Anomalieerkennung mit neuronalen Netzen in Fahrzeugnetzwerken

Mohammad Fazel Soltani (2308059)

Forschungswerkstatt 1 (FW1)
Betreuender Prüfer: Prof. Dr. Franz Korf
Hochschule für Angewandte Wissenschaften (HAW) Hamburg, 20099 Hamburg, DE
mohammadfazel.soltani@haw-hamburg.de

Zusammenfassung. Fahrzeuge sind auf Grund von Internetkonnektivität zu einem attraktiven Angriffsziel von Hackern geworden. Sicherheitskritische Komponenten wie Bremsen und Motor werden per Fernzugriff gesteuert, manipuliert oder vollständig ausgeschaltet. Die Detektion solcher Angriffe kann mit der Anomalieerkennung umgesetzt werden. Dadurch können Gegenmaßnahme zur Aufrechterhaltung der Funktionalität sicherheitskritischer Komponenten eingeleitet werden. Neuronale Netze können trainiert werden, um das normale Verhalten des Fahrzeugnetzwerkes zu erlernen. Jedoch ist nicht klar, welche Daten aus dem Fahrzeugnetzwerk für das Training eines neuronalen Netzes verwendet werden sollen. Außerdem ist auch nicht klar, welche Topologien von neuronalen Netzen zur Anomalieerkennung in Fahrzeugnetzwerke verwendet werden sollen. Im Rahmen dieser Literaturarbeit wurden verwandte Arbeiten ausgesucht und recherchiert, in denen neuronale Netze zur Anomalieerkennung in Computernetzwerke oder Fahrzeugnetzwerke ihren Einsatz fanden. Die Ergebnisse aus den verwandten Arbeiten zeigen, dass der Einsatz von verschiedenen neuronalen Netzen zur Anomalieerkennung, je nachdem welche Daten ausgewählt und wie diese verarbeitet wurden, umgesetzt werden kann. Die Anomalieerkennung mit neuronalen Netzen in zukünftigen Fahrzeugnetzwerken ist gegenwärtig ein offenes Thema.

Schlüsselwörter: Anomalieerkennung · Neuronale Netze · Fahrzeugnetzwerke

1 Einleitung

Ethernetbasierte Fahrzeugnetzwerke sind über Schwachstellen an den elektronischen Steuergeräten (ECUs) kompromittierbar [7]. Angriffe von Hackern an diesen Schwachstellen ermöglichen die Manipulation oder das Abschalten sicherheitskritischer Funktionen im Fahrzeug wie Bremse, Antrieb und Motor [14]. Daher ist die Detektion von Angriffen von besonderer Bedeutung, um mögliche Gegenmaßnahmen einzuleiten. Die Anomalieerkennung mit neuronale Netzen ermöglicht bekannte und unbekannte Angriffsmuster zu detektieren [17]. Jedoch verbergen sich Fragen im Kontext von Fahrzeugnetzwerken. Einerseits ist nicht ersichtlich, welche Daten aus einem Fahrzeugnetzwerk für Anomalieerkennung

trainiert und wie die Verarbeitung dieser Daten vollzogen werden sollen. Andererseits ist es auch nicht ersichtlich, welche neuronalen Netze für die Anomalieerkennung in Fahrzeugnetzwerk eingesetzt werden kann.

Diese Literaturarbeit verschafft einen Überblick über den Einsatz neuronaler Netze zur Anomalieerkennung in Fahrzeugnetzwerke. Hierbei wird der Fokus auf die verwendeten Daten sowie dessen Verarbeitung gerichtet. Außerdem welche neuronalen Netze für diese verarbeiteten Daten eingesetzt und welche Ergebnisse erzielt wurden.

Die Arbeit ist folgendermaßen gegliedert. In Abschnitt 2 wird auf Fahrzeugnetzwerke eingegangen sowie die Angriffsoberflächen erläutert. In Abschnitt 3 wird die Anomalieerkennung behandelt. Hier werden auf Eingabe- und Ausgabedaten, Arten und Lernverfahren von Anomalien eingegangen. In Abschnitt 4 werden neuronale Netze vorgestellt, in der auf Besonderheiten dieser Netze und zur Anomalieerkennung eingegangen wird. In Abschnitt 5 werden verwandte Arbeiten zur Anomalieerkennung mit neuronalen Netzen in Fahrzeugnetzwerke vorgestellt. Die Arbeit wird in Abschnitt 6 mit Fazit und Ausblick abgeschlossen.

