

Faculteit Industriële Ingenieurswetenschappen

master in de industriële wetenschappen: informatica

Masterthesis

**AI-ondersteunde webapplicatie voor datavisualisatie en analyse van
oplaadinfrastructuur voor elektrische voertuigen**

Jenthe Lowist

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: informatica

PROMOTOR :

Prof. dr. Davy VANACKEN

PROMOTOR :

Ing. Sebastiaan VANSPAUWEN

Gezamenlijke opleiding UHasselt en KU Leuven



Universiteit Hasselt | Campus Diepenbeek | Faculteit Industriële Ingenieurswetenschappen | Agoralaan Gebouw H - Gebouw B | BE 3590 Diepenbeek

Universiteit Hasselt | Campus Diepenbeek | Agoralaan Gebouw D | BE 3590 Diepenbeek
Universiteit Hasselt | Campus Hasselt | Martelarenlaan 42 | BE 3500 Hasselt



2023
2024

Faculteit Industriële Ingenieurswetenschappen

master in de industriële wetenschappen: informatica

Masterthesis

***AI-ondersteunde webapplicatie voor datavisualisatie en analyse van
oplaadinfrastructuur voor elektrische voertuigen***

Jenthe Lowist

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: informatica

PROMOTOR :

Prof. dr. Davy VANACKEN

PROMOTOR :

Ing. Sebastiaan VANSPAUWEN



KU LEUVEN

Woord vooraf

Met gepaste trots presenteer ik u hierbij mijn masterproef, waarmee ik een belangrijke mijlpaal bereik in mijn academische loopbaan. De weg naar dit moment was er één van hard werken, toewijding en doorzettingsvermogen, maar vooral ook van inspiratie, ondersteuning en waardevolle begeleiding. Dit voorwoord biedt mij de gelegenheid om mijn dank en waardering uit te spreken voor de personen die op verschillende manieren hebben bijgedragen aan het tot stand komen van deze masterproef.

Allereerst wens ik mijn oprechte dankbaarheid te uiten aan prof. dr. Davy Vanacken en de heer Sebe Vanbrabant. Hun expertise, geduld en inzichtvolle feedback hebben mij geholpen mijn onderzoek op een hoger niveau te tillen. Hun begeleiding was essentieel in het richting geven aan mijn ideeën en het verfijnen van de inhoud van deze studie. Zonder hun voortdurende steun en constructieve kritiek zou dit werk niet in zijn huidige vorm bestaan.

Daarnaast wil ik mijn erkentelijkheid betuigen aan ing. Sebastiaan Vanspauwen, voor uw waardevolle suggesties en ondersteuning tijdens het onderzoeksproces. Uw kritische blik en de zorgvuldige revisies hebben bijgedragen aan de kwaliteit van deze masterproef. Ik ben u bijzonder dankbaar voor de tijd en moeite die u heeft gestoken in het begeleiden van dit onderzoek.

Tot slot wil ik mijn familie en vrienden bedanken voor hun onvoorwaardelijke steun en begrip gedurende de afgelopen periode. Hun vertrouwen en aanmoediging gaven mij de kracht om door te zetten, zelfs op de momenten dat het moeilijk was.

Deze masterproef is een weerspiegeling van de kennis en vaardigheden die ik tijdens mijn academische reis heb opgedaan, maar is vooral het resultaat van de samenwerking met de mensen die ik hier heb genoemd. Aan hen allen mijn oprechte dank.

Inhoudsopgave

Woord vooraf	1
Lijst van tabellen	5
Lijst van figuren	7
Abstract.....	11
Abstract in English	13
Inleiding.....	15
1.1 Situering	16
1.2 Probleemstelling	17
1.3 Doelstellingen	18
1.4 Methode.....	19
1.5 Vooruitblik	21
Bronnenstudie	23
2.1 Inleiding.....	23
2.2 Vergelijking machinelearningalgoritmen	24
2.3 Vergelijking machine learning frameworks	27
2.4 Verminderen van false positives in fraudedetectie	31
2.5 Juridische aspecten	32
2.6 Relevante technologieën en frameworks	33
Gegevensbeheer en analyse	35
3.1 EEVEE Mobility Database	35
3.2 Visuele data-analyse	42
3.3 Nabespreking.....	49
Ontwikkeling webapplicatie.....	51
4.1 Ontwikkeling isolation forest-algoritme.....	51
4.2 Ontwikkeling API	55
4.3 Ontwikkeling frontend webapplicatie	55
Besluit	65
Referentielijst	69

Lijst van tabellen

Tabel 1: Vergelijkingstabel van machinelearningalgoritmen [1-5]	24
Tabel 2: Vergelijkingstabel van machine learning frameworks [6-9]	28
Tabel 3: Gedeeltelijke output van het isolation forest-algoritme toegepast op abnormaal gebruik	54

Lijst van figuren

Figuur 1: High-level ER diagram van de EEVEE dataset waarbij de relaties tussen de entiteiten getoont wordt.....	35
Figuur 2: addresses tabel van de EEVEE dataset.....	36
Figuur 3: brands tabel van de EEVEE dataset.....	37
Figuur 4: car_fleet tabel van de EEVEE dataset.....	37
Figuur 5: charges tabel van de EEVEE dataset.....	37
Figuur 6: car tabel van de EEVEE dataset.....	38
Figuur 7: events tabel van de EEVEE dataset.....	38
Figuur 8: fleets tabel van de EEVEE dataset.....	39
Figuur 9: locations tabel van de EEVEE dataset.....	39
Figuur 10: users tabel van de EEVEE dataset.....	40
Figuur 11: Screenshot van JetBrains DataGrip	40
Figuur 12: Verdeling van automerken (n=54149). 74,9% van de auto's in de dataset is van het merk Tesla, terwijl de overige 25,1% andere merken vertegenwoordigen	43
Figuur 13: Wereldkaart visualisatie van de oplaadsessies (voor het toepassen van de filter)	43
Figuur 14: Wereldkaart visualisatie van de oplaadsessies (gedetailleerde kijk op vierkante datapuntenwolk)	44
Figuur 15: Taartdiagram visualisatie van de distributie van automerken binnen het verdachte datapuntenwolk rond Rusland.....	45
Figuur 16: Taartdiagram visualisatie van de distributie van automerken buiten het verdachte datapuntenwolk rond Rusland.....	45
Figuur 17: Wereldkaart visualisatie van de oplaadsessies met enkel Skoda data.....	46
Figuur 18: Wereldkaart visualisatie van de oplaadsessies (na het toepassen van de filter, zonder Skoda data).....	46
Figuur 19: Adresclassificatie in de EV-dataset (n=601190). De overgrote meerderheid (82%) van de adressen is gecategoriseerd als 'onzichtbaar'	47
Figuur 20: Wagenclassificatie in de EV-dataset (n=40209). Een klein aantal (4,6 %) van de wagens is gecategoriseerd als hybrid.....	47
Figuur 21: Verdeling van event types in de dataset (n=135267943). Een substantieel deel van de events lijkt relevant voor het detecteren van V2G- of charge pass-fraude	48
Figuur 22: Grafiek met de resultaten van het V2G isolation forest-algoritme. De x-as geeft het startpercentage van de batterij aan tijdens een 'parked' event, de y-as het eindpercentage. De oranje punten stellen anomalieën voor (-1).....	53
Figuur 23: Diagram dat de architectuur en stroom van de FastAPI-API vertegenwoordigt	55
Figuur 24: Screenshot van de API documentatie.....	57

Figuur 25: Diagram van de project structuur binnen de Vue.js applicatie.....	58
Figuur 26: Diagram van de afhankelijkheden binnen de Vue.js applicatie	60
Figuur 27: Webpagina dat een tabel weergeeft van alle fleets met bijhorende data.....	61
Figuur 28: Webpagina dat een tabel weergeeft van alle EV-eigenaars met bijhorende data	62
Figuur 29: Webpagina dat een zijvenster weergeeft van alle individuele laadsessies voor een bepaalde EV eigenaar met bijhorende data.....	63
Figuur 30: Webpagina dat aantoont dat deze specifieke pagina niet deel is van het afstudeerproject.....	63

Abstract

De toenemende adoptie van elektrische voertuigen brengt nieuwe uitdagingen met zich mee op het gebied van fraudepreventie en -detectie. Dit onderzoek richt zich op het ontwikkelen van AI-ondersteunde datavisualisatietechnieken binnen een webapplicatie, om frauduleuze activiteiten, zoals '*vehicle-to-grid*'-detectie, '*charge pass fraud*', en abnormaal gebruik, te identificeren en te analyseren binnen de groeiende markt van elektrische voertuigen en oplaadinfrastructuur.

De methodologie omvat een literatuurstudie naar machinelearningalgoritmen en frameworks voor zowel de frontend als de backend van de webapplicatie. Relaties in de database werden opgezet met foreign keys en kolommen werden geïndexeerd om prestaties te verbeteren. Er werd een visuele data-analyse uitgevoerd, een *isolation forest*-algoritme werd ontwikkeld om anomalieën te detecteren, en een API werd gecreëerd voor toegang tot data en resultaten. Tot slot werd een webapplicatie ontwikkeld voor de visualisatie van anomalieën.

De resultaten van dit onderzoek omvatten de ontwikkeling van een operationele webapplicatie met behulp van het Vue.js *framework* en een REST API gemaakt met FastAPI. Deze webapplicatie is in staat om frauduleuze activiteiten te detecteren en te visualiseren met behulp van het *isolation forest* AI-algoritme. Deze resultaten dragen bij aan een betere detectie van frauduleuze activiteiten binnen het EV-oplaadecosysteem, wat essentieel is voor het behoud van de integriteit van de oplaadinfrastructuur en het vertrouwen van gebruikers.

Abstract in English

The increasing adoption of electric vehicles brings new challenges in fraud prevention and detection. This research focuses on developing AI-assisted data visualisation techniques within a web application, to identify and analyse fraudulent activities, such as vehicle-to-grid detection, charge pass fraud, and abnormal usage within the growing electric vehicle and charging infrastructure market.

The methodology includes a literature review of machine learning algorithms and frameworks for both the frontend and backend of the web application. Relationships in the database were set up with foreign keys and columns were indexed to improve performance. Visual data analysis was performed, an isolation forest algorithm was developed to detect anomalies, and an API was created to access data and results. Finally, a web application was developed for the visualisation of anomalies.

The results of this research included the development of an operational web application using the Vue.js framework and a REST API created with FastAPI. This web application is able to detect and visualise fraudulent activities using the isolation forest AI algorithm. These results contribute to better detection of fraudulent activities within the EV charging ecosystem, which is essential for maintaining the integrity of the charging infrastructure and user trust.

Hoofdstuk 1

Inleiding

De transitie naar duurzame mobiliteit is een steeds urgenter vraagstuk, vooral gezien de noodzaak om de klimaatverandering tegen te gaan en de luchtkwaliteit in stedelijke gebieden te verbeteren. Elektrische voertuigen (EV's) spelen een essentiële rol in deze transitie, aangezien ze bijdragen aan de vermindering van broeikasgasemissies en de afhankelijkheid van fossiele brandstoffen [1]. Volgens een studie van de Universiteit Antwerpen is het van cruciaal belang dat beleidsmaatregelen gericht zijn op het bevorderen van duurzame mobiliteit, met als doel de Europese doelstellingen voor emissiereductie tegen 2050 te bereiken [2].

Daarnaast is de opkomst van artificiële intelligentie (AI) een belangrijke factor die deze transitie kan versnellen. AI-technologieën kunnen helpen bij het optimaliseren van mobiliteitsdiensten, het verbeteren van verkeersmanagement en het bevorderen van het gebruik van gedeelde mobiliteitsoplossingen. Dit biedt een kans voor organisaties zoals Wisemen om de transitie naar duurzame mobiliteit te ondersteunen door innovatieve oplossingen te ontwikkelen die gebruikmaken van AI. Door het combineren van AI met duurzame mobiliteitsstrategieën kunnen we niet alleen de efficiëntie van transport verbeteren, maar ook de bredere sociale en economische effecten van deze transitie beter begrijpen en implementeren [3] [4].

In dit licht is het van groot belang dat de ontwikkeling en implementatie van duurzame mobiliteitsoplossingen niet alleen technologische vooruitgang omvatten, maar ook een integrale benadering van sociaal beleid en economische duurzaamheid. Hierdoor kan een veerkrachtiger en inclusiever mobiliteitssysteem ontstaan dat tegemoetkomt aan de behoeften van alle gebruikers, terwijl tegelijkertijd de milieu-impact wordt geminimaliseerd.

Wisemen, een vooraanstaand bedrijf in de digitalisering van de maatschappij, heeft zich tot doel gesteld de adoptie van elektrische voertuigen te stimuleren. Dit willen ze bereiken door de ontwikkeling van een AI-ondersteunde webapplicatie die datavisualisatie en analyse van oplaadinfrastructuur faciliteert.

Binnen de diensten van Wisemen ligt de focus op het verhogen van de omzet van hun klanten, het verbeteren van digitale efficiëntie en het voldoen aan klantverwachtingen in een digitale omgeving.

EEVEE Mobility, een dochterbedrijf van Wisemen, richt zich op het vereenvoudigen, transparanter maken en goedkoper maken van elektrisch rijden voor bedrijven. Het bedrijf biedt oplossingen die bedrijven helpen om hun elektrische laadkosten te begrijpen, te beheren en te reduceren. Dit doen ze door data rechtstreeks uit de voertuigen te verzamelen, wat hen in staat stelt om waardevolle inzichten te bieden en automatisch kosten terug te vorderen zonder dure hardware te hoeven gebruiken [5] [6] [7].

Dit onderzoek richt zich op het benutten van een uitgebreide dataset met gegevens over oplaadpunten, locaties en bijbehorende gebeurtenissen. Door middel van AI-ondersteunde

datavisualisatietechnieken wordt er inzicht verkregen in frauduleuze activiteiten en abnormaal gebruik. Het uiteindelijke doel is om niet alleen data te presenteren, maar ook om interactieve en intuïtieve visuele representaties te ontwikkelen die gebruikers helpen verschillende aspecten van de oplaadinfrastructuur beter te begrijpen en te analyseren. In het kader van dit onderzoek is een nieuwe webapplicatie ontwikkeld die aansluit bij deze doelstellingen door een prototype te bouwen om frauduleuze activiteiten binnen een elektrische oplaadinfrastructuur te detecteren. Met behulp van AI-technologieën zullen gebruikers diepgaande inzichten verkrijgen in het gebruik en de efficiëntie van laadpunten, wat niet alleen de klanttevredenheid zal verbeteren, maar ook zal bijdragen aan een duurzamere toekomst.

1.1 Situering

EV's spelen een centrale rol in het verminderen van de ecologische voetafdruk van het transport. Dit wordt ondersteund door verschillende statistieken en studies die de impact van EV's op de milieuvervuiling en CO₂-uitstoot aantonen. EV's stoten tijdens hun gebruik geen directe CO₂ uit, wat een significant voordeel is ten opzichte van traditionele voertuigen die op fossiele brandstoffen rijden. Volgens een studie van de Vlaamse overheid kan de ecologische voetafdruk van elektrische voertuigen tot 4 keer lager zijn dan die van conventionele voertuigen, afhankelijk van de energiebron die wordt gebruikt voor het opladen [8]. De ecologische voetafdruk van transport in Vlaanderen is een belangrijk aandachtspunt. De EV beschrijft de biologisch productieve oppervlakte die nodig is om de consumptie van een bevolking te ondersteunen. In Vlaanderen is de externe ecologische voetafdruk van hernieuwbare materialen geschat op ongeveer 90% van de totale consumptievoetafdruk, wat de noodzaak benadrukt om over te schakelen naar meer duurzame transportmiddelen zoals EV's [9].

De integratie van AI in EV's biedt aanzienlijke voordelen voor de duurzaamheid en efficiëntie van transport. AI-technologieën optimaliseren niet alleen de energiebeheersystemen van EV's, maar verbeteren ook de oplaadervaring door het analyseren van oplaadgedrag en het voorspellen van piekbelasting op het elektriciteitsnet. Dit draagt bij aan een betrouwbaardere en efficiëntere oplaadinfrastructuur, wat cruciaal is voor de acceptatie van EV's. Bovendien stelt AI EV's in staat om als energieopslag te fungeren, wat de interactie met het elektriciteitsnet optimaliseert en bijdraagt aan een duurzamer energiesysteem. Deze innovaties versterken de rol van EV's in het verminderen van de ecologische voetafdruk van transport, zoals eerder besproken in de context van Vlaanderen [10].

1.2 Probleemstelling

De groeiende vraag naar EV's en bijbehorende laadinfrastructuur vereist geavanceerde benaderingen om mogelijke frauduleuze activiteiten en abnormaal gebruik te identificeren en aan te pakken. Binnen het kader van deze uitdagingen in de snelgroeiende EV-markt en de evoluerende oplaadinfrastructuur luidt de onderzoeksvraag als volgt: **Hoe kunnen AI-ondersteunde datavisualisatietechnieken worden toegepast op uitgebreide datasets van oplaadpunten, locatiegegevens en gerelateerde gebeurtenissen om frauduleuze activiteiten zoals vehicle-to-grid-detectie, charge pass fraud en abnormaal gebruik te identificeren en te analyseren?**

Om een dieper begrip te verschaffen, wordt er onderzoek uitgevoerd naar deze specifieke frauduleuze praktijken die zich voordoen binnen het oplaadecosysteem.

In het geval van vehicle-to-grid (V2G) wordt fraudedetectie relevant wanneer een elektrisch voertuig is geparkeerd maar een significant percentage van de batterijcapaciteit verliest. Dit fenomeen kan wijzen op het ongeautoriseerd gebruik van de voertuigbatterij voor externe doeleinden, zoals het voeden van het elektriciteitsnet.

Charge pass fraud doet zich voor wanneer een laadtransactie wordt geregistreerd met een laadpas, maar de batterij van het voertuig vertoont geen tekenen van opladen. Dit kan duiden op misbruik van laadpassen, waarbij transacties worden vastgelegd zonder daadwerkelijke stroomoverdracht naar het voertuig.

Abnormaal gebruik verwijst naar situaties waarin het daadwerkelijke laadgedrag sterk afwijkt van voorspelde patronen. Verschillende factoren, zoals onverwachte drain, koude weersomstandigheden en andere omgevingsvariabelen, kunnen leiden tot abnormaal gebruik.

Traditionele datavisualisatietechnieken kunnen inzicht geven in patronen en afwijkingen, maar schieten tekort bij het detecteren van complexe en subtiele anomalieën die kenmerkend zijn voor frauduleus gedrag. Dit onderzoek richt zich daarom op de integratie van AI, specifiek machine learning-technieken zoals isolation forests, in datavisualisatieprocessen. Deze AI-gestuurde benadering maakt het mogelijk om diepere en minder voor de hand liggende patronen te ontdekken die met traditionele methoden over het hoofd worden gezien. Door AI-ondersteunde datavisualisatietechnieken toe te passen, kunnen we niet alleen patronen in het gebruik van de oplaadinfrastructuur en gedragskenmerken van gebruikers ontdekken, maar ook deze specifieke frauduleuze praktijken effectiever onderscheiden en aanpakken. Dit zal niet alleen bijdragen aan de optimalisatie van de laadinfrastructuur, maar ook aan het waarborgen van de integriteit en efficiëntie van het EV-oplaadecosysteem.

1.3 Doelstellingen

Het hoofddoel van dit onderzoek is het ontwikkelen van AI-ondersteunde datavisualisatietechnieken binnen een gebruiksvriendelijke webapplicatie, door middel van de implementatie van machinelearningalgoritmen voor het detecteren van frauduleuze activiteiten, zoals 'vehicle-to-grid'-detectie, 'charge pass fraud', en abnormaal gebruik, en het visualiseren van deze activiteiten om gebruikers te ondersteunen bij het identificeren en analyseren van fraude binnen de context van de groeiende vraag naar elektrische voertuigen en evoluerende oplaadinfrastructuur. In het kader hiervan werd deze masterproef uitgevoerd voor EEVEE Mobility, met als doel de visualisaties van de frauduleuze activiteiten te integreren in hun bestaande platform.

De deeldoelstellingen omvatten:

- Implementatie van fraudedetectie-algoritmen:

Het implementeren van AI-algoritmen gericht op het detecteren van frauduleuze activiteiten, met specifieke aandacht voor 'vehicle-to-grid'-detectie, 'charge pass fraud', en abnormaal gebruik binnen de dataset van oplaadpunten.

- Ontwikkeling van visualisatiemethoden voor fraude-identificatie met uitlegfunctie:

Het ontwikkelen van visualisatiemethoden die niet alleen frauduleus gebruikersgedrag in kaart brengen binnen de oplaadinfrastructuur, maar ook inzicht geven in waarom bepaalde activiteiten als verdacht worden beschouwd. Bijvoorbeeld, als het isolation forest-algoritme een activiteit als een outlier detecteert, wordt dit direct in de visualisatie uitgelegd. Hierdoor kunnen werkgevers die het EEVEE Mobility platform gebruiken om de EV fleets van hun werknemers te beheren, verdachte activiteiten niet alleen identificeren, maar ook begrijpen waarom ze als verdacht worden gemarkeerd.

