minder geheugen te gebruiken tijdens berekeningen, slaat Dask de volledige gegevens op de schijf op en gebruikt kleinere delen van de dataset vanaf de schijf voor verwerking. Tijdens de verwerking worden eventuele tussentijdse waarden die worden gegenereerd, zo snel mogelijk verwijderd om geheugenverbruik te besparen. Dask is met name nuttig voor het verwerken en analyseren van big data.

Wanneer je gebruik maakt van meerdere machine learning modellen voor sentiment analyse kan je Dask gebruiken om parallelle modeltraining uit te voeren op meerdere delen van de dataset. Dit kan de trainingstijd verkorten. Vooral bij diepgaande neurale netwerken of andere complexe modellen

Dask maakt gebruik van dezelfde soort interface als Pandas met een paar kleine verschillen vanwege het parallelle karakter van Dask. (https://docs.dask.org/en/latest/dataframe.html, 2023) Dask dataframes zijn ontworpen als een vervanging van voor Pandas dataframes. Terwijl Pandas goed is voor het werken met datasets die in het geheugen passen, Is Dask goed voor het bewerken van grotere datasets door parallel gedistribueerde rekenen te gebruiken.

Dask biedt een bibliotheek genaamd “dask-ml” die scikit-learn uitbreidt om te werken met datasets die groter zijn dan het geheugen, door taken voor machine learning te parallelliseren en te veredelen. Hiermee kunnen machine learning algoritmes gebruikt worden op Dask verzamelingen, zoals Dask-dataframes.

Verder biedt Dask ook arrays, vergelijkbaar met NumPY- arrays, maar geschikt voor grootschalige gegevensverwerking.

Dask bevat ook structuren genaamd “Bag” en “Delayed” die handig zijn voor verschillende dataverwerkingsscenario’s. Dask bag is een gegevensstructuur die handig voor werken met ongestructureerde of semigestructureerde gegevens, zoals JSON-bestanden, tekstbestanden of andere gegevensbronnen waarbij elke record anders kan zijn.

Dask delayed stelt je instaat om aangepaste berekeningen te maken en deze uit te stellen voor latere uitvoering. Dit kan handig zijn in situaties waarin je aangepaste complexe berekeningen wilt uitvoeren en de berekeningen wilt uitstellen totdat ze allemaal zijn gedefinieerd.

### 3.2.2. Toepassing van Dask op mijn code

Hieronder is een voorbeeld van mijn code. Hierbij vergelijk ik LogisticRegression van scikit-learn met LogisticRegression van Dask. Ik gebruik hierbij de dataset van de vorige opdracht.

Om te beginnen pas ik tfidf toe op de kolom review zodat de modellen ermee kunnen werken. Daarna maak ik twee functies. Een voor scikit-learn en een voor Dask.

Afbeelding met tekst, schermopname, software, scherm

Automatisch gegenereerde beschrijving

Code ml-modellen van scikit-learn en dask

Tot slot voer ik alle functies uit en print ik de nauwkeurigheid van de modellen en hoelang de modellen erover hebben gedaan. Ik gebruik ook de functie laad\_data dat ik in het MongoDB bestand heb gemaakt om de dataset in te laden.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Samenvoegen van de functies en printen nauwkeurigheid

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Resultaten modellen met en zonder dask

Zoals hierboven te zien is Dask een stuk sneller dan de standaard scikit-learn logistic\_regression. De nauwkeurigheid is wel iets lager. De snelheid is ongeveer 30% sneller.

## 3.3. Neurale netwerken

In dit onderdeel ga ik beschrijven wat een neuraal netwerk is en hoe het in elkaar zit. Vervolgens beschrijf ik ook de verschillende soorten neurale netwerken. Tot slot laat ik zien hoe ik in mijn code neurale netwerken heb toegepast. Het doel van de opdracht is om minstens 2 neurale netwerken te implementeren.