2 Fahrzeugnetzwerk

Gegenwärtige Fahrzeuge sind mit Sensoren, Aktoren, elektronischen Steuergeräten (ECUs), Bussystemen - beispielsweise Controller Area Network (CAN) - und einem Ethernet Backbone ausgestattet [18]. Durch die Vernetzung dieser Komponenten kann die interne Kommunikation im Fahrzeug statisch (z.B. Bremssteuerung) oder dynamisch (z.B. Rückfahrkamera) sein. Außerdem haben diese verschiedenen Steuerungs- und Fahrassistenzfunktionen unterschiedliche Quality-of-Service (QoS)-Anforderungen und müssen feste Latenzzeiten einhalten [9].

Wie in Abbildung 1 zu sehen ist, hat sich das Bordnetzwerk des Fahrzeuges im Laufe der Zeit verändert. In Abbildung 1 (a) ist die zentrale Gateway Topologie zu sehen. Die in funktionale Domänen gruppierten ECUs kommunizieren mit Hilfe von Bussystemen über ein zentrales Gateway miteinander. Dieses zentrale Gateway bildet einen Flaschenhals, welches durch die steigende Anzahl an vernetzten ECUs über das Gateway die domänenübergreifende Kommunikation erhöht [18].

In Abbildung 1 (b) ist die Domänen Controller Topologie zu sehen. Immer mehr Fahrassistenzfunktionen (z.B. Rückfahrkamera) verwenden Sensoren wie hochauflösende Kameras mit hohen Bandbreiten. Diese Topologie verwendet Automotive Ethernet [12]. Mit zentralen Rechencluster (HPC) lassen sich die in Software implementierten ECU-spezifischen Funktionen steuern. Außerdem bildet diese Topologie einen geswitchten Backbone, um eine schnelle domänenübergreifende Kommunikation zu ermöglichen [9].

In zukünftigen Fahrzeugenetzwerke wird das in Abbildung 1 (c) dargestellte Zonen-Modell [5] vorgeschlagen. Künftig sollen Fahrzeuge mit anderen Fahrzeugen oder mit dem Internet in Verbindung gebracht werden (V2X) [9]. Mit dem Zonen-Modell werden ECUs, je nach räumlichen Entfernung, zu einer Zonen zugewiesen. Diese sind mit einem Zonen-Gateway verbunden. Ein Zonen-Gateway ist für die Netzwerkverbindung und die Energieversorgung der ECUs zuständig.

Die Kommunikation von ECU zum Zonen-Gateway erfolgt mit Bussystemem, jedoch folgt die domänenübergreifende Kommunikation der Zonen-Gateways mit Ethernet. Die Einführung von Ethernet als Kommunikationstechnologie öffnet aber auch die Tür für Angriffe, die im folgenden Abschnitt erläutert werden.

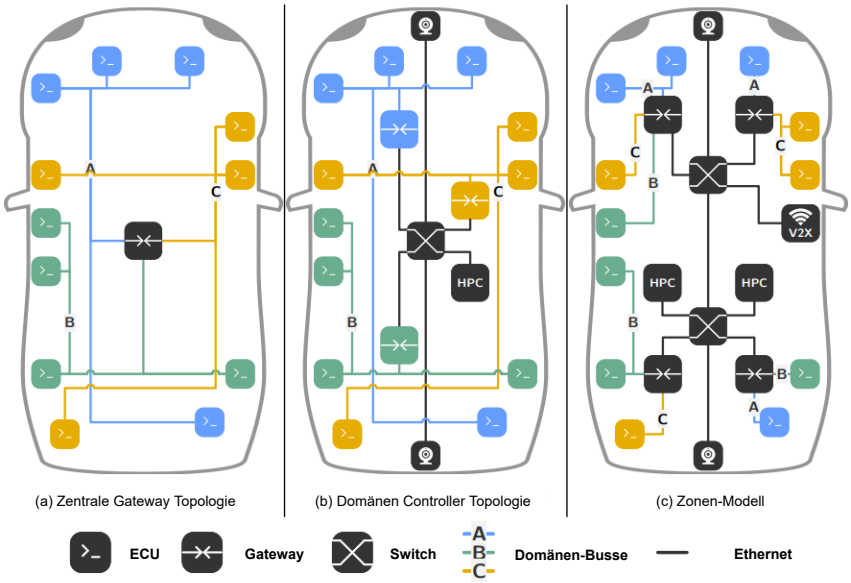

Abb. 1. Entwicklung der Fahrzeugnetzwerk-Topologien (Quelle: [9])