De voorwaarden waarbij de opdrachtgever het probleem als opgelost beschouwt:

- Operationele webapplicatie:

De webapplicatie moet operationeel zijn, waarbij gebruikers tegen begin september 2024 in staat zijn om interactieve visualisaties te genereren. De visualisaties worden geïntegreerd in het reeds bestaande EEVEE-platform. Dit platform biedt inzicht in laadkosten, laadverbruik en andere relevante gegevens, die nader worden geanalyseerd in de hoofdstukken betreffende de implementatie van de webapplicatie. Aan deze bestaande gegevens worden visualisaties van frauduleuze activiteiten toegevoegd, waarbij bijvoorbeeld een waarschuwingssymbool wordt weergegeven naast een reeks gegevens.

1.4 Methode

Werkpakket 1: Literatuurstudie

Een grondige literatuurstudie vormt het fundament van dit onderzoek. In werkpakket 1 (WP1) worden literatuurbronnen systematisch verzameld en geanalyseerd om een uitgebreide basis voor het onderzoek te creëren. De aanpak van het literatuuronderzoek omvat een zorgvuldige selectie en evaluatie van relevante publicaties. Hierbij ligt de focus op verschillende deelgebieden: machinelearningalgoritmen (taak 1.1), machinelearningframeworks (taak 1.2), ethische aspecten (taak 1.3) en relevante technologieën zoals frontend en backend frameworks (taak 1.4). Elk van deze deelgebieden wordt afzonderlijk onderzocht om een volledig en nauwkeurig overzicht van de bestaande kennis en technieken te waarborgen.

Werkpakket 2: Voorbereidingsfase

In WP2 werd gestart met het opstellen van een gedetailleerde onderzoeksopzet en planning, waarbij een GANTT-chart werd gebruikt om de verschillende taken en hun tijdslijnen overzichtelijk in kaart te brengen (taak 2.1). Na het vaststellen van de planning werd een database verkregen die als basis diende voor verdere analyses (taak 2.2). In deze fase moesten nog diverse voorbereidende handelingen worden uitgevoerd om de database volledig operationeel te maken. Dit omvatte onder andere het indexeren van kolommen om de prestaties van queries te optimaliseren, evenals het aanmaken van foreign keys om de relaties tussen verschillende datatabellen te waarborgen.

Werkpakket 3: Uitvoeringsfase

In WP3 werd gestart met een grondige visuele data-analyse (taak 3.1), waarmee belangrijke inzichten in de dataset werden verkregen en mogelijke patronen en anomalieën konden worden geïdentificeerd. Deze analyse vormde de basis voor de daaropvolgende ontwikkeling van het machine learning isolation forest-algoritme (taak 3.2), dat speciaal werd ontworpen om efficiënt afwijkingen in de data te detecteren. Na de succesvolle implementatie van het algoritme, werd er een API ontwikkeld om de data op een gestroomlijnde manier beschikbaar te stellen via specifieke endpoints (taak 3.3). Deze API speelde een cruciale rol in het verbinden van de backend met de frontend, zodat de data toegankelijk was voor verdere verwerking. De laatste stap in WP3 was de ontwikkeling van een bijhorende frontend webapplicatie (taak 3.4), die speciaal was ontworpen om de geanalyseerde data te consumeren en op een gebruiksvriendelijke manier te visualiseren. Hierdoor konden eindgebruikers eenvoudig interactie hebben met de data en de resultaten van de analyses inzien, wat de bruikbaarheid en toegankelijkheid van het systeem aanzienlijk vergrootte.

Werkpakket 4: Documentatie

Dit werkpakket is essentieel voor het delen van de onderzoeksresultaten. Elk onderdeel draagt bij aan het presenteren en verdedigen van het onderzoek waardoor de wetenschappelijke waarde van het project wordt versterkt.

Taak 4.1: Poster

Het ontwikkelen van een overzichtelijke poster is een belangrijk onderdeel van kennisverspreiding. Dit werkpakket omvat het ontwerp en de creatie van een visueel aantrekkelijke poster die de belangrijkste aspecten van het onderzoek toont.

Taak 4.2: Scriptie

De scriptie vormt het meest uitgebreide document van het onderzoek. Hier worden de theoretische achtergrond, methodologie, resultaten, discussie en conclusies uitgebreid behandeld.

Taak 4.3: Verdediging

Hierin wordt de presentatie van het onderzoek voorbereid, met aandacht voor heldere communicatie van de belangrijkste bevindingen, ondersteund door visuele hulpmiddelen. Deze presentatie wordt gebruikt bij de verdediging van de masterproef. De verdediging van de masterproef is het sluitstuk van het onderzoeksproject. Er wordt aandacht besteed aan het verdedigen van de methodologie, resultaten en conclusies.

1.5 Vooruitblik

In de komende hoofdstukken wordt eerst een grondige literatuurstudie uitgevoerd naar machine learning-algoritmen en -frameworks om een solide basis te leggen voor de toegepaste technieken in deze masterproef. Vervolgens wordt er een literatuurstudie gepresenteerd over de ethische aspecten die relevant zijn voor het project, evenals een studie naar relevante technologieën en frontend/backend frameworks die van belang zijn voor de ontwikkeling van de webapplicatie. Hierna wordt er ingegaan op gegevensbeheer en -analyse, waarbij de verwerking en interpretatie van data centraal staan. Vervolgens volgt de ontwikkeling van de webapplicatie, waarin de implementatie van de voorgestelde oplossingen wordt behandeld. Ten slotte worden de bevindingen besproken in een nabespreking en wordt de masterproef afgesloten met een conclusie die de belangrijkste resultaten samenvat en de centrale onderzoeksvraag beantwoordt.

Hoofdstuk 2

Bronnenstudie

2.1 Inleiding

In de wereld van EV's is het essentieel om frauduleuze activiteiten en anomalieën te detecteren om de integriteit van laadtransacties te waarborgen. Diverse machinelearningalgoritmen bieden mogelijkheden voor deze detectie. K-means clustering groepeerde data in clusters op basis van afstandscriteria [11], DBSCAN identificeert clusters op basis van dichtheid [12], isolation forests detecteren afwijkingen in data door te kijken naar de scheiding van data punten [13], proces mining analyseert en visualiseert processen op basis van event logs [14], en neurale netwerken zijn geavanceerde modellen die complexe patronen kunnen leren en voorspellen [15]. In deze vergelijkende analyse worden de prestaties van deze algoritmen onderzocht voor de detectie van frauduleuze EV-activiteiten en anomalieën. Om een grondig inzicht te krijgen in de voor- en nadelen van elk algoritme, is er literatuuronderzoek uitgevoerd. Verschillende wetenschappelijke artikelen en technische documentatie zijn geanalyseerd om een objectief beeld te vormen van de capaciteiten, beperkingen, en toepassingsgebieden van de verschillende machine learning algoritmen.

2.2 Vergelijking machinelearningalgoritmen

Algoritme	Voordelen	Nadelen	Toepassing op EV-data
K-means clustering	Eenvoudig te implementeren, efficiënt, interpreteerbare clusters	Vereist vooraf gedefinieerd aantal clusters, gevoelig voor outliers	Clustering van laadgedrag, identificatie van laadprofielen
DBSCAN	Geen van tevoren kennis van aantal clusters nodig, robuust tegen outliers	Kan traag zijn voor grote datasets, minder interpreteerbare clusters	Detectie van afwijkende laadpatronen, identificatie van "vehicle-to-grid"
Isolation forests	Zeer efficiënt in het detecteren van anomalieën, geen training nodig	Minder nauwkeurig dan andere algoritmen, gevoelig voor hyperparameter tuning	Detectie van frauduleuze laadtransacties, identificatie van "laadpasfraude"
Proces mining	Ontdekking van processen uit event logs, kan anomalieën in processen detecteren	Vereist event logs met gedetailleerde informatie	Analyse van laadsessies, detectie van onregelmatige laadpatronen
Neurale netwerken	Zeer flexibel en nauwkeurig, kunnen complexe patronen leren	Complexer te trainen en interpreteren, vereist veel data	Voorspelling van energieverbruik, detectie van complexe frauduleuze schema's

Tabel 1: Vergelijkingstabel van machinelearningalgoritmen [11-15]

Tabel 1 is samengesteld op basis van de informatie uit bronnen 11 tot en met 15. Voor meer gedetailleerde informatie over elk algoritme, zie de volgende paragrafen.

K-means Clustering

K-means clustering is een veelgebruikt ongesuperviseerd machine learning algoritme dat ongelabelde datasets groepeerd in verschillende clusters [16] [17]. Het algoritme probeert de data punten binnen een cluster zo gelijkaardig mogelijk te maken, terwijl de clusters onderling zo verschillend mogelijk zijn [18]. K-means is een centroïde-gebaseerd algoritme waarbij elke cluster geassocieerd wordt met een centroïde. Het doel is om de som van de afstanden tussen datapunten en hun overeenkomstige clusters te minimaliseren. Het algoritme neemt de ongelabelde dataset als input, verdeelt deze in k clusters en herhaalt dit proces totdat de beste clusters gevonden zijn. De waarde van k moet vooraf bepaald worden [17].

K-means wordt vaak gebruikt voor klantsegmentatie, data-analyse, beeldanalyse en semi-superviseerd leren [18].

K-means is bijzonder geschikt voor eenvoudige clustering omdat het algoritme relatief snel convergentie bereikt en eenvoudig te implementeren is. Het biedt een duidelijke structuur door clusters te definiëren op basis van hun afstand tot de centroiden, wat het ideaal maakt voor situaties waarbij een snelle en interpreteerbare clustering gewenst is. Bovendien is de efficiëntie van K-means in het omgaan met grote datasets een belangrijk voordeel, zolang de clusters globaal rond de centroiden zijn en de waarde van k correct is ingesteld.

DBSCAN

DBSCAN is een populair clustering algoritme dat wordt gebruikt voor het groeperen van punten in een dataset op basis van hun dichtheid en onderlinge afstand [19] [20] [21]. Het algoritme partitioneert de data in dichte regio's van punten die gescheiden worden door minder dichte gebieden [21]. DBSCAN definieert clusters als gebieden in de dataset waar veel punten dicht bij elkaar liggen, terwijl punten die ver van een cluster af liggen worden beschouwd als uitschieters of ruis [21].

Het algoritme heeft twee belangrijke parameters: ϵ (de maximale afstand tussen twee punten om als buren te worden beschouwd) en $\min_samples$ (het minimale aantal punten dat vereist is voor een punt om als kernpunt te worden beschouwd) [19] [20] [22]. DBSCAN kan clusters van willekeurige vorm identificeren en is robuust tegen ruis en uitschieters in de data [19] [22]. Het algoritme is echter gevoelig voor de keuze van de parameters en werkt niet goed met datasets die variërende dichtheden hebben [22].

DBSCAN heeft verschillende toepassingen, zoals anomalie detectie, satellietbeeldverwerking en kristallografie [22]. Hoewel het een krachtig algoritme is, is het soms moeilijker te tunen dan andere clustering algoritmes zoals K-Means [20].

DBSCAN is robuuster tegen outliers omdat het punten die niet voldoen aan de dichtheidscriteria als ruis beschouwt, wat helpt om de impact van uitschieters te minimaliseren. Echter, het algoritme kan traag zijn voor grote datasets, vooral omdat het de afstand tussen elk paar punten moet berekenen om te bepalen of ze tot hetzelfde cluster behoren. Dit kan leiden tot aanzienlijke rekentijd en geheugengebruik naarmate de dataset groter wordt.

Isolation forests

Isolation Forest is een geavanceerd algoritme voor anomaliedetectie, ontwikkeld door Fei Tony Liu en zijn collega's in 2008. Het algoritme werkt door gegevens te partitioneren in binaire bomen, waarbij het zich richt op het isoleren van afwijkende gegevenspunten. Anomalieën, die doorgaans zeldzaam en afwijkend zijn, vereisen minder splitsingen om geïsoleerd te worden, wat betekent dat ze sneller kunnen worden gedetecteerd. Dit gebeurt door willekeurig geselecteerde kenmerken en splitsingswaarden te gebruiken, wat resulteert in een lineaire

tijdcomplexiteit en een laag geheugengebruik, waardoor het bijzonder geschikt is voor grote en hoge-dimensionale datasets. Het algoritme berekent een anomalie-score op basis van de lengte van het pad dat een gegevenspunt door de boom aflegt; kortere paden duiden op een grotere waarschijnlijkheid dat het punt een anomalie is [23] [24].

Hoewel Isolation Forests efficiënt zijn in het detecteren van anomalieën, kunnen ze minder nauwkeurig zijn dan andere algoritmen omdat ze afhankelijk zijn van willekeurige splitsingen en kenmerken. Dit kan leiden tot variabiliteit in de resultaten en minder precisie bij het identificeren van subtiele anomalieën.

Proces mining

Process mining is een verzameling technieken en methoden die organisaties in staat stelt om hun werkprocessen te analyseren en te optimaliseren door gebruik te maken van event logs. Dit proces begint met het verzamelen van gegevens over de activiteiten die binnen een organisatie plaatsvinden, vaak vastgelegd in digitale systemen. De drie belangrijkste vormen van proces mining zijn:

1. **Discovery:** Hierbij wordt een model gecreëerd op basis van de event logs, zonder voorafgaande informatie.
2. **Conformity checking:** Dit houdt in dat de werkelijkheid zoals geregistreerd in de logs wordt vergeleken met een bestaand model om te controleren op overeenkomsten en afwijkingen.
3. **Enhancement:** Dit betreft het verbeteren van een bestaand procesmodel door inzichten te verkrijgen uit de event logs, zoals doorlooptijden en frequenties, om knelpunten te identificeren en de efficiëntie te verhogen.

Het doel van proces mining is om de efficiëntie van processen te verbeteren, kosten te verlagen en de complexiteit te verminderen, wat kan leiden tot significante operationele verbeteringen in verschillende sectoren zoals de gezondheidszorg, industrie en overheid [25] [26].

Process mining kan bijzonder nuttig zijn voor het analyseren van laadsessies en het detecteren van onregelmatige laadpatronen. Door de gedetailleerde event logs van laadstations te analyseren, kunnen afwijkingen van de verwachte laadprocessen snel worden geïdentificeerd. Dit maakt het mogelijk om onregelmatigheden in laadgedrag of technische problemen tijdig te signaleren en te verhelpen. Bovendien kunnen de verkregen inzichten helpen bij het optimaliseren van laadprocessen en het verbeteren van de algehele efficiëntie van laadnetwerken, wat cruciaal is voor het effectief beheren van energiebronnen en het verhogen van klanttevredenheid.

Neurale netwerken

Neurale netwerken zijn geïnspireerd op de structuur en werking van het menselijke brein en zijn ontworpen om complexe taken uit te voeren door middel van een netwerk van onderling verbonden 'neuronen'. Deze kunstmatige netwerken bestaan uit verschillende lagen: een inputlaag, verborgen lagen en een outputlaag. In de inputlaag worden gegevens, zoals afbeeldingen of tekst, ingevoerd. De verborgen lagen verwerken deze gegevens door verschillende kenmerken te analyseren, waarna de outputlaag een antwoord genereert. Een belangrijk kenmerk van neurale netwerken is hun vermogen om te leren en zich aan te passen door verbindingen tussen neuronen te versterken of te verzwakken, wat hen in staat stelt om patronen te herkennen en beslissingen te nemen op basis van grote hoeveelheden data. Dit maakt ze bijzonder nuttig in toepassingen zoals gezichtsherkenning, spraakherkenning en zelfrijdende voertuigen [27] [28].

Neurale netwerken zijn flexibel en nauwkeurig omdat ze complexe patronen in data kunnen herkennen door hun gelaagde structuur en aanpassingsvermogen. Deze veelzijdigheid maakt ze krachtig, maar tegelijkertijd is het trainen van deze netwerken vaak tijdrovend en moeilijk te interpreteren, doordat de interne werking vaak een 'black box' blijft.

Ter conclusie, voor eenvoudige clustering is K-means een goede keuze. DBSCAN is robuuster tegen outliers, maar kan traag zijn voor grote datasets. Isolation forests zijn efficiënt in het detecteren van anomalieën, maar minder nauwkeurig dan andere algoritmen. Proces mining is nuttig voor het analyseren van laadsessies en het detecteren van onregelmatige laadpatronen. Neurale netwerken zijn zeer flexibel en nauwkeurig, maar complexer te trainen en interpreteren.

2.3 Vergelijking machine learning frameworks

In dit hoofdstuk van deze bronnenstudie wordt de vergelijking gemaakt tussen verschillende machine learning frameworks die worden toegepast voor de detectie van frauduleuze EV-activiteiten en anomalieën. Een framework biedt de structuur en tools om machine learning algoritmen te implementeren en te trainen. De onderzochte frameworks zijn TensorFlow, een uitgebreid en flexibel framework voor machine learning [29], TensorFlow.js, een versie van TensorFlow geoptimaliseerd voor webapplicaties [30], Brain.js, een lichtgewicht framework gericht op eenvoudige neurale netwerken [31], en PyTorch, een krachtig en gebruiksvriendelijk framework met een sterke focus op onderzoek en ontwikkeling [32]. Deze vergelijkende analyse belicht de voor- en nadelen van elk framework en hun geschiktheid voor specifieke toepassingen op EV-data.

Om een grondig inzicht te krijgen in de voor- en nadelen van elk framework, is er literatuuronderzoek uitgevoerd. Verschillende artikelen zijn geanalyseerd om een objectief

beeld te vormen van de capaciteiten, beperkingen, en toepassingsgebieden van TensorFlow, TensorFlow.js, Brain.js, en PyTorch.

Framework	Voordelen	Nadelen	Toepassing op EV-data
TensorFlow	Populair, breed scala aan libraries, krachtig, goed gedocumenteerd	Complex, kan traag zijn voor sommige toepassingen	Zeer geschikt voor alle beoogde analyses (DBSCAN, isolation forests, proces mining)
TensorFlow.js	Gebaseerd op TensorFlow, kan in JavaScript worden gebruikt, eenvoudiger te implementeren in webapplicaties	Beperkte functionaliteit in vergelijking met TensorFlow, minder flexibel	Geschikt voor DBSCAN en isolation forests, minder geschikt voor proces mining
Brain.js	Lichtgewicht, eenvoudig te gebruiken, gericht op neurale netwerken	Beperkte functionaliteit, minder geschikt voor complexe analyses	Mogelijk geschikt voor DBSCAN, maar minder krachtig dan TensorFlow of PyTorch
PyTorch	Flexibel, gebruiksvriendelijk, krachtig, goed gedocumenteerd	Iets minder populair dan TensorFlow, kan traag zijn voor sommige toepassingen	Zeer geschikt voor alle beoogde analyses (DBSCAN, isolation forests, proces mining)

Tabel 2: Vergelijkingstabel van machine learning frameworks [29-32]

Tabel 2 is samengesteld op basis van de informatie uit bronnen 29 tot en met 32. Voor meer gedetailleerde informatie over elk framework, zie de volgende paragrafen.

TensorFlow

TensorFlow is een open-source platform voor machine learning, ontwikkeld door het Google Brain-team. Het werd voor het eerst uitgebracht in 2015 en is ontworpen om zowel onderzoekers als ontwikkelaars te ondersteunen bij het bouwen en implementeren van machine learning-modellen. TensorFlow biedt een uitgebreide ecosysteem van tools en bibliotheken, waaronder TensorFlow Lite voor mobiele en embedded apparaten, en TensorFlow.js voor het uitvoeren van modellen in de browser. Het platform maakt gebruik van een hiërarchie van API's, waarbij de hoge-niveau API's zoals Keras gebruikers in staat stellen om eenvoudig modellen te bouwen en te trainen. TensorFlow is veelzijdig en kan worden toegepast in verschillende programmeertalen, waaronder Python, JavaScript, en C++ [33] [34] [35].

TensorFlow.js

TensorFlow.js is een krachtig machine learning platform dat het mogelijk maakt om ML-modellen te ontwikkelen, trainen en inzetten in JavaScript, zowel in de browser als in Node.js. Met intuïtieve APIs kan je direct aan de slag met ML in JavaScript, zonder de complexiteit van tensorsmanipulatie. TensorFlow.js biedt ook ondersteuning om bestaande Python TensorFlow-modellen om te zetten en in de browser te draaien [36]. Dankzij WebGL-versnelling en binding met de C API levert TensorFlow.js indrukwekkende prestaties, waardoor het een serieuze contender is voor Python op het gebied van ML-ontwikkeling. Door de combinatie van JavaScript's eenvoud en TensorFlow's kracht, opent TensorFlow.js de deur voor JavaScript-ontwikkelaars om geavanceerde AI-toepassingen te bouwen, zowel client-side in de browser als server-side met Node.js [37].

Brain.js

Brain.js is een open-source JavaScript bibliotheek voor het bouwen van neurale netwerken. Het is ontworpen om gemakkelijk te gebruiken te zijn en biedt een eenvoudige API voor het maken en trainen van verschillende soorten neurale netwerken, waaronder feedforward netwerken, recurrente netwerken en convolutional netwerken [38] [39].

Met Brain.js kun je machine learning modellen bouwen die gebruikt kunnen worden voor een verscheidenheid aan taken, zoals tekstclassificatie, beeldherkenning en tijdreeksanalyse. De bibliotheek is volledig geschreven in JavaScript en draait zowel in de browser als aan de serverkant met Node.js [38] [39].

