Stefan Kubica, Hagen Ringshausen,
Jörg Reiff-Stephan, Marius Schlingelhof (Hrsg.)
2. Automobil Symposium Wildau:
Tagungsband Technische Hochschule Wildau 2017

Thomas Jacob
Technische Hochschule Wildau
thomas.jacob@th-wildau.de

Stefan Kubica
Technische Hochschule Wildau
stefan.kubica@th-wildau.de

EINSATZ VON STREAM-MACHINE-LEARNING AUF FAHRZEUGDATEN

Zusammenfassung

Daten liegen in modernen Fahrzeugen in einer großen Menge vor. Die Daten, welche die Sensoren des Fahrzeuges erzeugen, werden über Bus-Systeme zentral zur Verfügung gestellt und für die jeweiligen Anwendungen von den Steuergeräten ausgewertet. Nach diesem Einsatz werden die Sensordaten verworfen. Eine Herausforderung an die Verwendung jener Daten sind die Historisierung und systematische Auswertung dieser. Dafür werden Data-Mining-Methoden angewandt, welche die internen sowie die externen Daten (Car2X-Kommunikation) in Echtzeit auswerten und mit dem digitalem Gedächtnis abgleichen. Ziel dessen ist die Erkennung von Mustern, zur Identifikation von Situationen (z.B. Verkehrsunfall) und Fahrzuständen (z.B. Autobahnfahrt). Als Ergebnisse sollen der Komfort der Insassen und die Sicherheit des Straßenverkehrs verbessert werden. Potentiell bedrohliche Situationen sollen dabei vor dem Eintreffen erkannt werden, um Fahrer und Fahrzeug entsprechend zu konditionieren.
Das Vorhaben zielt konkret auf ein Echtzeit-Data-Mining von Fahrzeugdaten ab. Dabei wird Stream-Machine-Learning als möglicher Lösungsansatz verwendet, da große Datenmengen vorhanden sind und diese in Echtzeit ausgewertet werden müssen, um potentiell bedrohliche Situationen frühzeitig zu erkennen. Der klassische Batch-Learning-Ansatz ist dazu nicht geeignet, da jener alle Daten mit einmal verarbeiten will. Stream-Machine-Learning kombiniert das Konzept von Datenströmen mit dem Online-Lernen. Dadurch verwendet der Ansatz nur die im Stream empfangenden Daten und zudem lernt er während des Betriebs weiter. Nach jeder verarbeiteten Instanz ist es dem Algorithmus möglich ein Modell auszugeben bzw. das vorhandene Modell wird weiter trainiert. Ein selbstständiges Lernen wird ermöglicht.
Für die Umsetzung ist eine Toolchain erstellt worden, die das Vorhaben mit vom Fahrzeug erzeugten und über die OBD2-Schnittstelle bereitgestellten Datensätzen simuliert darstellt. Dabei ist besonderes Augenmerk auf die Komptabilität zu in Fahrzeugen genutzten Betriebssysteme (Linux, Autosar [WaPa12]) gelegt, damit eine Migration gewährleistet wird.

doi: 10.15771/ASW_2017_6

1. Einführung - Nutzen Data-Mining auf Fahrzeugdaten

In der Automobilbranche ist neben der Elektromobilität das autonome Fahren das Trendthema. Dadurch entstehen neue Anforderungen an die Fahrzeuge und die Sensorik dieser. Wobei die Ansprüche je nach Automatisierungsgrad ansteigen. [BAST12] Auch aus diesem Grund ist in modernen Fahrzeugen eine Vielzahl von Sensoren verbaut und eine große Datenmenge fällt an. Die Bus-Systeme, wie beispielsweise der CAN-Bus, stellen die Daten der Sensoren zentral zur Verfügung, sodass Steuergeräte diese nutzen können. Die Steuergeräte werten die Sensordaten aus, wobei auch per Sensorfusion Daten verschiedener Sensoren an einem Steuergerät verwendet werden können [Vonr06, S.25ff], und steuern gegebenenfalls Aktoren. Die Aktoren können aktiv eingreifen (Beispielswese das ABS) oder passiv (Beispielsweise eine ESP-Warnleuchte) wirken.

