Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.039583

ARTICLE

A Comprehensive Analysis of Datasets for Automotive Intrusion Detection Systems

Seyoung Lee[1], Wonsuk Choi[1], Insup Kim[2], Ganggyu Lee[2] and Dong Hoon Lee[1,*]

[1]School of Cybersecurity, Korea University, Seoul, 02841, Korea

[2]Memory Division, Samsung Electronics, Hwaseong, 18449, Korea

*Corresponding Author: Dong Hoon Lee. Email: donghlee@korea.ac.kr

Received: 06 February 2023 Accepted: 21 June 2023 Published: 08 October 2023

ABSTRACT

Recently, automotive intrusion detection systems (IDSs) have emerged as promising defense approaches to counter attacks on in-vehicle networks (IVNs). However, the effectiveness of IDSs relies heavily on the quality of the datasets used for training and evaluation. Despite the availability of several datasets for automotive IDSs, there has been a lack of comprehensive analysis focusing on assessing these datasets. This paper aims to address the need for dataset assessment in the context of automotive IDSs. It proposes qualitative and quantitative metrics that are independent of specific automotive IDSs, to evaluate the quality of datasets. These metrics take into consideration various aspects such as dataset description, collection environment, and attack complexity. This paper evaluates eight commonly used datasets for automotive IDSs using the proposed metrics. The evaluation reveals biases in the datasets, particularly in terms of limited contexts and lack of diversity. Additionally, it highlights that the attacks in the datasets were mostly injected without considering normal behaviors, which poses challenges for training and evaluating machine learning-based IDSs. This paper emphasizes the importance of addressing the identified limitations in existing datasets to improve the performance and adaptability of automotive IDSs. The proposed metrics can serve as valuable guidelines for researchers and practitioners in selecting and constructing high-quality datasets for automotive security applications. Finally, this paper presents the requirements for high-quality datasets, including the need for representativeness, diversity, and balance.

KEYWORDS

Controller area network (CAN); intrusion detection system (IDS); automotive security; machine learning (ML); dataset

1 Introduction

In recent years, integrating electronic control units (ECUs) in modern vehicles has greatly improved driver safety and convenience. An ECU is a specialized computer system regulating specific functions within the vehicle. A typical modern vehicle comprises around 70 to 100 ECUs, forming an in-vehicle network (IVN) [1]. The communication between these ECUs relies on the controller area network (CAN) protocol, widely used for IVN communication. Moreover, modern vehicles are now equipped with additional interfaces, such as Bluetooth, Wi-Fi, or cellular networks, to offer drivers

various services. Unfortunately, these interfaces also create opportunities for attackers to gain remote control of vehicles. If an attacker successfully compromises an external interface and injects malicious CAN messages into the IVN, they can remotely manipulate critical vehicle functions like steering and acceleration [2–4]. In academia, these attacks are commonly categorized as denial-of-service (DoS), fuzzy, replay, spoofing, and other similar attack types.

To mitigate the security risks posed by these attacks, numerous researchers have focused on developing security methods, including the utilization of cryptographic protocols [5,6] and automotive intrusion detection systems (IDSs) [7–9]. However, due to resource constraints, ECUs often struggle to perform heavy computations required by cryptographic protocols [10]. On the other hand, automotive IDSs have gained significant attention due to their self-adapting nature, which allows them to seamlessly extend their functionality to new vehicles and adapt to the automotive domain [11]. Automotive IDSs analyze CAN traffic to identify abnormal patterns that may indicate potential intrusions or attacks. These systems are generally classified into rule-based IDSs and machine learning (ML)-based IDSs.

Rule-based IDSs rely on predefined rules to detect discrepancies in CAN traffic. They are relatively simple to develop and widely adopted in the industry due to their straightforward nature [12–16]. These systems offer cost-effective and rapid detection of vehicular intrusions. However, a major limitation of rule-based IDSs is their reliance on comprehensive rulesets encompassing all possible CAN traffic characteristics. Consequently, rule-based IDSs excel in detecting known attacks but struggle to identify unknown or novel attack patterns [17]. In contrast, ML-based IDSs have the capability to effectively handle both known and unknown attacks by learning from normal CAN traffic [18]. ML-based IDSs leverage various machine learning techniques, such as convolutional neural network (CNN) [18], deep neural network (DNN) [19], artificial neural network (ANN) [20], and generative adversarial network (GAN) [21], to enhance their detection capabilities. These techniques enable ML-based IDSs to achieve high detection rates, often exceeding 98% accuracy with a false alarm rate of less than 1% [22]. Consequently, ML-based IDSs have shown promise in identifying both known and previously unseen vehicular intrusions, making them adaptable to emerging attack techniques [23].

However, the performance and effectiveness of ML-based IDSs heavily rely on the quality of the datasets used for training and evaluation. Ensuring dataset representativeness, diversity, and balance is crucial for accurate intrusion detection and adaptability to emerging threats in real-world automotive environments. In recent years, the vehicle security community has made significant advancements in providing publicly accessible datasets, including real-world and simulated data, to support the training and evaluation of ML-based IDSs [21,24–30]. These datasets play a crucial role in facilitating the development and evaluation of ML-based IDSs in the field of vehicle security.

Despite these advancements, there is a need to evaluate the quality of these published datasets to ensure their effectiveness and suitability for IDS training and evaluation purposes. However, existing studies have predominantly focused on qualitative aspects, overlooking the quantitative measures of attack dataset quality [29,31,32]. We demonstrate the limitations of currently available published datasets for automotive ML-based IDS to illustrate the importance of high-quality datasets. Furthermore, we assert that qualitative and quantitative measures of dataset quality should be considered during the construction of datasets. In light of these challenges, we propose qualitative and quantitative metrics to evaluate dataset quality effectively. By employing these proposed metrics, researchers and practitioners can assess and enhance the quality of datasets, ultimately advancing the development and evaluation of ML-based IDSs for automotive applications.

To demonstrate the necessity of high-quality datasets, we evaluate the detection performance of various ML-based IDSs trained on commonly used public datasets. Specifically, we investigate the performance of these IDSs when subjected to attacks with slight variations or novel attack types. The findings reveal limitations in these datasets, such as the lack of generalization across specific attack types and the absence of up-to-date attacks, resulting in decreased detection performance. To effectively evaluate these datasets, we propose both qualitative and quantitative metrics that can be utilized to assess their quality. The qualitative metrics encompass evaluating the dataset description document, dataset collection environment, and other relevant factors. On the other hand, the quantitative metrics focus on evaluating the complexity of performed attacks, balance, and other quantitative aspects. These metrics provide valuable guidelines for evaluating datasets used in developing automotive IDSs. Finally, we conduct an in-depth analysis of datasets for automotive IDS development by applying the proposed metrics. This analysis offers insights into the strengths and weaknesses of these datasets, enabling informed decisions regarding their suitability for training and evaluating automotive IDSs. The evaluation results demonstrate that the CarChallenge [28] and TTIDS [30] datasets, both typical attack datasets and enhanced attack datasets, are the most suitable for training and evaluating ML-based IDSs.

Our detailed contributions are as follows:

1) We demonstrate the effectiveness of publicly available datasets for ML-based IDSs in the automotive domain, highlighting their limitations and emphasizing the need for high-quality datasets.
2) We propose qualitative and quantitative metrics to fairly evaluate datasets when it comes to the development of automotive IDSs. Based on proposed metrics, the evaluation of publicly accessible datasets is presented to identify limitations and shortcomings.
3) We present recommendations for dataset compliance, aiming to guide the criteria that datasets should meet for improved performance and effectiveness of automotive IDSs.

2 Motivation

There has been a growing availability of datasets collected from real vehicles or simulated sources, aiming to facilitate the development of ML-based IDSs. These datasets have been widely used by researchers to train and evaluate ML-based IDSs. However, the performance evaluation in previous studies has primarily focused on specific datasets without considering the impact of different datasets on performance. Therefore, it is essential to investigate how the performance of ML-based IDSs, known for their excellent performance, changes when exposed to new attack types or minor variations in existing attack types. This highlights the significant role played by the quality of datasets used for training and evaluation.

One of the commonly used datasets for training and evaluating ML-based IDSs is the Car-Hacking dataset [21], developed by the Hacking and Countermeasure Research Lab (HCRL). This dataset includes both attack-free data and attack datasets collected from a real vehicle. Researchers have employed the Car-Hacking dataset to evaluate the performance of their ML-based IDS approaches [18,33]. For instance, Song et al. [18] proposed an automotive IDS based on a deep convolutional neural network (DCNN) and assessed its performance using the Car-Hacking dataset, with a specific focus on detection accuracy. The researchers reported an impressive F1 score of 99.9%, indicating a high level of accuracy in detecting attacks. Similarly, Minawi et al. [33] presented an ML-based IDS employing various algorithms, such as Naïve Bayes, Decision Tree, Stochastic Gradient

Descent (SGD), and Random Forest. They evaluated their IDS using the Car-Hacking dataset and reported an average F1 score of approximately 98%.

To further investigate the performance of ML-based IDSs, we conducted an experiment using new datasets. The experiment consisted of two steps. Firstly, the ML-based IDSs were evaluated using different datasets, including the same types of attacks present in the Car-Hacking dataset, such as DoS, fuzzing, and spoofing attacks. In 2020, HCRL released another CAN dataset called "CAR Hacking: Attack & Defense Challenge 2020" [28], referred to as CarChallenge, which also included these attack types. The ML-based IDSs were trained and evaluated using these datasets. Secondly, we evaluated the ML-based IDSs using a dataset that introduced new attack types. Lee et al. [30] published a dataset featuring masquerade attacks conducted by exploiting the diagnostic service or bus-off mode to suspend an ECU.

The F1 scores of the ML-based IDSs utilized in this research are presented in Table 1. Notably, each ML-based IDS exhibited distinctly different F1 scores depending on the attack type. This finding highlights the significant impact of dataset quality on the performance, accuracy, and adaptability of automotive IDSs. As a result, this thesis puts forth evaluation criteria specifically tailored for open datasets in the field of automotive IDS. This research aims to address the critical need for high-quality datasets by offering a set of fair and comprehensive metrics. The proposed evaluation criteria will facilitate the development of IDS models capable of effectively detecting and classifying various attack types.

Table 1: Results of the ML-based IDSs

Methods	DoS	Fuzzing	Spoofing	Masquerade (w/UDS)	Masquerade (w/bus-off)
Naïve Bayes [33]	100%	59.55%	4.59%	13.15%	15.06%
Decision tree [33]	100%	91.20%	0.00%	66.74%	71.39%
SGD [33]	100%	61.79%	0.00%	22.85%	24.63%
Random forest [33]	100%	98.28%	0.00%	66.74%	71.39%
DCNN [18]	99.99%	47.87%	0.51%	61.93%	71.68%
LSTM [18]	100%	98.9%	0.00%	37.86%	50.18%

3 Preliminaries

3.1 Controller Area Network (CAN) Protocol

Robert Bosch introduced the CAN protocol in the early 1980s to reduce the wiring complexity of automobiles. The CAN protocol is a message-based broadcast protocol through which each message is transmitted sequentially and received by every ECU on the CAN bus. The high-speed CAN variant of the CAN protocol has a maximum signaling rate of 1 Mbit per second. However, in practice, the typical bit rate used on a high-speed CAN is around 500 kbit per second for reliable data transmission.