Brain.js is een alternatief voor TensorFlow.js en stelt ontwikkelaars in staat om machine learning toe te passen in de browser. Dit biedt voordelen zoals verbeterde privacy, snellere reactietijden, offline functionaliteit en real-time interactiviteit [39]. Hoewel Brain.js een eenvoudige manier biedt om neurale netwerken te implementeren in JavaScript, heeft het ook enkele beperkingen zoals prestatieproblemen, beperkte functionaliteit en schaalbaarheidsuitdagingen in vergelijking met andere machine learning bibliotheken [39] [40].

PyTorch

PyTorch is een open-source machine learning bibliotheek gebaseerd op de Torch bibliotheek, ontwikkeld door Meta AI (voorheen Facebook AI Research). Het wordt veel gebruikt voor toepassingen zoals computer vision en natuurlijke taalverwerking. PyTorch biedt twee belangrijke functies: tensor computing (zoals NumPy) met sterke acceleratie via GPU's, en diepe neurale netwerken gebouwd op een tape-gebaseerd automatisch differentiatie systeem [41].

PyTorch is één van de twee populairste machine learning bibliotheken naast TensorFlow. Het biedt gratis en open-source software onder de aangepaste BSD-licentie. Hoewel de Python interface meer gepolijst is en de primaire focus van ontwikkeling, heeft PyTorch ook een C++ interface [41].

PyTorch wordt ondersteund op grote cloud platforms zoals Amazon Web Services, Google Cloud en Microsoft Azure, waardoor eenvoudige ontwikkeling en schaalvergroting mogelijk is [42]

[43]. Het heeft ook een rijk ecosysteem van tools en bibliotheken die PyTorch uitbreiden en ondersteuning bieden voor ontwikkeling in onder andere computer vision en natuurlijke taalverwerking [44].

TensorFlow is de beste keuze voor maximale flexibiliteit en kracht. TensorFlow.js is een goede keuze als de analyse geïntegreerd moet worden in een webapplicatie met JavaScript. Brain.js is een eenvoudige optie voor basale analyses met neurale netwerken. PyTorch is een krachtig en gebruiksvriendelijk alternatief voor TensorFlow.

Gezien de complexiteit van de analyses en de noodzaak voor integratie in een webapplicatie, worden TensorFlow en TensorFlow.js als de beste keuzes beschouwd binnen het kader van deze masterproef. TensorFlow biedt maximale flexibiliteit en kracht, terwijl TensorFlow.js de implementatie in een webapplicatie vereenvoudigt. TensorFlow kan gebruikt worden voor de data-analyse en het trainen van de machine learning modellen, terwijl TensorFlow.js de modellen kan implementeren in de webapplicatie.

2.4 Verminderen van false positives in fraudedetectie

Dit deel richt zich op een kritische bespreking van recente onderzoeken die zich specifiek richten op het verminderen van false positives in fraudedetectie. Het verminderen van false positives is een cruciale uitdaging in fraudedetectie, omdat deze foutieve alarmsignalen de geloofwaardigheid van het detectiesysteem kunnen aantasten. Verschillende benaderingen en technieken worden onderzocht.

In een studie uit [45] wordt de "red flag"-benadering gecombineerd met procesmining om false positives in fraudedetectie te verminderen. De "red flag"-benadering identificeert ongewone patronen, terwijl procesmining de processen binnen een organisatie visualiseert (zie sectie 2.2 voor een gedetailleerde uitleg van proces mining). Door deze te combineren, kunnen frauduleuze processen met bijbehorende "red flags" worden opgespoord. De auteurs slaagden erin om 15 van de 31 fraudegevallen in hun dataset te identificeren met een false positive rate van 0.37%. Het cijfer van 15 uit 31 kan op het eerste gezicht misschien niet indrukwekkend lijken, maar er zijn een paar belangrijke zaken om in overweging te nemen. De onderzoekers melden dat hun false positive rate (het percentage gevallen dat ten onrechte als fraude werd aangemerkt) slechts 0,37% is. Dit is aanzienlijk lager dan wat in vergelijkbare onderzoeken wordt gerapporteerd. Dit betekent dat hoewel ze misschien niet elk fraudegeval hebben opgespoord, ze heel nauwkeurig zijn in de gevallen die ze wel als fraude hebben aangemerkt. Het onderzoek suggereert ook dat de combinatie van technieken effectief is in het verminderen van het aantal onterechte fraudemeldingen, wat waardevol kan zijn voor bedrijven die willen voorkomen dat ze te veel tijd en middelen besteden aan het onderzoeken van false positives. Hoewel dus 15 van de 31 fraudegevallen lijkt alsof het slechts een gedeeltelijk succes is, is dit resultaat toch indrukwekkend gezien de lage false positive rate. De methode biedt een goede balans tussen het identificeren van fraude en het vermijden van onterechte verdenkingen.

In een onderzoek beschreven in [46] wordt er gefocust op het gebruik van machine learning om false positives te verminderen. De auteurs trainden een machine learning model op basis van historische data om frauduleuze transacties te identificeren. Het model behaalde een betere prestatie dan traditionele detectiemethoden en reduceerde het aantal false positives met 30%.

De bestudeerde papers tonen verschillende methodes om false positives in fraudedetectie te verminderen. Combineren van detectiemethoden door "red flags" te combineren met procesmining of machine learning, kan de nauwkeurigheid van fraudedetectie worden verhoogd. Machine learning modellen kunnen worden getraind om frauduleuze patronen te herkennen en false positives te verminderen. Hoewel deze methodes in deze masterproef niet rechtstreeks worden toegepast, bieden ze wel waardevolle inzichten die in toekomstige projecten van Wisemen kunnen worden gebruikt. Deze technieken kunnen dienen als een nuttige basis voor het ontwikkelen van geavanceerdere systemen die effectiever zijn in het verminderen van false positives. Hierdoor kunnen toekomstige fraudedetectiegerelateerde projecten profiteren van de lessen en methodes die in deze onderzoeken worden besproken, zelfs als ze in dit werk niet direct worden geïmplementeerd.

2.5 Juridische aspecten

De AI-ondersteunende webapplicatie die in deze masterproef wordt ontwikkeld, richt zich op fraudedetectie binnen elektrische voertuigen (EV's) en de bijbehorende oplaadinfrastructuur. Het gebruik van het isolation forest algoritme om outliers te detecteren in een omvangrijke dataset plaatst deze applicatie binnen een specifiek risiconiveau volgens de AI Act.

De applicatie valt onder de categorie van systemen met een hoog risico om de volgende redenen:

1. Impact op fundamentele rechten: De detectie van frauduleuze activiteiten in laadkosten en V2G-toepassingen kan aanzienlijke gevolgen hebben voor de financiële belangen van zowel gebruikers als aanbieders van oplaadinfrastructuur. Onjuiste classificaties of beslissingen kunnen leiden tot onterechte kosten of zelfs juridische geschillen [47] [48].
2. Complexiteit van data: De grote dataset die wordt gebruikt voor het trainen van het isolation forest algoritme bevat diverse variabelen en kan complexiteit met zich meebrengen. Dit verhoogt het risico op vooringenomenheid in de data, wat kan resulteren in onjuiste conclusies of discriminatie [49] [50].
3. Noodzaak van menselijk toezicht: Gezien de potentiële impact van de beslissingen die door de applicatie worden genomen, is het essentieel dat er menselijk toezicht is. Dit waarborgt dat de uitkomsten van de AI-systemen worden gecontroleerd en gevalideerd voordat ze worden toegepast in de praktijk [50] [51].
4. regelgeving en compliance: De AI Act vereist dat systemen met een hoog risico voldoen aan strikte eisen op het gebied van risico-mitigatie, kwaliteitsborging van datasets en transparantie. Dit maakt het noodzakelijk om een goed gedefinieerd compliance-plan op te stellen dat deze vereisten adresseert [47] [49] [48].

Samenvattend, de webapplicatie voor fraudedetectie binnen EV's en oplaadinfrastructuur valt onder een hoog risiconiveau volgens de AI Act. Het is van cruciaal belang dat de ontwikkeling en implementatie van deze applicatie zorgvuldig worden uitgevoerd, met aandacht voor de juridische en ethische implicaties, om te voldoen aan de gestelde eisen en om de integriteit van het systeem te waarborgen.

2.6 Relevante technologieën en frameworks

Dit deel richt zich op de keuze van frameworks voor zowel de frontend als de backend van de webapplicatie. De keuze van het juiste framework is essentieel voor het succesvol ontwikkelen en implementeren van de webapplicatie, waarbij rekening wordt gehouden met zowel technische als strategische overwegingen. Dit deel biedt een gedetailleerde analyse van de overwegingen die hebben geleid tot de selectie van specifieke frameworks en benadrukt de voordelen en compatibiliteit van deze frameworks met betrekking tot het onderzoeksproject.

Voor het frontend framework is de beslissing genomen om Vue.js te gebruiken. Een cruciaal argument dat deze keuze motiveert, is de uitgebreide ondersteuning van Vue.js door Wisemen. Wisemen heeft Vue.js geïdentificeerd als het primaire frontend framework voor hun projecten. Deze beslissing komt niet alleen voort uit de technische mogelijkheden van Vue.js, maar ook uit de overtuiging dat het framework de beste oplossing biedt voor het ontwikkelen van moderne en responsieve gebruikersinterfaces. Het gebruik van Vue.js in het onderzoek is niet alleen een technische keuze, maar ook een strategische beslissing om te profiteren van de ondersteuning en expertise die Wisemen biedt.

In de wereld van webapplicatieontwikkeling spelen backend frameworks een cruciale rol bij het bouwen van krachtige en schaalbare software. In deze vergelijking worden er drie frameworks onderzocht: Laravel (PHP), Express/NestJS (Node), en FastAPI (Python), met betrekking tot hun geschiktheid voor het onderzoek. Er is specifiek gekozen voor deze 3 frameworks aangezien ze allemaal reeds worden gebruikt intern bij Wisemen waardoor het bedrijf ondersteuning en expertise kan bieden tijdens de ontwikkeling van de webapplicatie.

Laravel, een PHP-gebaseerd framework, geniet populariteit en uitgebreide documentatie dankzij een grote community [52]. Met zijn MVC-architectuur bevordert het gestructureerde code en scheiding van belangen [53]. Laravel biedt een uitgebreide set ingebouwde functies, waaronder ondersteuning voor authenticatie, autorisatie en databasebeheer [54]. Hoewel het snel en schaalbaar is, kan de steile leercurve beginners ontmoedigen [55] [56].

Express en NestJS zijn JavaScript-gebaseerde frameworks, aantrekkelijk voor ontwikkelaars bekend met deze taal. Ze zijn lichtgewicht, modulair en geoptimaliseerd voor snelheid en efficiëntie. De actieve community biedt ondersteuning, maar ze hebben minder ingebouwde functionaliteit in vergelijking met Laravel. Handmatige configuratie kan ontwikkeling complexer maken.

FastAPI is een modern Python-gebaseerd framework dat zich richt op het bouwen van API's met hoge prestaties. Het biedt asynchrone programmeermogelijkheden, wat essentieel is voor het verwerken van grote hoeveelheden data in real-time, zoals bij het detecteren van fraude in EV-gegevens. FastAPI ondersteunt automatische documentatie en validatie van gegevens, wat de ontwikkeling versnelt en de betrouwbaarheid van de applicatie verhoogt [57] [58].

Bovendien integreert FastAPI naadloos met andere Python-bibliotheken, zoals Pandas en NumPy, die nuttig zijn voor data-analyse en -visualisatie. Dit maakt het een uitstekende keuze voor een project dat zich richt op het analyseren van complexe datasets en het detecteren van afwijkingen [57] [59].

- Laravel: Een goede keuze vanwege uitgebreide functionaliteit, een grote community en gedetailleerde documentatie.
- Express/NestJS: Te overwegen als een moderne JavaScript-stack en snelle prestaties belangrijk zijn.
- FastAPI: Te overwegen voor Python-gebaseerde oplossingen.

Een voordeel van het gebruik van Express/NestJS in combinatie met Vue.js als frontend framework is de naadloze integratie van JavaScript-technologieën over de gehele stack. Beide frameworks zijn JavaScript-gebaseerd, waardoor er geprofiteerd kan worden van een uniforme taal en architectuur. Deze consistentie vergemakkelijkt niet alleen de ontwikkeling, maar zorgt ook voor een soepele communicatie tussen de backend en frontend, wat resulteert in een gestroomlijnde ontwikkelingservaring.

Voor projecten gericht op Artificial Intelligence (AI) waarbij TensorFlow en Python worden gebruikt, biedt FastAPI specifieke voordelen. FastAPI, als een op Python gebaseerd framework, integreert naadloos met TensorFlow en biedt een stabiele en gestructureerde omgeving voor AI-ontwikkeling. Het gebruik van Python als de kern van zowel FastAPI als TensorFlow bevordert een consistente codebase en vergemakkelijkt de uitwisseling van gegevens tussen de webapplicatie en de AI-componenten.

Gezien de specifieke eisen en focus van het project, worden de voordelen die Express/NestJS in combinatie met Vue.js als frontend framework biedt, en het voordeel van FastAPI met TensorFlow en Python voor AI, als crucialer beschouwd dan de voordelen van Laravel. De naadloze integratie van JavaScript-technologieën over de gehele stack en de consistente Python-omgeving voor AI-ontwikkeling wegen zwaarder in dit scenario. Hierdoor is besloten Laravel direct uit te sluiten als optie, omdat de unieke voordelen die het biedt niet opwegen tegen de specifieke behoeften en efficiënte integratie die Express/NestJS met Vue.js en FastAPI met TensorFlow en Python kunnen bieden voor frontend- en AI-functionaliteiten.

Gezien de specifieke vereisten van het project, waaronder de noodzaak voor effectieve visualisatie van EV-gegevens, is besloten om FastAPI te gebruiken voor de backend en Vue.js voor de frontend. Deze keuze is verder onderbouwd door het feit dat Python gebruikt werd voor visuele datavisualisatie, wat de integratie en ontwikkeling van de applicatie vergemakkelijkt. De combinatie van deze frameworks biedt een krachtige en flexibele oplossing voor de uitdagingen van het project.

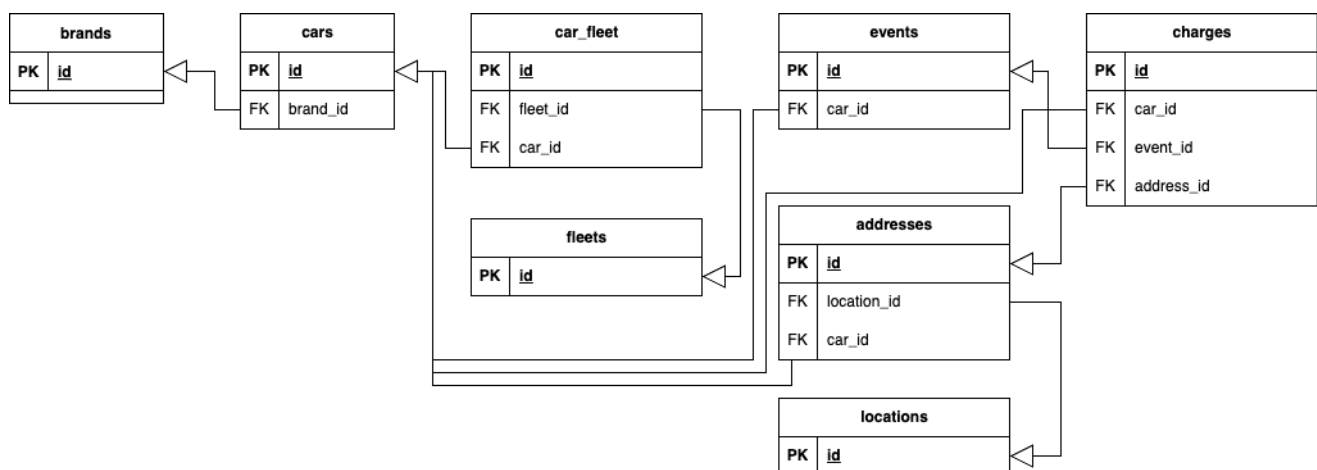
Hoofdstuk 3

Gegevensbeheer en analyse

In dit hoofdstuk wordt de database beschreven die als basis diende voor de verdere stappen van het onderzoek in deze masterproef. De eerste stap in dit proces was het ontvangen en analyseren van de EEVEE Mobility dataset. Deze dataset, aangeboden in een relationele MySQL-database, bevatte uitgebreide gegevens over elektrische voertuigen (EV's) en hun gebruik. Om de relevante data efficiënt te kunnen verwerken en te gebruiken voor verdere analyses, werd een subset van de volledige dataset geselecteerd en geoptimaliseerd. In de volgende secties wordt gedetailleerd beschreven hoe deze subset werd samengesteld, geanalyseerd en klaargemaakt voor gebruik in diverse toepassingen, zoals visuele data-analyse, machine learning-modellen en de webapplicatie.

3.1 EEVEE Mobility Database

Voor deze masterproef werd gewerkt met de EEVEE Mobility dataset, waarvan een diagram werd verkregen, zoals weergegeven in figuur 1. Deze diagram illustreert de high-level relaties tussen tabellen in de verkregen relationele MySQL-database.



Figuur 1: High-level ER diagram van de EEVEE dataset waarbij de relaties tussen de entiteiten getoont wordt

Een zorgvuldig door de auteur van deze masterproef geselecteerde subset van deze volledige dataset werd gekozen, waarbij de methodologie voor selectie werd ontwikkeld in overleg met Ing. Sebastiaan Vanspauwen en een collega binnen EEVEE Mobility. Voor dit onderzoek is een specifieke subset van data geselecteerd die kan helpen bij het detecteren van V2G-fraude, 'charge pass'-fraude en abnormaal gebruik. Een belangrijke bron van data is de 'events' tabel, waarin informatie is opgenomen over het type 'event', zoals 'parked', 'charging', en de bijbehorende batterijpercentages aan het begin ('percent_start') en einde ('percent_end') van het

'event'. Wanneer het type 'event' bijvoorbeeld 'parked' is en er een verschil is tussen 'percent_start' en 'percent_end', kan dit duiden op een verschil in batterijniveau gedurende het parkeerevent. Dit verschil kan een aanwijzing zijn voor mogelijke V2G-fraude, waarbij energie uit de batterij wordt gebruikt zonder toestemming.

Bij 'events' van het type 'charging' kan het ontbreken van een verschil in batterijpercentage wijzen op 'charge pass'-fraude, waarbij de laadtransactie mogelijk niet correct is geregistreerd of gemanipuleerd. Daarnaast zijn de locaties van de laadsessies meegenomen in de subset om te kunnen beoordelen of deze realistisch zijn.