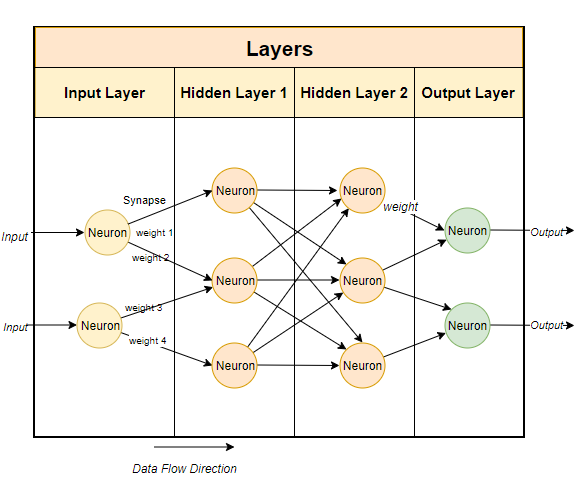
### 3.3.1. Wat is een neuraal netwerk

Een neuraal netwerk is een machine learning proces genaamd deep learning in artificial intelligence dat is geïnspireerd op de werking van het menselijk brein. Het wordt vaak gebruikt voor taken zoals patroonherkenning, classificatie, voorspellingen en taalherkenning. (aws.amazon.com, sd)

Een neuraal netwerk bestaat uit een verzameling van neuronen, ook wel knooppunten genoemd. Elk neuron ontvangt invoergegevens voert een berekening uit en geeft een uitvoer. Neuronen zijn georganiseerd in lagen. Een typisch neuraal netwerk bestaat uit drie soorten lagen. De input laag, één of meer verborgen lagen en de output laag. De input laag ontvangt de gegevens, de verborgen lagen voeren berekeningen uit en de output laag geeft het eindresultaat (aws.amazon.com, sd)

Neuronen zijn verbonden aan andere neuronen via weights. Elke verbinding heeft een gewicht dat bepaald hoe sterk de invoer van het ene neuron de uitvoer van het andere neuron beïnvloedt (deepai.org, sd). Elk neuron past de gewogen invoer toe op een activeringsfunctie, meestal een niet lineaire functie. Deze functie bepaalt of het neuron wordt geactiveerd op basis van de gewogen invoer. Populaire activatiefuncties zijn de sigmoid, relu en tahn. (Sharma, 2017)

De berekeningen van neurale netwerken beginnen bij de invoerlaag en gaan door elke laag via de verbindingen en activeringsfuncties, Dit wordt voorwaartse propagatie genoemd. De uitvoer van de laatste laag is het uiteindelijke voorspelde resultaat. Tijdens het trainen van het neurale netwerk worden de weights aangepast om de voorspelling dichter bij de gewenste resultaten te brengen. Dit wordt gedaan met behulp van een algoritme voor de achterwaartse propagatie, zoals de backward propagation-methode. Bij backward propagation wordt de fout tussen de voorspelde uitvoer en gewenste uitvoer gemeten en de fout wordt teruggeworpen door het netwerk om de weights in de verborgen laag aan te passen. Dit herhaalt tot de fout minimaal is. Na het trainen kan het netwerk voorspellen.



### 3.3.2. Convolutional neural networks

Convolutional neural networks (CNNs) zijn een speciaal type neuraal netwerk ontworpen voor beeldverwerkingstaken voor het herkennen van objecten of patronen in foto’s. Het belangrijkste kenmerk van CNNs is de toepassing van convolutielagen om kenmerken uit de input laag te halen. De convolutie laag bestaat uit een verzameling filters die over het invoerbeeld worden geschoven om lokale patronen te detecteren. Elk filter is verantwoordelijk voor het herkennen van bepaalde kenmerken in het beeld zoals randen, vormen, texturen en kleuren. Een filter is matrix van willekeurige getalswaarden. (telusinternational.com, 2022)

Na het berekenen van de convolutie laag wordt op elk element van de kenmerkaart een niet lineaire activatiefunctie toegepast. CNNs zijn erg goed in het interpreteren van visuele gegevens die niet in een specifieke volgorde worden gepresenteerd. Ze hebben echter moeite met het interpreteren van tijdsafhankelijke of sequentiële informatie.