The CAN protocol has four frame types: the data frame, which is used to transmit data; the remote frame, which is used to request data; the error frame, which is sent if an ECU detects an error; and the overload frame, which is used to insert a delay between data or remote frames. The most commonly used frame in the CAN protocol is the data frame, which is supported in both standard and extended formats. The standard format utilizes an 11-bit identifier, while the extended format

exclusively employs a 29-bit identifier. The 29-bit identifier is composed of an 11-bit base identifier and an additional 18-bit extended identifier. Commercial vehicles commonly employ data frames in the standard format for CAN communications, as illustrated in Fig. 1. The important fields within the data frame, namely the identifier field (ID) and the data field, are highlighted in green. These fields carry the relevant information for communication. Other fields are associated with the protocol structure. The data length code (DLC) indicates the number of bytes in the data field, with a maximum payload of 8 bytes in classical CAN. The CAN specification defines dominant and recessive bits, denoted by (0) and (1), respectively. If one ECU transmits a dominant bit (0) and another ECU transmits a recessive bit (1), the dominant bit will dominate the bus. Therefore, if more than one ECU wants to transmit to the bus, the message with the highest priority, which is encoded in the identifier field (i.e., the CAN ID), wins the arbitration and is allowed to transmit its message.

SOF (1-bit)	Identifier (11-bits)	RTR (1-bit)	IDE (1- bit)	RB0 (1-bit)	DLC (4-bits)	Data (0-8 bytes)	CRC (15-bits)	CRC Delimiter (1-bit)	ACK (1-bit)	ACK Delimiter (1-bit)	EOF (7-bits)

Figure 1: CAN data frame format (Standard frame)

For fault-tolerant communication in the CAN protocol, a robust error-handling mechanism is also defined to prevent erroneous messages from propagating and to stop faulty nodes from disrupting communications. ECUs can automatically detect and take appropriate action, such as discarding a frame, broadcasting an error flag, and retransmitting a frame. Each ECU can exist in one of three modes: error-active, error-passive, and bus-off modes. The determination of these states depends on the values of the transmit error counter (TEC) and the receive error counter (REC). Whenever an ECU encounters an error while transmitting a frame, its TEC increases by 8, while observing an error while not transmitting leads to an increment of 1 in the REC. Successful transmits or receives result in a decrement of 1 in the respective counters. Fig. 2 illustrates a state machine diagram that provides a visual representation of the state transitions. Initially, all ECUs begin in the error-active mode. Once either the TEC or REC exceeds 127, the ECU switches to the error-passive mode (Fig. 2-①). If both the TEC and REC remain below 128, the ECU switches back to the error-active mode (Fig. 2-②). Furthermore, if the TEC surpasses 255, the ECU enters the bus-off mode (Fig. 2-③), which restricts its influence on the bus. Importantly, an ECU is allowed to automatically revert to the error-active mode after monitoring 128 occurrences of 11 consecutive recessive bits (1) on the bus (Fig. 2-④).

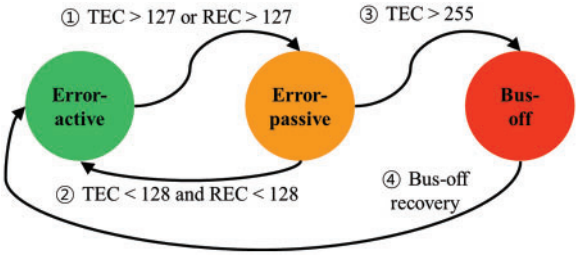

Figure 2: State machine of fault confinement in the CAN

3.2 Attacks on CAN Bus Communication

In general, CAN attacks can be categorized according to the network layer at which the attack is performed. For example, the application layer or the data link layer [34]. In the application layer, an adversary accessing an in-vehicle network can typically send and receive messages without any limitations regarding the ID or payload. Fig. 3 shows two access points of automotive CAN: OBD-II-based physical access and telematics unit-based remote access. The lack of security, such as encryption or authentication, allows an attacker with access to the CAN bus to take control of safety-critical functions related to the vehicle's essential driving functions, such as braking, accelerating, and steering. It is noted that the majority of existing research works on ML-based IDSs have focused on detecting application layer attacks. Accordingly, typical examples of attacks that can occur at the application layer are described.

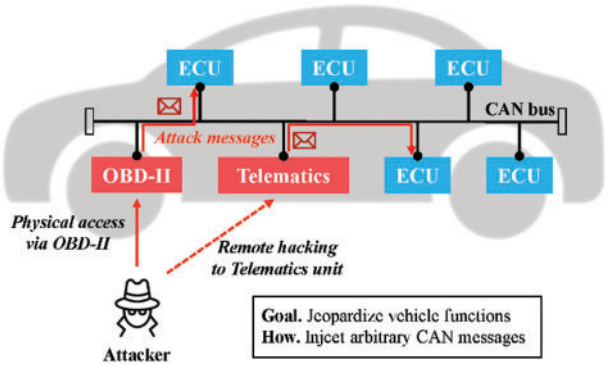

Figure 3: Access points to the in-vehicle network

3.2.1 DoS Attack (or Flooding Attack)

A DoS attack is the easiest way to conduct an attack on the CAN bus. CAN messages with a CAN ID of 0x000 and an arbitrary payload are injected at a high frequency, as shown by Miller and Valasek [35]. Since 0x000 is the highest priority in the arbitration decision, many other CAN messages with a priority lower than 0x000 are delayed or fail to deliver. The consequences of such an attack can be disruptive to the vehicle's operation and may lead to system malfunctions or even temporary loss of control.

3.2.2 Fuzzing Attack

In a fuzzing attack, CAN IDs and the data payload are randomly selected and injected at a high frequency. Unlike a DoS attack, injected messages may include the CAN IDs that appear in normal traffic. Accordingly, an ECU may receive these CAN messages and address functions according to the payload because they include legitimate CAN IDs. By analyzing the vehicle's reaction, the attacker can learn more about the target [2]. This type of attack can compromise the vehicle's safety and security, as it provides valuable information to the attacker about vulnerabilities and potential points of exploitation.

3.2.3 Replay Attack

For a replay attack, attackers can carefully observe message sequences during a specific time interval and save the observed message sequences [36]. Then, all or part of these legitimate message

sequences are injected. Each payload contains a valid CAN control message. Hence, it can cause possible damage or unexpected vehicular behavior. The impact of a successful replay attack can result in unauthorized control over vehicle systems, manipulation of critical functions, or even compromising the safety of the vehicle occupants.

3.2.4 Spoofing Attack (or Fabrication Attack)

If an attacker knows information regarding which CAN ID is used for a particular vehicular function, they could transmit a forged message with that specific CAN ID onto the bus to achieve the desired result. Miller et al. [35] executed a spoofing attack on a Ford Escape, in which the dashboard shows a Door Ajar notification even though the door is closed. However, since the legitimate ECU is still active and continues to transmit correct messages, an attacker must force the receiving ECU to react to the conflicting messages. To this end, the frequency of spoofed messages is much higher (typically up to 100 times) than the transmission rate for correct messages [4]. A successful spoofing attack can result in false information being displayed or acted upon by the vehicle's systems, potentially leading to dangerous situations or manipulation of critical functions.

3.2.5 Diagnostic Attack

The unified diagnostic services (UDS), codified in ISO-14229, is a diagnostic communication protocol used over the CAN protocol for vehicle diagnostics [37]. A diagnostic attack aims to acquire information about the vehicle or ECUs connected to the CAN bus using UDS. The range of identifiers in diagnostic messages is known to be from 0x700 to 0x7FF [38]. Miller et al. [35] transmitted some diagnostic messages to Ford and Toyota to obtain information that kills the engine or controls the fuel gauge. Lee et al. [30] forced certain ECUs to enter diagnostic mode by transmitting some diagnostic messages. The consequences of a successful diagnostic attack can range from unauthorized access to sensitive vehicle data to manipulation of critical functions, potentially compromising the safety and security of the vehicle.

3.2.6 Bus-Off Attack

A bus-off attack intentionally puts a specific ECU into bus-off mode by exploiting the error-handling mechanism. To make the target ECU's TEC exceed 255, the attacker generates a bit error while the target ECU is transmitting a message [39,40]. Although the ECU in the bus-off mode recovers to error-active mode after a certain time and resumes message transmission [30,41], it may cause unexpected vehicular behavior due to the bus-off mode. A successful bus-off attack can result in temporary loss of communication with the targeted ECU, disruption of critical functions, and potentially compromise the overall system's reliability.

3.2.7 Suspension Attack

A suspension attack attempts to stop or suspend the transmission of a legitimate ECU. The effect of a suspension attack is similar to that of a DoS attack. However, while the DoS attack significantly increases message traffic, the suspension attack does not increase the busload. Rather, message traffic is reduced such that it is equal to suspended transmission. Lee et al. [30] showed suspension attacks using diagnostic services and a bus-off attack. A successful suspension attack can disrupt the normal operation of the targeted ECU, potentially affecting critical vehicle functions or compromising system reliability.

3.2.8 Masquerade Attack

A masquerade attack can be seen as a combination of a suspension attack and a spoofing attack. An attacker suspends message transmission from a specific ECU and transmits manipulated messages at the same frequency to the CAN bus to maintain the original CAN busload [31]. Lee et al. [30] first published masquerade attack datasets performed on real vehicles by exploiting diagnostic services and a bus-off attack. A successful masquerade attack can result in unauthorized control over vehicle systems, manipulation of critical functions, and potentially compromising the safety and security of the vehicle occupants.

3.3 Intrusion Detection Systems (IDSs)

Intrusion detection has attracted the attention of many researchers seeking to address the ever-increasing issue of intrusive activities on the CAN bus. Automotive IDSs can be smoothly integrated into existing or new vehicles, as they do not require substantial changes to the CAN communication protocol. As a result, a large amount of research is being carried out related to these IDSs in the industry as well as in academia [12–21,33,42,43]. Automotive IDSs can be primarily divided into two categories based on the detection technique: rule-based and ML-based approaches. We describe the two categories as follows.