Checkoway et al. [7] sehen Schnittstellen am Fahrzeug als Angriffsoberflächen an. Wie in Abbildung 2 dargestellt, werden die Schnittstellen anhand ihrer Reichweite in Indirect Physical Access (IPA), Short-range Wireless Access (SrWA) und Long-range Wireless Access (LrWA) kategorisiert. Schnittstellen wie USB, SD-Anschlüsse sowie der On-Board Diagnostics (OBD)-II-Port sind dem IPA zugeordnet, welche durch einen direkten Zugriff auf das Fahrzeug angegriffen werden können. Jedoch ist es auch möglich Fernangriffe über diese Schnittstellen zu tätigen, wenn beispielsweise der OBD-II-Port bereits vom Angreifer kontrolliert wird. Im Gegensatz dazu sind Schnittstellen wie Bluetooth, Tire Pressure Monitoring System (TPMS), Remote Keyless Entry und Internet Hotspots dem SrWA zugeordnet, welche aus einer Reichweite von fünf bis 300 Meter angegriffen werden können. LrWA Schnittstellen erlauben die Kommunikation über noch größere Reichweiten zu denen das Radio oder mobile Daten, die eine individuelle Addressierung ermöglichen, gehören. Angriffe über diese Schnittstellen können durch die Anomalieerkennung detektiert werden.

3 Anomalieerkennung

Die Anomalieerkennung handelt vom Feststellen unregelmäßiger Muster (Anomalien) innerhalb von normalen Daten. Diese Anomalien oder Outlier [6] haben Eigenschaften wie seltenes Auftreten, Unregelmäßigkeit und Unbekanntkeit, bis

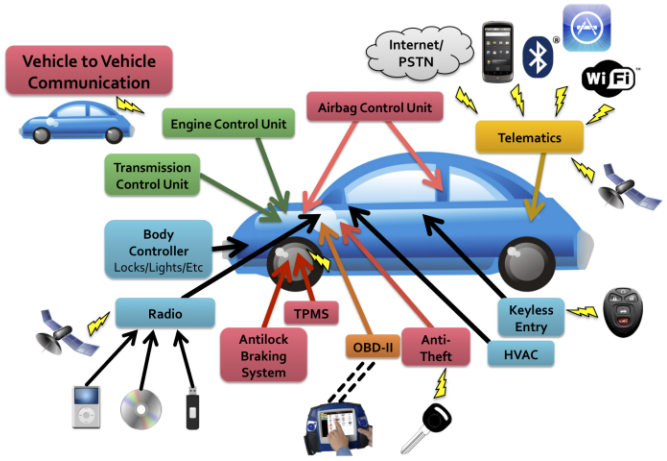

Abb. 2. Schnittstellen als Angriffsoberflächen an Fahrzeugen (Quelle: [7])

sie tatsächlich auftreten [16] - beispielsweise bei Netzwerkeinbrüchen in einem Fahrzeugnetzwerk. Anomlien können nicht nur aufgrund von Angriffen auftreten und spiegeln nicht das normale Verhalten des System wieder.

Je nach Anwendungsbereich unterscheiden sich die Eingabedaten bei der Anomalieerkennung. Eingabedaten sind eine Sammlung von Daten, die als Objekte, Vektoren, Beobachtungen oder Punkte bezeichnet werden kann. Diese wiederum bestehen aus einer Menge von Attributen - beispielsweise Variablen, Felder oder Features. Ein Attribut wird in binär, kategorisch oder kontinuierlich kategorisiert. Bestehen die Daten aus einem Attribut, werden diese als Univariate bezeichnet. Bestehen die Daten aus mehreren Attributen, werden diese als Mulitvariate bezeichnet. Die Kombination oder die Arten von Attributen innerhalb der Daten entscheiden, welche Algorithmen ein Anomalieerkennungssystem einsetzt oder wie diese Daten anzupassen sind [6]. Die Ausgabe von Anomalien kann neben der Klassifikation von anomalen und normalen Daten (labeling), durch das Zuweisen von Wertungen, die das Maß der Abnormalität darstellt (scoring). Je nach Anwendungsfall kann die Art der Ausgabe gewählt werden [6].

Es gibt verschiedene Arten von Anomalien. Punktanomalien sind einzelne Instanzen, die im Vergleich zu der Mehrheit der anderen einzelnen Instanzen anomal sind, beispielsweise die anormalen Gesundheitsindikatoren eines Patienten. Bedingte Anomalien, auch kontextuelle Anomalien genannt, beziehen sich ebenfalls auf einzelne anomale Instanzen, aber in einem bestimmten Kontext (z.B. plötzlicher Temperaturenabfall/-anstieg). Gruppenanomalien, auch bekannt als kollektive Anomalien, sind eine Teilmenge von Dateninstanzen, die in ihrer Gesamtheit im Vergleich zu den anderen Dateninstanzen anomal sind [6,16].

