Hauptseminar Winter 2020
Master Informatik

Anomalieerkennung im Fahrzeugnetzwerk

Wilhelm Schumacher

Eingereicht am 29.2.2020

Zusammenfassung Mit dem immer weiter voranschreitenden Ausbau der Kommunikation im internen Netzwerkes des Autos steigt auch gleichzeitig der Bedarf an IT Security Konzepten im Auto. Zum Beispiel verändert der Einsatz von Echtzeit-Ethernet mit TSN im zentralen Backbone des Netzes und auf lange Sicht die Entwicklung hin zu einem flachen Ethernet-Netzwerk die Kommunikationsstrukturen des Netzes. Zusätzlich erfolgt im Moment eine Öffnung des internen Netzes hinzu vermehrter Kommunikation mit der Außenwelt. Ein Security Konzept, das in diesem Kontext im Moment wachsende Aufmerksamkeit erhält, ist die Entdeckung von Angriffen durch Anomalieerkennung. Unterschiedliche Verfahren zur Anomalieerkennung werden bereits in anderen Domänen wie zum Beispiel bei der Erkennung von Kreditkartenbetrug, im medizinischen Bereich, für Fehlererkennungen im Raumschiff (Safety) oder klassisch für Intrusion Detection Systeme, die Angriffe auf Netzwerke entdecken sollen, erfolgreich eingesetzt. Das langfristige Ziel ist es Herauszufinden, wie praktikabel der Einsatz von Anomalieerkennung im Fahrzeug ist und welche der bekannten Algorithmen besonders gut für die Erkennung von Anomalien innerhalb der Automotive Security Domäne geeignet sind. Diese Ausarbeitung gibt erstmal einen geordneten Überblick zur den Grundlagen der Anomalieerkennung und grenzt die unterschiedlichen Methoden voneinander ab. Anschließend ist das Ziel dieser Hauptseminar Ausarbeitung wichtige Aspekte hinsichtlich der Anomalieerkennung im automotiven Kontext hervorzuheben und das Widerspiegeln des aktuellen Forschungsstandes aus diesem Gebiet.

Schlüsselwörter Automotive Security · Anomalieerkennung · Anomaly Detection · Anomalieerkennung im Fahrzeugnetzwerk · Methoden der Anomalieerkennung · Kategorien Anomalieerkennung

Wilhelm Schumacher
E-Mail: Wilhelm.Schumacher@haw-hamburg.com
Department Informatik
Fakultät Technik und Informatik
Hochschule für Angewandte Wissenschaften Hamburg

Inhaltsverzeichnis

1 Einleitung

Moderene Fahrzeugnetzwerke bestehen aus einer Vielzahl verteilter Steuereinheiten (ECUs) [vgl. SNA+13, Seite 9], die über verschiedene Kommunikationsmedien systemweit viele Informationen über Betriebszustände und weitere relevante Daten miteinander austauschen. Typischerweise enthält ein modernes Auto über 70 verschiedene ECUs. Zur Verbindung der ECUs werden als Kommunikationsmedien entweder verschiedene Systembusse wie CAN, LIN, MOST, FlexRay oder vor allem in neueren Fahrzeugen Ethernet-Netzwerke [HMVVDK13], die Systembusse schrittweise ersetzen sollen, eingesetzt. Auch die Angriffsoberfläche (attack surface) des Fahrzeuges hat sich durch neue Schnittstellen, wie einer Internetverbindung zur Außenwelt, Car-To-X, OBD-II, Bluetooth usw., mit der Zeit immer mehr erweitert [vgl. CMK+11, Seite 2-4]. Diese Interfaces bieten das Potential für die Umsetzung einer Vielzahl von neuen Funktionen. Mit Updates over the Air kann zum Beispiel das Kartenmaterial des autonomen Fahrzeuges aktualisiert werden, mit Car-To-X Kommunikation die Motorsteuerung an die aktuelle Verkehrslage angepasst werden oder mit Hilfe der Cloud der nächste Ladezyklus des Elektroautos inklusive Reservierung und Einbeziehung des Kalenders besser geplant werden.

Jedoch führt die Öffnung des internen Kommunikationsnetzes nach Außen auch zu einer erhöhten Verwundbarkeit der Informationssicherheit (Integrität, Vertraulichkeit, Verfügbarkeit) im internen Netzwerk des Fahrzeuges [vgl. MA11, Seite 1]. Denn durch die Einbindung der neuen Interfaces wird das Auto zu einem attraktiven Angriffsziel für Hacker. Hacker könnten potentielle Schwachstellen in einem der externen Interfaces ausnutzen, um böswillige Pakete in das Netzwerk einzuschleusen, die den normalen Betriebsablauf des Fahrzeuges stören sollen. Zum Beispiel sind die Steuergeräte (ECUs) angreifbar und können nach Kompromittierung genutzt werden, um die gesamte Kommunikation zu manipulieren. Deswegen sind IT Security Konzepte zum Schutz des Fahrzeuges notwendig. Ein Security Konzept, das in diesem Kontext im Moment wachsende Aufmerksamkeit erhält, ist die Entdeckung von Angriffen durch Anomalieerkennung. Unterschiedliche Verfahren zur Anomalieerkennung werden bereits in anderen Domänen wie zum Beispiel bei der Erkennung von Kreditkartenbetrug, im medizinischen Bereich, für Fehlererkennungen im Raumschiff oder klassisch für Intrusion Detection Systeme, die Angriffe auf Netzwerke entdecken sollen, erfolgreich eingesetzt. Einige Algorithmen wurden dabei speziell für diese bestimmten Anwendungsbereiche entwickelt, abhängig von Art der Daten, der Verfügbarkeit von gelabelten Trainingsdaten, den Typen von Anomalien und dem gewünschten Outputformat [vgl. CBK09, Seite 6-7]. In der automotiven Domäne könnte sich der Einsatz von Anomalieerkennung lohnen, da die Verarbeitung auf Grund der Menge an Daten automatisiert durchgeführt werden muss und die Nachrichtenmuster genug Ähnlichkeiten und Zusammenhänge aufweisen um sie von anormalen Daten zu unterscheiden. Weiterhin sind die unterschiedlichen Ebenen, wie z.B. auf Anwendungsebene, im Netzwerk, in den Sensoren oder im GPS Signal auf denen die Erkennung der Anomalien durchgeführt werden kann, eine Besonderheit der Anomalieerkennung aus dem automotiven Bereich.