### 3.3.3. Recurrent neural networks

Een Recurrent neural networks (RNNs) is een type neuraal netwerk dat is ontworpen om te werken met sequentiële gegevens, waarbij volgorde het verband tussen de gegevens van cruciaal belang zijn. RNNs hebben een unieke architectuur die hen instaat stelt om de context en de tijdsafhankelijkheid van gegevens te begrijpen. In plaats van traditionele feedforward neuronen die in gewone neurale netwerken worden gebruikt, hebben RNNs herhalende neuronen. Deze neuronen nemen input op te nemen en de activatie van eerdere of latere knooppunten te hergebruiken om de output te beïnvloeden. (telusinternational.com, 2022)

### 3.3.4 Toepassing van neurale netwerken in mijn code

Na het inladen van de dataset splits ik de data in een training set en een test set. Vervolgens heb ik een tokenizer toegepast om de tekst om te zetten naar numerieke gegevens die ik vervolgens omzet naar sequenties van getallen. Ik pas ook padding toe aan de sequenties zodat ze allemaal dezelfde lengte hebben. De maximale lengte wordt beperkt tot Max\_sequence\_length. Dit is handig omdat machine learning modellen meestal verwachten dat de invoergegevens van gelijke lengte zijn. Hiermee is de tekst voorbereid zodat de neurale netwerken er gebruik van kunnen maken.

Afbeelding met tekst, schermopname

Automatisch gegenereerde beschrijving

Eerste CNN

Ik heb voor de neurale netwerken voorbeelden gebruikt die op DLO staan onder applied artificial intelligence. Ik heb deze netwerken niet aangepast en gekeken wat de resultaten zijn. De nauwkeurigheid is al erg goed voor het CNN met een nauwkeurigheid van 94% en een loss van 20%.

Afbeelding met tekst, Lettertype, lijn, schermopname

Automatisch gegenereerde beschrijving

Ik heb voor de CNN wat lagen toegevoegd en gekeken of dit effect heeft op hoe nauwkeurigheid en de validatie set. Ik heb ook gekeken wat voor effect het veranderen van de batch grootte voor effect heeft.

Ik heb ervoor gekozen om de batch size op te verhogen naar 128 omdat dit het trainen iets sneller maakt. Als ik de batch size nog groter zou maken, zou het veel GPU-geheugen kosten.

Ik heb gekeken wat voor effect het aantal epochs heeft op de nauwkeurigheid. Ik heb een plot gemaakt die de nauwkeurigheid en de nauwkeurigheid van de validatie set laat zien. Na meer dat 5 epochs gaat de nauwkeurigheid van de validatie set omlaag. Daarom heb ik ervoor gekozen om het aantal epochs op 5 te laten.

Verder heb ik ook gekeken of het toevoegen van een aantal Conv1D en Maxpooling1D lagen effect zou hebben op de nauwkeurigheid. De nauwkeurigheid bleef ongeveer hetzelfde. Wat op de plot te zien is, is dat de nauwkeurigheid van de validatie set na de 2e epoch daalt.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Tweede CNN

Afbeelding met tekst, Lettertype, schermopname

Automatisch gegenereerde beschrijving

Resultaten tweede CNN

Afbeelding met tekst, lijn, Perceel, schermopname

Automatisch gegenereerde beschrijving

CNN nauwkeurigheid plot

Ik gebruikte daarna de RNN model van DLO. De eerste RNN gaf ook goede resultaten met een nauwkeurigheid van 94% en loss van 17%. De nauwkeurigheid van en de validatie set nauwkeurigheid blijven ongeveer hetzelfde. Er is te zien dat na de 2e epoch de nauwkeurigheid iets daalt en daarna weer omhooggaat.