De 'charges' tabel is geselecteerd om inzicht te krijgen in de laadkosten en het laadverbruik van verschillende laadsessies. Hiermee kunnen afwijkingen in het laadgedrag worden gedetecteerd, die mogelijk wijzen op abnormaal gebruik of andere vormen van fraude. Andere relevante tabellen, zoals 'brands', 'car_fleet', 'fleets', en 'cars', die reeds aanwezig zijn in het EEVEE Mobility platform, zijn eveneens opgenomen om extra context te bieden bij de geanalyseerde data en zo een vollediger beeld te krijgen van de situatie. De subset, te zien in figuur 2 tot en met 10, bevat alle essentiële data voor de initiële visuele data-analyse, het daaropvolgende toepassen van machine learning, het beschikbaar stellen van de data aan een frontend via een API en uiteindelijk de visualisatie van de data via een webapplicatie.

addresses	
☐ location_id	bigint unsigned
☒ car_id	bigint unsigned
☒ type	enum('home', 'work', 'public', 'other', 'invisible', 'parked')
☐ latitude	decimal(10,7)
☐ longitude	decimal(10,7)
☐ zipcode	varchar(255)
☐ city	varchar(255)
☐ country	varchar(255)
☐ first_charge_time	timestamp
☐ last_charge_time	timestamp
☐ created_at	timestamp
☐ updated_at	timestamp
☐ country_code	varchar(3)
☐ locale	varchar(6)
☒ id	bigint unsigned

Figuur 2: addresses tabel van de EEVEE dataset

brands	
name	varchar(255)
logo	varchar(255)
id	bigint unsigned

Figuur 3: brands tabel van de EEVEE dataset

car_fleet	
fleet_id	bigint unsigned
car_id	bigint unsigned
driver_id	bigint unsigned
created_at	timestamp
updated_at	timestamp
id	bigint unsigned

Figuur 4: car_fleet tabel van de EEVEE dataset

charges	
car_id	bigint unsigned
event_id	bigint unsigned
address_id	bigint unsigned
percent_added	double(8,2)
charge_added	double(8,2)
percent_started	double(8,2)
percent	double(8,2)
range_ideal	double(8,2)
range_estimated	double(8,2)
started_charging_at	timestamp
finished_charging_at	timestamp
price	double(8,2)
outside_temp	double(8,2)
status	enum('state', 'charging', 'finished')
type	enum('electric', 'fossil')
created_at	timestamp
updated_at	timestamp
charge_limit	int
estimated_finished_at	timestamp
id	bigint unsigned

Figuur 5: charges tabel van de EEVEE dataset

cars	
brand_id	bigint unsigned
user_id	bigint unsigned
account_id	bigint unsigned
charge_profile_id	bigint unsigned
vin	varchar(255)
model	varchar(255)
image	varchar(255)
battery_capacity	double(8,2)
battery_level	double(8,2)
odometer	double(8,2)
is_charging	tinyint(1)
type	enum('electric', 'hybrid', 'fossil')
created_at	timestamp
updated_at	timestamp
outside_temp	double(8,2)
inside_temp	double(8,2)
id	bigint unsigned

Figuur 6: car tabel van de EEVEE dataset

events	
carconnect_event_id	bigint unsigned
car_id	bigint unsigned
type	enum('driving', 'charging', 'refueling', 'parked', 'pluggedin', 'plugged_in', 'sleeping', 'in_service', 'offline')
started_at	timestamp
finished_at	timestamp
created_at	timestamp
updated_at	timestamp
distance_driven_start	double(8,2)
percent_start	double(8,2)
percent_end	double(8,2)
range_ideal_start	double(8,2)
range_ideal_end	double(8,2)
range_estimated_start	double(8,2)
range_estimated_end	double(8,2)
avg_outside_temp	double(8,2)
avg_inside_temp	double(8,2)
id	bigint unsigned

Figuur 7: events tabel van de EEVEE dataset

fleets	
created_at	timestamp
updated_at	timestamp
country	varchar(255)
city	varchar(255)
region	varchar(255)
postal_code	varchar(255)
currency_code	varchar(255)
currency_symbol	varchar(255)
id	bigint unsigned

Figuur 8: fleets tabel van de EEVEE dataset

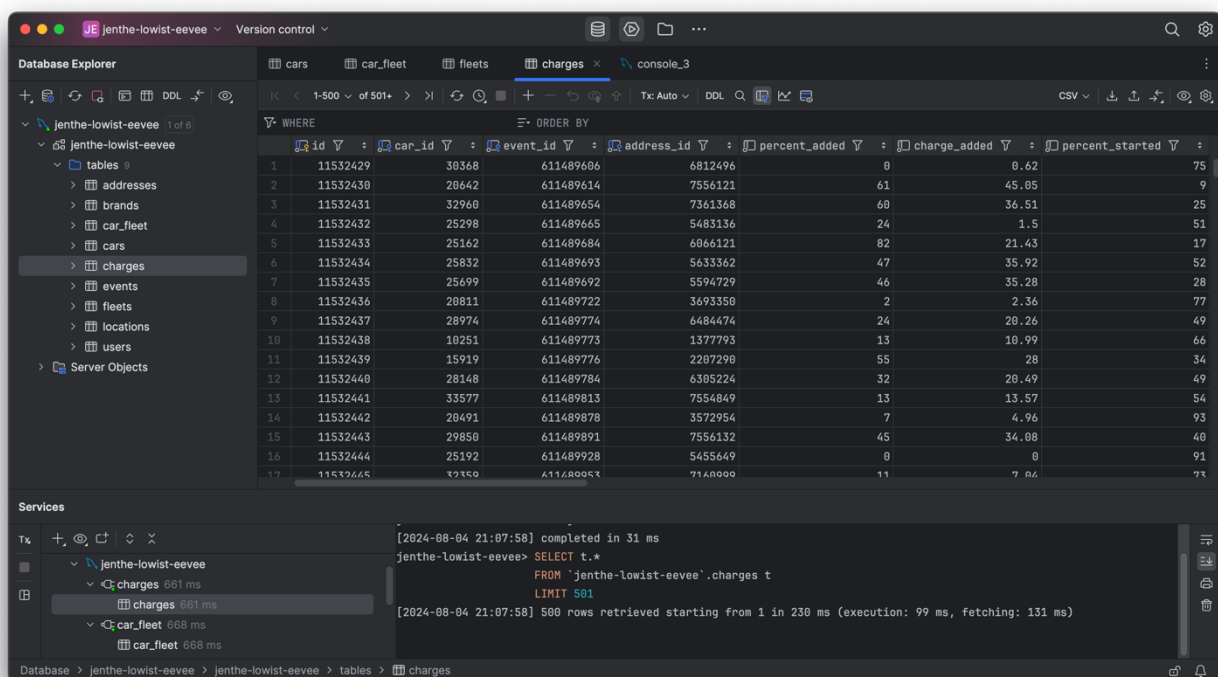
locations	
uuid	char(36)
name	varchar(255)
street	varchar(255)
postcode	varchar(255)
city	varchar(255)
state	varchar(255)
country	varchar(255)
country_code	varchar(255)
type	varchar(255)
timezone	varchar(255)
created_at	timestamp
updated_at	timestamp
latitude	decimal(10,7)
longitude	decimal(10,7)
id	bigint unsigned

Figuur 9: locations tabel van de EEVEE dataset

users	
currency_code	varchar(255)
currency_symbol	varchar(255)
uses_miles	tinyint(1)
created_at	timestamp
updated_at	timestamp
locale	varchar(10)
metric_temperature	enum('celsius', 'fahrenheit')
id	bigint unsigned

Figuur 10: users tabel van de EEVEE dataset

Na de selectie van de subset werd de uiteindelijke data geleverd via een MySQL-verbinding . Voor de interactie met de database werd gebruikgemaakt van JetBrains DataGrip, een krachtige cross-platform software tool voor zowel relationele als NoSQL-databases [60]. Een screenshot van deze software is te vinden in figuur 11.



Figuur 11: Screenshot van JetBrains DataGrip

Voordat de data geanalyseerd kon worden, moesten er foreign key-relaties worden aangemaakt tussen de verschillende tabellen. Een foreign key-relatie zorgt ervoor dat een kolom in een tabel verwijst naar de primaire sleutel van een andere tabel, waardoor gerelateerde data op een consistente manier kan worden gekoppeld en opgehaald. Hoewel deze foreign key-relaties al zichtbaar zijn in de diagram van figuur 1, waren ze bij de levering van de relationele database

nog niet aanwezig. Hiertoe werden bepaalde kolommen eerst geïndexeerd in een balanced tree. Deze kolommen waren noodzakelijk voor het opzetten van foreign key-relaties en werden tevens frequent opgevraagd, bijvoorbeeld via een WHERE-clausule in een SQL-query. Het indexeren van deze kolommen was cruciaal voor de prestatieverbetering van de database-interactie, aangezien de database tabellen bevatte met miljoenen rijen. Zonder deze database-optimalisatie zou het uitvoeren van de query, benodigd voor het aanmaken van foreign key-relaties, meerdere uren in beslag hebben genomen.

Deze stappen vormden een essentieel onderdeel van het voorbereidingsproces voor de latere fasen van het project, waarbij het cruciaal was dat de data efficiënt en effectief toegankelijk was voor verdere analyse en toepassing in de ontwikkelde webapplicatie.

3.2 Visuele data-analyse

De visuele data-analyse werd uitgevoerd met het doel om diepgaand inzicht te verkrijgen in de aard en betrouwbaarheid van de beschikbare data. Door middel van visuele representaties, zoals taartdiagrammen en geografische kaarten, werd de samenstelling van de dataset onderzocht om bijvoorbeeld te begrijpen welke merken van EV's het meest vertegenwoordigd waren en om eventuele anomalieën in de locatiegegevens van laadsessies te identificeren. Deze analyse was essentieel om te bepalen met wat voor data er gewerkt werd en om de integriteit van de dataset te waarborgen, wat cruciaal is voor toekomstige analyses en conclusies.

De data in de database werd visueel geanalyseerd, waarbij door middel van SQL queries de focus werd gelegd op bijvoorbeeld de merken van de opgenomen EV's. Dit werd gedaan om inzicht te krijgen in de verdeling van verschillende automerken binnen de database. Uit een taartdiagram in figuur 12 blijkt dat Tesla-voertuigen bijna 75 % van de database uitmaken. Dit is niet verrassend, aangezien er verschillende redenen zijn waarom EEEVEE Mobility zoveel gegevens over Tesla bezit:

1. Ondersteuning van Tesla: Tesla was een van de eerste EV-merken die EEEVEE Mobility ondersteunde [61] [62]. Tesla's integreren naadloos met het EEEVEE-platform, waardoor een groot aantal Tesla-bestuurders het platform reeds gebruiken.
2. Geavanceerde telematica-systemen: Tesla's zijn uitgerust met geavanceerde telematica-systemen die gedetailleerde voertuiggegevens verzamelen en delen [63]. Dit stelt EEEVEE in staat om zeer gedetailleerde data over laadsessies, energieverbruik, bereik en andere relevante parameters te verzamelen.

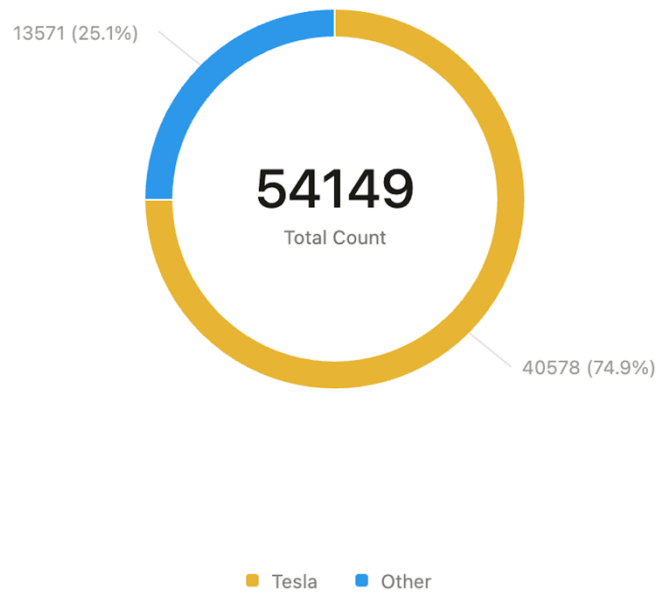
Tesla's zijn uitgerust met geavanceerde telematica-systemen die gedetailleerde voertuiggegevens verzamelen en delen. Dit stelt EEEVEE in staat om zeer gedetailleerde data over laadsessies, energieverbruik, bereik en andere relevante parameters te verzamelen.

Andere merken zijn ook in toenemende mate uitgerust met vergelijkbare telematica-systemen. Bijvoorbeeld, Ford heeft zijn telematica-systeem uitgebreid zodat het ook gegevens van voertuigen van andere merken kan verzamelen, wat de functionaliteit voor wagenparkbeheer vergroot [64]. Daarnaast zijn er bedrijven zoals Radius Telematics en Verizon Connect die geavanceerde telematica-oplossingen aanbieden die niet beperkt zijn tot één merk, maar geschikt zijn voor verschillende soorten voertuigen en merken [65] [66].

Dit toont aan dat hoewel Tesla een pionier is op het gebied van geavanceerde telematica, andere merken en oplossingen ook beschikbaar zijn die vergelijkbare gegevensverzameling en -analyse mogelijk maken.

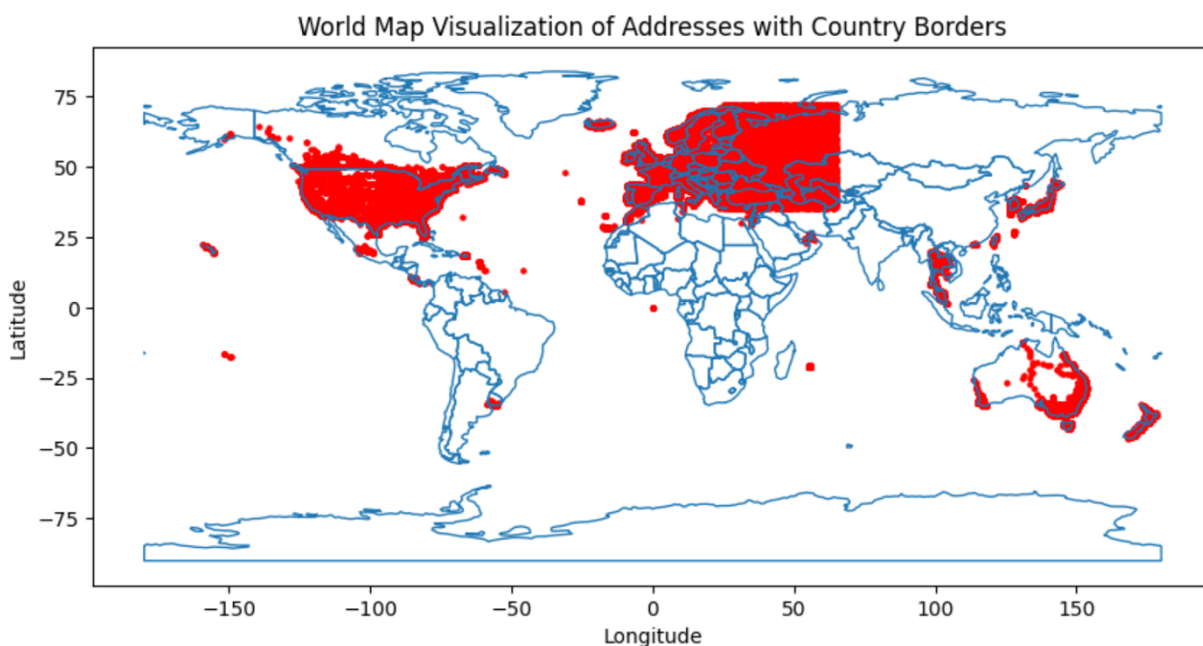
3. Populariteit van Tesla: Tesla is momenteel het populairste en best verkochte EV-merk in veel markten [64]. Met meer Tesla's op de weg genereert het merk logischerwijs meer data voor EEEVEE.

Hoewel Tesla momenteel de meerderheid van de data levert, is EEVEE voortdurend bezig met het toevoegen van meer merken en het verbeteren van de integratie met andere EV-fabrikanten [65]. Naarmate het bedrijf groeit, zal de datamix naar verwachting diverser worden.



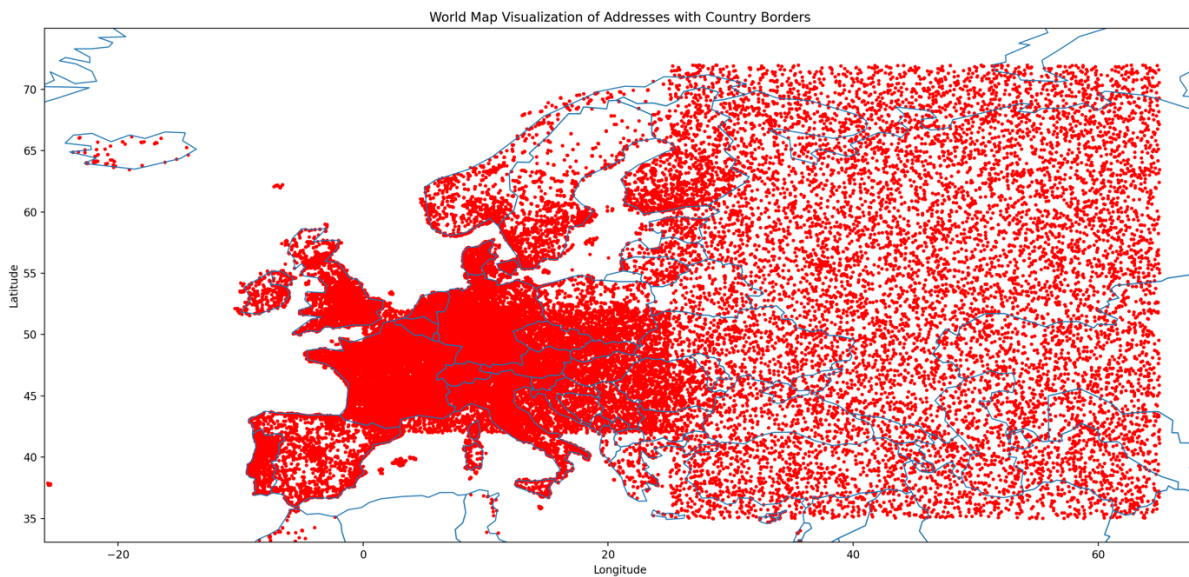
Figuur 12: Verdeling van automerken (n=54149). 74,9% van de auto's in de dataset is van het merk Tesla, terwijl de overige 25,1% andere merken vertegenwoordigen

De lengte- en breedtegraden van de laadsessies werden gevisualiseerd op een wereldkaart met behulp van de Geopandas-library. Dit werd gedaan om de betrouwbaarheid van de geografische gegevens te verifiëren en eventuele afwijkingen in de data op te sporen. Dit resulteerde in figuur 13.



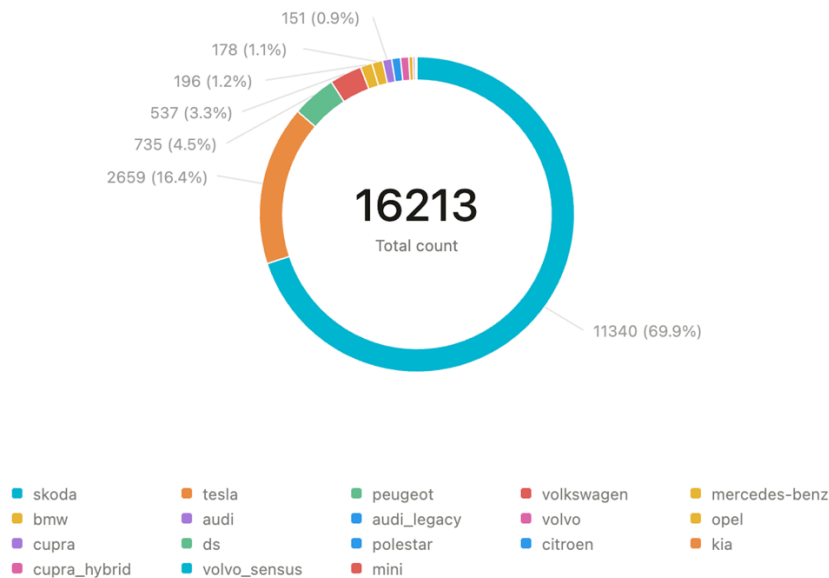
Figuur 13: Wereldkaart visualisatie van de oplaadsessies (voor het toepassen van de filter)

Uit deze visualisatie bleek dat laadsessies op diverse locaties wereldwijd plaatsvonden. Echter, er viel een perfect vierkante datapuntenwolk op rondom Rusland, die in meer detail te zien is in figuur 14.

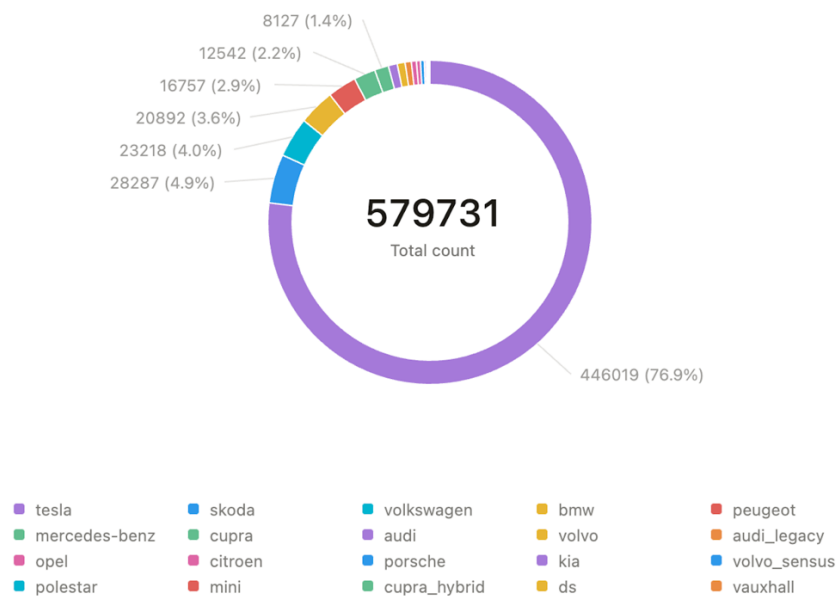


Figuur 14: Wereldkaart visualisatie van de oplaadsessies (gedetailleerde kijk op vierkante datapuntenwolk)

Na overleg met EEVEE Mobility bleek dit valse data te zijn, afkomstig van één van hun klanten, namelijk Skoda. Om de herkomst van de verdachte gegevens te bevestigen, werd een grondige analyse uitgevoerd om te bepalen of deze gegevens daadwerkelijk afkomstig waren van Skoda. In figuur 15 wordt een taartdiagram gepresenteerd waarin de verdeling van automerken binnen de vierkante datapuntenwolk rond Rusland wordt weergegeven. Uit deze figuur blijkt dat 69,9% van de gegevens binnen deze datapuntenwolk afkomstig is van Skoda. In figuur 16 wordt een vergelijkbaar taartdiagram getoond, maar nu voor de verdeling van automerken buiten de vierkante datapuntenwolk rond Rusland. Hieruit blijkt dat 76,9% van de gegevens buiten deze wolk afkomstig is van Tesla, hetgeen in lijn is met de resultaten van een eerdere analyse (zie figuur 12).