3.3.1 Rule-Based IDS

A rule-based IDS defines rulesets representing behavior patterns over the CAN bus and detects an intrusion whose behavior is different from the ruleset [12]. The IDS send an alert when an intrusion is detected based on the observed events and the known attack patterns. The simplest approach is to define rulesets that can be gathered from the formal specifications of a given vehicle or by monitoring messages on the CAN bus [13,14]. For example, the ruleset can be established based on a set of valid message IDs, DLC, and data fields transmitted on a CAN bus. This type of IDS can easily and accurately detect attacks injected with CAN messages that fall outside the specifications but can be easily thwarted by an attacker who injects attack messages that conform to the specifications. Hence, while this approach can be useful against DoS or fuzzing attacks, it becomes useless for specification-compliant attacks such as replay or spoofing attacks. Since almost every CAN message is periodically transmitted on the CAN bus, the periodicity or time interval of CAN messages is also one of the rulesets used in automotive IDSs. The injection of CAN messages by an attacker will increase the transmission frequency or change the statistical properties related to periodicity [15,16]. Song et al. [15] analyzed the time interval of the CAN bus message with the same ID and were able to detect a message as an injected message when the time interval was below half of the normal using their proposed approach. Although these methods are relatively simple and cost-effective, they fail to cover more sophisticated attacks that precisely mimic the original message frequency and even comply with the specification.

3.3.2 Machine Learning-Based IDS (ML-Based IDS)

Machine learning algorithms are powerful mathematical tools used extensively in computer and artificial intelligence. These tools have helped with classification, regression, and clustering, and thus, they can also be used to develop automotive IDSs. These algorithms are especially suitable for detecting novel or previously unknown attacks [9]. An ML-based IDS can be designed along with a classification algorithm like a support vector machine (SVM), K-nearest neighbor network (KNN), or tree-based approach to learning the normal (or attack) behavior of network traffic [42,43]. These IDSs

monitor the current traffic, and any significant deviation from that will be considered an intrusion on the CAN bus. Taylor et al. [43] utilized a supervised one-class support vector machine (OCSVM) in detecting any deviations from normal frequencies of the CAN messages.

In recent years, many ML-based techniques such as convolutional neural network (CNN) [18], deep neural network (DNN) [19], artificial neural networks (ANN) [20], and generative adversarial networks (GAN) [21] have been adopted and introduced to create enhanced ML-based IDSs. Kang et al. [19] proposed an intrusion detection method utilizing DNN, which effectively extracted low-dimensional features from IVN traffic exchanged between ECUs. The method achieved a high detection rate (99.8%) in distinguishing normal and attack packets. Building upon this, Song et al. [18] introduced an IDS based on DCNN for CAN bus intrusion detection. By training the model with 29 consecutive CAN IDs, the approach exhibited lower false negative rates for DoS, fuzzing, and spoofing attacks. However, these ML-based IDSs lack explainability and struggle to extract effective features from multi-dimensional time-series data [44]. Moreover, they often require substantial computational resources for training and execution, posing challenges for real-time detection on ECUs with limited memory and operating speed [45]. Furthermore, the efficacy of ML-based IDSs relies on the availability of diverse network traffic samples encompassing both normal and attack instances for comprehensive training and validation of the models [31]. Notably, in practical scenarios, even minor changes to datasets or new attack types can lead to significant performance degradation, as discussed in Section 2. Therefore, it is crucial for security researchers involved in IDS development and evaluation to conduct thorough dataset analyses, considering relevant security properties.

4 Related Work

Many researchers focus on analyzing benchmark datasets for IDSs in Ethernet networks. In contrast, only a few researchers have presented an analysis of recent datasets for automotive IDSs. This section summarizes some studies that analyzed datasets for automotive IDSs.

Swessi et al. [31] presented the review results for six publicly accessible CAN datasets that were labeled. They presented 11 metrics as the criteria for evaluating each dataset, including a labeled dataset size, a diversity of real traced attacks, online availability, detailed documentation, and so on. Based on the qualitative evaluation, the Real Oak Ridge National Laboratory (ORNL) Automotive Dynamometer (ROAD) and Survival analysis datasets were recommended for in-vehicle ML-based IDSs. Although the authors presented various criteria, the given criteria are all qualitative, and detailed explanations about these remain insufficient. This can lead to different assessments of the dataset depending on the experience and understanding of the researchers.

Verma et al. [29] analyzed several automotive IDS datasets and presented the ROAD CAN dataset, consisting of over 3.5 h of one vehicle's CAN data. The researchers pointed out some drawbacks of the datasets by focusing on discrepancies between the documentation and the actual data for the existing CAN datasets. Moreover, the authors pointed out that simple attacks are only included in the datasets, which any automotive IDS can easily detect based on the CAN ID or periodicity verification. Because in-vehicle attacks are becoming more complicated, the datasets should include state-of-art attacks like a masquerading attack to support ML-based IDSs. We propose quantitative metrics to measure the degree of similarity of an attack against normal traffic. We believe that datasets including attacks that performed similarly to normal traffic are more important when it comes to training and evaluating ML-based IDSs.

Most recently, Vahidi et al. [32] investigated various characteristics of datasets for automotive Intrusion Detection Systems (IDSs) and developed several metrics based on Lawrence's data readiness

levels concept [46]. The researchers proposed a heuristic method to estimate the level of complexity captured by a dataset, indicating the difficulty of the problem it presents. They trained four ML-based IDSs using the Car-Hacking DoS) dataset to evaluate their approach. Subsequently, they assessed the IDSs' performance using a new DoS attack dataset, where all attack messages were modified to employ a different high-priority ID. Remarkably, the authors demonstrated that even this minor alteration significantly degraded the performance of the IDSs. While their method emphasizes the significance of the dataset employed in ML-based IDS training, we introduce criteria to measure attack difficulty based on the similarity between the attack dataset and the normal dataset. Our criteria offer a quantitative assessment of attack complexity, focusing exclusively on the dataset.

5 Evaluation Metrics for Automotive IDS Datasets

Many automotive IDSs to detect attacks on the CAN bus have been extensively studied. However, the performance evaluation of these automotive IDSs is hindered by the absence of standardized datasets. As discussed in Section 2, an IDS may exhibit favorable results on one dataset but yield unsatisfactory outcomes on another. Therefore, there is a need for metrics that enable a fair and comprehensive evaluation of automotive IDS datasets. In this section, we present a comprehensive set of qualitative and quantitative metrics meticulously designed to facilitate an equitable assessment of published datasets for automotive IDSs. Fig. 4 provides an overview of our proposed evaluation framework, including quantitative and qualitative metrics. Each dataset is evaluated based on the proposed quantitative and qualitative metrics to identify its strengths and shortcomings.

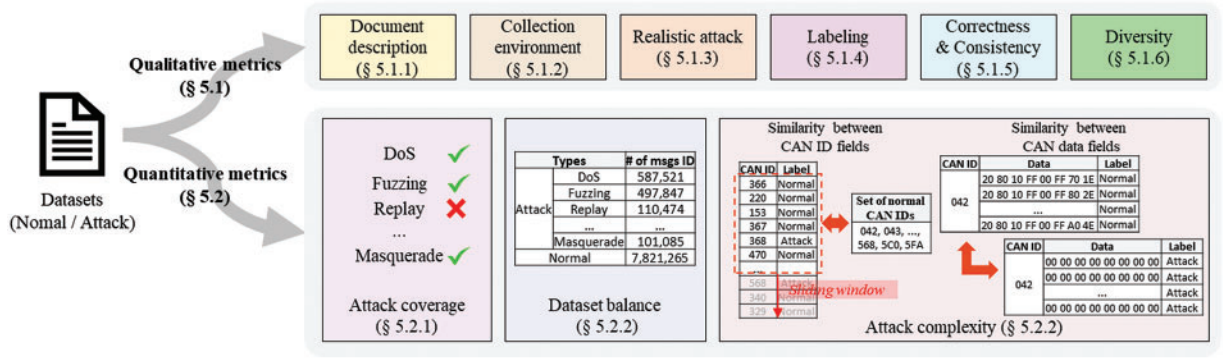

Figure 4: Overview of the proposed evaluation framework

5.1 Qualitative Metrics

5.1.1 Dataset Documentation

The presence of comprehensive documentation describing a dataset plays a pivotal role in assessing its quality and usability. Adequate documentation accompanying the dataset facilitates an evaluation of the dataset's creation methodology and its potential contributions to IDSs. Insufficient information regarding network configuration during dataset creation, recording procedures, and included attack scenarios can diminish the dataset's usefulness. Moreover, datasets collected from real vehicles often encompass diverse environments. Consequently, integrating vehicle contexts, such as driving conditions (e.g., driving or idling) and status information (e.g., speed and gear position), becomes crucial for correlation and the development of context-aware IDSs [47]. Hence, we propose four key criteria for evaluating dataset documentation:

1) General description: This includes details about the content, age, size, origin, and objective of the dataset.
2) Environment description: It encompasses information on where and how the dataset was recorded or attacks are executed.
3) Attack description: This entails providing insights into the type of attacks, attack setups, and the effects of the attacks.
4) Vehicle context description: This involves describing the vehicle context in which the data was collected (e.g., driving, parking) and the features of the dataset (e.g., the number of IDs, the values of signals).

5.1.2 Dataset Collection Environment

The collection of datasets for automotive IDSs can be categorized into two main types: synthetic generation (e.g., simulations) and measurement from real vehicles. Synthetic datasets created within simulated environments offer the advantage of being easily controlled and tailored by their creators. However, these datasets often exhibit a higher prevalence of unrealistic data. In order to create an ideal automotive dataset that encompasses both normal and abnormal conditions, it is crucial to incorporate a diverse range of network properties that are inherent to real vehicles. These properties may include the arbitration process, transmission delay, and network delay. Although acquiring data from real vehicles can be a time-consuming endeavor, such datasets provide valuable insights into a wide range of meaningful features that can be utilized for classifying normal and abnormal samples. Therefore, automotive datasets collected from real vehicles are considered to be of high quality and hold significant value for research and analysis purposes.

5.1.3 Realistic Attack Dataset

There are two methods for creating datasets that accurately represent attacks on vehicles. The first method involves capturing real-time data from a vehicle while executing various attack scripts. This approach provides authentic attack scenarios and captures the actual effects on the in-vehicle network (IVN). The second method involves synthesizing abnormal samples based on a previously collected dataset to generate diverse attack datasets. This method offers the advantage of easier creation of varied attack scenarios. However, the synthetic approach has limitations. Modifying timestamps of CAN messages manually to create synthetic attacks may result in an unrealistic dataset, as these timestamps are influenced by factors such as transmission frequency and priority. Moreover, the synthesized attacks may not fully capture the complex dynamics of the CAN protocol, including natural variations in message timing, content, and events. Consequently, the quality of the dataset may be compromised in unknown ways. Additionally, physically verifying the impact of synthesized attacks on vehicles is not feasible. Therefore, when training or testing automotive IDSs, it is crucial to consider the representativeness and accuracy of the events occurring on the IVN. Collecting datasets while executing actual attacks against vehicles is considered to yield high-quality datasets as they closely resemble the real effects of the attacks and the corresponding changes in the IVN.

5.1.4 Labeled Dataset

Labels play a crucial role in datasets by providing meaningful and informative tags that offer data context [48]. Each sample in a dataset can be associated with one or more labels that indicate whether it represents normal or attack conditions. These labels are essential training features for ML-based IDSs. The accuracy and representativeness of the labels significantly impact the performance of IDSs.