Das Ziel dieser Hauptseminar Ausarbeitung ist es einerseits den aktuellen wissenschaftlichen Stand zum Thema Anomalieerkennung im automotiven Kontext hervorzuheben und andererseits wichtige Aspekte hinsichtlich der Anomalieerkennung im Fahrzeug darzustellen. Langfristig soll so ein Beitrag zu dem Ziel Herauszufinden, wie praktikabel der Einsatz von Anomalieerkennung im Fahrzeug ist und welche der bekannten Algorithmen besonders gut für die Erkennung von Anomalien innerhalb der Automotive Security Domäne geeignet sind, entstehen (weiterer Verlauf meines Masters). Dazu gibt diese Ausarbeitung erstmal einen geordneten Überblick zu Methoden der Anomalieerkennung aus der Literatur und zeigt auf wie diese in Kategorien unterteilt werden können. Weiterhin wird noch ein kurzer Überblick zur Anomalieerkennung allgemein und dem Aufbau eines Fahrzeugnetzwerkes gegeben. Auf Basis dieser Grundlagen wird dann ein Blick in die Literatur vorgenommen und Aspekte, die spezifisch für Anomalieerkennung in automotiven Kontext sind, herausgearbeitet. Die Arbeit findet innerhalb des SecVI Forschungsprojekt [siehe Pro19] statt. Das SecVI Projekt wird in Verbindung mit Industriepartnern durchgeführt und vom Federal Ministry of Education und Research gefördert und hat als Ziel eine sichere Netzwerkarchitektur für das Auto zu entwickeln.

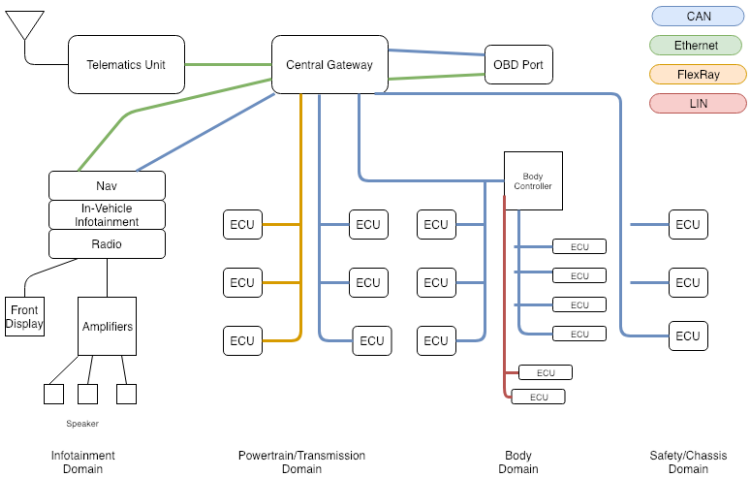
Die folgende Ausarbeitung ist so strukturiert, dass zuerst das Kapitel **Grundlagen** relevante Grundlagen zum Thema Anomalieerkennung und dem Aufbau eines Fahrzeugnetzwerkes beschreibt. Anschließend zeigt das Kapitel **Methoden der Anomalieerkennung** wie Verfahren der Anomalieerkennung innerhalb der Literatur in Kategorien geordnet werden können und hebt die wichtigsten Charakteristika der einzelnen Kategorien hervor. Das Kapitel **Aspekte der Anomalieerkennung im automotiven Kontext** erläutert Besonderheiten, die die Anomalieerkennung im Fahrzeug betreffen, und diese von anderen Domänen unterscheidet. Daraufhin spiegelt das Kapitel **Anomalieerkennung in der Literatur im automotiven Kontext** den aktuellen Stand der Forschung wieder und das letzte Kapitel **Ausblick und Zusammenfassung** schließt die Ausarbeitung dann mit einer kurzen Zusammenfassung der wichtigsten Punkte der Ausarbeitung und einem Ausblick hinsichtlich des Hauptprojektes und des weiteren Verlaufs meines Masters ab.