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Eerste RNN

Afbeelding met tekst, Lettertype, schermopname

Automatisch gegenereerde beschrijving

Eerste RNN resultaten

Afbeelding met tekst, diagram, lijn, Perceel

Automatisch gegenereerde beschrijving

RNN nauwkeurigheid plot

Ik heb een 2 RNN netwerken gemaakt waarbij ik een laag met GRU of LTSM toevoeg. GRU (gated recurrent Unit) en LTSM (Long Short-term memory) zijn types RNN-netwerkarchitectuur die zijn ontworpen om problemen met vanishing gradients te verminderen. Vanishing gradients is een probleem dat zich voordoet tijdens het trainen van diepe neurale netwerken, met name bij RNNs.

Afbeelding met tekst, schermopname

Automatisch gegenereerde beschrijving

GRU en LTSM RNNs

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Resultaten GRU RNN

Afbeelding met tekst, Lettertype, schermopname

Automatisch gegenereerde beschrijving

LTSM RNN

Afbeelding met tekst, Perceel, lijn, schermopname

Automatisch gegenereerde beschrijving

Er is te zien dat er niet veel verschil is in nauwkeurigheid. De nauwkeurigheid zit in alle 3 gevallen boven 94%.

## 3.4. Visualisatie dashboard

Volgens de opdracht moeten de volgende elementen in het visualisatie dashboard komen volgens de zogenaamde Visualisatie-mantra:

• Overzicht: krijg een overzicht van de gehele collectie   
• Zoom: zoom in op interessante items   
• Filteren: interessante items eruit filteren of interessante items erin filteren  
• Details op aanvraag: Selecteer een item of groep en ontvang relevante informatie overeenkomstig

Voor het maken van het dashboard in python gebruik ik Dash. Dash is een python framework voor het bouwen van interactieve webapplicaties. Het visualiseren van de data kan helpen met inzichten krijgen. Visualisaties van data kan helpen bij het identificeren van uitschieters of het ontdekken van ontbrekende en inconsistenties gegevens binnen de data.

### 3.4.1. Overzicht

Ik heb een overzicht gemaakt met alle hotels in een tabel. In deze tabel zijn de volgende velden te zien: De naam van het hotel, de gemiddelde score, het aantal reviews, het land, de stad, het adres en de coördinaten van waar het hotel zich bevind.

Afbeelding met tekst, schermopname, Lettertype, nummer

Automatisch gegenereerde beschrijving

Overview tabel

Afbeelding met tekst, schermopname, diagram, software

Automatisch gegenereerde beschrijving

Overview scatterplot en piechart

### 3.4.2. Zoom

Voor de zoom functie heb ik een kaart met de locaties van de hotels. Je kan in en uitzoomen om te zien in welke steden de hotels zich bevinden. Elk bolletje in de kaart staat voor een hotel. Met de kleur is te zien wat de gemiddelde score is van het hotel. Wanneer je met je muis over een bolletje gaat zie je het totaal aantal reviews, de coördinaten en de gemiddelde score. Het is ook mogelijk om in de scatterplot een gedeelte te selecteren om op in te zoomen.

Afbeelding met tekst, kaart, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Kaart

Afbeelding met tekst, Perceel, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Scatterplot ingezoomd

### 3.4.3. Filteren

Er is de mogelijkheid om te filteren op basis van land en of score van de hotels. Het is ook mogelijk om hotels in de tabel aan te vinken om specifieke gegevens te zien zoals hieronder

Afbeelding met tekst, Lettertype, lijn, nummer

Automatisch gegenereerde beschrijving

Afbeelding met tekst, Lettertype, schermopname, wit

Automatisch gegenereerde beschrijving

### 3.4.4. Details op aanvraag

Om extra gegevens te zien is er de mogelijkheid om met de muis over de scatterplot en piechart te gaan, waardoor er meer info wordt weergegeven. Bijvoorbeeld in de scatterplot worden de scores en het aantal reviews weergegeven. In de piechart is te zien hoeveel reviews uit het land komt waar je de muis boven houdt.