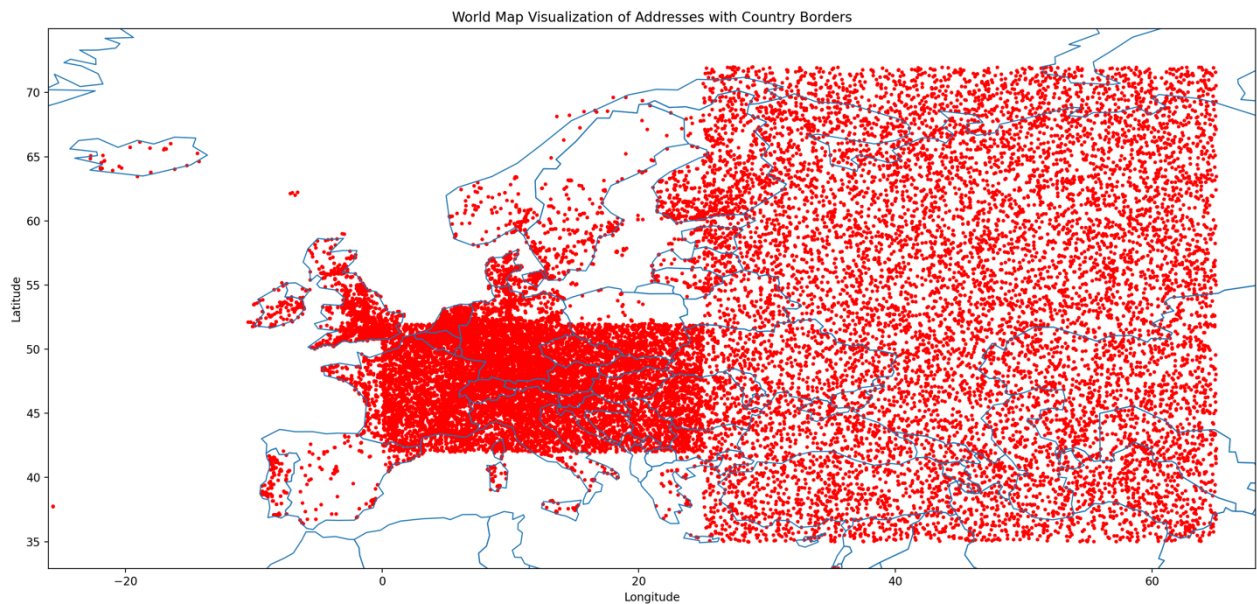


Figuur 15: Taartdiagram visualisatie van de distributie van automerken binnen het verdachte datapuntenwolk rond Rusland

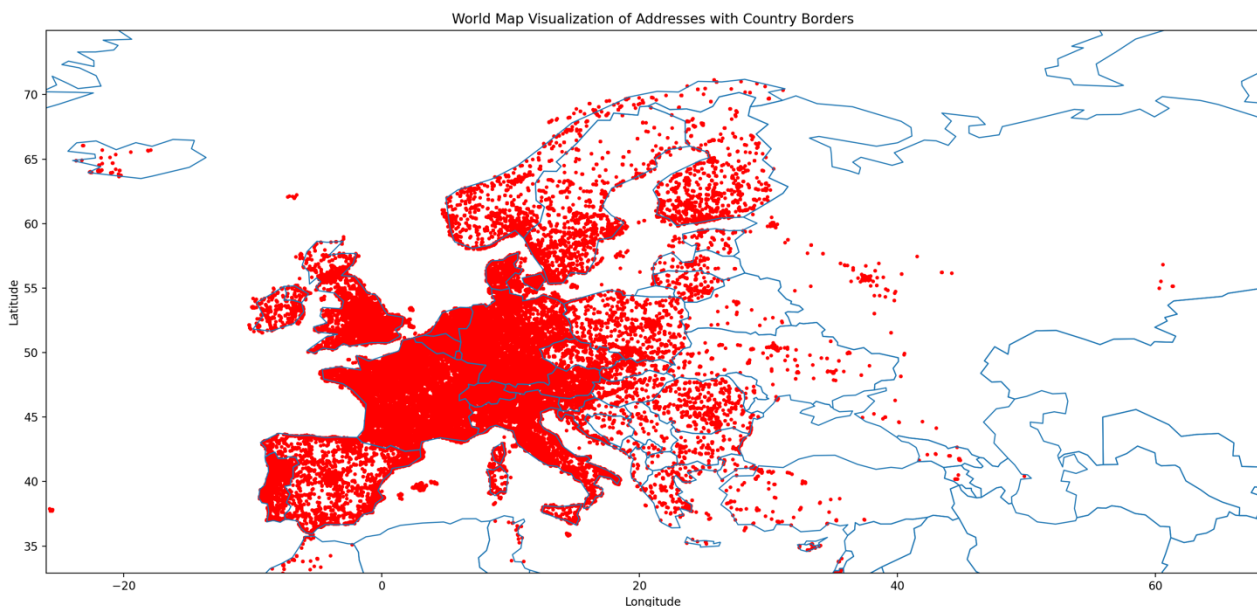


Figuur 16: Taartdiagram visualisatie van de distributie van automerken buiten het verdachte datapuntenwolk rond Rusland

Verder werd een wereldkaart gevisualiseerd waarop uitsluitend de laadpunten afkomstig van Skoda zijn weergegeven. Dit resulteerde in figuur 17, waarin een tweede datapuntenwolk duidelijk zichtbaar is, wat wijst op de aanwezigheid van onjuiste gegevens. Deze foutieve Skoda-gegevens werden vervolgens uitgefilterd, wat resulteerde in een aangepaste wereldkaartvisualisatie van de laadpunten, zoals weergegeven in figuur 18.

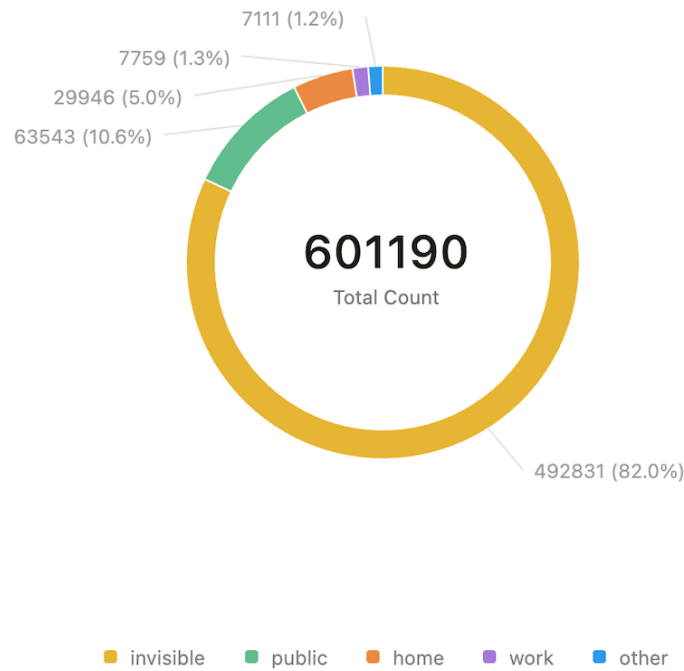


Figuur 17: Wereldkaart visualisatie van de oplaadsessies met enkel Skoda data



Figuur 18: Wereldkaart visualisatie van de oplaadsessies (na het toepassen van de filter, zonder Skoda data)

De verschillende types adressen van de laadsessies werden gevisualiseerd in een taartdiagram met behulp van de Matplotlib-library. Dit werd gedaan om de verdeling van de laadsessies over verschillende soorten locaties te analyseren en om eventuele onregelmatigheden in de gegevens te identificeren. Dit leidde tot figuur 16. Uit deze visualisatie bleek dat 82 % van de laadsessies een onzichtbaar type hadden. Dit is handig omdat het inzicht biedt in de volledigheid en kwaliteit van de data. Daarnaast werd er ook een analyse gedaan van de verschillende types wagens. Er werd vastgesteld dat er ook een kleine hoeveelheid (4,6 %) hybride wagens aanwezig is, zoals te zien is in figuur 17.

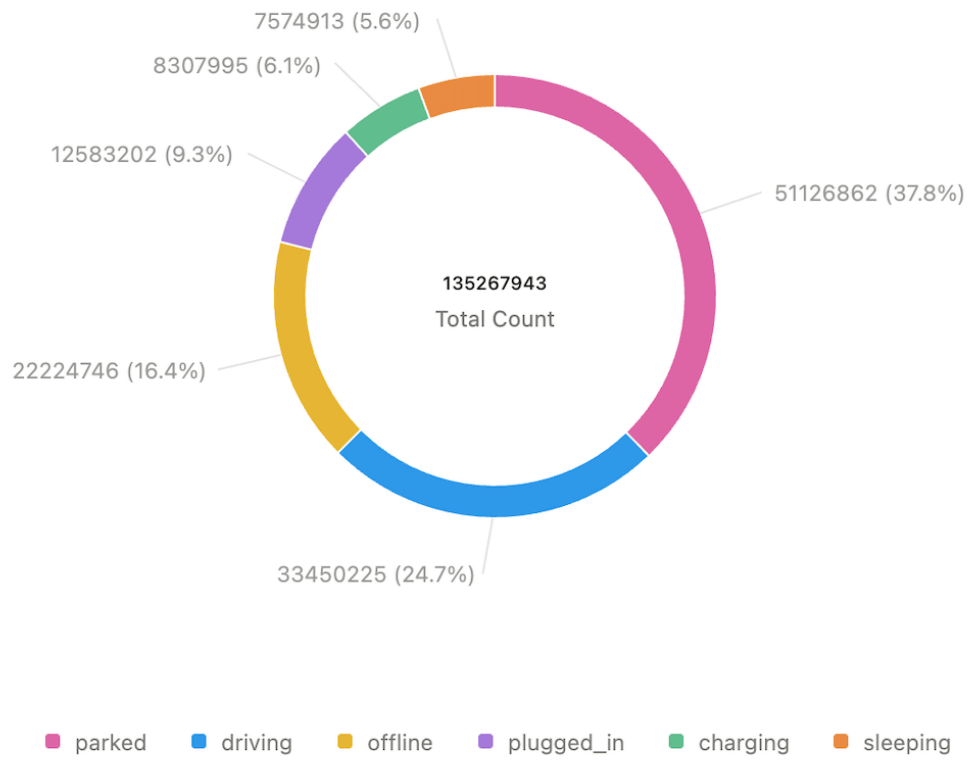


Figuur 19: Adresclassificatie in de EV-dataset (n=601190). De overgrote meerderheid (82%) van de adressen is gecategoriseerd als 'onzichtbaar'



Figuur 20: Wagenclassificatie in de EV-dataset (n=40209). Een klein aantal (4,6 %) van de wagens is gecategoriseerd als hybrid

De verschillende soorten event types in de dataset werden gevisualiseerd in een taartdiagram met behulp van de Matplotlib-library. Dit werd gedaan om de verdeling van de event types te analyseren en te beoordelen of een relevant deel van de data geschikt was voor het detecteren van V2G-fraude of charge pass-fraude aangezien we hier dan kijken naar het type 'parked' en 'charging'. Dit leidde tot figuur 18. Uit deze visualisatie bleek dat een aanzienlijk deel van de event types potentieel relevant is voor verder onderzoek naar fraudedetectie.



Figuur 21: Verdeling van event types in de dataset ($n=135267943$). Een substantieel deel van de events lijkt relevant voor het detecteren van V2G- of charge pass-fraude

3.3 Nabespreking

In Hoofdstuk 3 wordt een gedetailleerde beschrijving gegeven van het gegevensbeheer en de analyseprocessen die ten grondslag liggen aan dit onderzoek. De EEVEE Mobility dataset vormt de basis voor deze analyse. Het hoofdstuk beschrijft hoe een subset van deze dataset zorgvuldig is geselecteerd en geoptimaliseerd om specifieke vormen van fraude, zoals V2G-fraude en ‘*charge pass*’-fraude, te detecteren.

De bespreking van het gegevensbeheer omvat het opzetten van foreign key-relaties en het indexeren van kolommen om de prestaties van de database te verbeteren. Deze stappen waren cruciaal voor een efficiënte data-analyse en verdere toepassing in machine learning-modellen en webapplicaties.

Daarnaast biedt dit hoofdstuk inzicht in de visuele data-analyse die werd uitgevoerd om de integriteit en betrouwbaarheid van de gegevens te waarborgen. Door middel van visuele representaties, zoals taartdiagrammen en geografische kaarten, werden patronen in de dataset geïdentificeerd, waaronder merkverdeling van EV's en anomalieën in locatiegegevens. Opmerkelijke ontdekkingen, zoals de vierkante datapuntenwolk rond Rusland, werden onderzocht en gevalideerd, waarbij Skoda-gegevens als bron van de afwijking werden geïdentificeerd en uiteindelijk gefilterd.

Kortom, hoofdstuk 3 illustreert de complexiteit van gegevensbeheer en analyse in het kader van fraudeopsporing binnen het domein van elektrische voertuigen, en legt de basis voor de verdere stappen in dit onderzoek.

Hoofdstuk 4

Ontwikkeling webapplicatie

4.1 Ontwikkeling isolation forest-algoritme

Na een visuele data-analyse werd het isolation forest-algoritme gekozen voor de detectie van uitschieters in de dataset van EEVEE Mobility. Dit machine learning-algoritme is specifiek geschikt voor het identificeren van afwijkingen zoals V2G-transacties, laadpasfraude en abnormaal gebruik, zoals hoge kosten in een specifieke maand of laadsessie. Het isolation forest-algoritme is effectief vanwege zijn vermogen om anomalieën snel te isoleren, zelfs in grote datasets. Daarnaast is het schaalbaar en snel, wat essentieel is voor de continue stroom van gegevens bij EEVEE Mobility. Voor de implementatie werd de isolation forest module uit de scikit-learn bibliotheek in Python gebruikt.

4.1.1

Het isolation forest-algoritme werd toegepast om 'vehicle-to-grid' fraude te detecteren. Deze vorm van fraude houdt in dat energie onterecht teruggeleverd wordt aan het elektriciteitsnet, bijvoorbeeld wanneer een bestuurder van een EV zijn thuisbatterij oplaadt met energie die hij van zijn werkgever heeft verkregen. Voor de ontwikkeling van het algoritme werd de 'events' tabel van de database geanalyseerd, waarbij specifiek werd gekeken naar 'events' van het type 'parked'. Dit hield in dat alle EV-gebeurtenissen werden onderzocht waarbij de auto geparkeerd stond. Indien de batterij van de wagen meer dan 5% van zijn capaciteit verliest terwijl deze geparkeerd staat, markeert het isolation forest-algoritme dit als een uitschieter, oftewel een anomalie. Deze grens van 5% is afkomstig van een werknemer van EEVEE Mobility, die veel met EV-data werkt en dit als advies heeft meegegeven.

Voor de anomaliedetectie werd een lijst opgesteld op basis van de kolom 'percent_difference' en de kolom 'avg_outside_temp'. De 'percent_difference' kolom is een kolom in de dataset die het percentageverschil tussen de start en het einde van een event berekent. De 'avg_outside_temp' kolom stelt de gemiddelde buitentemperatuur voor tijdens het geparkeerde event. Vervolgens werd de contaminatieratio bepaald. Dit is het percentage rijen in de dataset waarbij de waarde in de kolom 'percent_difference' groter is dan 5%. Met andere woorden, het geeft aan hoeveel van de data mogelijk afwijkend of ongewoon is in vergelijking met de rest van de dataset.

Op basis van deze contaminatieratio werd een isolation forest-model gecreëerd. In eenvoudige bewoordingen werkt het door de dataset op te splitsen in kleinere delen, net zoals een bos wordt opgesplitst in individuele bomen (vandaar de naam "forest"). Het idee is dat afwijkende gegevenspunten sneller geïsoleerd worden dan normale punten. Door deze splitsingen meerdere keren te herhalen, kan het model leren welke punten waarschijnlijk anomalieën zijn.

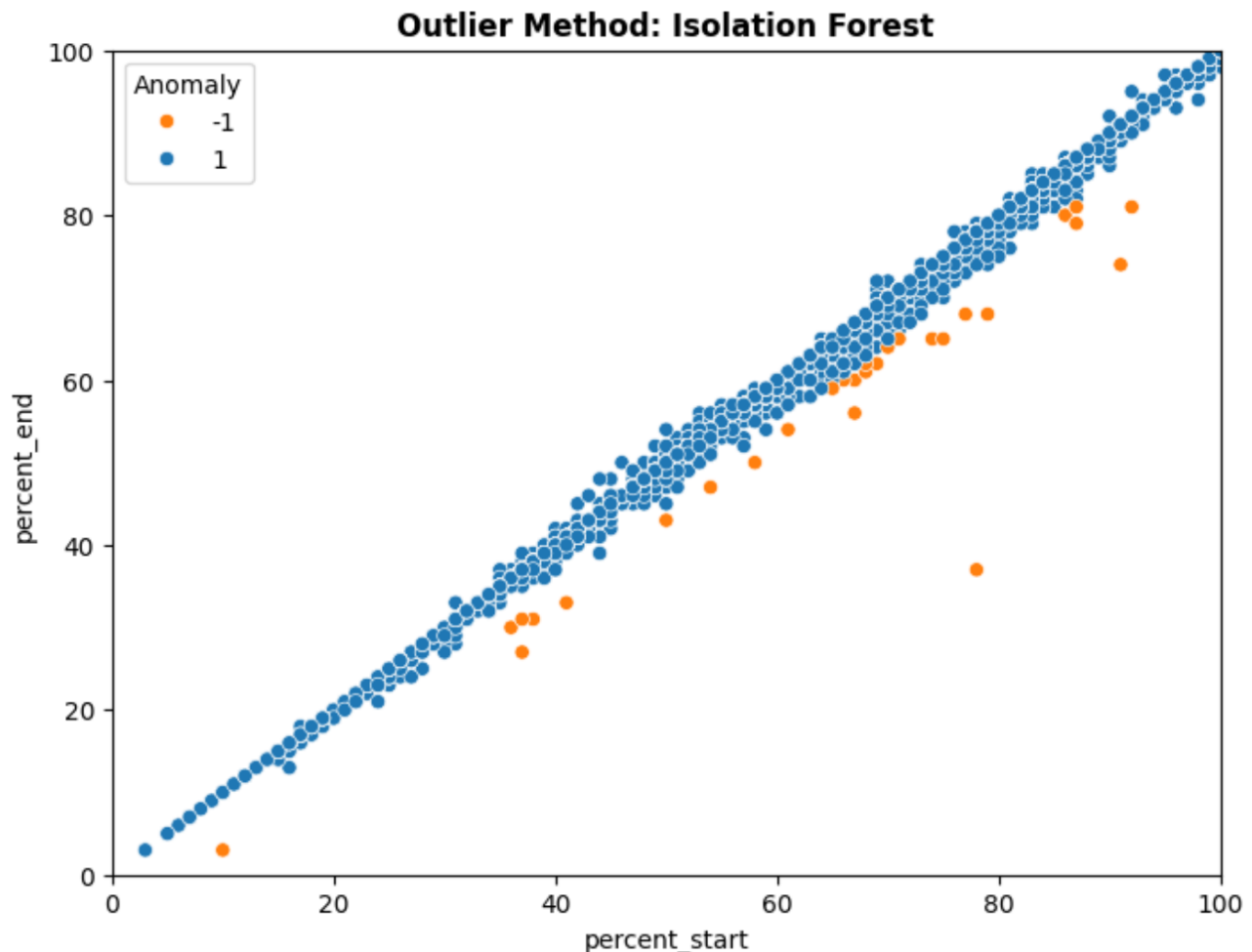
Om ervoor te zorgen dat de resultaten van het model consistent zijn bij elke uitvoering, werd een 'random_state' parameter ingesteld. Deze parameter fungeert als een zogeheten 'willekeurige

zaadwaarde' die het algoritme leidt bij het maken van willekeurige keuzes. Hierdoor worden dezelfde resultaten verkregen, zelfs als het model meerdere keren opnieuw wordt uitgevoerd met dezelfde gegevens. Dit is belangrijk voor de reproduceerbaarheid van de resultaten, zodat de uitkomsten van het model bij herhaalde experimenten gelijk blijven.

Dit model werd vervolgens getraind met behulp van de kolommen '*percent_difference*' en '*avg_outside_temp*' uit de dataset. 'Training' in dit geval betekent dat het model de gegevens in de kolom analyseert en patronen leert herkennen die op normale of abnormale waarden wijzen. Na deze trainingsfase berekent het model voor elke rij in de dataset een anomaliescore. Deze score geeft aan hoe waarschijnlijk het is dat een rij een anomalie is, waarbij een hogere score betekent dat de kans op een anomalie kleiner is. Daarnaast maakt het model ook een voorspelling voor elke rij: -1 als de rij een anomalie is, en 1 als het een normale rij is.

Als uit analyse zou blijken dat 5% altijd een betrouwbare indicator is van fraude, zou men kunnen denken dat een eenvoudigere regel (zoals het handmatig filteren op >5%) volstaat. Echter, de isolation forest is een geavanceerder model dat niet alleen naar één drempel kijkt, maar naar de algemene structuur van de data en hoe afwijkend een bepaalde rij is in vergelijking met de rest van de dataset. In het geval van dit onderzoek hebben we het over de training van het algoritme met behulp van de kolom '*percent_difference*' maar ook op de '*avg_outside_temp*' kolom. Het model kan dus complexere patronen en anomalieën detecteren die niet alleen gebaseerd zijn op de 5% drempel, maar ook op andere factoren of combinaties van kenmerken in de dataset. Dit maakt de isolation forest waardevol, zelfs als 5% een belangrijke drempel blijft.

De resultaten van het algoritme zijn visueel weergegeven in figuur x, waar de gegevens op een grafiek zijn uitgezet.