Moderne Fahrzeuge nutzen vermehrt auch externe Daten, wie exemplarisch über die Car2X-Kommunikation empfangende Werte, für die Analyse und Steuerung. Bei der Car2X-Kommunikation wird auf den Standard IEEE 802.11p zurückgegriffen [IEEE10]. Die externen Daten ermöglichen Erkenntnisse, die fahrzeugintern durch die Sensorik nicht ermittelbar sind. Als Beispiel ist ein Hindernis hinter einer Kurve zu nennen. Trotz vorhandener Sensorik, die auch Frontkameras und Radar-Systeme beinhaltet, ist dies für das Fahrzeug alleine nicht festzustellen. Durch den zusätzlichen Informationsgewinn bietet die Car2X-Kommunikationen ein großes Potential.

Ein weiteres Potential, welches das Kernthema dieser Veröffentlichung ist, bietet der Einsatz von Data-Mining-Methoden auf die vorhandenen Fahrzeugdaten. Dabei werden interne und externe Daten verwendet. Heutzutage werden ausschließlich die aktuellen Sensordaten im Fahrzeug betrachtet und nach deren Verwendung werden jene gelöscht. An dieser Stelle setzt die Nutzung von Data-Mining an.

Die folgende Definition beschreibt den aus der Wirtschaft kommenden Begriff: „Unter Data Mining versteht man die Anwendung von Methoden und Algorithmen zur möglichst automatischen Extraktion empirischer Zusammenhänge zwischen Planungsobjekten, deren Daten in einer hierfür aufgebauten Datenbasis bereitgestellt werden.“ [LACKoJ] Diese extrahierten Daten und eine Historisierung dieser, bieten neue Möglichkeiten für das autonome und für das fahrergelenkte Fahren. Durch den Einsatz von Data Mining werden Muster innerhalb der Sensordaten erkannt und Situationen, wie das Abkommen von der Straße, sowie Fahrzustände, wie schlechter Fahruntergrund, detektierbar. Eine Wiedererkennung der Muster ermöglicht eine Rekonstruktion von Situationen. Dadurch soll ein kurzer Blick (Millisekundenbereich) in die Zukunft geworfen werden und proaktiv agiert werden. Bei Situationen, die ein Eingreifen benötigen, kann somit unterstützend eingegriffen werden. Im Idealfall wird diese Situation abgewandt aber auch das richtige Reagieren, auf die Lage, wenn sie nicht mehr zu verhindern ist, kann helfen, um im Extremen Schäden und Verletzungen zu mindern.

Der Abgleich der aktuellen Lage mit einem digitalen Gedächtnis und der Einsatz von maschinellen Lernen (Machine-Learning) sind entscheidende Bestandteile des Ansatzes.

2. Einsatz Stream-Machine-Learning auf Fahrzeugdaten

Die Anforderungen an ein System, welches die Sensordaten von den Bus-Systemen verarbeiten soll sind hoch. Grund für die erhöhten Ansprüche sind einerseits die großen Datenmengen und anderseits die Tatsache, dass eine Echtzeitanalyse notwendig ist, um zeitnah konditionieren zu können, wenn es angebracht ist. Beispielsweise schätzt Prof. Dr.-Ing. Christian Wietfield den täglichen Datentransfer über den CAN-Bus auf 12 GByte [Nest15]. Es wird deutlich, dass einfaches Batch-Machine-Learning (klassische Stapelverarbeitung) bei diesen Datenmengen an die Grenzen stößt. Die Tatsache, dass bei dieser Methode alle Daten auf einmal gelernt werden, ist der Grund für das Nichterfüllen der Anforderungen.
Die Verbindung von Datenströmen, sogenannte Streams, mit dem Online-Learning ist eine potentielle Lösung für diese Problemstellung bzw. die erhöhten Ansprüche. Datenströme definieren sich als eine geordnete, in der Länge meist unbeschränkte, Sequenz von x1…xn Datenelementen, die in Echtzeit verarbeitet werden müssen, nicht wahlfrei zugreifbar sind und nur einmal oder in geringer Zahl erneut gelesen werden können.
Durch das Stream-Learning werden kontinuierlich Daten verarbeitet und jede Instanz wird dazu genutzt das vorhandene Modell weiter zu trainieren, um somit die Ergebnisse zu verbessern. Das weitere Trainieren des Modells mit jeder Instanz wird als Online-Learning bezeichnet. Durch die Kombination von Stream- und Online-Learning entsteht der Ansatz des Stream-Machine-Learnings.
Stream-Machine-Learning wird in den letzten Jahren vermehrt angewendet und mithilfe des Prinzips werden beispielsweise Finanztransaktionen oder Netzwerkbewegungen untersucht. Diese Szenarien lassen sich durch das Konzept der Datenströme besser abbilden und beschreiben. Bei jenen Einsatzfällen entstehen, ebenso in diesem Vorhaben, große Datenmengen in kurzer Zeit. Auch hier ist die herkömmliche maschinelle Verarbeitung überfordert. Stream-Learning-Algorithmen dienen dagegen der Datenstromanalyse, indem sie das zugrundeliegende Modell zyklisch mit neuen Daten aktualisieren. Nach jeder verarbeiteten Instanz erzeugen die Algorithmen ein neues Modell, was als Basis dient.
Das Datenstromparadigma ist in gewisser Hinsicht eine weitere Evolutionsstufe der Datenanalyse. Machine Learning hat die Aufgabe, automatisiert Informationen aus großen Daten-beständen zu extrahieren [GrLi05, S.11]. Es ist in der Lage, Regeln aus einer kleinen Menge Trainingsdaten zu lernen, setzt allerdings voraus, dass alle zu verarbeitenden Daten im Speicher liegen.
Da dies unter Umständen nicht ge-währleistet werden kann, beschäftigt sich das Data Mining mit Wegen, die Rechenzeit und den Speicherbedarf großer statischer Datenmengen zu reduzieren. Dazu können beispielsweise die Datensätze unterteilt oder ausgelagert werden.