Bhuyan et al. [4] unterschieden drei Lernverfahren, die bei einer Anomalieerkennung eingesetzt werden können. Das überwachte Lernen (supervised Learning) erstellt anhand gelabelter Daten einen Klassifikator, welcher anomales und normales Verhalten unterscheidet. Beim halbüberwachten Lernen (semi-

supervised Learning) gibt es keine gelabelten Daten für das anomale Verhalten. Daher kann das resultierende Modell nur mit normalen Verhalten trainiert werden. Im unüberwachten Lernen (unsupervised Learning) erfolgt das Training mit ungelabelten Daten. Hierbei wird angenommen, dass Anomalien in den Daten kaum oder überhaupt nicht vorhanden sind.

4 Neuronale Netze

Neuronale Netze sind eine Technik zur Entwicklung von Deep Learning Modellen. Deep Learning ist eine Technik des maschinellen Lernen, welches wiederum eine Untermenge der künstlichen Intelligenz ist [15]. Die Idee neuronaler Netze stammt aus der Informationsverarbeitung des menschlichen Gehirns [8]. Es kann als ein Computersystem angesehen werden, welches aus einer Anzahl miteinander verbundener Neuronen besteht, die Informationen durch ihren dynamischen Zustand als Reaktion auf externe Eingaben verarbeiten [2].

Neuronale Netze nutzen komplexe Zusammensetzungen linearer/nicht linearer Funktionen, die durch einen Computergraphen dargestellt werden können, um aussagekräftige Darstellungen zu erlernen [8]. Aktivierungsfunktionen und Schichten in neuronale Netze bilden hierbei die grundlegenden Bestandteile. Aktivierungsfunktionen, wie sigmoid, tanh, ReLU (Rectified Linear Unit) und ihre Varianten, können lineare und nichtlineare Funktionen sein und bestimmen die Ausgabe von Neuronen in neuronalen Netzen bei bestimmten Eingaben, die einen Schwellenwert erreichen. Schichten, wie vollständig verbundene Schichten (siehe Abbildung 3), Faltungs- und Pooling-Schichten sowie rekurrente Schichten, können zum Aufbau verschiedener gängiger neuronaler Netze genutzt werden. Eine

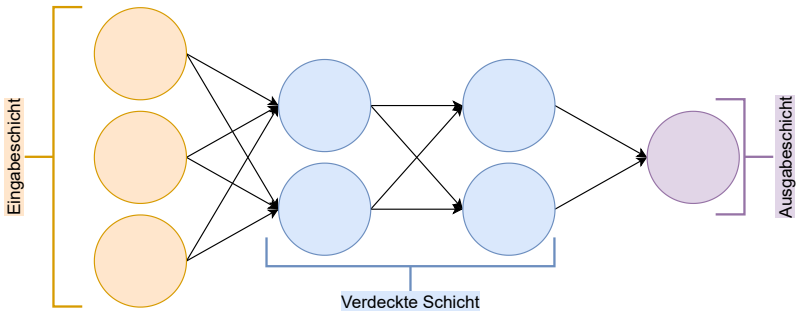

Abb. 3. Beispiel eines neuronalen Netzes (Vgl. Quelle: [8])

Schicht in neuronalen Netzen bezieht sich auf eine Gruppe von Neuronen, die in irgendeiner Form gestapelt sind [16].

Ein Neuron, wie in Abbildung 4 zu sehen ist, ist folgendermaßen aufgebaut. Es werden eine Menge von Eingaben x_1 bis x_m (hellblaue Kreise) mit den jeweiligen Gewichten w_1 bis w_m multipliziert und zueinander addiert. Die daraus berechnete Zahl wird durch an einer Aktivierungsfunktion übergeben. Mit Hilfe einer Aktivierungsfunktion wird die berechnete Zahl auf nichtlineare Weise transformiert [15], die aus $\hat{y}$ resultiert und wieder als Eingabe dienen kann.

