西南交通大学学报

第 56 卷第 5 期
2021 年 10 月

JOURNAL OF SOUTHWEST JIAOTONG UNIVERSITY

Vol. 56 No. 5
Oct. 2021

ISSN: 0258-2724 DOI：10.35741/issn.0258-2724.56.5.14

Research article

Computer and Information Science

ANOMALY-BASED INTRUSION DETECTION FOR A VEHICLE CAN BUS: A CASE FOR HYUNDAI AVANTE CN7

基于异常的汽车能够总线入侵检测：以现代先锋中国网 7 为例

Kennedy Okokpujie [a], Grace Chinyere Kennedy [b], Vingi Patrick Nzanzu [a, c], Mbasa Joaquim Molo [a, c], Emmanuel Adetiba [a, d], Joke Badejo [a, c]

[a] Department of Electrical and Information Engineering, Covenant University
Ota, Ogun-State, Nigeria, kennedy.okokpujie@covenantuniversity.edu.ng

[b] Department of Computer and Science Engineering, Kyungdong University
Sokcho, South Korea

[c] Covenant Applied Informatics and Communication African Center of Excellence, Covenant University
Ota, Ogun State, Nigeria

[d] HRA, Institute for Systems Science, Durban University of Technology
P.O. Box 1334, Durban, South Africa

Received: June 23, 2021 ▪ Review: July 21, 2021 ▪ Accepted: August 19, 2021 ▪ Published: October 30, 2021

Abstract

Flooding, spoofing, replay, and fuzzing are common in various types of attacks faced by enterprises and various network systems. In-vehicle network systems are not immune to attacks and threats. Intrusion detection systems using different algorithms are proposed to enhance the security of the in-vehicle network. We use a dataset provided and collected in "Car Hacking: Attack and Defense Challenge" during 2020. This dataset has been realized by the organizers of the challenge for security researchers. With the aid of this dataset, the work aimed to develop attack and detection techniques of Controller Area Network (CAN) using different algorithms such as support vector machine and Feedforward Neural Network. This research work also provides a comparison of the rendering of these algorithms. Based on experimental results, this work will help future researchers to benchmark their results for the given dataset. The results obtained in this work show that the model selection does not depend only on the model's accuracy that is explained by the accuracy paradox. Therefore, for the overall result accuracy of 62.65%, they show that the support vector machine presents the most satisfying output in terms of precision and recall. The Radial basis kernel gives 65% and 67% precision for fuzzing and flooding and the recall of 64% and 100% for replay and spoofing, respectively.

Keywords: In-Vehicle Network, Controller Area Network, Spoofing, Replay, Fuzzing

摘要 泛洪、欺騙、重放和模糊測試在企業和各種網絡系統面臨的各類攻擊中很常見。車載網絡系統不能免受攻擊和威脅。提出了使用不同算法的入侵檢測系統來增強車載網絡的安全性。我們使用了 2020 年在“汽車黑客：攻擊和防禦挑戰”中提供和收集的數據集。該數據集已由安全研究人員挑戰的組織者實現。借助該數據集，該工作旨在開發使用支持向量機和前饋神經網絡等不同算法的控制器局域網攻擊和檢測技術。這項研究工作還對這些算法的渲染進行了比較。基於實驗結果，這項工作將幫助未來的研究人員對給定數據集的結果進行基準測試。在這項工作中獲得的結果表明，模型選擇不僅僅取決於模型的準確性，這是由準確性悖論所解釋的。因此，對於 62.65% 的整體結果準確率，他們表明支持向量機在精度和召回率方面表現出最令人滿意的輸出。径向基礎內核為模糊測試和泛洪提供 65% 和 67% 的精度，對於重放和欺騙分別提供 64% 和 100% 的召回率。