Dadurch wird der statische teure Prozess des maschinellen Lernens zu einem idealerweise mit der Anzahl der Instanzen skalierenden Prozess umgeformt. Es wird allerdings nicht das Problem der kontinuierlichen Daten, wie sie in der Realität in Anwendung auftreten, gelöst. Üblicherweise kann ein Modell, einmal trainiert, nicht durch neue Instanzen erweitert werden.
Hier setzt das Konzept der Datenströme an, das mit Echtzeitdaten, die den vorhandenen Speicher weit überschreiten arbeiten kann. Eingehende Instanzen werden einmalig betrachtet und dazu genutzt das Modell inkrementell zu trainieren. Der Vorteil dieses System liegt darin, dass das Model, welches nur aus wenigen Beispielen trainiert wurde, stetig weiter ausgebaut werden kann.
In diesem Vorhaben wird die Kombination von Datenströmen in Verbindung mit Online-Learning genutzt, um die großen Datenmengen, welche im Fahrzeug anfallen, zu verarbeiten und zu analysieren. Die Einsparung von Zeit sowie Speicherplatz sind dabei die Hauptargumente für die Verwendung des Stream-Machine-Learnings.

3. Vorhabenbeschreibung

In dem Vorhaben sollen mithilfe des Stream-Machine-Learnings die kontinuierlich auftretenden Daten über die Bus-Systeme eines Fahrzeuges analysiert werden. Als Ergebnis der Analyse sollen Muster erkannt werden, die aktuelle Fahrzustände erkennbar machen. Darüber hinaus soll ein kurzer Blick in die Zukunft geworfen werden, um potentiell gefährliche Situationen frühzeitig zu erkennen. Dadurch sollen Gegenmaßnahmen eingeleitet werden, die den Fahrer (z.B. optische, akustische oder haptische Warnungen) und/oder das Fahrzeug (z.B. Steuerungen der Aktoren, wie beispielsweise die Bremse oder der Gurtstraffer) konditionieren. Als Folge der Konditionierung sollen mögliche Verletzungen der Insassen oder Schäden am Fahrzeug minimiert werden.
Die folgende Abbildung zeigt die schematische Darstellung der Integration im Kontext des Fahrzeuges.