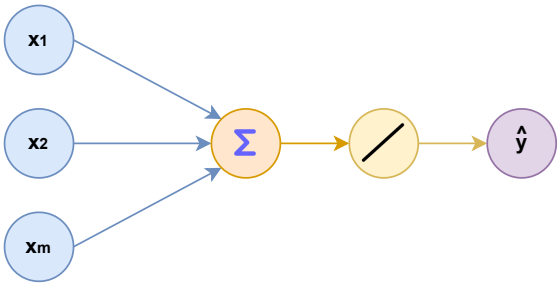

Abb. 4. Der Aufbau eines Perzeptrons (Vgl. Quelle: [2], The perceptron)

4.1 Neuronale Netze zur Anomalieerkennung

Pang et al. [16] schlagen in ihrer Arbeit drei Paradigmen der Anomalieerkennung mit neuronalen Netzen vor:

- Merkmalsextraktion (Feature Extraction)
- Ende-zu-Ende Lernen von Anomaliewerten (End-to-End Anomaly Score Learning)
- Lernen von Merkmalsrepräsentation der Normalität (Learning Feature Representation of Normality)

Neuronale Netze zur Merkmalsextraktion werden angewendet, um niedrigdimensionale Merkmalsrepräsentationen aus hochdimensionalen und/oder nicht linear trennbaren Daten für die Anomalieerkennung zu extrahieren. Die Merkmalsextraktion und die Erkennung von Anomalien sind vollständig voneinander getrennt und unabhängig. Der niedrigere dimensionale Raum hilft, versteckte Anomalien aufzudecken und die Zahl der falsch-positiven Ergebnisse zu reduzieren [16]. Das Ende-zu-Ende Lernen von Anomaliewerten (End-to-End Anomaly Score Learning) zielt darauf ab, skalare Anomaliewerte in einer Ende-zu-Ende-Methode zu lernen [16]. Die Anomaliewerte sind nicht vom Maß der Anomalien abhängig, da ein neuronales Netz die Anomaliewerte direkt lernt. Dieses Paradigma bietet eine einfache Erklärung von Anomalien, indem die Aktivierungsgewichte von Anomalie-Scores zurückverfolgt werden, um die Merkmale zu finden, die für große Anomalie-Scores verantwortlich sind [16].

Das Lernen von Merkmalsrepräsentation der Normalität. Es wird das Lernen von Merkmalen mit der Anomalieerkennung teilweise gekoppelt. Dies kann mit den Ansätzen Generisches Lernen von Normalitätsmerkmalen (Generic Normality Feature Learnings) und Anomaly Measure-dependent Feature Learning vollzogen werden. Beim generischen Lernen von Normalitätsmerkmalen werden Repräsentationen von Dateninstanzen durch Optimierung einer generischen Merkmalslernzielfunktion erlernt, die nicht in erster Linie für die Erkennung von Anomalien konzipiert ist [16]. Die gelernten Repräsentationen können die Erkennung von Anomalien dennoch unterstützen, da sie gezwungen sind, einige wichtige zugrundeliegende Datenregularitäten zu erfassen. Dieser Ansatz kann Methoden wie Datenrekonstruktion, generative Modellierung, Modellierung der Vorhersagbarkeit und selbstüberwachte Klassifizierung umfassen. Dadurch können Anomalien in hochdimensionalen Daten sowie in nicht-unabhängigen Daten

erkannt werden. Außerdem können die Zahl der falsch-positiven Ergebnisse verringert werden, wenn die gelernten Darstellungen aussagekräftiger sind [16]. Anomaly Measure-dependent Feature Learning zielt auf das Lernen von Merkmalsrepräsentationen ab, die speziell für ein bestimmtes bestehendes Anomaliemaß optimiert sind [16]. Das Lernen der Repräsentationen kann durch distanzbasierte, einklassige Klassifikationsverfahren und clusterbasierte Verfahren realsiert werden.

Liu et al. [11] haben in ihrer Arbeit ein Faltungsneuronales Netze (CNN) zur Anomalieerkennung angewendet. Ihr Ansatz entspricht dabei dem Lernen der Merkmalsextraktion. Aus dem Anomalierkennungsproblem haben sie ein Bildklassifizierungsproblem entwickelt, indem sie aus dem NSL-KDD-Datensatz [19] Daten des Netzwerkverkehrs mit der schnellen Fourier Transformation (FFT) zu Bildern umgeneriert haben. Daten, wie protocol type, flag und service wurden in einem 1*41 Vektor durch Zahlen von 0 bis 69 ersetzt. Beispielsweise hat protocol type drei Zeichenvariablen: tcp, udp und icmp, die durch 0, 1 und 2 ersetzt wurden. Diese Daten wurden mit FFT berechnet und in 4096 Abtastpunkten abgetastet. Da die 4096 Abtastpunkten komplexe Zahlen beinhalten, wurden diese in zweidimensionale 64*64 Kanäle aufgeteilt. In einem Kanal wurde der Realteil, in einem anderem Kanal der Imaginärteil und in einem weiteren Kanal die Summer aus Real- und Imaginärteil zusammengefasst. Die drei Kanäle wurden danach normalisiert, jeder normalisierte Wert zwischen 0 und 255 eingeordnet und aufgerundet. Diese Bilder spiegeln sowohl das normale, als auch anomale Verhalten des Netzwerkverkehrs aus den Daten des NSL-KDD Datensatzes wieder.