关键词: 車載網絡、控制器局域網、欺騙、重放、模糊測試

I. INTRODUCTION

The backbone of all electrical features of today’s vehicles is in-vehicle networking technology. The automotive industry has developed specialized communication protocols or extended them to existing standards to meet the demanding requirements of the automotive industry in conjunction with technology providers and standards authorities. Many of them are standardized and maintained by standardization bodies such as ISO, IEEE, or SAE [1]. A major step towards integrating a number of computer systems called Electronics Control Unit (ECU) has recently been achieved in automotive systems. ECU is used to control and monitor the energy efficiency improvement of the vehicle subsystem and decrease noise and vibration. More recently, automotive networking services such as Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) require intra-vehicle communication and inter-vehicle communication by computer devices [2], [3]. In order to support communication, various communication protocols are developed. The Controller Area Network (CAN) protocol is such a simple communication protocol, which supports connecting ECU sensors and actuators, and the adoption of CAN allows new automotive applications to emerge [4], [5]. Important information is often provided through a CAN bus for automotive services, such as self-driving and advanced driver assistance systems (ADAS), such as diagnostic, informational, and control data. For the safety of a driver, the information must be secured. However, the increase in network capacity is accompanied by major safety concerns, with several security defects in the in-vehicle network (IVN) [6]. Examples include flooding the bus with messages intended to override legitimate messages or using arbitrary bus spoofed identifiers. Moreover, much more sophisticated and stealthy attacks, including replay attacks and fuzzing attacks that appear legitimate and difficult to distinguish from ordinary traffic that can be generated, had been carried out.

A. Flooding

Flooding attempts to send massive messages in order to consume the CAN bus bandwidth. Flood attacks on the network were long a standard component of an attacker's denying service toolbox. The basic concept of flooding includes i) either send much traffic to a certain server or service, in order to use all of its resources to respond to fabricated traffic in order not to process legitimate service requests; ii) or to generate so much network congestion that legal traffic cannot reach a specific server or service, sends massive amounts of traffic to a specific network segment. This type of attack is not specific since there could be any traffic sent to the network. For the first type of flooding attack, many tools are available to send thousands of SIP INVITE simultaneous messages to somebody's system to see how the bomb is handled. Several tools that make such an attack easier attempt to exceed the number of simultaneous sessions a server can handle, which may lead to server failure or server reboot or cease operations in some cases.

B. Spoofing

Controller Area Network messages are injected to control certain desired functions. Spoofing is the act of disguising a communication or identity in order to appear to be linked to a confident, authorized source. Spoofing attacks may take many forms, from common phishing attacks to caller ID spoofing attacks often used for fraud. Additional technical networking elements such as IP addresses, the

DNS server, and the Address Resolution Protocol (ARP) service could also be targeted as part of a spoofing attack by the attackers, if necessary. Spoofing attacks typically benefit from trustworthy relationships by impersonating a person or organization known to the victim. In certain cases—like whale phishing attacks with e-mail spoofing or website spoofing—these e-mails can even be customized to convince the victim that communications are legitimate. If the user does not know that internet communications can be a fake, they will likely be spoofed. Researchers have proved that spoofing attacks may exploit the CAN protocol's vulnerability, often used for IVN attacks [7].

C. Replay

A replay attack is to extract normal traffic at a specific time and replay (injects) it into the CAN bus. A data replay attack is characterized by an attacker who wants to disrupt the system’s continuous functioning [8]. The aim is to perform hijacking to replace sensors' measurements and replace them with former signals generated by the system. The attacker observes and records reading while a system is in a continuous state, then feeds the control system with the measurements recorded while its attack is performed. This is a good strategy for an attack when the system's dynamic is not well known, and the system stays in a constant state for a certain time [9].

D. Fuzzing

The system is injected with random messages to cause unexpected behavior. The random messages include CAN IDs and data. Moreover, invalid and unexpected data may be injected if the attacker does not have enough information about CAN IDs and data. This procedure is repeated over and over and over again by the attacker until a considerable vulnerability is met. The concept of fuzzing can also be seen as a brute-forcing attack [9], [10].

Cyber attacks on automobiles can have disastrous results, including the loss of human life. Vehicles must behave securely, predictably, and reliably in order to avoid those mentioned above. Hence, Critical hypotheses are made in this research work. They include i) it may be necessary to use intrusion and threat detection techniques such as anomaly-based to improve the security of the IVN; ii) the use of an Intrusion Detection Sensor (IDS) might gain much attention due to the efficiency and simplicity in detecting different threats and attacks.