Figuur 22: Grafiek met de resultaten van het V2G isolation forest-algoritme. De x-as geeft het startpercentage van de batterij aan tijdens een 'parked' event, de y-as het eindpercentage. De oranje punten stellen anomalieën voor (-1)

De grafiek in figuur x toont dat wanneer er een verschil is van meer dan 5% tussen het start- en eindbatterijpercentage van de EV, het isolation forest-algoritme deze datapunten detecteert als uitschieters, welke vervolgens oranje gemarkeerd worden.

4.1.2

Na de ontwikkeling van dit algoritme werd het isolation forest-algoritme ingezet om abnormaal gebruik te detecteren. Dit gebeurde op verzoek van EEEVE Mobility, aangezien dit een prioriteit was voor hun onderzoek. Abnormaal gebruik werd gedefinieerd als uitzonderlijk hoge maandelijkse laadkosten voor EV-bestuurders. Hierbij werd gekeken naar de totale maandelijkse laadkosten en het verbruik van bestuurders, evenals naar specifieke laadsessies van individuele gebruikers. Dit algoritme werkt op een vergelijkbare manier als het algoritme voor het detecteren van 'vehicle-to-grid' fraude, maar in dit geval werd het model getraind met de kolom 'monthly_cost', welke is berekend op basis van alle 'price'-waarden voor een specifieke maand, afkomstig uit de 'charge'-tabel van de dataset. Een deel van de output van dit algoritme wordt weergegeven in tabel x. De kolom 'finished_charging_month' vertegenwoordigt de maand waarin een laadsessie is beëindigd. De kolom 'monthly_kWh' wordt op een vergelijkbare wijze berekend als de kolom 'monthly_cost', namelijk door het totale maandelijkse verbruik voor een specifieke maand te bepalen.

◊	finished_charging_month	◊	user_id	◊	fleet_id	◊	monthly_cost	◊	monthly_kWh	◊	price_per_kWh	◊	anomaly	◊
7	2023-12		23399		19		58.34		218.17		0.267406		1	
18	2023-12		21135		19		7.74		254.19		0.030450		-1	
29	2023-12		30146		28		12.38		35.35		0.350212		1	
37	2023-12		29279		190		2.21		8.83		0.250283		1	
48	2023-12		30181		28		124.17		354.54		0.350228		1	
58	2023-12		32575		28		66.19		331.03		0.199952		-1	
69	2023-12		37996		29		104.00		297.12		0.350027		1	
80	2023-12		38003		29		82.59		284.04		0.290769		1	

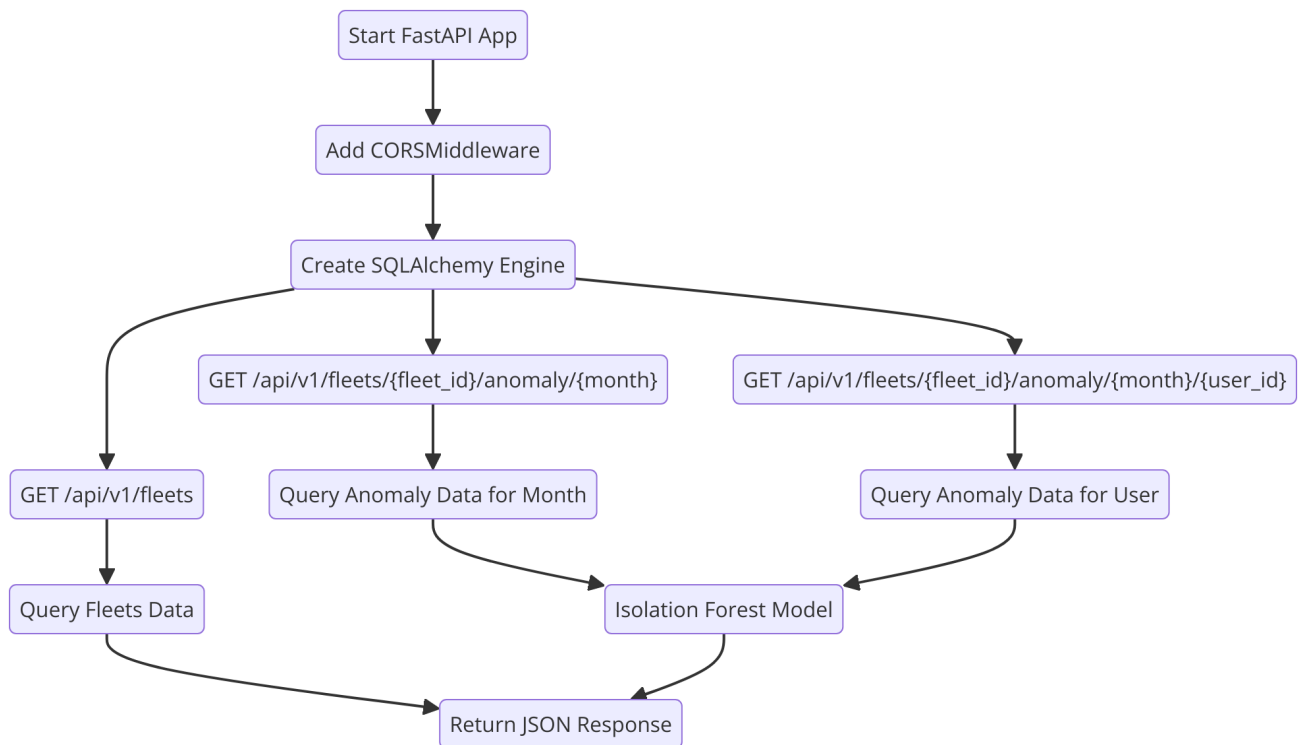
Tabel 3: Gedeeltelijke output van het isolation forest-algoritme toegepast op abnormaal gebruik

Het isolation forest-algoritme werd overwogen als methode om 'charge pass fraude' te identificeren, een frauduleuze praktijk waarbij laadpassen oneigenlijk worden gebruikt, zoals bij laadsessies zonder merkbare stijging van het batterijpercentage van de elektrische voertuigen. Hoewel het onderzoek gericht was op het toepassen van deze aanpak, bleek het uiteindelijk niet mogelijk om dit specifieke doel binnen de beschikbare onderzoeksperiode te realiseren.

Samenvattend, er zijn twee isolation forest-algoritmen ontwikkeld: één voor het detecteren van 'vehicle-to-grid' fraude en één voor het detecteren van abnormaal gebruik met betrekking tot laadkosten van EV's.

4.2 Ontwikkeling API

Na de ontwikkeling van het isolation forest-algoritme werd een API ontworpen om de gegevens en resultaten toegankelijk te maken voor een frontend webapplicatie. Deze API werd gerealiseerd met behulp van het Python-gebaseerde FastAPI-framework en is schematisch te zien in figuur x.



Figuur 23: Diagram dat de architectuur en stroom van de FastAPI-API vertegenwoordigt

Initialisatie

Het proces begint met het 'Start FastAPI App'-knooppunt, dat de FastAPI-applicatie initialiseert.

Het 'Add CORSMiddleware'-knooppunt vertegenwoordigt de opname van 'Cross-Origin Resource Sharing' (CORS)-middleware, waardoor de server verzoeken van gespecificeerde oorsprongen kan afhandelen, wat cruciaal is voor het mogelijk maken van frontend-applicaties die op verschillende domeinen worden gehost.

Database-engine

Het 'Create SQLAlchemy Engine'-knooppunt is cruciaal omdat het een verbinding tot stand brengt met de EEVEE Mobility MySQL-database met behulp van SQLAlchemy. Deze engine is verantwoordelijk voor het uitvoeren van SQL-query's en het ophalen van gegevens uit de database.

API-eindpunten

In de applicatie zijn drie primaire API-eindpunten gedefinieerd:

GET /api/v1/fleets: Deze route, vertegenwoordigd door het knooppunt 'RouteFleets', verwerkt verzoeken voor het ophalen van vlootgegevens. Het knooppunt 'Query Fleets Data' geeft aan dat

een SQL-query wordt uitgevoerd om de vlootinformatie uit de database op te halen, die vervolgens wordt geretourneerd als een JSON-antwoord.

GET /api/v1/fleets/{fleet_id}/anomaly/{month}: Het *'RouteAnomaly'*-knooppunt vertegenwoordigt een eindpunt dat afwijkende gegevens ophaalt voor een specifieke vloot binnen een bepaalde maand. Het knooppunt *'Query'*-anomaliegegevens voor maand laat zien dat gegevens uit de database worden opgevraagd. Deze gegevens worden vervolgens doorgegeven aan het isolation forest model voor detectie van afwijkingen, waar ongebruikelijke patronen in de maandelijkse kosten worden geïdentificeerd.

GET /api/v1/fleets/{fleet_id}/anomaly/{month}/{user_id}: Deze route, aangegeven door het knooppunt *'RouteUserAnomaly'*, is verantwoordelijk voor het ophalen van afwijkende gegevens die specifiek zijn voor een gebruiker binnen een vloot voor een bepaalde maand. Net als bij de vorige route worden gegevens opgehaald via het knooppunt *'Query Anomaly Data for User'* en verwerkt door het isolation forest model om afwijkingen te detecteren.

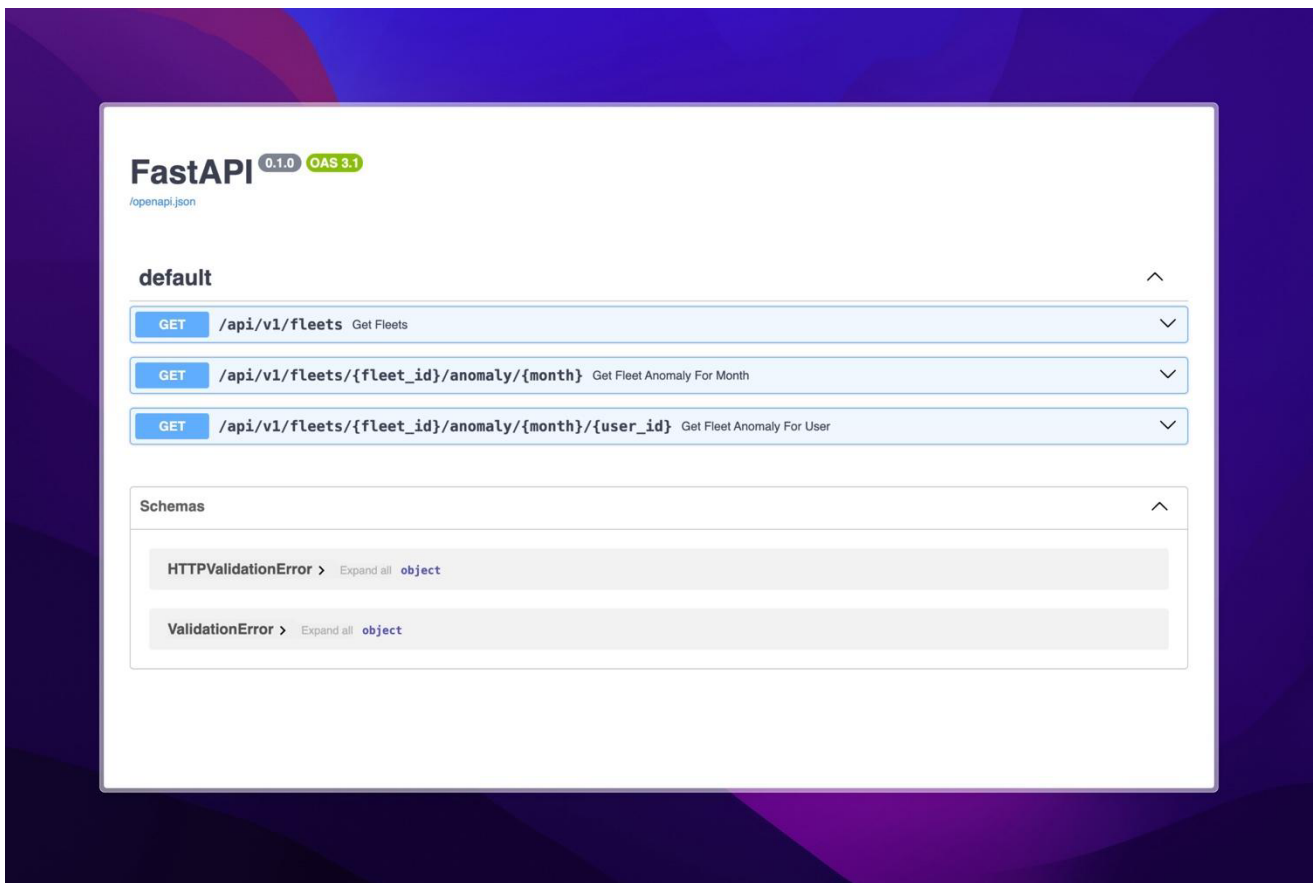
Machine learning-integratie

Het knooppunt *'isolation forest model'* vertegenwoordigt de toepassing van het isolation forest-algoritme. Het verwerkt de gegevens die uit de database worden opgehaald en berekent scores die bepalen of specifieke gegevenspunten (bijvoorbeeld maandelijkse kosten) als afwijkend worden beschouwd.

Reactieafhandeling

Het *'return JSON response'*-knooppunt geeft aan dat de verwerkte gegevens, of het nu om vlootinformatie of om afwijkingendetectie gaat, worden geconverteerd naar een JSON-indeling en worden teruggestuurd naar de client, waarmee de verzoek-antwoordcyclus wordt voltooid.

Het FastAPI-framework genereert automatisch de benodigde interactieve API-documentatie, geleverd door Swagger UI [69]. Een screenshot van deze documentatie is te zien in figuur x.



Figuur 24: Screenshot van de API documentatie

Deze API voorziet een dienst die door de webapplicatie, welke later in deze scriptie wordt besproken, kan worden geconsumeerd om zodoende de correcte gegevens op een visuele en interactieve manier weer te geven.

4.3 Ontwikkeling frontend webapplicatie

In dit deel van het onderzoek wordt de ontwikkeling van de frontend webapplicatie uitvoerig beschreven. Deze applicatie werd gerealiseerd met behulp van het Vue.js framework en heeft als doel de visualisatie van de EEVEE Mobility data te faciliteren, waardoor gebruikers op interactieve wijze met de data kunnen omgaan.

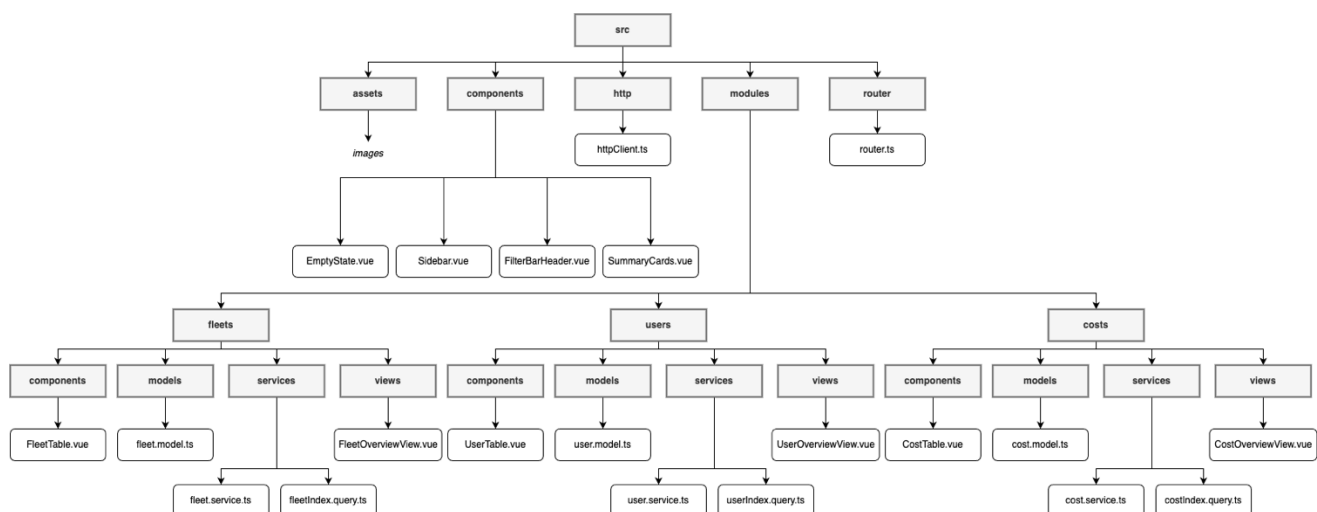
Vorbereiding en ontwerpfase

Er werd een frontend onboarding proces doorlopen om te verzekeren dat de code voldeed aan de normen en waarden van Wisemen. Een ontwerpdocument werd ontvangen, waarin de styling van de uiteindelijke webapplicatie gedetailleerd werd beschreven. Dit ontwerp vormde de basis voor de ontwikkeling, waardoor er met duidelijke richtlijnen met betrekking tot de structuur en lay-out van de applicatie geprogrammeerd kon worden.

Principes van programmeren

Een essentieel onderdeel van softwarearchitectuur is de strikte scheiding van zorgen. Dit principe, samen met het *'Don't Repeat Yourself'*-principe (DRY), zorgt ervoor dat de code goed georganiseerd, herbruikbaar en gemakkelijk te onderhouden is. Concreet is deze scheiding van zorgen terug te zien in het diagram van figuur x. In deze figuur zien we:

1. Componenten: Bevatten de visuele elementen van de gebruikersinterface (UI) en handelen de interactie met de gebruiker af.
2. Models: Definieren de structuur van de data die wordt gebruikt in de applicatie.
3. Services: Bevatten de logica voor het communiceren met de backend, het verwerken van data en het uitvoeren van complexe taken.
4. Views: Zorgen voor de presentatie van de data in de UI.



Figuur 25: Diagram van de project structuur binnen de Vue.js applicatie

Door deze duidelijke scheiding:

1. Wordt het makkelijker om wijzigingen aan te brengen in een bepaald onderdeel van de applicatie zonder dat dit invloed heeft op andere onderdelen.
2. Kunnen we dezelfde logica (bijvoorbeeld het ophalen van gegevens) hergebruiken in verschillende componenten door middel van services.
3. Verbetert de leesbaarheid van de code aanzienlijk, omdat elke laag een specifieke verantwoordelijkheid heeft.

Het introduceren van een HTTP-service

Om deze scheiding van zorgen verder te versterken, is er een HTTP-service geïntroduceerd. Deze service is verantwoordelijk voor alle communicatie met de backend. Door deze service te gebruiken, voorkomen we dat de componenten direct met de backend hoeven te communiceren. Dit heeft als voordeel dat:

1. De componenten zich kunnen concentreren op de presentatie van de data. Ze hoeven niet meer bezig te zijn met de complexiteit van HTTP-verzoeken.
2. De service als een abstractielaag fungeert. Wijzigingen in de backend kunnen worden doorgevoerd zonder dat de componenten aangepast hoeven te worden.
3. De code beter te testen is. We kunnen de service isoleren en testen zonder de volledige applicatie op te starten.

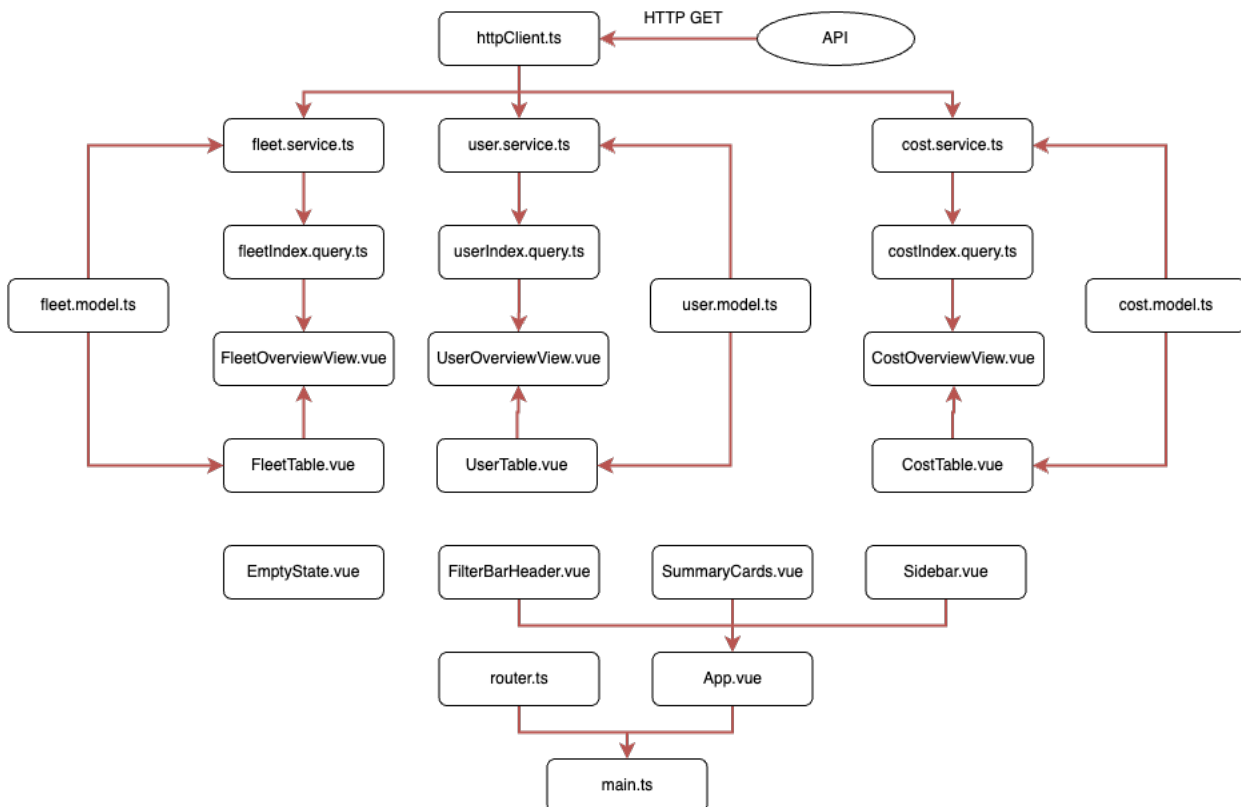
Voorbeeld: In het diagram van figuur y, zien we dat de *'FleetTable'* component gebruik maakt van de *fleet.service.ts* om gegevens op te halen. Dit is een duidelijke illustratie van hoe de scheiding van zorgen in de praktijk wordt toegepast.