Without correct labels, the dataset becomes unusable, and any analysis results derived from it lack validity and reliability. Therefore, it is imperative for high-quality datasets to provide accurate labels.

5.1.5 Correctness and Consistency

Datasets should be presented in a standardized format, such as text or CSV, to ensure compatibility with machine learning frameworks, facilitating tasks like feature selection and data normalization. Additionally, it is crucial that all messages within the dataset adhere to the correct format and are free from corruption. Inconsistencies in the dataset format or the presence of corrupt messages impose unnecessary burdens on researchers, necessitating additional analysis or correction efforts. Therefore, it is essential to provide datasets in a consistent format without any corrupt data, promoting smoother research processes and reliable analysis.

5.1.6 Dataset Diversity

The diversity of a dataset refers to the inclusion of data collected from various environments during its creation. These environments encompass different vehicle types, driving conditions, and the involvement of multiple drivers. An ML-based IDS trained solely on a dataset derived from a limited environment may not perform effectively in real-world scenarios. For instance, if the training data only consists of samples gathered while the vehicle is parked, normal driving events such as shifting gears or using turn signals could be mistakenly identified as attacks. Hence, a high-quality dataset should encompass data obtained from diverse real-world driving conditions, multiple vehicles, multiple drivers, and a range of attack scenarios to ensure its reliability and applicability.

5.2 Quantitative Metrics

5.2.1 Attack Coverage

In the domain of computer networks, Sharafaldin et al. [49] introduced a metric known as attack diversity, which quantifies the number of distinct attack types present in a dataset. In line with this concept, we propose a similar metric called attack coverage, which pertains to the number of attack types specifically targeting the CAN bus. Attack coverage can be expressed as follows:

$$\textit{attack coverage} = \frac{|A \cap N|}{|N|} \tag{1}$$

where A is a set of attack types in a dataset and $N \in \{\text{DoS, Fuzzing, Replay, } \ldots\}$ corresponds to the set of eight possible attack types described in Section 3.2. We assert that a high-quality dataset should encompass a broad range of CAN attack types.

5.2.2 Dataset Balance

This metric is a quantitative balance between the number of CAN messages for benign and attack types. In the field of machine learning, it is typically desirable to have roughly equal class sizes [50]. However, it is noteworthy that the published vehicle dataset contains a significantly larger proportion of normal messages compared to attack messages. While certain techniques, such as segmented federated learning [51] or SMOTE [52], exist to address the challenges posed by imbalanced datasets, their applicability to automotive IDSs remains uncertain. Therefore, achieving data balance remains an important consideration for dataset creators and ML-based IDS developers. To assess the extent

of this balance issue, Shannon's entropy [53] is utilized as a measure of dataset balance.

$$H = -\sum_{i=1}^{k} \frac{c_i}{n} log\ \frac{c_i}{n} \tag{2}$$

where n represents the total number of messages in the dataset, and k denotes the number of different types contained within it. Each type is labeled with a specific identifier i, and c_i represents the number of messages belonging to that particular type. If the dataset comprises solely normal messages, the dataset balance is expected to be 0, indicating an imbalance. Conversely, in a balanced dataset containing an equal distribution of normal and attack messages, the dataset balance would be $log\ k$, considering a dataset size of $\frac{n}{k}$. Thus, the equation for evaluating the dataset balance is updated as follows:

$$dataset\ balance = \frac{-\sum_{i=1}^{k} \frac{c_i}{n} log\ \frac{c_i}{n}}{log\ k} \tag{3}$$

It is worth noting that for the suspension attack type, c_i should be zero since no injected messages are present. Assessing the real-world implications of dataset imbalance is a complex and challenging task that can vary depending on various factors [32]. Nonetheless, ensuring dataset balance remains a significant concern for developers of automotive IDSs.

5.2.3 Attack Complexity

Attack complexity is defined as a metric that focuses on the ability of ML-based IDSs to detect attacks. The level of similarity between an injected attack and the normal behavior pattern directly affects the difficulty of detection by ML-based IDSs. To measure attack complexity, we employ the widely used Jaccard similarity measurement, which compares the similarity between the normal and attack datasets. The Jaccard similarity is calculated as the ratio of the intersection of the sets to the union of the sets and is expressed as follows:

$$J\ (A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{4}$$

When two sets are identical, the Jaccard similarity is 1, while a value of 0 indicates no common elements between the sets. We present methods for measuring similarity using the two crucial fields in the CAN data frame. These methods enable the assessment of how closely the dataset collected during an attack resembles the normal dataset. Consequently, if the similarity value obtained from the attack dataset is close to that derived from the normal dataset, it indicates that the performed attack successfully emulates the normal behavior pattern.

(1) *Similarity between CAN ID fields*

Let N represent the set of CAN IDs in a normal dataset, and let A be the set of CAN IDs in a dataset containing a specific attack type. In the case of a fuzzing attack dataset, random CAN IDs ranging from 0x000 to 0x7FF are injected, resulting in N is a subset of A, and the size of A is significantly larger than that of N. As a result, the similarity value between the two sets approaches zero, indicating that the attack deviates from the normal pattern. On the other hand, in a DoS attack, the attack messages consist solely of CAN ID 0x000 and are injected at a high frequency. In this case, N is a subset of A, but the difference in size between the sets is only 1 (i.e., $|A| = |N| + 1$). Consequently, the similarity value of the two sets approaches 1. However, since DoS attacks can be detected by examining the CAN ID validation check alone, the Eq. (4) does not yield the desired similarity result. To address this issue, a proposal is made to calculate the Jaccard similarity using a sliding window approach.

$$J_i(N, A_i) = \frac{|N \cap A_i|}{|N \cup A_i|} \tag{5}$$

where N represents the set of CAN IDs in the normal dataset, which remains constant, and A_i the set of CAN IDs that appear in window W_i of the dataset containing a specific attack type. Analyzing the distribution of these similarities in the attack dataset allows us to assess the similarity of the attacks to the normal behavior pattern.

(2) *Similarity between CAN data fields*

An attacker injects CAN messages containing various data into the CAN bus to induce vehicle malfunctions. The concept of similarity is employed to assess the resemblance between the data frames of the injected attack messages and those of normal messages.

$$J_{ID_i}\left(N_{ID_i}, A_{ID_i}\right) = \frac{\left|N_{ID_i} \cap A_{ID_i}\right|}{\left|N_{ID_i} \cup A_{ID_i}\right|} \tag{6}$$

Here, N_{ID_i} represents the set of data with CAN ID i in the normal dataset, while A_{ID_i} denotes the set of data labeled as an attack with CAN ID i in the dataset specific to a particular attack type. In the scenario of a spoofing attack, the attacker injects malicious messages by modifying a specific signal different from the normal data, thereby inducing malfunctions. The detection of such attacks is easily possible through data validation checks. This is because the IDS monitor previously unobserved signal values associated with the respective CAN ID. Consequently, a small similarity value, approaching 0, suggests that the attack is likely to be detected using a straightforward method. This indicates that the attack was performed without considering the normal data patterns.

6 Analysis of Automotive IDS Datasets

In this section, we review the existing open CAN datasets for training and evaluating ML-based IDSs. Each dataset is analyzed using the proposed qualitative and quantitative metrics.

6.1 Datasets for Automotive IDS

Among the published datasets, we focused on eight publicly accessible in-vehicle CAN datasets. These datasets predominantly capture traffic from the vehicle's diagnostic port (OBD-II). Table 2 presents the automotive IDS datasets that have been included in our analysis.

6.1.1 OTIDS [24]

The OTIDS dataset, initially published by the Hacking and Countermeasure Research Lab (HCRL), is designed for intrusion detection purposes. It consists of datasets collected from a Kia Soul vehicle, encompassing scenarios of DoS attacks, fuzzing attacks, and attack-free conditions. In addition to CAN data frames, the attack datasets incorporate remote frames and response data frames. In the DoS attack dataset, the attackers inject messages with the dominant CAN ID 0x000 at short intervals. Although the authors of the dataset refer to the attack as an impersonation attack, which aligns with the masquerade attack described in Section 3.2, a detailed examination of the datasets reveals that the targeted ECU was not suspended. Instead, it appears that the spoofed remote frame responses were injected. Based on this observation, the attack can be classified as a spoofing attack.

Table 2: Open CAN IDS datasets

Dataset	Organization	Year	Dataset URL
OTIDS [24]	HCRL	2017	https://ocslab.hksecurity.net/Dataset/CAN-intrusion-dataset
Car-Hacking [21]	HCRL	2018	https://ocslab.hksecurity.net/Datasets/car-hacking-dataset
Survival analysis [25]	HCRL	2018	https://ocslab.hksecurity.net/Datasets/survival-ids
Automotive CAN v2 [26]	TU Eindhoven	2019	https://data.4tu.nl/articles/dataset/Automotive_Controller_Area_Network_CAN_Bus_Intrusion_Dataset/12696950/2
SynCAN [27]	BOSCH	2019	https://github.com/etas/SynCAN
CarChallenge [28]	HCRL	2020	https://ocslab.hksecurity.net/Datasets/carchallenge2020
ROAD [29]	ORNL	2020	https://0xsam.com/road/
TTIDS [30]	EMSEC	2022	https://github.com/EmbbededSecurity/AutomotiveSecurity

6.1.2 Car-Hacking [21]

Another dataset released by HCRL, the Car-Hacking dataset, was collected from a Hyundai YF Sonata. This dataset includes three types of attacks: DoS, fuzzing, and spoofing attacks. In the case of spoofing attacks, the attack was focused on altering the information related to Gera/RPM. Each dataset within the Car-Hacking dataset consists of 30 to 40 min of traffic and contains 300 attacks performed through injected messages. Each attack lasts for three to five seconds. For the DoS attack dataset, attacks were conducted by injecting messages with the dominant message CAN ID 0x000 every 0.3 milliseconds. In the fuzzing attack dataset, randomly injected messages with various CAN IDs were performed every 0.5 milliseconds. Lastly, in the spoofing attack dataset, the respective CAN ID message related to RPM/gear information was injected every millisecond to simulate the spoofing attacks.

6.1.3 Survival Analysis [25]

HCRL has published multiple datasets derived from three different vehicles. For each vehicle, they collected normal driving data along with three types of injection attacks that caused the vehicle to malfunction. The datasets consist of three attack categories: DoS, fuzzing, and malfunction attacks, which are equivalent to spoofing attacks. For each attack category and vehicle, the capture duration ranges from 25 to 100 s. Within each capture, there are one to four five-second injection intervals where the attacks are performed.

6.1.4 Automotive CAN Bus Intrusion Dataset v2 [26]

The Eindhoven University of Technology Lab has released datasets specifically designed for evaluating automotive IDS. The data was collected from real-world driving scenarios using an Opel Astra and Renault Clio in a city environment. Additionally, the researchers also built a CAN bus prototype to capture data. This dataset encompasses various types of attacks, including DoS, fuzzing, replay, suspension, and diagnostic attacks.