Efficient detectors and defense mechanisms must be developed and assessed for various attacks in response to these challenges. This work focuses on flooding, spoofing, replay, and fuzzing attacks detection in the CAN. We provide two comprehensive attack and anomaly detection methods using a support vector machine for the first and Feedforward Neural Network (FNN) for the second. We also evaluate the performance and accuracy of these methods with the CAN dataset provided and collected in "Car Hacking: Attack and Defense Challenge" during 2020. This dataset is useful in informing security experts about situational awareness when they detect new abnormal events, incidents, or attacks. Our research shows highly promising results in detecting various types of serious attacks and as an effective defense mechanism integrated into a vehicle. There have been some works conducting various researches on detecting ECU and CAN target attacks; however, none of them has used the dataset in use here. This research will contribute to helping benchmark the work of future researchers to determine the suitable algorithm and model for detecting ECU and CAN target attacks. Figure 1 shows the working principle of the attacks/intrusion detection framework.

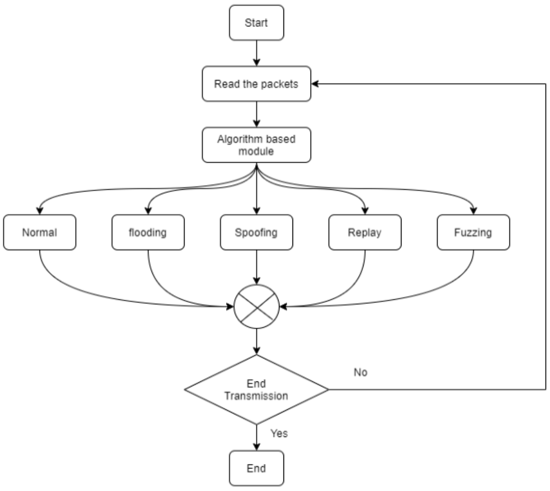

Figure 1. The functioning principle of the intrusion detection framework

The remainder of this work is organized as follows: the literature review is provided in Section 2, the work methodology, including details of the approach and algorithm of detection, is presented in Section 3. In Section 4, we present our experiments and results. We conclude the work in Section 5.

II. Literature Review

This section provides an overview of potential security attacks related to IVNs based on the various automotive features an attacker can target.

A. Sensor Attacks

[11] provided insight into autonomous cars' problems and safety considerations. This study offered greater sensor potential vulnerability for automobiles with more challenging embedded computing applications. The authors carried out a case study on automotive computing and sensing. Wireless access sensors, navigator sensors, and actuator controls as autonomous vehicle components were diagnosed. They also dealt with steganographic attacks, hardware and firmware attacks, eavesdropper ambushes on information leakage, and physical attacks based on chip-tampering.

[12] explored various attacks on car communication channels and analyzed the sensor functioning of linked automobiles that may execute Cooperative Adaptive Cruise Control (CACC). The simulation results in this study show that insider attacks can cause major problems in CACC vehicle networks. [13] said that sensor security in autonomous vehicles is a key concern and a significant safety risk in the same optic. They analyzed Tesla autonomous driving and three crucial sensors: ultrasonic sensors, millimeter-wave radar, and autopilot cameras. Radio interference and spoofing were achievable revealed the results of the study.

B. IVN Attacks

Many researchers have investigated attacks performed on internal networks such as CAN [14]–[17]. [17] conducted auto control systems sniffing and replay CAN bus network attacks. Vehicle safety risks on the CAN bus were studied. The authors submitted a classification on four scenarios based on their test setting. The target of the attack, style of attack, and related vulnerabilities was identified for each scenario.

[18] suggested a topology of the conceptual network for a CAN bus with the gateway. The offenders in their model referred to six sorts of attacks, including passive (like reading messages) and active attacks (particularly injection attacks, such as flooding, replay, and spoofing). The authors also proved that selective Denial of Service (DoS) attacks are conceivable for automobiles communicating using CAN. They further stressed that such an attack could not be detected at the level of the frame.

[19] suggested a CAN simulator for attacks. Control Area Network vulnerabilities have been demonstrated and proven using onboard diagnostic (OBD) scanners readily available after buying a car. Under the J3061 guidelines generally employed by the automobile industry, the authors implemented the design safety.

[20] conducted research on CAN communication threats of four sorts using the fuzzy neural network methodology (DoS, fuzzy, spoofing the drive gear and spoofing the RPM gauge). The authors suggested a mechanism for the detection of CAN protocol attacks. In this work, the real data in the CAN packet was confirmed as a feature vector. The authors also concluded that all fuzzy classification methods performed had precision in the range of 0.85 to 1 for detecting attacks targeting the CAN protocol identification.

To explore the susceptibility of CAN prototypes, [21] used fuzzy-based testing. They have carried out fuzzy black-box tests of ECUs in test cars to show the car system design vulnerability. Fuzzing is a standard software vulnerability detection method.