Implementatie van kernfunctionaliteiten

Er werd een router opgezet, een cruciaal onderdeel van Vue.js-applicaties dat de navigatie tussen verschillende weergaven mogelijk maakt. De router speelt ook een rol bij het beheren van authenticatie en rechten voor specifieke routes door middel van zogenaamde 'guards'. Hoewel 'guards' in dit onderzoek niet aan bod komen omdat Wisemen geen authenticatiesysteem nodig had, zijn ze een belangrijke feature voor toekomstige ontwikkelingen. Ze zorgen ervoor dat toegangsbeheer in webapplicaties efficiënt en veilig kan worden geregeld, wat van grote waarde kan zijn bij het uitbreiden van de applicatie.

Weergave van fleet data

Een model voor het beheer van een fleet werd ontwikkeld, dat dient als interface voor een enkele fleet en de bijbehorende eigenschappen. Vervolgens werd een fleet-service ontwikkeld die een GET-verzoek verstuurt via de HTTP-client naar de API. Daarna werd er een query opgezet, die de getAll-functie van de service aanroept en de fleets ophaalt van de backend. De query dient als een belangrijke abstractielaag die de Vue-componenten ontlast van de complexiteit van data-ophaling en asynchrone processen, terwijl het ook zorgt voor een flexibele en onderhoudbare architectuur. De service zorgt dan voor de daadwerkelijke communicatie met de API. Het bevat een eenvoudige methode, getAll, die een HTTP GET-request uitvoert via de httpClient. Dit is allemaal schematisch te zien in figuur y.



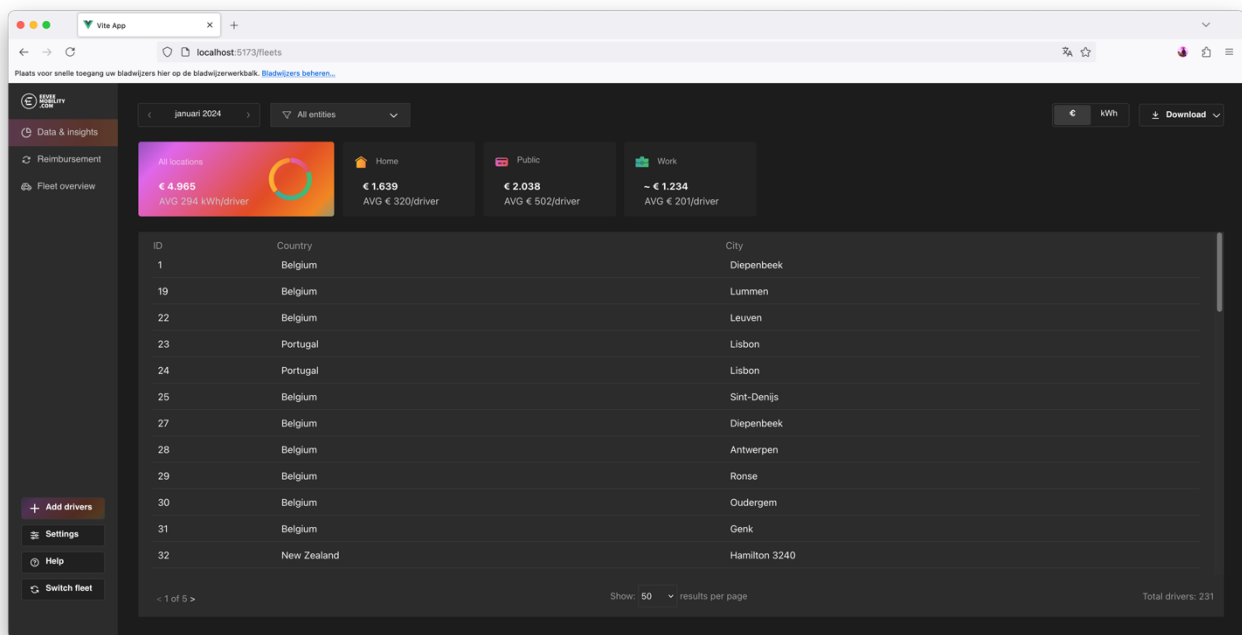
Figuur 26: Diagram van de afhankelijkheden binnen de Vue.js applicatie

Ontwikkeling van de fleet tabel component

Er werd een fleet tabel component gecreëerd. Deze component werd ontwikkeld met behulp van HTML, CSS en Typescript. Gedurende de gehele ontwikkeling van de webapplicatie werden continu modules gebruikt. Deze modules bestaan uit verzamelingen van componenten, weergaven, stores, services, enzovoort, die gezamenlijk een specifieke functie van de applicatie realiseren. Via deze manier werd ook de gebruikers tabel en de kosten lijst ontwikkeld. Deze modules zijn terug te vinden in figuur x. Onder de modules folder van deze figuur zijn er 3 modules gedefinieerd. De fleets module realiseert alle functies van de applicatie die te maken hebben met het modelleren, opvragen en weergeven van fleet data. Dit geldt ook zo voor de users en de costs modules maar dan respectievelijk voor user en cost data.

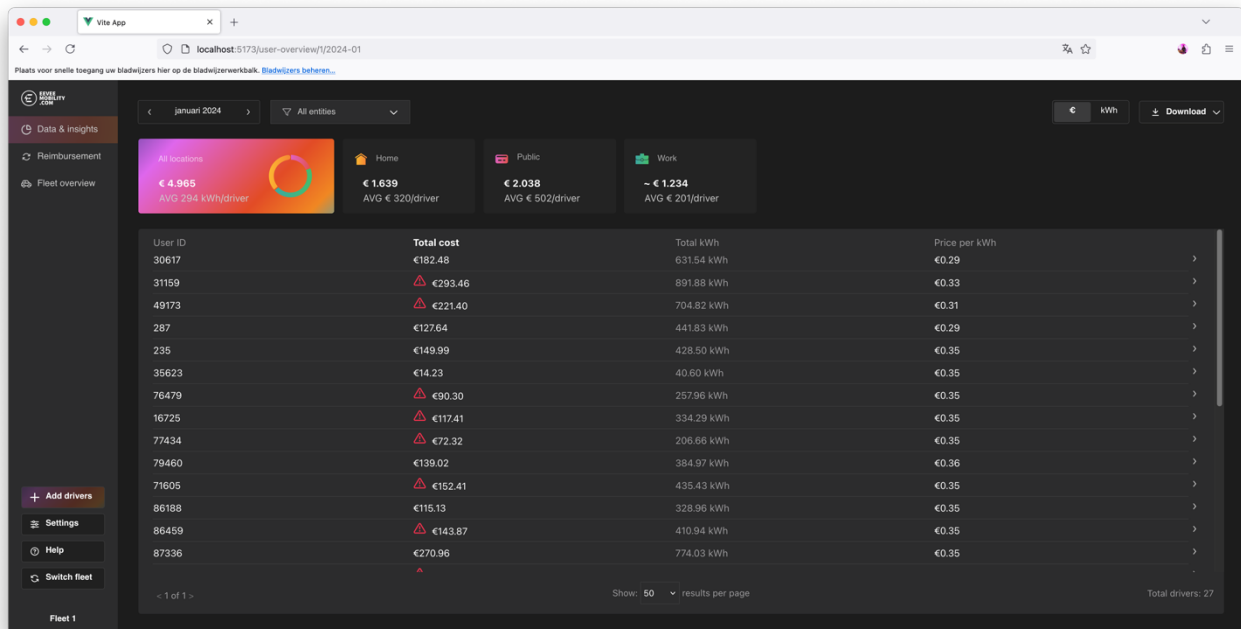
Resultaten

De uitvoering van de geschreven code resulteert in de volgende interfaces. De gebruiker begint op een webpagina waarop een overzicht wordt geboden van alle fleets, inclusief het bijbehorende land en de betreffende stad, zoals weergegeven in figuur 1. In plaats van de naam van de fleet, wordt de ID weergegeven omwille van de vereiste anonimiteit volgens de richtlijnen van EEEVE Mobility. Deze anonimiteit is ook van toepassing op de daaropvolgende webpagina's. Door op een van de fleets in de tabel te klikken, wordt de gebruiker doorgestuurd naar een webpagina waar een tabel wordt weergegeven met verschillende EV-eigenaren en de bijbehorende gegevens, zoals geïllustreerd in figuur 2. Tot op dit punt zijn de beschreven functionaliteiten ontworpen voor medewerkers van EEEVE Mobility, met als doel inzicht te bieden in verschillende fleets (klanten van EEEVE Mobility).



ID	Country	City
1	Belgium	Diepenbeek
19	Belgium	Lummen
22	Belgium	Leuven
23	Portugal	Lisbon
24	Portugal	Lisbon
25	Belgium	Sint-Denijs
27	Belgium	Diepenbeek
28	Belgium	Antwerpen
29	Belgium	Ronse
30	Belgium	Oudergem
31	Belgium	Genk
32	New Zealand	Hamilton 3240

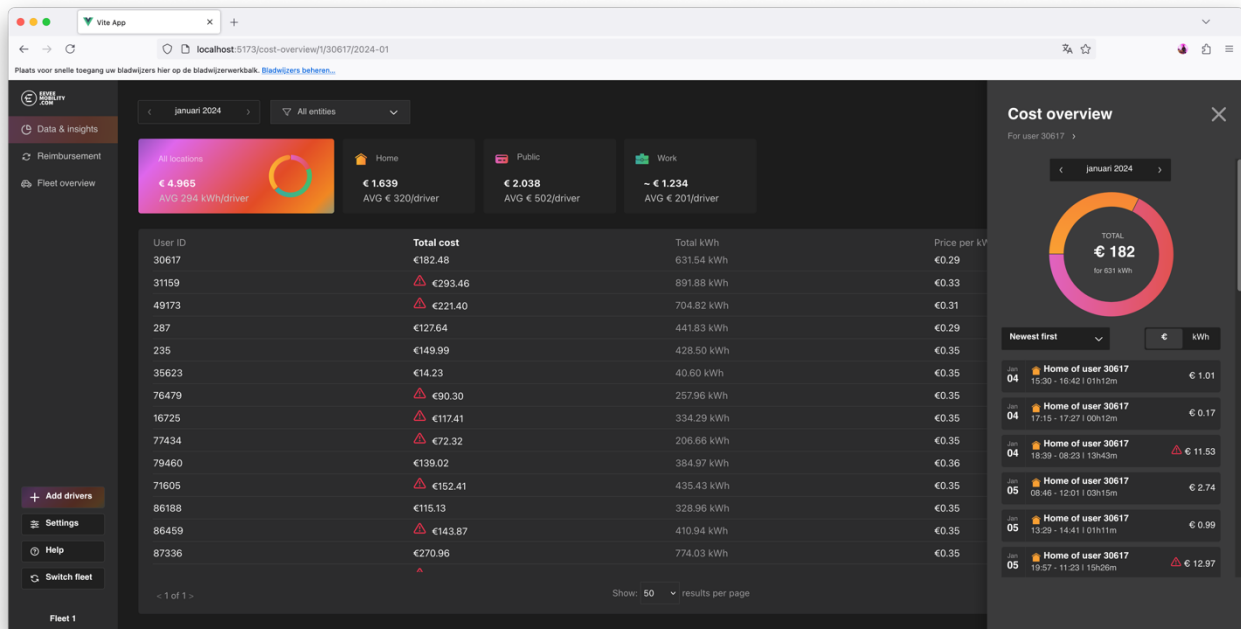
Figuur 27: Webpagina dat een tabel weergeeft van alle fleets met bijhorende data



Figuur 28: Webpagina dat een tabel weergeeft van alle EV-eigenaars met bijhorende data

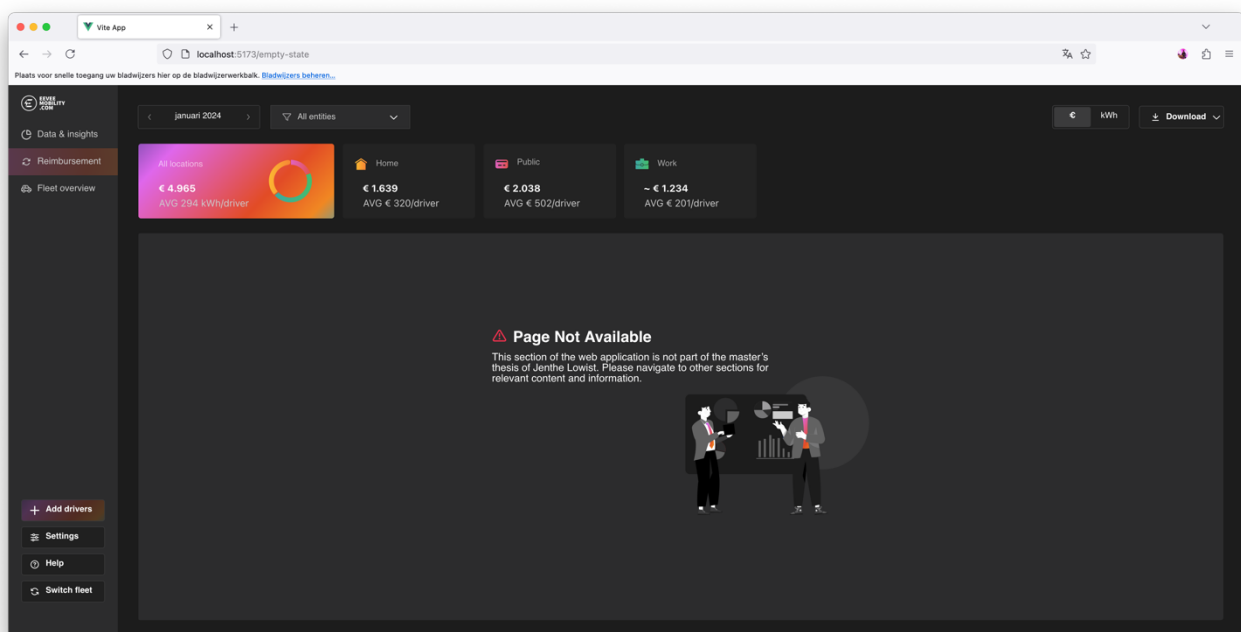
Vanaf de tabel van EV-eigenaren vertegenwoordigt de gebruiker van de webapplicatie een werkgever van het bedrijf dat eigenaar is van een specifieke fleet en dus klant is van EEEVE Mobility. De tabel van EV-eigenaren (werknemers van een bedrijf) bevat gebruikersgegevens voor een bepaalde maand, waarbij de gebruiker de mogelijkheid heeft deze maand naar wens aan te passen. Tevens kan de gebruiker het aantal weergegeven resultaten in de tabel aanpassen en navigeren door de gegevens. De tabel van EV-eigenaren toont ook de totale maandelijkse laadkosten voor elke specifieke EV-eigenaar, samen met het totale verbruik van die maand en de gemiddelde prijs per kWh. Indien er een rood waarschuwingssymbool naast een bepaalde maandelijkse laadkost verschijnt, betekent dit dat deze waarde door het isolation forest-algoritme is gedetecteerd als een mogelijke anomalie. De gebruiker kan over dit symbool hoveren om de verklaring te zien waarom deze waarde als anomalie wordt beschouwd.

Verder kan de gebruiker op een rij in de tabel van EV-eigenaren klikken, wat leidt tot een nieuwe webpagina met een zijvenster waarin de laadkostgegevens van de betreffende EV-eigenaar worden getoond, zoals weergegeven in figuur 3. Deze lijst geeft alle individuele laadsessies van een specifieke EV-eigenaar weer voor een bepaalde maand, waarbij de gebruiker de mogelijkheid heeft om de maand naar wens aan te passen. De totale kosten en het verbruik worden opnieuw weergegeven, naast een gedetailleerde lijst van afzonderlijke laadkosten en verbruik per sessie, inclusief de datum en duur van elke sessie. Ook hier kan een waarschuwingssymbool verschijnen naast een specifieke laadsessiekost, wat aangeeft dat deze gegevens door het isolation forest-algoritme als een anomalie zijn gedetecteerd. Door over het waarschuwingssymbool te hoveren, kan de gebruiker een toelichting zien op deze detectie.



Figuur 29: Webpagina dat een zijvenster weergeeft van alle individuele laadsessies voor een bepaalde EV eigenaar met bijhorende data

De sidebar van de applicatie bevat daarnaast een aantal tabs die standaard aanwezig zijn op het reeds bestaande EEEVEE Mobility-platform. Deze tabs zijn visueel geïmplementeerd, maar niet functioneel. Om de gebruiksvriendelijkheid te waarborgen, kan de gebruiker op deze tabs klikken, waarna hij of zij wordt doorgestuurd naar een lege webpagina, zoals weergegeven in figuur 4. Op deze pagina wordt duidelijk vermeld dat deze specifieke pagina geen onderdeel uitmaakt van het afstudeerproject en dat de gebruiker kan navigeren naar andere wel geïmplementeerde informatie.



Figuur 30: Webpagina dat aantoont dat deze specifieke pagina niet deel is van het afstudeerproject

4.4 Nabespreking

In hoofdstuk 4 is een uitgebreide beschrijving gegeven van de ontwikkeling van een webapplicatie voor EEVEE Mobility, waarbij specifiek gebruik is gemaakt van machine learning en software-ontwikkelingsprincipes. De focus lag voornamelijk op het ontwikkelen van het isolation forest-algoritme en de bijbehorende API, gevolgd door de ontwikkeling van een frontend webapplicatie.

Ontwikkeling van het Isolation Forest-algoritme

Het isolation forest-algoritme werd geselecteerd na een zorgvuldige analyse van de data en bleek effectief te zijn voor het identificeren van anomalieën in de EEVEE Mobility dataset. Twee kernfunctionaliteiten werden geïmplementeerd: de detectie van ‘vehicle-to-grid’ fraude en het opsporen van abnormaal gebruik op basis van laadkosten. Deze algoritmen zijn specifiek afgestemd op de noden van EEVEE Mobility en hebben geholpen om patronen te herkennen die op mogelijke fraude of abnormaal gedrag wijzen.

Ontwikkeling van de API

Een API werd ontwikkeld met behulp van het FastAPI-framework. Deze API fungeert als schakel tussen de machine learning-modellen en de frontend webapplicatie. De architectuur van de API is gedetailleerd uitgewerkt en zorgt voor een gestroomlijnde toegang tot data en resultaten voor de eindgebruikers.

Ontwikkeling van de Frontend Webapplicatie

In het derde deel van dit hoofdstuk is de ontwikkeling van de frontend webapplicatie beschreven. De applicatie, gebouwd met Vue.js, biedt een interactieve en visueel aantrekkelijke manier om de data van EEVEE Mobility te presenteren. Belangrijke ontwikkelingsprincipes zoals de scheiding van zorgen en het DRY-principe werden strikt nageleefd, wat resulteerde in een onderhoudsvriendelijke en modulaire codebase. De webapplicatie biedt gebruikers de mogelijkheid om gedetailleerde inzichten te krijgen in verschillende fleets en de bijbehorende laadsessies, met speciale aandacht voor anomalieën die door het isolation forest-algoritme zijn gedetecteerd.

Conclusie

Hoofdstuk 4 illustreert een systematische aanpak van het ontwikkelen van een complexe webapplicatie met geïntegreerde machine learning-functionaliteiten. De combinatie van machinelearningalgoritmen, een API, en een gebruiksvriendelijke frontend heeft geresulteerd in een robuust en praktisch hulpmiddel voor EEVEE Mobility.

Hoofdstuk 5

Besluit

In deze masterproef is een uitgebreide webapplicatie ontwikkeld voor EEVEE Mobility, met als kern de integratie van machine learning-algoritmen en een gebruikersvriendelijke frontend. Dit hoofdstuk bespreekt de conclusies die kunnen worden getrokken op basis van de voorgaande hoofdstukken, evalueert de doelstellingen van het project, en biedt een kritische discussie over het werk, inclusief toekomstige uitdagingen en mogelijkheden voor verder onderzoek.

Behaalde Doelstellingen

In Hoofdstuk 1 werden de doelstellingen van dit project uiteengezet:

1. Ontwikkeling van een effectief machine learning-algoritme voor anomaliedetectie.
2. Ontwikkeling van een robuuste API voor gegevensuitwisseling tussen backend en frontend.
3. Implementatie van een interactieve frontend webapplicatie voor data-visualisatie.