6.1.5 SynCAN [27]

Hanselmann et al. [27] from Bosch GmbH recognized the scarcity of standardized and comprehensive CAN datasets. In response, they released a synthesized and signal-translated CAN dataset called SynCAN. This dataset serves as a benchmark for evaluating and comparing different CAN IDSs in various attack scenarios within the signal space. SynCAN includes attacks such as DoS, suspension, and masquerade attacks, with each attack captured for multiple intervals ranging from 2 to 4 s. Notably, the authors provided detailed classification for the masquerade attack, distinguishing it into three types: continuous, plateau, and replay, based on the method of data tampering employed in each masquerade attack instance.

6.1.6 Car-Hacking: Attack & Defense Challenge [28]

HCRL has released a new dataset called CarChallenge as part of the Car-Hacking: Attack & Defense Challenge. This dataset includes various types of attacks, such as DoS, fuzzing, spoofing, and replay attacks. It was collected from a real vehicle and covers attacks that occurred both when the vehicle was paring and driving. Unlike previous datasets that focused on specific attack types, CarChallenge combines all these attacks into a comprehensive dataset. The dataset was used in the "Car-Hacking: Attack & Defense Challenge 2020" event in South Korea, allowing researchers and participants to assess the effectiveness of their intrusion detection and defense mechanisms against real-world attack scenarios.

6.1.7 ROAD [29]

Verma et al. [29] introduced the ROAD dataset specifically designed for evaluating automotive IDSs. Notably, it was the dataset to include a masquerade attack, allowing for physical verification of its impact on the vehicle. The dataset consists of captures collected from a single vehicle, comprising 12 attack-free captures spanning approximately three hours, and 33 attack captures lasting around 30 min. The dataset focuses on three primary attack types: fuzzy, spoofing, and masquerade attacks.

6.1.8 TTIDS [30]

The TTIDS dataset, recently introduced by Lee et al. [30], stands out as the first dataset to conduct a masquerade attack on a real vehicle utilizing a diagnostic service or a bus off attack. Unlike previous datasets that relied on artificially generated attack data, the authors obtained the data by intentionally halting the message transmission of a specific ECU in a real vehicle and injecting malicious CAN messages with the same frequency.

6.2 Qualitative Evaluation

In this subsection, we proceed with the evaluation of the eight automotive IDS datasets described earlier, utilizing the proposed qualitative metrics. The essential features of these datasets, as presented in Table 3, will be further discussed in the following section.

Table 3: Summary of main characteristics of datasets

Dataset	Environment	Attack types	Attack	Data attributes	Label	Format	Diversity*
OTIDS [24]	1 vehicle	DoS Fuzzing	Real	Timestamp, CAN ID, DLC, Data	No label	.txt	P
Car-Hacking [21]	1 vehicle	DoS Fuzzing Spoofing	Real	Timestamp, CAN ID, DLC, Data	Label	.txt, .csv	P
Survival analysis [25]	3 vehicles	DoS Fuzzing Spoofing	Real	Timestamp, CAN ID, DLC, Data	Label	.txt	P
Automotive CAN v2 [26]	2 vehicles and 1 testbed	DoS Fuzzing Replay Diagnostic Suspension	Synthetic	Timestamp, CAN ID, Data	Unclearness	.txt	P
SynCAN [27]	1 simulation	Spoofing Suspension Masquerade	Synthetic	Timestamp, CAN ID, Signals	Label	.csv	P
CarChallenge [28]	1 vehicle	DoS Fuzzing Replay Spoofing	Real	Timestamp, CAN ID, DLC, Data	Label	.csv	P, ND
ROAD [29]	1 vehicle	Fuzzing Spoofing Masquerade	Real Synthetic	Timestamp, CAN ID, Data	No label	.txt	P, ND, AD
TTIDS [30]	1 vehicle	Masquerade	Real	Timestamp, CAN ID, DLC, Data	Label	.csv	P

Notes: *: P-Parking, ND-Normal driving, AD-Abnormal driving.

6.2.1 Dataset Documentation

To ensure a fair assessment, we split the datasets into five overlapping sets so that three of the five authors would analyze each dataset. Table 4 presents a summary of the qualitative evaluation of dataset documentation based on four criteria points. All documentation related to the datasets was obtained either online or from other research theses. A general description of the dataset objective, content, and source was provided by all experts, along with information on the data collection location and the types of attacks included. However, it was noted that only a few datasets provided details on the data collection process, such as the hardware and software used, or a comprehensive explanation of the attack, including injection tools, procedures, and their effects. Only two datasets, Automotive CAN v2 and ROAD, describe the tools and libraries used for data collection. Descriptions of the effects of

the attacks on real vehicles are included in only two datasets: ROAD and TTIDS. Most datasets had no description relating to the context of the vehicles, and only SynCAN and ROAD provided the value of signals in which raw CAN data was decoded. In light of the documentation evaluation, only ROAD satisfies all documentation criteria.

Table 4: Qualitative evaluation of dataset documentation

Criteria/Dataset		[24]	[21]	[25]	[26]	[27]	[28]	[29]	[30]
General description	Objective	✓	✓	✓	✓	✓	✓	✓	✓
	Content	✓	✓	✓	✓	✓	✓	✓	✓
	Origin	✓	✓	✓	✓	✓	✓	✓	✓
Environment description	Where*1	✓	✓	✓	✓	✓	✓	✓	✓
	How*2	–	–	–	✓	–	–	✓	–
Attack description	Type	✓	✓	✓	✓	✓	✓	✓	✓
	Setup	✓	✓	✓	✓	–	–	✓	✓
	Effects*3	–	–	–	–	–	–	✓	✓
General description	Objective	✓	✓	✓	✓	✓	✓	✓	✓
	Content	✓	✓	✓	✓	✓	✓	✓	✓
	Origin	✓	✓	✓	✓	✓	✓	✓	✓
Environment description	Where*1	✓	✓	✓	✓	✓	✓	✓	✓
	How*2	–	–	–	✓	–	–	✓	–
Attack description	Type	✓	✓	✓	✓	✓	✓	✓	✓
	Setup	✓	✓	✓	✓	–	–	✓	✓
	Effects*3	–	–	–	–	–	–	✓	✓
Vehicle context description	Context*4	–	–	–	–	–	✓	✓	–
	Features*5	–	–	–	–	✓	–	✓	✓

Notes: *1: Description of the data collection environment, *2: Description of data collection tool, *3: Description of the actual vehicle impact caused by the attack, *4: Description of vehicle environments, and *5: Description of mapping information between IDs and ECUs.

6.2.2 Dataset Collection Environment

The majority of the datasets reviewed in this section were collected from real vehicles, providing valuable insights into real-world scenarios. Notably, the Automotive CAN v2 dataset emerges as a prominent contender, encompassing data from two distinct vehicles, namely an Opel Astra and a Renault Clio, along with a CAN testbed comprising an instrument cluster and two Arduino boards. Significantly, the Survival Analysis dataset affords an opportunity to analyze identical attacks across three different vehicles. The amalgamation of data sets derived from such diverse sources enhances the overall comprehensiveness of the study. Conversely, the SynCAN dataset holds a distinct attribute as it originates from simulations, thereby facilitating a controlled environment to investigate CAN bus attacks. While this synthetic data set possesses the potential to complement the real vehicle data set, it is essential to acknowledge its limitations stemming from its collection under controlled conditions.

6.2.3 Realistic Attack Dataset

The majority of the datasets reviewed in this analysis were acquired during active attacks on genuine vehicles. The DoS, fuzzing, and spoofing attacks predominate due to their ease of injecting attack messages into real vehicles. Conversely, datasets incorporating suspension attacks and masquerade attacks are relatively scarce, as these types necessitate the suspension of ECU's CAN message transmission in real time. Regarding the Automotive CAN v2 and SynCAN datasets, manual replacement of the data field of a specific message within the collected dataset or the deliberate addition of arbitrary messages with adjusted timestamps were employed. However, it is important to note that the effectiveness of these synthetic attacks on real vehicles cannot be verified, rendering them unrealistic. Additionally, the timing of messages is contingent upon individual ID frequencies and priorities during the arbitration. While the SynCAN and ROAD datasets do contain masquerade attacks, they are considered unrealistic due to the manual and random modification of specific message data fields. Notably, the TTIDS dataset stands as the sole dataset encompassing masquerade attacks that are challenging to replicate on real vehicles.

6.2.4 Labeled Dataset

The majority of the datasets under examination include labels for distinguishing between normal messages and attack messages, serving as valuable resources for training ML-based IDSs. However, it should be noted that the OTIDS and ROAD datasets do not provide specific labels for attack messages. In the case of the Automotive CAN v2 dataset, manual acquisition of labels is possible for DoS and diagnostic attacks based on the attack description. Nevertheless, for other types of attacks, the presence of explicit labels is unclear, making it challenging to verify their labeling accuracy.

6.2.5 Correctness and Consistency

The Car-Hacking dataset presents a discrepancy in file formats, where the normal dataset is provided in .txt format and the attack datasets are in .csv format. Furthermore, there is a disagreement in field separators within the datasets. Consequently, IDS developers are required to perform post-processing tasks to address feature selection and data normalization. During the analysis of the Survival datasets, two format discrepancies were identified. Firstly, the normal datasets lack explicit labels, although this can be considered a minor issue that can be addressed by assigning normal tags. Secondly, the field separators in the attack datasets remain consistent, whereas those in the normal datasets exhibit incoherence. Consequently, IDS developers must undertake post-processing steps to rectify this discrepancy.

Most datasets furnish Timestamp, CAN ID, DLC, and CAN Data field as essential data attributes. However, the SynCAN dataset offers decoded signal values instead of raw CAN data. This format presents challenges for many automotive IDSs, as they are typically trained and evaluated using data fields of up to 8 bytes. The discrepancy in data format may require additional preprocessing or adaptation to ensure compatibility between the dataset and IDS requirements. On the other hand, the ROAD dataset provides a valuable feature by including additional decoded signal values through the utilization of CAN reverse technology [54].

6.2.6 Dataset Diversity

The majority of datasets derived from real vehicles exhibited limitations in terms of encompassing only a single vehicle and stationary scenarios. However, the Survival Analysis and the Automotive CAN v2 datasets distinguished themselves by providing datasets involving more than two vehicles.

Additionally, the CarChallenge dataset released datasets capturing both parked and in-motion situations. Notably, the ROAD dataset stands out as it was collected with consideration given to a diverse range of normal and abnormal driving activities, such as seatbelt unbuckling or door opening while the vehicle is in motion. Nevertheless, it is important to acknowledge that none of the datasets under analysis incorporate multiple drivers, which is a noteworthy aspect that remains unaddressed in the existing datasets.