5 Anomalieerkennung mit neuronalen Netzen in Fahrzeugnetzwerken

In diesem Abschnitt werden vergangene Arbeiten über die Erkennung von Anomalien mit neuronalen Netzen in Fahrzeugnetzwerken behandelt. Die vergangenen Arbeiten wurden so ausgewählt, dass sie unterschiedliche neuronale Netze verwenden und sich auf unterschiedliche Netzwerke (also CAN oder Automotive Ethernet) fokussieren. Zuvor wird das normale Verhalten von Fahrzeugnetzwerke diskutiert.

Um die Anomalieerkennung mit neuronalen Netze in Fahrzeugnetzwerken vollziehen zu können, ist es zunächst wichtig das normale Verhalten in einem Fahrzeugnetzwerk zu bestimmen. Während des Designprozesses von Fahrzeugnetzwerken werden die fahrzeuginternen Verbindungen zwischen ECUs in einer Kommunikationsmatrix des Fahrzeuges vordefiniert und mit Verhaltenseigenschaften wie Bandbreite, Timing und Bursts versehen [13]. Außerdem müssen auch Daten aus Sensoren wie Kameras und LIDARs in Betracht gezogen werden [13]. Diese Daten können variable Größen haben und werden domänenübergreifend übermittelt (z.B. von Sensorfusion über Swichtes zum Infotainment [13]). Diese Verhaltenseigenschaften können wichtige Merkmale für das Trainieren des Normalverhalten eines Automotive Ethernet-basierten Netzwerkes sein.

Seo et al.[17] haben in ihrer Arbeit ein generatives adversarisches Netz IDS (GIDS) für ein Fahrzeugnetzwerk entwickelt, um eine hohe Genauigkeit von

falsch positiven Meldungen zu erzielen. GANs sind ein auf Deep Learning basierendes generatives Modell und erzeugen neue Dateninstanzen, die ihren Trainingsdaten ähneln [8]. Hierfür haben sie aus einem Hyundai YF Sonata die CAN-Daten aufgezeichnet. CAN-IDs in CAN-Daten zeigen sich wiederholende Muster und diese wurden für das Training extrahiert. Die extrahierten CAN-IDs wurden durch die One-Hot-Vektor Kodierung in CAN-Bilder generiert. Aus den hexadezimalen Zahlen einer CAN-ID wird jede hexadezimale Zahl in binärer Form mit 16 Ziffern ausgedrückt. Danach werden die binären Formen jeder hexadezimalen Zahl der CAN-IDs in einen One-Hot-Vektor kodiert. Bei der Kodierung mit One-Hot-Vektor wird eines der Bits zu einer 1 und die restlichen Bits sind alle 0, wodurch CAN-Bilder resultieren. In ihrem GAN haben Seo et al. zwei neuronale Netze (Generator und Diskriminator) mit den CAN-Bildern trainiert. Für das Training von bekannten Angriffen (DoS, FUZZY, RPM/GEAR) und deren Unterscheidung erhält der erste Diskriminator normale CAN-Bilder und abnormale CAN-Bilder. Hierbei erzielte der erste Diskriminator Erkennungsgenauigkeiten von 99,9% bei DoS, 98,7% bei FUZZY, 99,7% bei RPM und 99,8% bei GEAR. Für unbekannte Angriffe werden der zweite Diskriminator und der Generator gleichzeitig durch einen gegnerischen Prozess trainiert. Der Generator erzeugt gefälschte Bilder mit Hilfe von Zufallsrauschen. Der zweite Diskriminator empfängt normale CAN-Bilder, die vom Generator erzeugten Bilder und lernt diese voneinander zu unterscheiden. Hierbei erzielte der zweite Diskriminator die Erkennungsgenauigkeiten von 99,6% bei DoS, 99,5% bei FUZZY, 99,0% bei RPM und 96,5% bei GEAR durch das Training mit Rauschbildern und CAN-Bildern. In ihrer Arbeit fokussierten sich Seo et al. nur auf CAN, aber zukünftige Fahrzeugnetzwerke wie Automotive Ethernet wurden nicht behandelt.