Op basis van de beschrijving in de voorgaande hoofdstukken kunnen we concluderen dat deze doelstellingen zijn behaald:

1. Machinelearningalgoritme: In hoofdstuk 4 is een '*isolation forest*'-algoritme toegepast om anomalieën zoals '*vehicle-to-grid*'-fraude en abnormaal gebruik van laadkosten te detecteren. De keuze voor dit algoritme bleek effectief na visuele data-analyse en het behalen van de gestelde doelen. De anomaliedetectie op basis van batterijverlies tijdens parkeerperiodes en maandelijks laadkosten werd succesvol geïmplementeerd, wat aantoonde dat het algoritme in staat is om relevante afwijkingen te identificeren in de dataset van EEVEE Mobility.
2. API ontwikkeling: De ontwikkeling van de API met het FastAPI-framework zorgde voor een gestructureerde en schaalbare manier om data en resultaten beschikbaar te stellen voor de frontend applicatie. Het ontwerp van de API met verschillende eindpunten voor vlootdata en anomaliedetectie voldoet aan de gestelde eisen voor data-integratie en interactie tussen backend en frontend.
3. Frontend webapplicatie: De frontend webapplicatie, gerealiseerd met Vue.js, biedt een gebruiksvriendelijke interface voor de visualisatie van data. De applicatie stelt gebruikers in staat om data te bekijken, anomalieën te identificeren, en gedetailleerde informatie over laadsessies en kosten te verkrijgen. De scheiding van zorgen in de applicatie-architectuur, evenals het gebruik van een HTTP-service, heeft geleid tot een goed gestructureerde en onderhoudsvriendelijke codebase.

Kritische discussie

Hoewel het project succesvol is in het behalen van zijn doelstellingen, zijn er verschillende aspecten die kritisch bekeken moeten worden:

1. Onderzoeksvragen en concept: Het onderzoek richtte zich op het ontwikkelen van een systeem voor anomaliedetectie in EV-data en het visualiseren van deze data. Het gekozen isolation forest-algoritme bleek geschikt voor het detecteren van afwijkingen in grote datasets. De keuze van dit algoritme werd gerechtvaardigd door zijn snelheid en effectiviteit in het isoleren van anomalieën. Echter, een grondigere vergelijking met andere algoritmen, zoals statistische methoden, had mogelijk extra inzichten kunnen bieden in de prestaties en geschiktheid van de gekozen methode.
2. Algoritmen: Het isolation forest-algoritme was effectief voor de beoogde toepassingen, maar er zijn beperkingen. Bijvoorbeeld, de keuze van de parameter voor contaminatie (5%) was gebaseerd op een advies, maar een gedetailleerdere validatie en afstemming van deze parameter op basis van historische data had de nauwkeurigheid kunnen verbeteren. De implementatie van anomaliedetectie voor '*charge pass*'-fraude werd niet voltooid binnen de onderzoeksperiode. Dit wijst op de noodzaak van verdere verfijning en onderzoek om deze toepassing te realiseren.
3. Architectuur en implementatie: De gekozen architectuur met een scheiding van concerns en het gebruik van FastAPI en Vue.js biedt een solide basis voor de applicatie. De API is goed gestructureerd, maar de prestaties kunnen verder worden geoptimaliseerd door middel van cachingstrategieën of asynchrone data-verwerking om de responsiviteit te verbeteren. De frontend-applicatie is intuïtief en gebruikersvriendelijk, maar er kunnen extra functionaliteiten worden toegevoegd, zoals uitgebreide filters en zoekmogelijkheden.

Open uitdagingen en toekomstig onderzoek

Er zijn verschillende uitdagingen en mogelijkheden voor toekomstig onderzoek die naar voren zijn gekomen:

1. Verbetering van algoritmes: Verdere studie is nodig om andere anomaliedetectiemethoden te verkennen en te vergelijken met het isolation forest-algoritme. Methoden zoals deep learning of hybride benaderingen kunnen mogelijk betere resultaten opleveren voor specifieke soorten anomalieën.
2. Uitbreiding van functionaliteiten: De frontend webapplicatie kan worden uitgebreid met extra functionaliteiten, zoals geavanceerdere analysetools en interactie-opties. Bijvoorbeeld, integratie van real-time data-analyse en meldingssystemen voor detectie van anomalieën kan de bruikbaarheid verbeteren.
3. Optimalisatie van API: De API kan worden geoptimaliseerd voor betere prestaties en schaalbaarheid. Het onderzoeken van technieken zoals load balancing en microservices kan de efficiëntie en responsiviteit verbeteren.
4. Beveiliging en privacy: Het waarborgen van de beveiliging en privacy van de gegevens is cruciaal, vooral gezien de gevoelige aard van de EV-data. Verdere aandacht moet worden besteed

aan beveiligingsmaatregelen en gegevensversleuteling om te voldoen aan de regelgeving en beste praktijken.

5. Evaluatie en validatie: Voortdurende evaluatie van de modellen en de applicatie in diverse operationele omstandigheden en met verschillende datasets is noodzakelijk om de robuustheid en effectiviteit te waarborgen. Feedback van eindgebruikers kan ook waardevolle inzichten bieden voor verdere verbetering.

Conclusie

Deze masterproef heeft met succes een webapplicatie ontwikkeld die machinelearningalgoritmen integreert voor anomaliedetectie in EV-data en deze visualiseert via een gebruiksvriendelijke frontend. De doelstellingen zijn behaald en de resultaten tonen aan dat de ontwikkelde oplossingen effectief zijn voor het detecteren van anomalieën en het bieden van inzichtelijke data-visualisaties. Desondanks zijn er nog uitdagingen en mogelijkheden voor verbetering en toekomstig onderzoek, die zullen bijdragen aan het verfijnen van de huidige oplossingen en het uitbreiden van de functionaliteiten. Het project heeft een solide basis gelegd voor verdere ontwikkeling en implementatie van geavanceerdere data-analysetools en applicaties in de mobiliteitssector.

Referentielijst

- [1] "Duurzame mobiliteit," CBS, 2023. [Online]. Available: <https://longreads.cbs.nl/klimaatverandering-en-energietransitie-2023/duurzame-mobiliteit/>. [Accessed: Aug. 16, 2024].
- [2] Universiteit Antwerpen, "Duurzame mobiliteit," 2024. [Online]. Available: <https://www.uantwerpen.be/nl/projecten/global-change-ecology/duurzame-mobiliteit/>. [Accessed: 10-Aug-2024]., [Online].
- [3] "J. Baguet, E. Köse, R. Qaf, and Z. Hublet, "Literatuurstudie en resultaten enquête," M pact, 15-Nov-2023. [Online]. Available: https://www.duurzame-mobiliteit.be/sites/default/files/inline-files/SMEP_Literatuurstudie_en_enqu%C3%AAt_met_bijlage.pdf. [Acces," [Online].
- [4] "European Parliament, "Vervoer en toerisme," 2011. [Online]. Available: https://www.europarl.europa.eu/RegData/etudes/etudes/join/2011/460064/IPOL-TRAN_ET%282011%29460064%28SUM01%29_NL.pdf. [Accessed: 10-Aug-2024].," [Online].
- [5] ""EEVEE Mobility | LinkedIn," LinkedIn, [Online]. Available: <https://be.linkedin.com/company/eeveemobility>. [Accessed: 11-Aug-2024].," [Online].
- [6] ""EEVEE Mobility | Wisemen," Wisemen, [Online]. Available: <https://wisemen.digital/case/eevee-mobility/>. [Accessed: 11-Aug-2024].," [Online].
- [7] ". W. [A. h.-o.-e.-m. [1.-A.-2. "Product Owner - EEVEE Mobility. [Online].
- [8] S. Bruers en B. Verbeeck, "De berekening van de ecologische voetafdruk voor Vlaanderen," Ecolife, MIRA/2010/01, juni 2010. [Online]. Available: http://www.agripress.be/_STUDIOEMMA_UPLOADS/downloads/218_0.pdf.
- [9] "Analyse van het concept voetafdruk in een Vlaamse beleidscontext," Vlaamse overheid, november 2021. [Online]. Available: <https://omgeving.vlaanderen.be/sites/default/files/2021-11/Analyse%20van%20het%20concept%20voetafdruk%20in%20een%20Vlaamse%20beleidscontext.pdf>.
- [10] "Calma, J. (2024, May 29). How AI could change EV charging. The Verge.] <https://www.theverge.com/2024/5/29/24162389/ai-ev-charging-pilot-study-university-of-michigan-utilidata>," [Online].

- [11 "Scikit Learn, "sklearn.cluster.KMeans," 2024. [Online]. Available: <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>. [Geopend 20 maart 2024].," [Online].
- [12 "Scikit Learn, "sklearn.cluster.DBSCAN," 2024. [Online]. Available: <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html>. [Geopend 20 maart 2024].," [Online].
- [13 "Scikit Learn, "sklearn.ensemble.IsolationForest," 2024. [Online]. Available: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html>. [Geopend 20 maart 2024].," [Online].
- [14 "Google Scholar, "Mieke Jans," 2024. [Online]. Available: https://scholar.google.com/citations?view_op=list_works&hl=nl&hl=nl&user=2F8kQzYAAA. [Geopend 21 maart 2024].," [Online].
- [15 "TensorFlow, "Basic classification: Classify images of clothing," 2023. [Online]. Available: <https://www.tensorflow.org/tutorials/keras/classification>. [Geopend 21 maart 2024].," [Online].
- [16 A. Author, "K means Clustering - Introduction," GeeksforGeeks, Mar. 11, 2024. [Online]. Available: <https://www.geeksforgeeks.org/k-means-clustering-introduction/>. [Accessed: Aug. 16, 2024].
- [17 A. Author, "K-Means Clustering Algorithm," Javatpoint. [Online]. Available: <https://www.javatpoint.com/k-means-clustering-algorithm-in-machine-learning>. [Accessed: Aug. 16, 2024].
- [18 A. Author, "K-Means Clustering in Machine Learning: A Deep Dive," Datascientest. [Online]. Available: <https://datascientest.com/en/k-means-clustering-in-machine-learning-a-deep-dive>. [Accessed: Aug. 16, 2024].
- [19 A. Author, "DBSCAN Clustering in ML: Density-Based Clustering," GeeksforGeeks. [Online]. Available: <https://www.geeksforgeeks.org/dbscan-clustering-in-ml-density-based-clustering/>. [Accessed: Aug. 16, 2024].
- [20 A. Author, "DBSCAN Clustering Algorithm in Machine Learning," KDnuggets, Apr. 2020. [Online]. Available: <https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html>. [Accessed: Aug. 16, 2024].
- [21 A. Smith, "DBSCAN Clustering Algorithm Demystified," Built In, [Online]. Available: <https://builtin.com/articles/dbscan>. [Accessed: Aug. 16, 2024].
- [22 "DBSCAN Clustering in ML | Density based clustering," Tutorialspoint, [Online]. Available: <https://www.tutorialspoint.com/dbscan-clustering-in-ml-density-based-clustering>. [Accessed: Aug. 16, 2024].
- [23 "What is Isolation Forest," GeeksforGeeks, [Online]. Available: <https://www.geeksforgeeks.org/what-is-isolation-forest/>. [Accessed: Aug. 16, 2024].

- [24 "Anomaly detection using Isolation Forest," GeeksforGeeks, Apr. 2024. [Online]. Available:
] <https://www.geeksforgeeks.org/anomaly-detection-using-isolation-forest/>. [Accessed:
16-Aug-2024].
- [25 "Overview of process mining showing the different process mining perspectives,"
] ResearchGate, [Online]. Available: https://www.researchgate.net/figure/Overview-of-process-mining-showing-the-different-process-mining-perspectives_fig2_220477622.
[Accessed: 16-Aug-2024].
- [26 "Process mining," Passionned, [Online]. Available: <https://www.passionned.com/process-mining/>. [Accessed: 16-Aug-2024].
- [27 "Neuraal netwerk," Wikipedia, [Online]. Available:
] https://nl.wikipedia.org/wiki/Neuraal_netwerk. [Accessed: 16-Aug-2024].
- [28 "Is neuraal netwerk?" Jarnoduursma, [Online]. Available: <https://jarnoduursma.nl/blog/is-neuraal-netwerk/>. [Accessed: 16-Aug-2024].
- [29 "TensorFlow, "An end-to-end platform for machine learning," 2024. [Online]. Available:
] <https://www.tensorflow.org/>. [Geopend 22 maart 2024].," [Online].
- [30 "TensorFlow, "TensorFlow.js is a library for machine learning in JavaScript," 2024. [Online].
] Available: <https://www.tensorflow.org/js/>. [Geopend 22 maart 2024].," [Online].
- [31 "Brain.js, "Brain.js: GPU accelerated Neural networks in JavaScript," 2024. [Online].
] Available: <https://brain.js.org/>. [Geopend 22 maart 2024].," [Online].
- [32 "PyTorch, "PyTorch," 2024. [Online]. Available: <https://pytorch.org/>. [Geopend 22 maart
] 2024].," [Online].
- [33 "TensorFlow," Wikipedia, [Online]. Available: <https://en.wikipedia.org/wiki/TensorFlow>.
] [Accessed: Aug. 16, 2024].
- [34 "TensorFlow," Google Open Source, [Online]. Available:
] <https://opensource.google/projects/tensorflow>. [Accessed: Aug. 16, 2024].
- [35 "First Steps with TensorFlow," Google Developers, [Online]. Available:
] <https://developers.google.com/machine-learning/crash-course/first-steps-with-tensorflow/toolkit>. [Accessed: Aug. 16, 2024].
- [36 "TensorFlow.js," TensorFlow, [Online]. Available: <https://www.tensorflow.org/js>.
] [Accessed: Aug. 16, 2024].
- [37 "Introduction to Machine Learning with TensorFlow.js," Borstch, [Online]. Available:
] <https://borstch.com/blog/development/introduction-to-machine-learning-with-tensorflowjs>. [Accessed: Aug. 16, 2024].
- [38 "Brain.js," Brain.js, [Online]. Available: <https://brain.js.org>. [Accessed: Aug. 16, 2024].
]

- [39 "Machine Learning in JavaScript using Brain.js," Scribbler, [Online]. Available:
] <https://scribbler.live/2023/04/30/Machine-Learning-in-JavaScript-using-Brainjs.html>.
[Accessed: Aug. 16, 2024].
- [40 G. Fish, "Brain.js for Beginners," Dev.to, 2024. [Online]. Available:
] <https://dev.to/gfish94/brainjs-for-beginners-1g77>. [Accessed: Aug. 16, 2024].
- [41 "PyTorch," Wikipedia, 2024. [Online]. Available: <https://en.wikipedia.org/wiki/PyTorch>.
] [Accessed: Aug. 16, 2024].
- [42 "PyTorch," PyTorch.org, 2024. [Online]. Available: <https://pytorch.org>. [Accessed: Aug. 16,
] 2024].
- [43 "Cloud Partners," PyTorch.org, 2024. [Online]. Available: [https://pytorch.org/get-](https://pytorch.org/get-started/cloud-partners/)
] [started/cloud-partners/](https://pytorch.org/get-started/cloud-partners/). [Accessed: Aug. 16, 2024].
- [44 "Ecosystem," PyTorch.org, 2024. [Online]. Available: <https://pytorch.org/ecosystem/>.
] [Accessed: Aug. 16, 2024].
- [45 "G. Baader en H. Krcmar, „Reducing false positives in fraud detection: Combining the red flag
] approach with process mining,” Elsevier, vol. 31, pp. 1-16, 2018.,” [Online].
- [46 "R. Wedge, J. M. Kanter, K. Veeramachaneni, S. M. Rubio en S. I. Perez, „Solving the False
] Positives Problem in Fraud Prediction Using Automated Feature Engineering,” Springer
Link, vol. 11053, 2019.,” [Online].
- [47 T. op den Brouw, "AI wet in aantocht: wat is de AI Act en hoe ga je ermee om?"
] CertificeringsAdvies Nederland. [Online]. Available: [https://certificeringsadvies.nl/ai-wet-](https://certificeringsadvies.nl/ai-wet-in-aantocht-wat-is-de-ai-act-en-hoe-ga-je-ermee-om/)
in-aantocht-wat-is-de-ai-act-en-hoe-ga-je-ermee-om/. [Accessed: Aug. 16, 2024].
- [48 "AI-verordening aangenomen door het Europees Parlement," Digitale Overheid. [Online].
] Available: [https://www.digitaleoverheid.nl/nieuws/ai-verordening-aangenomen-door-](https://www.digitaleoverheid.nl/nieuws/ai-verordening-aangenomen-door-het-europees-parlement/)
het-europees-parlement/. [Accessed: Aug. 16, 2024].
- [49 "EU AI Act treedt in werking op 1 augustus 2024: aandachtspunten en verplichtingen voor
] ondernemingen," ZigZagHR. [Online]. Available: [https://zigzaghr.be/eu-ai-act-treedt-in-](https://zigzaghr.be/eu-ai-act-treedt-in-werking-op-1-augustus-2024-aandachtspunten-verplichtingen-voor-ondernemingen/)
werking-op-1-augustus-2024-aandachtspunten-verplichtingen-voor-ondernemingen/.
[Accessed: Aug. 16, 2024].
- [50 "AI Act," DDMA. [Online]. Available: <https://ddma.nl/legal/wetgeving/ai-act/>. [Accessed:
] Aug. 16, 2024].
- [51 "Een eerste kennismaking met de AI Act," Smals. [Online]. Available:
] <https://www.smals.be/nl/content/een-eerste-kennismaking-met-de-ai-act>. [Accessed:
Aug. 16, 2024].
- [52 "Statik, “Waarom wij zo'n fan zijn van Laravel,” 2024. [Online]. Available:
] <https://www.statik.be/blog/waarom-wij-zon-fan-zijn-van-laravel>. [Geopend 4 maart
2024],” [Online].

- [53 "Conversal, "Laravel development," 2024. [Online]. Available:
] <https://www.conversal.be/webdevelopment-laravel/>. [Geopend 4 maart 2024].," [Online].
- [54 "Working Talent, "Wat is Laravel," [Online]. Available: <https://www.workingtalent.nl/wat-is-laravel>. [Geopend 4 maart 2024].," [Online].
- [55 "Paqt, "De 4 belangrijkste redenen om software te ontwikkelen met Laravel," 2024. [Online].
] Available: <https://paqt.com/de-4-belangrijkste-redenen-om-software-te-ontwikkelen-met-laravel/>. [Geopend 4 maart 2024].," [Online].
- [56 "Kinsta, "Django of Laravel: Welk framework is het beste in 2024?," 2023. [Online].
] Available: <https://kinsta.com/nl/blog/django-of-laravel/>. [Geopend 4 maart 2024].," [Online].
- [57 T. M. M. de Oliveira, "FastAPI," [Online]. Available: <https://fastapi.tiangolo.com>. [Accessed:
] 16-Aug-2024].
- [58 A. M. G. de Oliveira, "Using FastAPI to Build Python Web APIs," Real Python, [Online].
] Available: <https://realpython.com/fastapi-python-web-apis/>. [Accessed: 16-Aug-2024].
- [59 "FastAPI Documentation," APIDog, [Online]. Available: <https://apidog.com/blog/fastapi-documentation/>. [Accessed: 16-Aug-2024].
- [60 "DataGrip," JetBrains, [Online]. Available: <https://www.jetbrains.com/datagrip/>.
] [Accessed: 16-Aug-2024].
- [61 "Eevee Mobility," [Online]. Available: <https://www.eeveemobility.com>. [Accessed: 16-Aug-
] 2024].
- [62 "Connect Tesla," Intercom, [Online]. Available: <https://intercom.help/eevee-mobility/en/articles/5785083-connect-tesla>. [Accessed: 16-Aug-2024].
- [63 "Eevee App," Google Play, [Online]. Available:
] <https://play.google.com/store/apps/details?hl=en&id=com.eevee.app>. [Accessed: 16-Aug-2024].
- [64 "Ford Telematics nu voor alle merken," Bestelauto.nl, [Online]. Available:
] <https://www.bestelauto.nl/nieuws/ford-telematics-nu-voor-alle-merken/17328/>.
[Accessed: 16-Aug-2024].
- [65 "Radio Telematics," [Online]. Available: <https://www.radiustelematics.com/nl-be/>.
] [Accessed: 16-Aug-2024].
- [66 "Wat is telematics?," Verizon Connect, [Online]. Available:
] <https://www.verizonconnect.com/nl/kennisbank/artikel/wat-is-telematics/>. [Accessed:
16-Aug-2024].
- [67 "Eevee Mobility Case," Wisemen Digital, [Online]. Available:
] <https://wisemen.digital/case/eevee-mobility/>. [Accessed: 16-Aug-2024].

- [68 "Home Charging," Eevee Mobility, [Online]. Available:
] <https://www.eeveemobility.com/home-charging>. [Accessed: 16-Aug-2024].
- [69 T. M. M. de Oliveira, "FastAPI," [Online]. Available: <https://fastapi.tiangolo.com>. [Accessed:
] 16-Aug-2024].
- [70 "Download," Agripress, [Online]. Available:
] http://www.agripress.be/_STUDIOEMMA_UPLOADS/downloads/218_0.pdf. [Accessed:
16-Aug-2024].