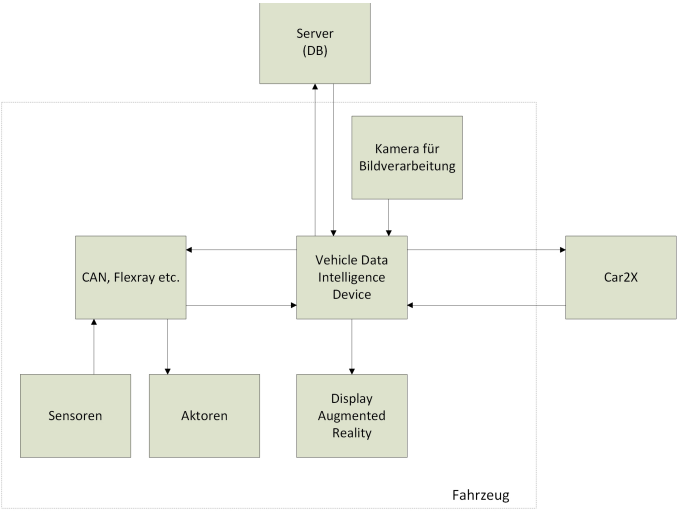

Abbildung 1: Systemintegration und -Aufbau des Vehicle Data Intelligence Devices [JaKu16a]

3.1 Toolchain von Erzeugung bis Auswertung der Daten

Für die Umsetzung des Vorhabens werden im ersten Schritt Beispieldatensätze benötigt, um mögliche Algorithmen und Systeme auf ihre Nutzbarkeit zu testen. Für die Erzeugung dieser Datensätze wird die in den meisten modernen Fahrzeugen frei zugängliche OBD2-Schnittstelle verwendet [GURSoJ]. Über die Schnittstelle können die Sensordaten in nahezu Echtzeit ausgelesen und archiviert werden. Dafür können unter anderem spezielle Adapter genutzt werden, welche die Daten per Bluetooth oder WLAN an Endgeräte wie Smartphones oder Laptops senden. Dort nimmt sie eine App oder eine Software auf und ein Datensatz entsteht. Allerdings sind die Datensätze unvollständig, da nicht alle Sensoren angesprochen werden und die Ergebnisse unterscheiden sich stark nach Fahrzeugtyp sowie -alter. Trotzdem genügen die aufgenommenen Daten für eine Analyse und die Anwendung von Data-Mining-Algorithmen.

Für die bessere Nutzung sollen diverse Fahrszenarien aufgezeichnet werden, um die Algorithmen in verschiedenen Situationen zu testen. Dadurch entstehen chronologisch sortierte Datensätze der über die OBD2-Schnittstelle zur Verfügung gestellten Sensordaten. Dies ist die Basis für die vorgestellte Toolchain.
Der Einsatz der Toolchain dient der Simulation für die angestrebte Umsetzung innerhalb des Fahrzeuges. Die Simulation soll auf einem normalen Arbeitsplatzcomputer durchgeführt werden. Dabei ist das Ziel die aus der OBD2-Schnittstelle gewonnenen Datensätze in einem Stream bereitzustellen, diese in Echtzeit zu analysieren und Muster zu erkennen. Der Schritt der Aktorensteuerung ist nicht Bestandteil der Kette.
Bei der Toolchain ist die höchste Priorität, dass eine Migration in das Fahrzeug möglich ist. Für in Fahrzeug eingesetzte Analyseboxen wird in der Regel ein Linux-Betriebssystem und (Adaptive) Autosar verwendet. Daher ist die Grundvoraussetzung an die verwendeten Werkzeuge und Software-Lösungen, dass sie auch in einem solchen System migriert werden können. Zudem müssen die eingesetzten Programme die genannten Anforder-ungen an das Stream-Machine-Learning erfüllen.
Die schematische Abbildung zeigt die Bestandteile der Toolchain. In der Folge werden die einzelnen Programme kurz präsentiert und deren Bedeutung für die Kette dargestellt:

- Apache Maven: stellt die Schnittstelle zwischen der Programmierumgebung und dem Computing-System Apache Spark dar. Es stellt sicher, dass die Programme standardisiert bereitgestellt werden und es erzeugt den richtigen Programmaufbau. In diesem Vorhaben dient es für die Verbindung der Java-Api von Apache Spark mit dem Programmierframework Eclipse. Durch die lizenzfreie Nutzbarkeit, den offenen Quellcode, die Plattformunabhängigkeit und die Tatsache, dass Apache Maven Teil der Apache Software Foundation ist, gilt, ebenso wie bei den anderen Toolchain-Elemente, das eine Migration auf andere Systeme möglich ist. [APACoJd]