2 Grundlagen

Das Kapitel Grundlagen ist in die beiden Unterkapitel 2.1 **Fahrzeugnetzwerk und Angriffsoberfläche** und 2.2 **Überblick Anomalieerkennung** unterteilt. Das erste Kapitel erläutert die notwendigen Grundlagen zum Aufbau des internen Fahrzeugnetzwerkes und zeigt anschließend noch die Vielzahl an Schnittstellen zur Außenwelt des Autos auf. Im zweiten Teil folgt ein Überblick zur Anomalieerkennung mit allen notwendigen Grundlagen, die für das Verständnis der folgenden Kapitel benötigt werden. Unter anderem beschreibt es wichtige Aspekte zu den Zielen, Input Daten, Typen von Anomalien und verschiedenen Lernmethoden.

2.1 Fahrzeugnetzwerk und Angriffsoberfläche

Eine wichtige Komponente im internen Kommunikationsnetzwerk der aktuellen Oberklasse Autos sind die **Electrical Control Units (ECU)** [vgl. SNA+13, Seite 2-3]. Sie sind verantwortlich für die Steuerung bestimmter Funktionalitäten, wie zum Beispiel für die Lenkung, die Bremse oder auch für die Komfortfunktionen des Sitzes. Je nach Aufgabe der ECU, verfügt sie auch über sehr unterschiedliche Mengen an Rechenleistung. Typischerweise enthält ein modernes Auto über 70 verschiedene ECUs. Zur Kommunikation der einzelnen Steuereinheiten werden Bussysteme mit mehreren Teilnehmern im internen Netzwerk eingesetzt. Das am meisten verwendete Bussystem ist der **CAN-Bus (Controller Area Network)** [siehe Bos19]. Weitere Bussysteme, die im Auto zum Einsatz kommen, sind **LIN (Local Interconnect Network)** [siehe Aut19], **MOST (Media Oriented Systems Transport)** [siehe Coo19] und **FlexRay** [siehe Fle19]. Abbildung 1 stellt beispielhaft eine domänenbasierte Architektur mit ECUs gruppiert nach Funktionen und verschiedenen Bussystemen dar. In der Zukunft spielt **Automotive Ethernet** in der Netzwerkarchitektur des Autos eine immer größere Rolle [vgl. Lev18]. In der abgebildeten Architektur wird Ethernet hauptsächlich für Infotainment, die Verbindung zum OBD Port und für die Verbindung zur Telematics Unit verwendet. Dieser Einsatzbereich stellt nur die Anfänge von Ethernet in der Netzwerkarchitektur des Fahrzeuges dar. Langfristig wird sich die dargestellte domänenbasierte Architektur hinzu einer zonalen Architektur mit flachen Ethernet-Netzwerk entwickeln [HMVVDK13].



Quelle: basierend auf https://www.iot-now.com/2017/02/27/59018-securing-automotive-air-updates

Abb. 1 Die Abbildung stellt eine typische domänenbasierte Netzwerkarchitektur mit Central Gateway von einem aktuellen Oberklasse Auto dar.

Innerhalb der letzten Jahre hat sich das interne Netzwerk des Autos von einem komplett abgeschotteten Netzwerk hin zu einem aktiven Teilnehmer im Internet der Dinge entwickelt. Durch neue Schnittstellen und Technologien können immer mehr Funktionen umgesetzt werden, bei denen Kommunikation mit anderen Teilnehmern benötigt wird. Zum Beispiel V2V- (Vehicle to Vehicle), V2I- (Vehicle to Infrastructure) oder V2C- (Vehicle to Cloud) Kommunikation. Dabei konsumiert das Auto Dienste von außen und bietet gleichzeitig auch Dienste nach außen hin an. Generell können die Schnittstellen des Fahrzeuges zur Außenwelt anhand ihrer Reichweite in die drei Kategorien **Indirect Physical Access**, **Short-range Wireless Access**, und **Long-range Wireless Access** unterteilt werden [vgl. CMK+11, Seite 2-4] und bilden die Angriffsoberfläche (Surface) des Autos. Zu den Indirect Physical Access Schnittstellen gehören z.B. der Diagnoseport OBD-II, USB-Schnittstellen, Dockingstations, CD-Player oder möglicherweise sogar ein direkter Ethernet-Anschluss. Diese Schnittstellen haben alle gemeinsam, dass direkter Zugriff auf das Fahrzeug benötigt wird, um einen Angriff ausführen zu können. Die Short-range Wireless Schnittstellen bieten Zugriff innerhalb einer Reichweite von 5-300 Meter an. Dazu zählen zum Beispiel Bluetooth, Remote Keyless Entry, Tire Pressure Monitoring System (TPMS), V2V Kommunikation und Internet Hotspots. Als Letztes gibt es noch die Long-range Wireless Schnittstellen, die Kommunikation über große Reichweiten ermöglichen, wie zum Beispiel Radio-Kanäle oder Mobile Daten, die sogar individuelle Adressierung erlauben.