6.3 Quantitative Evaluation

In this subsection, we analyze automotive IDS datasets based on the proposed quantitative metrics. It should be noted that, except for attack coverage among the quantitative metrics, the datasets must have labels for normal and attack messages to enable message-by-message analysis.

6.3.1 Attack Coverage

Fig. 5 illustrates the calculated attack coverage for each dataset. Notably, none of the published datasets have comprehensively covered all the attack types mentioned in Section 3.2. Among the attack types included in the dataset, DoS, fuzzing, and spoofing attacks are the most common, while the bus-off attack is not included in any dataset. The Automotive CAN v2 dataset, which encompasses five attack types, achieves the highest score of 0.625 in terms of attack coverage. It is important to acknowledge, however, that this dataset has limitations as the attacks were not conducted on real vehicles, thereby raising concerns about their applicability to real-world scenarios. On the other hand, the TTIDS dataset achieves the lowest score of 0.125, as it includes only one attack type, namely the masquerade attack. Nevertheless, it is noteworthy that the TTIDS dataset is the first publicly available dataset to feature masquerade attacks conducted on a real vehicle, making it a significant contribution to the field.

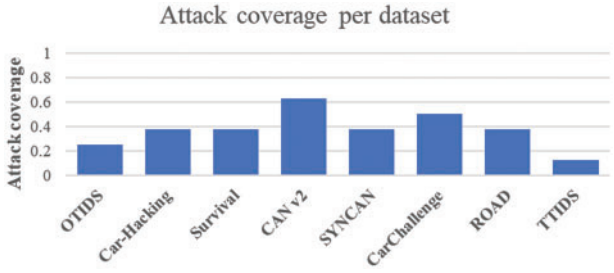

Figure 5: Attack coverage according to attack type

6.3.2 Dataset Balance

Among the eight publicly available datasets, only five of them provide both normal and attack labels, the dataset balance is analyzed for these five datasets. Table 5 displays the balance scores for these five datasets, revealing that all of them exhibit a score below 0.4. The primary reason for this imbalance is that the publicly accessible datasets contain a significantly larger proportion of normal messages in comparison to attack messages. While simulated environments have the potential to generate a greater number of attack messages compared to real vehicle environments, they still tend to display lower average balance scores than the other datasets. Note that despite the inherent imbalance, these datasets can still be utilized effectively. However, it becomes imperative to address this

issue during the training and evaluation processes by employing appropriate techniques to mitigate the effects of class imbalance.

Table 5: Results of dataset balance

Dataset	Type (# of messages)	Balance
Car-Hacking [21]	DoS (587,521)/fuzzing (497,847)/spoofing (1,252,149)/normal (15,226,830)	0.3799
Survival analysis [25]	DoS (88,150)/fuzzing (63,742)/spoofing (31,422)/normal (1,552,526)	0.3211
SynCAN [27]	Spoofing (779,224)/suspension (0)/masquerade (1,264,540)/normal (40,914,617)	0.1608
CarChallenge [28]	DoS (345,859)/fuzzing (216,571)/replay (110,474)/spoofing (200,338)/normal (7,821,265)	0.2844
TTIDS [30]	Masquerade (41,710)/normal (3,574,109)	0.0908

6.3.3 Attack Complexity

The attack complexity serves as an indicator specifically tailored for assessing the likelihood of bypassing ML-based IDSs. A higher resemblance between the attack dataset and the normal dataset presents a greater challenge for ML-based IDS in effectively detecting such attacks. The degree to which an executed attack mimics normal behavioral patterns directly correlates with the difficulty faced by an ML-based IDS in its detection. To evaluate the similarity between the dataset acquired during the injection of an attack and the normal dataset, particular emphasis is placed on the two most crucial fields found within the CAN data frame. As previously mentioned, only five datasets possess both normal and attack labels. Therefore, the examination of attack complexity is exclusively focused on these five datasets.

(1) Similarity between CAN ID fields

The similarity between the sets of CAN IDs in the attack and normal datasets is calculated using a sliding window approach. The number of unique IDs transmitted on the CAN bus of each vehicle is determined from the normal dataset. The sliding window size is adjusted accordingly to match the number of unique IDs for each vehicle, enabling the computation of similarity between the ID sets of the attack and normal datasets. If the similarity scores of the sliding windows containing the attack messages closely resemble those of the normal messages, it indicates that the attack has successfully imitated a normal behavior pattern. Fig. 6 shows the similarity distribution of CAN ID sets for the Car-Hacking dataset, which is extensively utilized in ML-based IDS literature.

This dataset includes DoS, fuzzing, and spoofing attacks, which are the most prevalent attack types. Each attack is executed for a duration of 3 to 5 s, resulting in multiple similarity values at the same elapsed time. In the case of the DoS attack, a message with a CAN ID of 0x000 is injected every 0.3 milliseconds. As depicted in the left section of Fig. 6, normal CAN IDs are infrequently transmitted during the attack, resulting in a dissimilar distribution of similarity values compared to the normal interval. It should be noted that intermittent occurrence of similarity values closes to normal can be observed due to the application of the sliding window. In the fuzzing attack, a message from a random CAN ID is injected every 0.5 milliseconds. Similar to the DoS attack, the distribution of similarity

values significantly deviates from the normal similarity distribution, as shown in the middle section of Fig. 6. In the case of a spoofing attack, where one normal CAN ID message is injected every 1 millisecond, the distribution of similarity values during the attack time is notably lower than that of the normal interval.

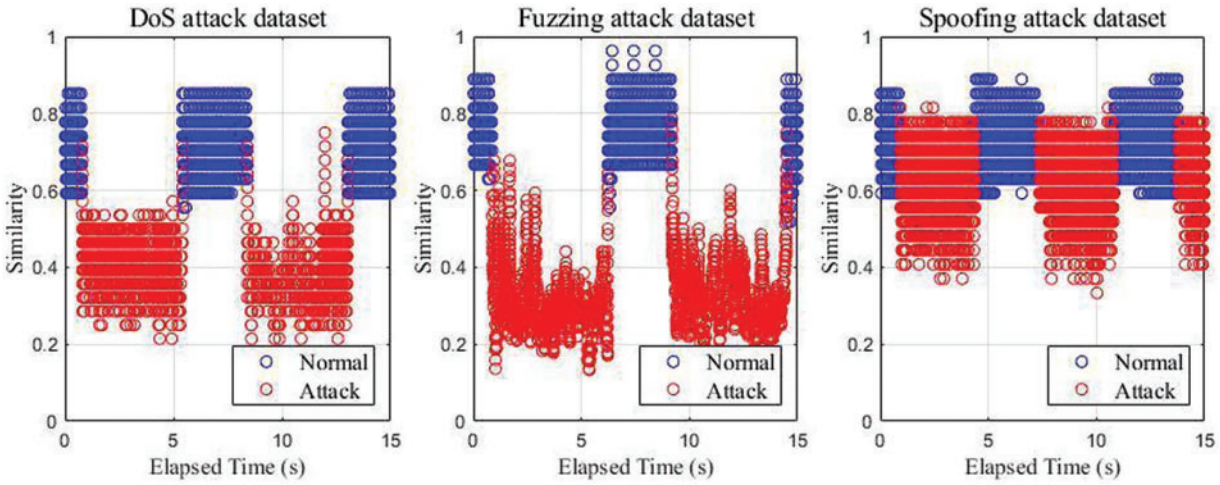

Figure 6: Similarity between CAN ID sets for three attack types

We observed that the similarity distributions of specific attack types in certain datasets differed from our initial expectations. Fig. 7 illustrates the similarity distribution for the CarChallenge attack dataset, which encompasses four attack types: DoS, spoofing, replay, and fuzzing.

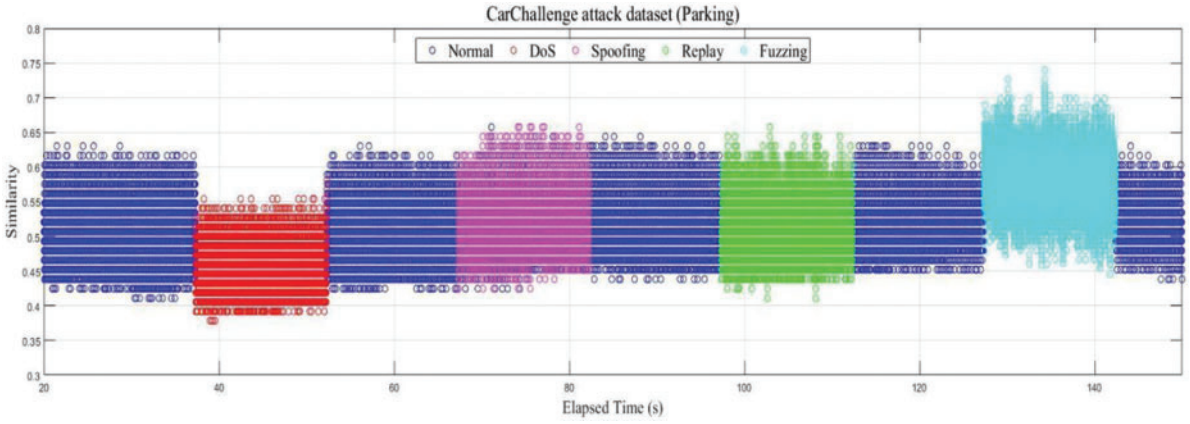

Figure 7: Similarity between CAN ID sets in the CarChallenge attack dataset

In contrast to well-known DoS attacks that aim to disrupt communication between normal ECUs, this dataset demonstrates the injection of a CAN ID 0x000 message every 0.8 milliseconds. This DoS attack allows uninterrupted transmission of normal CAN ID messages. Consequently, during a DoS attack, the similarity values are slightly lower than the normal interval, as indicated by the red point in Fig. 7. The fuzzing attack exhibits a significantly different similarity distribution, represented by

the cyan point in the same figure. Unlike conventional fuzzing attacks that inject random CAN ID messages, this dataset's fuzzing attack employs randomly selected CAN IDs from the normal CAN ID set for injecting attack messages. Consequently, the similarity values during the fuzzing attack turn out to be higher than those during the normal interval. In the case of spoofing attacks, where one normal CAN ID message is injected every 10 milliseconds, the resulting similarity distribution closely aligns with normal conditions. The replay attack exhibits similar characteristics.

Table 6 presents the average similarity values between the CAN ID field sets of the datasets and the deviation from the normal dataset. A close similarity value between the attack dataset and the normal dataset indicates that the attack successfully imitates the normal pattern. Therefore, a smaller deviation signifies a more suitable dataset for evaluating ML-based IDSs.