Afbeelding met tekst, schermopname, Lettertype, Perceel

Automatisch gegenereerde beschrijving

Scatterplot hover

Afbeelding met tekst, diagram, Perceel, Lettertype

Automatisch gegenereerde beschrijving

Pie chart hover

# 4 Conclusie

In dit verslag ben ik begonnen met het aanmaken van een MongoDB. Hierin heb ik alle data opgeslagen. Verder heb ik ook een aantal functies gemaakt voor het inladen van de data.

Een belangrijk onderdeel van het verslag was het vergelijken van Dask met de standaard machine learning modellen van scikit-learn. Dask is een library voor parallelle computing in python. Ik heb vastgesteld dat Dask sneller is dan scikit-learn en erg nuttig is voor data-analyse en machine learning taken van grote datasets. Dask had wel een iets lagere nauwkeurigheid van 73%.

Daarnaast heb ik verschillende neurale netwerken gemaakt en de nauwkeurigheid vergeleken. Ik heb verschillende architecturen geïmplementeerd waaronder Convolutional neuraal networks en recurrent neural networks. De netwerken hadden een hoge nauwkeurigheid van 94%.

Tot slot heb ik een dashboard gemaakt met behulp van DASH waarmee waardevolle inzichten uit de data gehaald kunnen worden.

Bronnen

Gillis, A. S., & Botelho, B. (2023). MongoDB. *Data Management*. <https://www.techtarget.com/searchdatamanagement/definition/MongoDB>

*Aanbevolen werkwijzen voor databasebeveiliging*. (z.d.). <https://www.oracle.com/nl/database/nosql/what-is-nosql/>

*What is Dask and how does it work? | Saturn Cloud Blog*. (2023, 4 oktober). <https://saturncloud.io/blog/what-is-dask/>

*DASK DataFrame — DASK Documentation*. (z.d.). <https://docs.dask.org/en/latest/dataframe.html>

*DASK — DASK Documentation*. (z.d.). <https://docs.dask.org/en/stable>

*BAG — Dask Documentation*. (s. d.). <https://docs.dask.org/en/stable/bag.html>

*Dask delayed — Dask Documentation*. (s. d.). <https://docs.dask.org/en/stable/delayed.html#:~:text=The%20Dask%20delayed%20function%20decorates,arguments%20into%20a%20task%20graph>.

*Dask\_Ml.Linear\_Model.LogisticRegression — DasK-ML 2022.5.28 documentation*. (z.d.). <https://ml.dask.org/modules/generated/dask_ml.linear_model.LogisticRegression.html>

<https://aws.amazon.com/what-is/neural-network/#:~:text=It%20is%20a%20type%20of,their%20mistakes%20and%20improve%20continuously>.

3Bplus. (2022, 24 maart). *Hoe werkt een neuraal netwerk – Neurale netwerken uitgelegd - 3BPLUS*. <https://3bplus.nl/machine-learning/hoe-werkt-een-neuraal-netwerk-neurale-netwerken-uitgelegd/>

DeepAI. (2020, 25 juni). *WEIGHT (Artificial neural Network)*. DeepAI. <https://deepai.org/machine-learning-glossary-and-terms/weight-artificial-neural-network>

Sharma, S. (2022, 20 november). Activation functions in neural networks - towards data science. *Medium*. <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>

TELUS International. (2022, 17 januari). What’s the difference between CNN and RNN? *TELUS International*. <https://www.telusinternational.com/insights/ai-data/article/difference-between-cnn-and-rnn>

Wang, C. (2021, 7 december). The vanishing gradient problem - towards data science. *Medium*. <https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484>

*DASH Documentation & User Guide | Plotly*. (z.d.). <https://dash.plotly.com/>