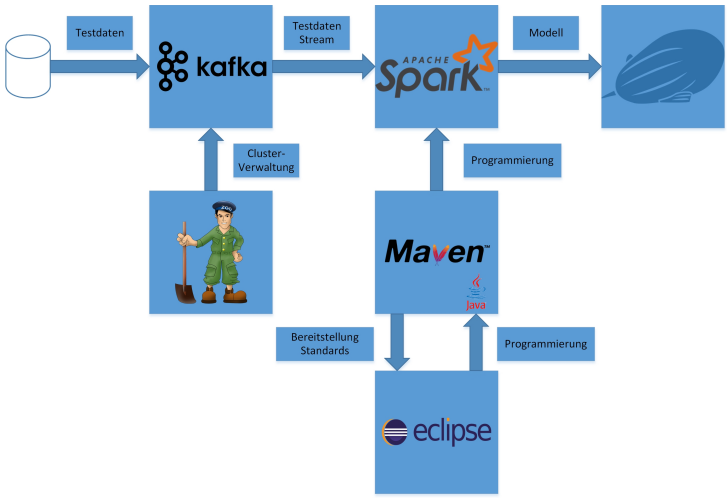

Abbildung 2: Toolchain für Simulation der Datenanalyse von Fahrzeugtestdaten

- Bei Apache kafka handelt es sich um eine Streaming Plattform. Die Software-Lösung ermöglicht das kontinuierliche Versenden von Datenströmen. Die Software ist lizenzfrei nutzbar, open source, plattformunabhängig und gehört der Apache Software Foundation an [APACoJa]. Im Vorhaben soll sie die Testdaten im Stream bereitstellen. Dadurch ist sie neben Apache Spark der wichtigste Bestandteil der Toolchain.
- Eclipse ist das Programmier-framework für das Vorhaben. In Eclipse wird Apache Maven integriert und auf diese Weise wird sichergestellt, dass standardisierte Programmaufbauten und die Verbindung zu Apache Spark gegeben sind. Das Framework gibt es angepasst für diverse Programmiersprachen und es ist durch Plugins modular erweiterbar. In diesem Fall wird es für die Programmierung von Java eingesetzt. Eclipse ist platformunabhängig, lizenzfrei nutzbar und open source. [THEEoJ]
- Apache Zeppelin ist optionaler Bestandteil der Toolchain. Die Software dient der Veranschaulichung für die Ergebnisse. Apache Zeppelin erhält das Modell und die Werte von Apache Spark und ermöglicht eine Aufbereitung dieser. Dadurch können die Ergebnisse anschaulicher dar-gestellt werden. Auch für Apache Zeppelin gilt, dass es lizenzfrei nutzbar, open source, plattform-unabhängig und Teil der Apache Soft-ware Foundation ist [APACoJe].
- Apache Spark ist ein universell einsetzbares Cluster-Computing-System, welches u.a. Stream Learning sowie Machine Learning ermöglicht. Dafür stellt es APIs in den Program-miersprachen Scala, Java, Python und R zur Verfügung. Dadurch stellt das System den zentralen logischen Punkt dieser Toolchain dar. Auch für Apache Spark gilt, dass es lizenzfrei nutzbar, open source, plattformunabhängig und Teil der Apache Software Foundation ist. [APACoJc]

- Die eigentliche Aufgabe des Apache ZooKeepers ist die Verwaltung und Synchronisation von verteilten Systemen sowie Diensten. Er wird für die Verwaltung von Clustern eingesetzt. Die Software ist lizenzfrei nutzbar, open source, plattformunabhängig und gehört ebenso der Apache Software Foundation an [APACoJb]. Der Apache ZooKeeper wird zwingend für die Nutzung von Apache kafka benötigt, um den Dienst bereitzustellen. Obwohl im Vorhaben nur ein Computer zum Senden und Empfangen des Streams eingesetzt wird, ist die Software unabdinglich.

Die vorgestellte Toolchain stellt eine Möglichkeit dar, wie Testdaten aus einem Fahrzeug im Datenstrom versandt, mithilfe von Stream-Machine-Learning ausgewertet und die Ergebnisse dargestellt werden können. Sie ist auf beliebige Systeme migrierbar und erfüllt die gestellten Grundvoraussetzungen.