Table 6: Results of dataset similarity values between sets of CAN ID field

Dataset	Environment (# of CAN IDs)		Types	Similarity value	Deviation (%)
Car-Hacking [21]	Vehicle A (27)		Normal	0.7555	0.00
			DoS	0.4179	44.69
			Fuzzing	0.3441	54.45
			Spoofing	0.6344	16.03
Survival analysis [25]	Vehicle A (27)		Normal	0.7634	0.00
			DoS	0.5434	28.82
			Fuzzing	0.5069	33.60
			Spoofing	0.6919	9.37
	Vehicle B (45)		Normal	0.5783	0.00
			DoS	0.4756	17.76
			Fuzzing	0.3860	33.25
			Spoofing	0.5621	2.80
	Vehicle C (83)		Normal	0.5572	0.00
			DoS	0.4587	17.68
			Fuzzing	0.3859	30.74
			Spoofing	0.5109	8.31
SynCAN [27]	Simulation (10)		Normal	0.8325	0.00
			Spoofing	0.5487	34.09
			Suspension	0.7817	6.10
			Masquerade	0.8325	0.00
CarChallenge [28]	Vehicle A (73)	Driving	Normal	0.5186	0.00
			DoS	0.4573	11.82
			Fuzzing	0.5796	11.86
			Replay	0.5123	1.21
			Spoofing	0.5236	0.96
		Parking	Normal	0.5178	0.00
			DoS	0.4558	11.97
			Fuzzing	0.5177	0.02

(Continued)

Table 6 (continued)

Dataset	Environment (# of CAN IDs)	Types	Similarity value	Deviation (%)
		Replay	0.5082	1.85
		Spoofing	0.5820	12.40
TTIDS [30]	Vehicle A (73)	Normal	0.5167	0.00
		Masquerade	0.5179	0.23

With the exception of the CarChallenge dataset, it is evident that the DoS and fuzzing attack datasets exhibit significant deviations from the similarity value of the normal dataset as a whole. This is primarily due to the injection of messages with a high-priority CAN ID (0x000) or random CAN IDs, which are typically not extensively monitored, at a high frequency. Notably, the fuzzing dataset collected during the parking situation in the CarChallenge dataset is particularly suitable for evaluating ML-based IDSs that extract features from CAN ID fields. Attack messages with regular CAN IDs were observed to be injected within approximately 1.3 milliseconds in this dataset. In contrast, the spoofing attack datasets, specifically Vehicle B from Survival Analysis and Driving from CarChallenge, demonstrate the highest similarity values compared to the normal dataset. This suggests that these datasets are highly effective for training and evaluating ML-based IDSs. The deviation from the normal dataset in terms of similarity values is relatively minor for the replay, abort, and spoof attack datasets. This indicates that datasets containing these attacks are suitable for training and evaluating ML-based IDSs that extract features from CAN IDs.

(2) Similarity between CAN data fields

In the context of data fields, we assess the similarity between the injected CAN ID's data fields and the data fields present in the normal dataset. If an attack message is injected with data fields that do not exist in the normal dataset, a simple ruleset, such as data field validation, can effectively detect the attack. Consequently, datasets that solely contain attacks performed without considering normal traffic are inadequate for training and evaluating ML-based IDSs. To calculate the similarity, we first extract the CAN ID of messages labeled as attacks from a specific attack dataset. We then gather the data fields from both the normal and attack entries associated with that CAN ID. The similarity between the two sets of data fields is subsequently computed. It is important to note that the similarity value cannot be calculated if the injected attack message's CAN ID does not exist in normal circumstances. Similarly, for suspension attacks, where no attack messages are injected, the similarity value cannot be computed. Then, we calculate the similarity values between the attack and normal datasets, excluding these two cases.

Table 7 provides the similarity values between sets of attack and normal data field messages for each dataset. A similarity value closer to 1 indicates that the dataset contains sophisticated attacks that closely imitate the normal pattern. Conversely, if the attack is executed with data fields that do not exist in the normal dataset, the similarity score approaches 0.

Table 7: Similarity score results between sets of data frames for datasets

Dataset	Attack (# of injected CAN IDs)		Similarity value
Car-Hacking [21]	Fuzzing (21)		0
	Spoofing (1)		0
Survival analysis [25]	Vehicle A	Fuzzing (27)	0
		Spoofing (2)	0
	Vehicle B	Fuzzing (45)	0
		Spoofing (1)	0
	Vehicle C	Fuzzing (78)	0
		Spoofing (1)	0
SynCAN [27]	Spoofing (10)		0.1602
	Masquerade (continuous) (9)		0.0671
	Masquerade (plateau) (9)		0.0772
	Masquerade (playback) (9)		0.0676
CarChallenge [28]	Driving	Fuzzing (73)	0
		Spoofing (2)	0
		Replay (73)	0.4896
	Parking	Fuzzing (73)	0
		Spoofing (2)	0
		Replay (73)	0.5233
TTIDS [30]	Masquerade using UDS (3)		1
	Masquerade using bus-off (3)		1

The Car-Hacking and Survival Analysis datasets have been extensively utilized for model training and evaluation in various ML-based IDSs [55–57]. Our analysis of attack complexity reveals that the injected messages intended to generate the fuzzing and spoofing attack datasets consist of data fields that are not present in the normal dataset. As a result, these injected messages can be effectively detected using a simple ruleset-based approach, such as data field validation. Consequently, they are insufficient for evaluating ML-based IDSs that rely on extracting features from CAN data fields. The same holds true for the attack datasets, with the exception of the CarChallenge replay attack dataset. On the other hand, only the two masquerade attack datasets from the TTIDS dataset exhibit the highest similarity score of 1. Since all the data fields of the injected messages for a masquerade attack are already present in the normal dataset, it can be inferred that these sophisticated attacks successfully mimic the normal pattern. Therefore, these datasets are suitable for training and evaluating ML-based IDSs.

7 Discussion

Extended Frame Format. The CAN protocol defines the use of an 11-bit CAN ID for the standard frame format, while the extended frame format utilizes a 29-bit CAN ID. Currently, the

extended frame format is not commonly observed in most real vehicles since the number of ECUs present in an in-vehicle network is typically less than 2,048, allowing for unique CAN IDs to be assigned to each ECU. Consequently, we did not include a metric in our evaluation criteria specifically addressing the extended frame format. However, as in-vehicle networks are expected to become larger and more complex in the future, the limited number of available CAN IDs may necessitate the adoption of the extended frame format. Recognizing this potential future requirement, we plan to update our evaluation metrics to account for this concern. By considering the evolving nature of in-vehicle networks and the potential significance of the extended frame format, we can enhance the comprehensiveness of our evaluation metrics and better address the requirements of future automotive systems.

8 Conclusion

In recent years, automotive IDSs have been studied a lot; accordingly, many approaches have been presented. In particular, the ML-based IDS incorporating machine learning or deep learning algorithms outperforms the other approaches. However, we have shown that the detection performance of the ML-based IDSs significantly degrades with a new dataset even though they greatly perform with the selected datasets. This result implies that the ML-based IDS significantly depends on the datasets. Accordingly, a high-quality dataset is required to develop a general ML-based IDS. In this paper, we have proposed qualitative and quantitative metrics for evaluating publicly available datasets. We subsequently evaluated the datasets using these metrics. Based on our observations, we propose the following recommendations for future dataset creation:

1) To improve a document for the specification of the dataset with proper labels.
2) To collect balanced datasets between normal and attack conditions. Also, the diversity of the conditions from real vehicles is necessary.
3) To execute a variety of attacks on real vehicles when the dataset is collected.
4) To need sophisticated attacks that maintain normal CAN traffics.

In our future work, we are planning to build a useful tool that automatically evaluates the datasets for automotive IDSs. Also, since we evaluated dataset balance and attack complexity metrics only for datasets with labels, we plan to find independent metrics regardless of whether the dataset has a label. We believe that this work would contribute to validating newly released datasets for ML-based IDSs.

Acknowledgement: The authors have expressed gratitude to the colleagues of the Embedded Security Lab (EMSEC) and reviewers that contributed to the article.

Funding Statement: This work was supported in part by the 2021 Autonomous Driving Development Innovation Project of the Ministry of Science and ICT, 'Development of Technology for Security and Ultra-High-Speed Integrity of the Next-Generation Internal Net-Work of Autonomous Vehicles' (No. 2021-0-01348) and in part by the National Research Foundation of Korea (NRF) grant funded by the Korean Government Ministry of Science and ICT (MSIT) under Grant NRF-2021R1A2C2014428.

Author Contributions: The authors confirm their contribution to the paper as follows: study conception and design: Seyoung Lee; data collection: Wonsuk Choi, Insup Kim, Ganggyu Lee; analysis and interpretation of results: Seyoung Lee, Wonsuk Choi, Dong Hoon Lee; draft manuscript preparation: Seyoung Lee, Dong Hoon Lee. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] W. Choi, K. Joo, H. J. Jo, M. C. Park and D. H. Lee, "Voltageids: Low-level communication characteristics for automotive intrusion detection system," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 2114–2129, 2018.

[2] K. Koscher, A. Czeskis, F. Roesner, S. Pater, T. Kohno *et al.,* "Experimental security analysis of a modern automobile," in *Proc. of IEEE Symp. on Security and Privacy (S&P)*, Oakland, CA, USA, pp. 447–462, 2010.

[3] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham *et al.,* "Comprehensive experimental analyses of automotive attack surfaces," in *Proc. of USENIX Security Symp.*, San Francisco, USA, pp. 447–462, 2011.

[4] C. Miller and C. Valasek, "Remote exploitation of an unaltered passenger vehicle," *Black Hat USA*, vol. 2015, pp. 1–91, 2015.

[5] A. I. Radu and F. D. Garcia, "LeiA: A lightweight authentication protocol for CAN," in *Proc. of Int. Conf. European Symp. on Research in Computer Security (ESORICS)*, Heraklion, Greece, pp. 283–300, 2016.

[6] B. Groza, S. Murvay, A. V. Herrewege and I. Verbauwhede, "Libra-CAN: A lightweight broadcast authentication protocol for controller area networks," in *Proc. of Int. Conf. on Cryptology and Network Security (CANS)*, Darmstadt, Germany, pp. 185–200, 2012.

[7] H. J. Jo and W. Choi, "A survey of attacks on controller area networks and corresponding countermeasures," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 6123–6141, 2021.

[8] S. F. Lokman, A. T. Othman and M. H. Abu-Bakar, "Intrusion detection system for automotive controller area network (CAN) bus system: A review," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 184, pp. 1–17, 2019.

[9] W. Wu, R. Li, G. Xie, J. An, Y. Bai *et al.,* "A survey of intrusion detection for in-vehicle networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 919–933, 2019.

[10] R. Kurachi, H. Takada, H. Ueda and S. Takimoto, "Towards minimizing MAC utilization for controller area network," in *Proc. of Second ACM Workshop on Automotive and Aerial Vehicle Security (AutoSec)*, New York, NY, USA, pp. 45–50, 2020.

[11] H. Olufowobi, S. Hounsinou and G. Bloom, "Controller area network intrusion prevention system leveraging fault recovery," in *Proc. of ACM Workshop on Cyber-Physical Systems Security & Privacy (CPS-SPC)*, New York, NY, USA, pp. 63–73, 2019.