2.2 Überblick Anomalieerkennung

Die Anomalieerkennung hat als Aufgabe unregelmäßige Muster innerhalb von normalen Daten festzustellen. Diese Unregelmäßigkeiten entsprechen dabei nicht dem normalen Verhalten des Systems, treten seltener auf und unterscheiden sich signifikant vom Rest der Daten. Sie werden hauptsächlich als **Anomalien** oder **Outlier** (Ausreißer) [CBK09] bezeichnet. Häufig ist es aber schwierig präzise Grenzen zwischen den normalen und anomalen Daten zu definieren und stellt eine der großen Herausforderungen der Anomalieerkennung dar. Zum Beispiel können Anomalien, die dicht an der Grenze eines normalen Datenbereiches liegen, nur sehr schwierig als Anomalie oder normaler Datensatz eingeordnet werden. Anomalieerkennung wird bereits in einer Vielzahl von unterschiedlichen Anwendungsbereichen eingesetzt. Zum Beispiel bei dem Erkennen von Kreditkartenbetrug, zum Entdecken von Tumoren in MRT-Bilder (Magnetresonanztomographie), im Cyber Security Bereich, dem Erkennen von Cheatern in Videospielen [Pat] oder für die Fehlererkennung bei sicherheitskritischen Systemen [FYM05]. In vielen dieser Anwendungsbereiche spielt die Identifikation von Anomalien eine große Rolle, denn sie ermöglicht es kritische Situationen zu erkennen und daraufhin die notwendigen Gegenmaßnahmen zu veranlassen. Zum Beispiel können ungewöhnliche Nachrichtenmuster in einem Computernetzwerk darauf hinweisen, dass von einem gehackten Computer aus, sensitive Daten in das Netzwerk an einen unautorisierten Host geschickt werden oder anomales Verhalten in einem Sensor eines Raumschiffes könnte auf Fehler in einer Komponente hinweisen.

Abhängig vom Anwendungsbereich kann sich die Art der Input Daten deutlich voneinander unterscheiden. Im Allgemeinen sind die Input Daten erstmal eine Sammlung von Daten (bezeichnet als Objekte, Vektoren, Beobachtungen, Punkte etc.), die wiederum aus einer Menge von Attributen (bezeichnet als Variablen, Felder, Feature etc.) bestehen [vgl. CBK09, Seite 6-7]. Ein Attribut stammt dabei aus einer von drei Kategorien **binär**, **kategorisch** oder **kontinuierlich**. Wenn die Daten nur aus einem Attribut bestehen, werden sie als **Univariate** bezeichnet und wenn die Daten aus mehreren Attributen bestehen als **Multivariate**. Je nachdem welche Arten von Attributen oder Kombinationen von Attributen innerhalb der Daten vorhanden sind, können andere Algorithmen angewendet werden oder die Daten müssen zuvor entsprechend angepasst werden.

Eine Anomalie kann in eine von drei Kategorien [Gmb18] eingeordnet werden. Die erste und einfachste Kategorie wird als **Punktanomalie** bezeichnet. Bei einer Punktanomalie wird ein einziger Datenpunkt in Anbetracht zu den restlichen Daten als Anomalie eingestuft. Ein Beispiel aus der realen Welt für eine Punktanomalie wäre zum Beispiel die Höhe des ausgegebenen Betrages bei Bezahlung mit Kreditkarte. Wenn über einen längeren Zeitraum nur sehr kleine Beträge bezahlt wurden und dann ein im Vergleich zu diesen Beträgen sehr großer Betrag auftritt, handelt es sich um eine Punktanomalie. In der zweiten Kategorie **kollektive Anomalien** verhält sich eine Reihe von Datenpunkten abweichend vom Rest der Daten. Dabei sind die individuellen Datenpunkte aus der kollektiven Anomalie nicht unbedingt anormal, sondern nur deren gemeinsames Auftreten. Ein Beispiel für eine kollektive Anomalie ist das Signal eines Elektrokardiogramms [vgl. CBK09, Seite 9] . Wenn das Signal über eine längeren Zeitraum stabil bleibt und keine steilen Anstiege enthält wie in den anderen Intervallen, handelt es sich um eine kollektive Anomalie. Die dritte Kategorie von Anomalien sind die **kontextuellen** Anomalien. Bei einer kontextuellen Anomalie wird ein Wert nur in einem spezifischen Kontext als Anomalie betrachtet. Zum Beispiel könnte eine kontextuelle Anomalien in Temperatur-Zeitreihen auftreten. Ein sehr niedriger Temperaturwert im Winter wäre nicht ungewöhnlich, aber im Sommer wäre es eine kontextuelle Anomalie.

Bei der Implementierung von Anomalieerkennungen wird zwischen den drei Methoden **überwachtes Lernen**, **semiüberwachtes Lernen** und **unüberwachtes Lernen** unterschieden [vgl. BBK14, Seite 7]. Welche Methode angewendet werden kann, hängt hautsächlich davon ab, ob gelabelte Trainingsdaten oder nur ungelabelte Trainingsdaten zur Verfügung stehen. Bei gelabelten Trainingsdaten sind die Datenpunkte entweder als normal oder anormal markiert. Häufig ist es aber sehr schwierig und aufwendig, genug korrekt gelabelte Trainingsdaten zu erhalten, weil diese oft manuell von menschlichen Experten erarbeitet werden müssen. Beim überwachten Lernen wird vorausgesetzt das Datenpunkte sowohl für die normale Klasse als auch für die anormale Klasse verfügbar sind. Auf Basis dieser Eingabedaten soll das System dann ein Model erstellen, dass es ermöglicht möglichst zielsicher vorauszusagen, ob ein neuer Datenpunkt eine Anomalie darstellt oder nicht. Dagegen wird beim semiüberwachten Lernen nur auf Grundlage von Daten aus der normalen Klasse bestimmt, ob eine neue Dateninstanz anormal ist oder nicht. Ein Vorteil des semi-

überwachten Lernens ist die bessere Anwendbarkeit, weil keine Daten für die Klasse der Anomalien benötigt werden. Für das unüberwachte Lernen werden überhaupt keine gelabelten Trainingsdaten benötigt. Es wird davon ausgegangen das normale Dateninstanzen in den Testdaten deutlich häufiger auftreten als Anomalien. Anomalien können dann dadurch identifiziert, indem Datenpunkte gefunden werden, die von den am häufigsten auftretenden Mustern abweichen. Ein großer Nachteil dieser Technik ist, dass wenn die Annahme nicht stimmt, das System sehr viele falsch positive Rückmeldungen bekommt.