4. Ausblick

Die Forschung im Bereich des Data Minings auf Fahrzeugdaten in Echtzeit ist nur gering vorhanden, weshalb ein hohes Potential in der Anwendung existiert. Die frühzeitige Erkennung von Situationen, die ein Eingreifen benötigen, ist ein großes Ziel der Automobilbranche. Durch den Ansatz wird das kooperative Fahren unterstützt, was essentiell für das automatisierte Fahren ist.
Die vorgestellte Toolchain ist fertig eingerichtet und die einzelnen Bestandteile funktionieren. Somit ist der theoretische Grundstein für die Entwicklung der Simulation von Echtzeit-Data-Mining auf Fahrzeugdaten gelegt. In Zukunft muss die Kette vollständig mit Daten durchgetestet werden. Dazu werden Testdaten unterschiedlicher Fahrszenarien über die OBD2-Schnittstelle aufgenommen. Als Ergebnis wird eine vollständige Simulationsum-gebung geschaffen.

5. Literaturverzeichnis

[APACoJa]
Apache Software Foundation: Apache kafka Documentation. https://kafka.apache.org/docu-mentation.html. Abruf am 2016-01-30.

[APACoJb]
Apache Software Foundation: Apache Zoo-Keeper.https://zookeeper.apache.org/. Abruf am 2016-01-30.

[APACoJc]
Apache Software Foundation: Spark Overview., http://spark.apache.org/docs/latest/index.html. Abruf am 2016-01-30.

[APACoJd]
Apache Software Foundation: Apache Maven Project. http://spark.apache.org/docs/latest/ index.html. Abruf am 2016-01-30.

[APACoJe]
Apache Software Foundation: Apache Zeppelin. , https://zeppelin.apache.org/. Abruf am 2016-01-30.

[BAST12] Bundesanstalt für Straßenwesen: Rechtsfolgen zunehmender Fahrzeugautoma-tisierung. 2012, http://www.bast.de/DE/ Publikationen/Foko/Downloads/2012-11.pdf. Abruf am 2016-01-25.

[GrLi05] Großekathöfer, Ulf; Lingner, Thomas: Neue Ansätze zum maschinellen Lernen von ignments (Diplomarbeit). Universität Bielefeld, Bielefeld, 2005.

[GURSoJ]
Gurskij, Wladimir: obd-2.de. – OBD-2 Allgemeines, technische Informationen. https://www.obd-2.de/obd-2-allgemeine-infos.html. Abruf am 2017-02-15

[IEEE10]
IEEE Computer Society: Amendment 6: Wireless Access in Vehicular Enviroments. 2010, https://www.ietf.org/mail-archive/web/its/current/pdfqf992dHy9x.pdf. Abruf am 2015-12-12

[JaKu16a]
Jacob, Thomas; Kubica, Stefan: Einsatz von Data-Mining-Methoden auf Fahrzeugdaten, für die frühzeitige Erkennung bedrohlicher Situationen – In: 17.Nachwuchswissenschaftlerkonferenz Tagungsband. Schmalkalden, 2016. S 324-328

[JaKu16b]
Jacob, Thomas; Kubica, Stefan: Realtime-Data-Warehouse für Fahrzeugdaten nutzen – In: Prozesse, Technologie, Anwendungen, Systeme und Management 2016. Schmalkalden, 2016. S 173-182

[LACKoJ]
Lackes, Richard: Gabler Wirtschaftslexikon. – Data Mining. http://wirtschaftslexikon.gabler.de/Definition/data-mining.html. Abruf am 2017-02-09

[NEST15]
Nestler, Yvonne: Funkschau.de. - Echtzeit für das vernetzte Auto. 2015, http://www.funkschau.de/mobile-solutions/artikel/122746/1/. Abruf am 2017-01-26

[THEEoJ]
The Eclipse Foundation. – Desktop IDEs. http://www.eclipse.org/ide/. Abruf am 2017-01-30

[VONR06]
von Rosenberg, Harald: Sensorfusion zur Navigation eines Fahrzeugs mit low-cost Inertialsensorik (Diplomarbeit). Universität Stuttgart, Stuttgart, 2006.

[WaPa12]
Wandling, Florian; Pallierer, Roman: All-electronics.de. – Autosar 4.0 kommt ins Rollen. 2012, http://www.all-electronics.de/autosar-4-0-kommt-ins-rollen/. Abruf am 2017-02-13