Alkhatib et al.[1] untersuchen in ihrer Arbeit Abweichungen der Protokollspezifikation von Scalable service-Oriented Middle-warE over IP (SOME/IP)[3] mit einem rekurrenten neuronalen Netz (RNN). SOME/IP wird zunehmend in Fahrzeugnetzwerken eingesetzt, um den Austausch verschiedener Dienste zwischen Anwendungen auf unterschiedlichen ECUs zu koodinieren. Die Abweichungen werden durch einen sogenannten SOME/IP Paketgenerator generiert. Dieser ist sowohl in der Lage das Verhalten von den vier Angriffen (Request without Response, Response without Request, Error on Error sowie Error on Event), als auch das normale Verhalten zwischen Clients und Server zu modellieren. Das normale Verhalten ist nach AUTOSAR spezifiziert. Der SOME/IP Paketgenerator erzeugt pcap-Dateien, die aus ungelabelte Paketen bestehen und aus dem gesamten Netzwerk gesammelt wurden. Die Pakete wurden mit einem Label wie 0 (nach AUTOSAR-Spezifikation) oder 1, 2, 3 oder 4 für die benannten Angriffe markiert. Jedes Paket wird durch 16 Merkmale dargestellt. Dazu zählen Merkmale wie Service ID, Methoden ID, Client ID Nachrichten Typ, Session ID, Interface und Protokoll Version, Return Code, Ziel- und Quell-IP Adresse, Ziel- und Quell-Port, Ziel- und Quell-IP MAC-Adresse, Protokoll und Label. Die 16 Merkmale werden durch die One-Hot-Vektor Kodierung in binäre Vektoren umgewandelt. 15 Merkmale eines Paketes (Eingabedaten) wurden auf 195 Merkmale erweitert und das Label (Ausgabedaten) auf 5 Klassen erweitert.

Zur Erkennung von Angriffen wurden Pakete ihrer entsprechenden Reihenfolge in Sequenzen gruppiert und untersucht. Die Sequenzen wurden auf eine feste Länge von 60 Paketen transformiert und dem RNN als Eingabedaten übergeben. Das RNN führt Multi-Klassifikationen durch. In ihren Messungen habe sie drei trainierte RNN-Modelle verwendet und beim besten Modell folgende Ergebnisse erzielt. Pakete aus Request without Response werden mit einer Genauigkeit von 0.93 erkannt. Pakete aus Response without Request werden mit einer Genauigkeit von 0.96 erkannt. Pakete aus Error on Error werden mit einer Genauigkeit von 0.93 erkannt. Pakete aus Error on Event werden mit einer Genauigkeit von 0.95 erkannt. In ihrer Arbeit gehen Alkhatib et al. zwar auf Automotive Ethernet ein, jedoch richtet sich der Blick auf Abweichungen auf der Protokollspezifikation und nicht auf das Verhalten des automotiven Ethernet-Netzwerkes.

Jeong et al.[10] haben ein CNN entworfen, um Anomalien paketweise in einem Audio-Video-Transport-Protokoll (AVTP) Strom zu detektieren. Hierfür haben sie ein Testbed entworfen, bei der kontinuierliche Videodaten von einer Kamera von einem Audio-Video-Bridging (AVB)-Talker über ein BroadR-Reach-Switch zu einem AVB-Listener aufgezeichnet werden. In ihren Analysen haben sie zwischen der Kommunikation der AVB-Teilnehmer einen zusätzlichen AVB-Talker (für das Fluten von Videodaten-Pakete) hinzugefügt zum AVB-Listener, um Angriffe wie DoS damit zu simulieren. Auffälig waren die ersten 58 Bytes der aus 438 Bytes bestandene Nutzdaten aus den übertragenen Videodaten, in denen Anomalien erfasst wurden. Diese 58 Bytes aus den AVTP Data Units (AVTPDU) wurden als Eingabe für das CNN mit einem Feature Generator umgeformt. Außerdem verwendet der Feature-Generator eine Fenstergröße w. Dieser aggregiert den kürzlich eingetroffenen Verkehr mit w und erzeugt daraus einen zweidimensionalen Vektor. Der zweidimensionale Vektor wird dem CNN übergeben und damit auf Anomalien trainiert. In ihren Messungen habe sie fünf trainierte CNN-Modelle verwendet und beim besten Modell folgende Ergebnisse erzielt. Angriffe über das AVTP werden mit einer Genauigkeit von 0.9953 detektiert. Die Arbeit von Jeong et al. ähnelt zum Teil das Forschungsvorhaben dieser Arbeit. Jedoch richten sich Jeong et al. auf zwischen AVB-Talkern und -Listener übertragene anomale Videodaten und nicht anomale Ethernet-Paketen, die zwischen ECUs über Switches übertragen werden.