3 Methoden der Anomalieerkennung

Das Kapitel Methoden der Anomalieerkennung blickt auf Methoden, die zur Anomalieerkennung verwendet werden und ordnet diese in Kategorien ein. Dazu betrachtet es verschiedene Ansätze zur Kategorisierung aus der Literatur. Es erläutert für die Kategorien Classification, Statistisch, Clustering und Nearest Neighbor die grundlegende Funktionsweise und nennt Beispiele für Algorithmen aus dieser Kategorie.

3.1 Klassen von Algorithmen

Bhuyan et al.: Network Anomaly Detection: Methods, Systems and Tools	Chandola et al.: Anomaly Detection : A Survey	P. García-Teodoro et al.: Anomaly-based network intrusion detection: Techniques, systems and challenges
1. Statistical	1. Statistical	1. Statistical
2. Classification Based	2. Classification Based	2. Machine Learning Based
3. Clustering and Outlier Based	3. Clustering Based	3. Knowledge Based
4. Soft Computing	4. Nearest Neighbor Based	-
5. Knowledge Based	5. Information Theoretic	-
6. Combination Learners	6. Spectral	-

Tabelle 1 zeigt beispielhaft Kategorien in die Algorithmen/Methoden zur Anomalieerkennung eingeordnet werden können.

Verfahren zur Anomalieerkennung klar voneinander abzugrenzen ist ziemlich schwierig. Aus diesem Grund findet man auch verschiedene Ansätze von Kategorisierungen in der Literatur wieder. Drei Beispiele für verschiedenen Kategorien in die die Algorithmen eingeordnet werden können, sind in der Tabelle 1 dargestellt. In der Tabelle werden die Kategorien aus den Arbeiten von [BBK14], [CBK09] und [GTDVMFV09] einander gegenübergestellt. Gemeinsam ist bei allen drei Ausarbeitungen, dass jeweils eine Kategorie für die statistischen Verfahren existiert. Unterschiede finden sich vor allem in der Kategorie Clustering. Während bei [GTDVMFV09] Clustering-Verfahren zu der Kategorie Machine Learning zählen, haben [BBK14] und [CBK09] diese Art von Algorithmen in eine separate Kategorie eingeordnet.

Zusätzlich werden bei [CBK09] aus den Clustering-Verfahren nochmal die Nearest Neighbor Based Verfahren in eine extra Kategorie abgetrennt. Die Trennung basiert auf der Annahme, dass Clustering-Verfahren den Fokus auf die Bildung des Clusters und die anschließende Einordnung legen. Im Gegensatz dazu fokussieren sich die Nearest Neighbor Based Verfahren auf die lokale Nachbarschaft der Dateninstanzen. Eine weitere Kategorie die in allen drei Arbeiten zu finden ist, sind die Klassifikationsalgorithmen (Machine Learning bei [GTDVMFV09]).

3.2 Classification Based Verfahren

Die erste Kategorie von Verfahren stellen die Classification Based Verfahren dar. Classification Based Verfahren versuchen neu auftretende Dateninstanzen anhand eines zuvor gelernten Models in verschiedene Klassen einzuordnen [vgl. BBK14, Seite 12]. Dies geschieht in 2 Phasen. Zuerst erfolgt eine Trainingsphase in der gelabelte Trainingsdaten benutzt werden, um das Model zu lernen. Anschließend wird in einer Testphase durch einen Klassifikator mit Hilfe des Models entschieden, ob es sich um eine normale oder anormale Instanz handelt. Dabei wird zwischen Verfahren mit nur einer normalen Klasse (one-class) und Multi-Klassen (multi-class) unterschieden. Bei Multi-Klassen Verfahren wird angenommen, dass er mehrere normale Klassen gibt, in welche die Dateninstanzen eingeordnet werden können. Wenn der Klassifikator der Dateninstanz keine der normalen Klassen sicher zuordnen kann, wird diese Dateninstanz als anormal eingestuft. Beispiele für Klassifikationsalgorithmen sind verschiedene Varianten von neuronalen Netzen, bayessche Netze, Support Vector Machines oder regelbasierte Systeme.

3.3 Nearest Neighbor Based Verfahren

Nearest Neighbor Based Verfahren [vgl. CBK09, Seite 28] benutzen die Distanz bzw. die Ähnlichkeit (bei kategorischen Attributen) zwischen zwei Datenpunkten als Grundlage zur Anomalieerkennung. Abhängig von den Distanzen/Ähnlichkeiten ihrer Attribute liegen Datenpunkte verschieden weit auseinander. Mit der Annahme, dass normale Daten gesammelt in kurzer Distanz zueinander liegen und anormale Daten weit von ihren Nachbarn entfernt liegen, sollen Anomalien identifiziert werden. Welche Verfahren dabei zur Distanzbestimmung verwendet werden können, hängt von der Art der Attribute der Daten ab. Zum Beispiel ist ein einfaches Verfahren zur Distanzbestimmung der Euklidische Abstand. Der Euklidische Abstand kann aber nur auf kontinuierliche Daten angewendet werden. Die Funktionsweise von Nearest Neighbor Based Verfahren kann in die zwei Kategorien **k^{th} Nearest Neighbor** und **Relative Density** unterteilt werden. Bei k^{th} Nearest Neighbor Verfahren wird für eine Dateninstanz deren Distanz zu den nächsten k-Nachbarn als Anomalie-Score betrachtet. Anhand eines Schwellenwertes (threshold) wird die Dateninstanz dann als normal oder anormal eingestuft. Bei Relative Density Verfahren wird eine Anomalie an der Dichte der Nachbarschaft erkannt. Es wird davon ausgegangen, dass Anomalien