[12] N. Salman, "Design and implementation of an intrusion detection system (IDS) for in-vehicle networks," M.S. Dissertation, Chalmers University of Technology, University of Gothenburg, Gothenburg, Sweden, 2017.

[13] U. E. Larson, D. K. Nilsson and E. Jonsson, "An approach to specification-based attack detection for in-vehicle networks," in *Proc. of IEEE Intelligent Vehicles Symp. (IV)*, Eindhoven, Netherlands, pp. 220–225, 2008.

[14] M. Müter, A. Groll and F. C. Freiling, "A structured approach to anomaly detection for in-vehicle networks," in *Proc. of Sixth Int. Conf. on Information Assurance and Security (IAS)*, Atlanta, GA, USA, pp. 92–98, 2010.

[15] H. M. Song, H. R. Kim and H. K. Kim, "Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network," in *Int. Conf. on Information Networking (ICOIN)*, Kota Kinabalu, Malaysia, pp. 63–68, 2016.

[16] M. Müter and N. Asaj, "Entropy-based anomaly detection for in-vehicle networks," in *Proc. of IEEE Intelligent Vehicles Symp. (IV)*, Baden-Baden, Germany, pp. 1110–1115, 2011.

[17] V. K. Kukkala, S. V. Thiruloga and S. Pasricha, "Indra: Intrusion detection using recurrent autoencoders in automotive embedded systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 11, pp. 3698–3710, 2020.
[18] H. M. Song, J. Woo and H. K. Kim, "In-vehicle network intrusion detection using deep convolutional neural network," *Vehicular Communications*, vol. 21, no. 100198, pp. 1–13, 2020.
[19] M. J. Kang and J. W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," *PLoS One*, vol. 11, no. 6, pp. e0155781, 2016.
[20] A. Wasicek, M. D. Pese, A. Weimerskirch, Y. Burakova and K. Singh, "Context-aware intrusion detection in automotive control systems," in *Proc. of 5th ESCAR USA Conf.*, Ypsilanti, MI, USA, pp. 21–22, 2017.
[21] E. Seo, H. M. Song and H. K. Kim, "GIDS: GAN based intrusion detection system for in-vehicle network," in *Proc. of Annual Conf. on Privacy, Security and Trust (PST)*, Belfast, Ireland, pp. 1–6, 2018.
[22] R. Panigrahi and S. Borah, "A detailed analysis of CICIDS2017 dataset for designing intrusion detection systems," *International Journal of Engineering & Technology*, vol. 7, no. 3.24, pp. 479–482, 2018.
[23] S. Seng, J. Garcia-Alfaro and Y. Laarouchi, "Why anomaly-based intrusion detection systems have not yet conquered the industrial market?" in *Proc. of Foundations and Practice of Security Symp. (FPS)*, Paris, France, pp. 341–354, 2021.
[24] H. Lee, S. H. Jeong and H. K. Kim, "OTIDS: A novel intrusion detection system for in-vehicle network by using remote frame," in *Proc. of Annual Conf. on Privacy, Security and Trust (PST)*, Calgary, AB, Canada, pp. 57–5709, 2017.
[25] M. L. Han, B. I. Kwak and H. K. Kim, "Anomaly intrusion detection method for vehicular networks based on survival analysis," *Vehicular Communications*, vol. 14, no. 2018, pp. 52–63, 2018.
[26] G. Dupont, A. Lekidis, J. D. Hartog and S. Etalle, *Automotive Controller Area Network (CAN) Bus Intrusion Dataset v2*, 4TU. Centre for Research Data, 2019. [Online]. Available: https://doi.org/10.4121/uuid:b74b4928-c377-4585-9432-2004dfa20a5d
[27] M. Hanselmann, T. Strauss, K. Dormann and H. Ulmer, "CANet: An unsupervised intrusion detection system for high dimensional CAN bus data," *IEEE Access*, vol. 8, pp. 58194–58205, 2020.
[28] H. K. Kim, *Car Hacking: Attack & Defense Challenge 2020 Dataset*, IEEE Dataport, 2021. [Online]. Available: https://doi.org/10.21227/qvr7-n418
[29] M. E. Verma, M. D. Iannacone, R. A. Bridges, S. C. Hollifield, P. Moriano *et al.,* "Addressing the lack of comparability & testing in CAN intrusion detection research: A comprehensive guide to CAN IDS data & introduction of the ROAD dataset," arXiv preprint arXiv: 2012.14600, 2020.
[30] S. Lee, H. J. Jo, A. Cho, D. H. Lee and W. Choi, "TTIDS: Transmission-resuming time-based intrusion detection system for controller area network (CAN)," *IEEE Access*, vol. 10, pp. 52139–52153, 2022.
[31] D. Swessi and H. Idoudi, "A comparative review of security threats datasets for vehicular networks," in *Proc. of Int. Conf. on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, Zallaq, Bahrain, pp. 746–751, 2021.
[32] A. Vahidi, T. Rosenstatter and N. I. Mowla, "Systematic evaluation of automotive intrusion detection datasets," in *Proc. of ACM Computer Science in Cars Symp. (CSCS)*, New York, NY, USA, pp. 1–12, 2022.
[33] O. Minawi, J. Whelan, A. Almehmadi and K. El-Khatib, "Machine learning-based intrusion detection system for controller area networks," in *Proc. of ACM Symp. on Design and Analysis of Intelligent Vehicular Networks and Applications*, Alicante, Spain, pp. 41–47, 2020.
[34] A. de Faveri Tron, S. Longari, M. Carminati, M. Polino and S. Zanero, "CANflict: Exploiting peripheral conflicts for data-link layer attacks on automotive networks," in *Proc. of ACM SIGSAC Conf. on Computer and Communications Security (CCS)*, Los Angeles, USA, pp. 711–723, 2022.
[35] C. Miller and C. Valasek, "Adventures in automotive networks and control units," *DEFCON*, vol. 21, pp. 15–31, 2013. [Online]. Available: http://iotsecuritylab.com/wp-content/uploads/2014/08/Adventures-in-Automotive-Networks-and-Control-Units.pdf

[36] T. Hoppe, S. Kiltz, A. Lang and J. Dittmann, "Exemplary automotive attack scenarios: Trojan horses for electronic throttle control system (ETC) and replay attacks on the power window system," *VDI-Berichte (VIB)*, vol. 23, pp. 165–183, 2016.
[37] *ISO 14229-1:2020, Road Vehicles–Unified Diagnostic Services (UDS)—Part 1: Application Layer*, 2020. [Online]. Available: https://www.iso.org/standard/72439.html
[38] S. Woo, H. J. Jo and D. H. Lee, "A practical wireless attack on the connected car and security protocol for in-vehicle can," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 993–1006, 2014.
[39] K. T. Cho and K. G. Shin, "Error handling of in-vehicle networks makes them vulnerable," in *Proc. of ACM SIGSAC Conf. on Computer and Communications Security (CCS)*, Vienna, Austria, pp. 1044–1055, 2016.
[40] K. Serag, R. Bhatia, V. Kumar, Z. B. Celik and D. Xu, "Exposing New vulnerabilities of error handling mechanism in CAN," in *Proc. of USENIX Security Symp.*, Virtual event, pp. 4241–4258, 2021.
[41] S. Kulandaivel, T. Goyal, A. K. Agrawal and V. Sekar, "CANvas: Fast and inexpensive automotive network mapping," in *Proc. of USENIX Security Symp.*, Santa Clara, CA, USA, pp. 389–405, 2019.
[42] A. Alfardus and D. B. Rawat, "Intrusion detection system for CAN bus in-vehicle network based on machine learning algorithms," in *Proc. of IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conf. (UEMCON)*, New York, USA, pp. 944–949, 2021.
[43] A. Taylor, N. Japkowicz and S. Leblanc, "Frequency-based anomaly detection for the automotive CAN bus," in *Proc. of World Congress on Industrial Control Systems Security (WCICSS)*, London, UK, pp. 45–49, 2015.
[44] M. H. Shahriar, W. Lou and Y. T. Hou, "CANtropy: Time series feature extraction-based intrusion detection systems for controller area networks," in *Proc. of Symp. on Vehicles Security and Privacy (VehicleSec)*, San Diego, CA, USA, pp. 1–8, 2023.
[45] B. Lampe and W. Meng, "IDS for CAN: A practical intrusion detection system for CAN bus security," in *Proc. of IEEE Global Communications Conf. (CLOBECOM)*, Rio de Janeiro, Brazil, pp. 1782–1787, 2022.
[46] N. D. Lawrence, "Data readiness levels," arXiv preprint arXiv:1705.02245, 2017.
[47] S. Rajapaksha, H. Kalutarage, M. O. Al-Kadri, G. Madzudzo and A. V. Petrovski, "Keep the moving vehicle secure: Context-aware intrusion detection system for in-vehicle CAN bus security," in *Proc. of Int. Conf. on Cyber Conflict: Keep Moving! (CyCon)*, Tallinn, Estonia, pp. 309–330, 2022.
[48] J. Bernard, M. Hutter, M. Zeppelzauer, D. Fellner and M. Sedlmair, "Comparing visual-interactive labeling with active learning: An experimental study," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 298–308, 2018.
[49] I. Sharafaldin, A. Gharib, A. H. Lashkari and A. A. Ghorbani, "Towards a reliable intrusion detection benchmark dataset," *Software Networking*, vol. 2017, no. 1, pp. 177–200, 2018.
[50] V. C. Nitesh, J. Nathalie and K. Aleksander, "Editorial: Special issue on learning from imbalanced data sets," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 1–6, 2004.
[51] Y. Sun, H. Ochiai and H. Esaki, "Intrusion detection with segmented federated learning for large-scale multiple lans," in *Proc. of IEEE Int. Joint Conf. on Neural Networks (IJCNN)*, Glasgow, UK, pp. 1–8, 2020.
[52] A. Fernandez, S. Garcia, F. Herrera and N. V. Chawla, "SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary," *Journal of Artificial Intelligence Research*, vol. 61, pp. 863–905, 2018.
[53] C. E. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.
[54] W. Choi, S. Lee, K. Joo, H. J. Jo and D. H. Lee, "An enhanced method for reverse engineering CAN data payload," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3371–3381, 2021.
[55] J. Ashraf, A. D. Bakhshi, N. Moustafa, H. Khurshid, A. Javed *et al.,* "Novel deep learning-enabled lstm autoencoder architecture for discovering anomalous events from intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4507–4518, 2020.

[56] M. D. Hossain, H. Inoue, H. Ochiai, D. Fall and Y. Kadobayashi, "LSTM-based intrusion detection system for in-vehicle can bus communications," *IEEE Access*, vol. 8, pp. 185489–185502, 2020.
[57] L. Yang, A. Moubayed and A. Shami, "MTH-IDS: A multitiered hybrid intrusion detection system for internet of vehicles," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 616–632, 2021.