6 Fazit und Ausblick

Fahrzeuge sind auf Grund von Schwachstellen angreifbar. Die Anomalieerkennung kann Angriffe über diese Schwachstellen detektieren. Neuronale Netze können je nach Struktur und Paradigma für die Anomalieerkennung eingesetzt werden. Die Anomalieerkennung mit neuronalen Netzen in Fahrzeugnetzwerken kann je nach Datenlage, der Vorverarbeitung der Daten und die Auswahl eines geeigneten neuronalen Netzes angewendet werden. Daten wie CAN-IDs, Inhalte aus SOME/IP oder Nutzdatenbereiche von AVTP Datenpakete können eine Basis sein, aus denen Merkmale gewonnen werden. Die Vorverarbeitung der Daten kann durch Kodierungsmechanismen wie One-Hot Vektor oder als reine Daten als Eingabe für neuronale Netze wie GANs, RNNs oder CNNs übergeben werden. Diese Daten können trainiert werden, um bekannte oder unbekannte

Angriffe, Protokollabweichungen oder Videodateninjektionen zu erkennen. Das Verhalten des Netzwerkes aus Ethernet-Paketen wurde in keiner dieser Arbeiten behandelt und kann als Forschungsvorhaben angesehen werden.

Literatur

1. Alkhatib, N.*et al.*: SOME/IP Intrusion Detection using Deep Learning-based Sequential Models in Automotive Ethernet Networks. In: 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). pp. 0954–0962 (2021)
2. Amini, A.: Introduction to Deep Learning (Jan 2022), `http://introtodeeplearning.com/slides/6S191_MIT_DeepLearning_L1.pdf`
3. AUTOSAR: SOME/IP Protocol Specification. Tech. Rep. 696, AUTOSAR (Dec 2017)
4. Bhuyan, M.H.*et al.*: Network Anomaly Detection: Methods, Systems and Tools. IEEE Communications Surveys Tutorials **16**(1), 303–336 (2014)
5. Brunner, S.*et al.*: Automotive E/E-architecture enhancements by usage of ethernet TSN. In: 2017 13th Workshop on Intelligent Solutions in Embedded Systems (WISES). pp. 9–13 (2017)
6. Chandola, V.*et al.*: Anomaly Detection: A Survey. ACM Comput. Surv. **41**(3) (jul 2009)
7. Checkoway, S.*et al.*: Comprehensive Experimental Analyses of Automotive Attack Surfaces. In: 20th USENIX Security Symposium (USENIX Security 11). USENIX Association, San Francisco, CA (Aug 2011)
8. Goodfellow, I.*et al.*: Deep Learning. MIT Press (2016)
9. Häckel, T.*et al.*: Secure Time-Sensitive Software-Defined Networking in Vehicles (Jan 2022)
10. Jeong, S.*et al.*: Convolutional Neural Network-based Intrusion Detection System for AVTP Streams in Automotive Ethernet-based Networks. CoRR **abs/2102.03546** (2021)
11. Liu, W.*et al.*: A novel network intrusion detection algorithm based on Fast Fourier Transformation. In: 2019 1st International Conference on Industrial Artificial Intelligence (IAI). pp. 1–6 (2019)
12. Matheus, K.*et al.*: Automotive Ethernet. Cambridge University Press, Cambridge, United Kingdom (Jan 2015)
13. Meyer, P.*et al.*: Network Anomaly Detection in Cars: A Case for Time-Sensitive Stream Filtering and Policing (Dec 2021)
14. Miller, C.*et al.*: Remote Exploitation of an Unaltered Passenger Vehicle. Black Hat USA **2015**, 91 (2015)
15. Mueller, J.P.*et al.*: Deep Learning kompakt für dummies. Wiley (2020)
16. Pang, G.*et al.*: Deep Learning for Anomaly Detection: A Review. ACM Comput. Surv. **54**(2) (mar 2021)
17. Seo, E.*et al.*: GIDS: GAN based Intrusion Detection System for In-Vehicle Network. In: 2018 16th Annual Conference on Privacy, Security and Trust (PST). IEEE (aug 2018)
18. Steinbach, T.: Ethernet-basierte Fahrzeugnetzwerkarchitekturen für zukünftige Echtzeitsysteme im Automobil. Springer Vieweg, Wiesbaden (Oct 2018)
19. Tavallaee, M.*et al.*: A detailed analysis of the KDD Cup 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications. pp. 1–6 (2009)