in Bereichen mit geringer Dichte an Nachbarn liegen, während normale Daten eine hohe Nachbarschaftsdichte besitzen.

3.4 Statistische Verfahren

Auch statistische Methoden können zur Erkennung von Anomalien eingesetzt werden. Als Grundlage dazu wird ein statistisches Model benutzt [vgl. BBK14, Seite 9-11]. Im ersten Schritt muss das Model dann mit normalen Daten befüllt werden. Mit Hilfe dieses Models können dann Anomalien erkannt werden, indem geschaut wird, wie wahrscheinlich es ist, dass eine neue Dateninstanz von diesem Model generiert wurde. Bei statistischen Anomalieerkennungsverfahren wird zwischen **parametrisierten** und **nicht-parametrisierten** Varianten unterschieden. Bei der parametrisierten Variante ist die Verteilungsfunktion der Daten schon bekannt und die Parameter werden durch die gegebenen normalen Daten eingestellt. Der Anomalie-Score wird durch das Inverse einer Wahrscheinlichkeitsdichtefunktion $f(x, \Theta)$ mit Parameter Θ und neuer Dateninstanz x bestimmt. Weiterhin unterscheiden sich die parametrisierten Verfahren, darin welche Verteilungsfunktion benutzt wird (z.B. Gaußsche Verteilung oder Regressionsmodell) und der Art und Weise wie Anomalien erkannt werden. Ein Beispiel für ein parametrisiertes Anomalieerkennungsverfahren ist die Box Plot Rule. Bei den nicht-parametrisierten Methoden werden statistische Modelle ohne Parameter benutzt, wie zum Beispiele Methoden, die auf Histogrammen basieren. Dabei wird zuerst durch die Daten ein Histogramm aufgebaut, das abhängig von den Werten eines Features verschiedene Bereiche mit unterschiedlichen Höhen (stellen Häufigkeiten dar) enthält. Für neue Testinstanzen wird dann überprüft in welchen Bereich des Histogramms der Datensatz fällt. Die Höhe des Bereiches bestimmt dann den Anomalie-Score der Dateninstanz.

3.5 Clustering-Verfahren

Ein weitere Möglichkeit zur Anomalieerkennung bieten Clustering-Verfahren. Das Ziel von Clustering ist es eine Menge von ähnlichen Objekten in gemeinsame Cluster einzuordnen [vgl. CBK09, Seite 30]. Die meisten Clustering-Verfahren operieren im unsupervised Modus. Zur Erkennung von Anomalien beim Clustering können unterschiedliche Annahmen genutzt werden. Eine einfache Annahme ist zum Beispiel, dass nur normale Daten zu den Clustern gehören und Anomalien nicht zu einem Cluster gehören. Algorithmen dieser Kategorie ordnen nicht zwangsläufig jeder Dateninstanz ein Cluster zu und die übriggebliebenen Daten werden als Anomalien eingestuft. Ein weitere Annahme besagt, dass normale Daten dichter am Mittelpunkt des Clusters (cluster centroid) liegen, während anormale Daten weit vom Mittelpunkt entfernt liegen. Ein Algorithmus aus dieser Kategorie wäre zum Beispiel K-means Clustering. Die letzte Annahme geht davon aus, dass normale Dateninstanzen sehr große und dichte Cluster bilden und daher nicht mit kleinen Clustern aus Anomalien zu verwechseln sind.

4 Aspekte der Anomalieerkennung im automotiven Kontext

Das folgende Kapitel erläutert Anforderungen und Aspekte, die spezifisch für die automotive Domäne sind und Auswirkungen auf das Design eines Anomalieerkennungssystem im Fahrzeug haben.

Grundsätzlich sind viele verschiedene Verfahren von Anomalieerkennungen bereits seit Jahren in anderen Bereichen erfolgreich im Einsatz, wie zum Beispiel im Bereich der Desktop Computer. Offensichtlich unterscheidet sich das Fahrzeug von vielen dieser anderen Anwendungsbereiche in der Hinsicht, dass es sich bei einem Fahrzeug um ein sicherheitskritisches System handelt und fehlerhaftes Verhalten zu Personen- und Umweltschäden führen kann. Weiterhin werden klassische Gegenmaßnahmen wie Firewall oder Virenscanner alleine nicht als ausreichender Schutz des Fahrzeugnetzwerkes angesehen, da sie einen Fokus auf die präventive Abwehr von Angriffen setzen [vgl. MGF10, Seite 1]. Denn das Fahrzeug besitzt in der Regel eine lange Lebenszeit und wird in vielen verschiedenen örtlichen und zeitlichen Bedingungen eingesetzt. Auch ein System zur Anomalieerkennung alleine stellt nur eine Ebene eines gesamten Sicherheitssystems dar.

Ein wichtiger Aspekt bei der Anomalieerkennung im Fahrzeug ist die **Auswahl der Daten**, die zum Finden der Anomalien benutzt werden sollen. In einem Fahrzeug gibt es eine Vielzahl von möglichen Datenquellen wie z.B. Daten aus der ECU, Daten aus dem Netzwerk oder Daten von der Sensorik, die verwendet werden könnten. Dabei ist die entscheidende Frage, welche dieser Daten oder Kombinationen von verschiedenen Datenquellen wirklich relevant sind um Angriffe effektiv erkennen zu können. In diesem Kontext spielt auch die Relation zwischen **Performance und Kosten** eine große Rolle. Denn in einem Fahrzeug wird oft sehr spezielle und kostenoptimierte Hardware, die zum Beispiel auf bestimmte physische Bedingungen ausgelegt ist, verwendet und weiterhin muss sichergestellt werden das alle Echtzeitanforderungen im Fahrzeug auch erfüllt werden können. So steht möglicherweise nicht genug Leistung zur Verfügung um alle Daten im gesamten Netzwerk des Fahrzeuges zu beobachten und auszuwerten. Ein weiterer wichtiger Aspekt ist die **eingesetzte Methodik bei der Erkennung von Anomalien**. Angriffe können einerseits mit signaturbasierten Verfahren identifiziert werden, die sicher alle bekannten Angriffe entdecken können und daher nur wenige falsch positive Alarme verursachen. Andererseits bieten sie im Gegensatz zur Anomalieerkennung nicht die Möglichkeit auch unbekannte Angriffsarten entdecken zu können und die regelmäßigen Updates, die für signaturbasierte Verfahren erforderlich sind, stellen auch einen technischen Aufwand dar. Hybride Ansätze könnten auch eine Lösung für diese Problemstellung sein. Ein weiterer wichtiger Aspekt ist auch **Intelligenz der Sensoren**. Denn entweder ist es möglich sehr kostengünstige Sensorik zu verwenden, die nur die Daten aus der entsprechenden Quelle einsammelt und anschließend an eine zentrale Verarbeitungseinheit weiterleitet oder Intelligenz könnte auch bereits in die Sensorik implementiert werden. Der große Vorteil des zweiten Ansatzes wäre es, dass der Traffic im Netzwerk geringer wäre und schon Teile der Anomalieerkennung übernommen werden könnten. In diesem Fall wären aber die Sensoren deutlich kostenintensiver. Eine weitere Pro-

blemstellung von der Anomalieerkennung im Fahrzeug stellen noch die **Gegenmaßnahmen auf die Entdeckung eines Angriffes** dar. Die klassische Möglichkeit des Benachrichtigen des Benutzers wäre im Auto relativ schwierig umzusetzen und nur im absoluten Notfall eine gute Option, da Benachrichtigungen oder Aufforderungen zur Reaktion auf dem Display den Fahrer ablenken würden. Deswegen müssten Systeme mit automatischen Gegenmaßnahmen auch im Netz vorhanden sein [vgl. MGF10, Seite 7].

5 Anomalieerkennung in der Literatur im automotiven Kontext

Der Einsatz von Anomalieerkennung im Fahrzeug zur Erhöhung der Security und Safety ist im Allgemeinen schon in einigen Arbeiten betrachtet worden. Das Survey Paper **A Survey of Anomaly Detection for Connected Vehicle Cybersecurity and Safety** [vgl. RMWH18] von Rajbahadur et al. gibt einen guten einen guten Überblick über alle Arbeiten bis zum Juni 2018 auf dem Themengebiet, indem durch eine Schneeballmethode alle relevanten Paper aus dem Gebiet der Anomalieerkennung im Fahrzeug gesammelt wurden. Die Wissenschaftler kommen am Ende ihrer Arbeit zu dem Schluss, dass die Arbeiten auf diesem Gebiet in großen Teilen nur Datensätze aus Simulationen und keine Datensätze aus der realen Welt benutzen. Weiterhin wurde festgestellt, dass Safety weniger Beachtung erhält als die Security im Fahrzeug und neue entwickelte Methoden zur Erkennung von Anomalien nur sehr selten gegen ein Basisset von Algorithmen zur Anomalieerkennung verglichen werden. Die Arbeit **A structured approach to anomaly detection for in-vehicle networks** [vgl. MGF10] von Müter et al. diskutiert den Einsatz von Anomalieerkennung im Fahrzeug unter Betrachtung einiger spezifischer Aspekte des automotiven Anwendungsbereiches und gibt Vorschläge für das zukünftige Design von Systemen zur Angriffsentdeckung im Fahrzeugen ab. In dem drauf folgenden Paper **Entropy-based anomaly detection for in-vehicle networks** [vgl. MA11] auch von Müter et al. wurde ein entropiebasiertes System zur Anomalieerkennung entwickelt. Zur Evaluation wurden verschiedene Angriffsszenarien durchgeführt, die anschließend auf dem CAN-Bus beobachtet wurden. Dabei konnte festgestellt werden, dass eine Vielzahl von Angriffen erkannt werden konnte. Probleme traten nur bei der Erkennung von „small-scale“ Angriffen auf, da diese sich nur geringfügig von dem normalen Verhalten unterschieden haben. Mo et al. beschreiben in ihrer Arbeit **Anomaly Detection of Vehicle CAN Network Based on Message Content** [vgl. MCWW19] eine neue Methode zur Anomalieerkennung basierend auf der Analyse des Inhaltes von Nachrichten aus einem CAN-Bus und der anschließenden Vorhersage einen Kofidenzintervalls, das eine Wahrscheinlichkeit angibt ob es sich um eine Anomalie handelt. In der Arbeit **An intrusion detection method for securing in-vehicle CAN bus** [vgl. GGT16] wird ein Fokus auf den Schutz des CAN-Bus gelegt und zuerst erläutert wieso ein zusätzlicher Schutz in Form eines Systems zur Anomalieerkennung sinnvoll ist. Anschließend wird ein Vorschlag für ein Anomalieerkennungssystem basierend auf den Zeitintervallen der CAN-Nachrichten dargestellt, das auch keine Modifizierungen auf Hardware Ebene erfordert.

6 Ausblick und Zusammenfassung

Die beständige Weiterentwicklung von modernen Fahrzeugnetzwerken und die Vielzahl an Kommunikationen mit der Außenwelt erfordern neue Security Konzepte zum Schutz des Fahrzeuges. Ein vielversprechender Ansatz stellt die Anomalieerkennung dar. In dieser Ausarbeitung wurden die Grundlagen eines Fahrzeugnetzwerkes und der Anomalieerkennung erläutert. Darauf aufbauend wurden Methoden der Anomalieerkennung und Aspekte, die beim der Entwicklung eines System zur Entdeckung von Anomalien im Fahrzeug beachtet werden müssen, aus der Literatur herausgearbeitet. In der Betrachtung der Literatur auf diesem Gebiet wurde in einem Survey Paper angemerkt, dass nur wenige Arbeiten Daten aus der realen Welt benutzen und neu entwickelte Methoden selten gegen ein Basisset verglichen wurden. In meinen Grundprojekt vergleiche ich gerade verschiedene Basisalgorithmen in einer OMNeT++ (Objective Modular Network Testbed in C++) Simulation um zu schauen welche Methoden der Anomalieerkennung sich für weitere Analysen eignen. Im weiteren Verlauf meines Masters (Hauptprojekt und später Masterarbeit) möchte ich mit Anomalieerkennung aus Basis von Daten aus der realen Welt befassen. Das SecVI-Projekt bietet mir dafür die Möglichkeit an einem realen Testauto mitzuarbeiten.

Literatur

Aut19. Autosar. Specification of lin interface, 2019.

BBK14. M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita. Network anomaly detection: Methods, systems and tools. *IEEE Communications Surveys Tutorials*, 16(1):303–336, First 2014.

Bos19. Bosch. Can specification, 2019.

CBK09. Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3):15:1–15:58, July 2009.

CMK+11. Stephen Checkoway, Damon Mccoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno. Comprehensive Experimental Analyses of Automotive Attack Surfaces. *USENIX Security*, 2011.

Coo19. Most Cooperation. Most specification, 2019.

Fle19. Flexray. Flexray communications system protocol specification, 2019.

FYM05. Ryohei Fujimaki, Takehisa Yairi, and Kazuo Machida. An approach to spacecraft anomaly detection problem using kernel feature space. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, pages 401–410, New York, NY, USA, 2005. ACM.

GGT16. M. Gmiden, M. H. Gmiden, and H. Trabelsi. An intrusion detection method for securing in-vehicle can bus. In *2016 17th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pages 176–180, Dec 2016.

Gmb18. Wirecard Technologies GmbH. Data & analytics – anomalieerkennung in zeitreihen, 2018.

GTDVMFV09. P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez. Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers and Security*, 28(1):18 – 28, 2009.

HMVVDK13. Peter Hank, Steffen Müller, Ovidiu Vermesan, and Jeroen Van Den Keybus. Automotive ethernet: In-vehicle networking and smart mobility. In *Proceedings of the Conference on Design, Automation and Test in Europe*, DATE '13, pages 1735–1739, San Jose, CA, USA, 2013. EDA Consortium.

MA11. M. Müter and N. Asaj. Entropy-based anomaly detection for in-vehicle networks. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 1110–1115, June 2011.

MCWW19. Xiuliang Mo, Pengyuan Chen, Jianing Wang, and Chundong Wang. Anomaly detection of vehicle can network based on message content. 6 2019.

MGF10. M. Müter, A. Groll, and F. C. Freiling. A structured approach to anomaly detection for in-vehicle networks. In *2010 Sixth*

International Conference on Information Assurance and Security, pages 92–98, Aug 2010.

Pat. Andrew Patterson. Outlier detection methods for detecting cheaters in mobile gaming.

Pro19. Secvi Projekt. Security for vehicular information - research project for secure automotive network architectures, 2019.

RMWH18. G. K. Rajbahadur, A. J. Malton, A. Walenstein, and A. E. Hassan. A survey of anomaly detection for connected vehicle cybersecurity and safety. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 421–426, June 2018.

SNA+13. I. Studnia, V. Nicomette, E. Alata, Y. Deswarte, M. Kaâniche, and Y. Laarouchi. Survey on security threats and protection mechanisms in embedded automotive networks. In *2013 43rd Annual IEEE/IFIP Conference on Dependable Systems and Networks Workshop (DSN-W)*, pages 1–12, June 2013.