ECUs and CANs are still targets for attackers. They were first physically connected and targeted, but advanced tactics like side-channel and fuzzing have been deployed jointly to perform more sophisticated attacks.

III. Methodology and Algorithmic Approach

This section presents the methodology and the algorithmic approaches we used to achieve our work.

A. Methodology

First of all, the dataset used in this work is a free dataset for security researchers provided and collected during the Car Hacking: Attack and Defense Challenge in 2020. The target vehicle of competition was Hyundai Avante CN7. The dataset contains different classes that specify whether a message sent inside the car network is normal or an attack. The different classes are given as follow:

- Normal: represent the type the normal in the bus
- Flooding: is a bus attack that consumes bandwidth to keep the bus busy by sending an intensive data information
- Spoofing: represent messages injected in the bus to control a certain number of functionalities of the car.
- Replay: It is all about extracting the CAN bus's information and reinjecting it later into the CAN bus.
- Fuzzing represents random messages introduced in the CAN bus to provoke an unexpected car's behavior.

The dataset contains 3,672,151 rows with 3,372,743 rows for normal traffic in CAN bus and 229408 rows for attacks traffic injected in the CAN bus.

The different features considered in the dataset for data analysis include the Arbitration ID, DLC, Data, and Class.

- The arbitration ID: is the unique ID assigned to every transaction that is run in the CAN bus.
- DLC: contain the number of bits that are transmitted in the bus.
- Data: represents the transmitted payload.
- Class: gives the classification of data as normal or attack.

The Data analysis was carried out using python run on Google Colab.

In existing works, the advanced machine learning algorithms are barely used for a vehicular network because the computing power of the conventional ECU is limited to process the complex process. However, in current days, the computing power of ECU has been notably increasing to process enormous real-time tasks in the most recent vehicular system.

We considered two different approaches to perform the classification of data for the detection of attacks.

- Deep learning approach: in this approach, we used the FNN.
- Machine learning approach: in this approach, we used the Support Vector Machine.

Figure 2 represents the flow chart of the data analysis and prediction model for intrusion detection used in this work:

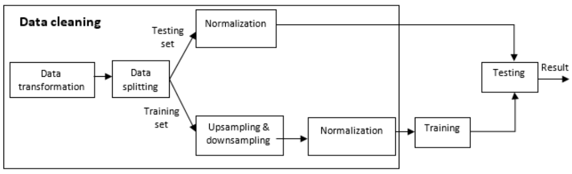

Figure 2. Data analysis and prediction flow chart

The Data cleaning is composed of some operations that take place:

- Data Transformation: this helps in converting the data format, values, and structure. The data transformation process helps have a set of data that is easy for both humans and computers to use or manage.
- Data splitting: helps to divide the dataset into training and testing sets.
- Normalization: aims to change the values of numeric data in the dataset to end up with a standard scale without altering the difference in range of those values.
- Up-weighting and Down-sampling: the data set at our disposal contains imbalanced data. Down-sampling and Up-weighting – the majority class – is an excellent approach to deal with unbalanced data. Down-sampling means training on a disproportionately low subset of the majority class example. In contrast, Up-weighting means adding an example weight to the down-sampled class equal to the factor somebody down-sampled.

After data cleaning, two operations are carried out in order to build the model:

- Training: This means adjusting the model's parameter going through the samples by minimizing the error function.
- Testing: To evaluate the trained model's efficiency or performance by providing a new set of samples.

B. Presentation of the Approaches Used

1) FNN

A FNN is an artificial neural network where connections between the nodes do not form a cycle. In its most basic form, a FNN is a single layer perceptron. Sequences of inputs enter the layer and are multiplied by the weights in this model. The weighted input values are then summed together to form a set. If the sum of the values is more than a predetermined threshold, which is normally set at zero, the output value is usually 1, and if the sum is less than the threshold, the output value is generally minus 1. The single-layer perceptron is a popular FNN model frequently used for classification. Single-layer perceptrons can also contain machine learning features. The neural network can correlate the outputs of its nodes with the desired values using a property known as the delta rule, permitting the network to alter its weights via training to create more accurate output values. This training and learning result in the gradient descent. The technique of updating weights in multi-layered perceptrons is virtually the same. However, the method is referred to as back-propagation. In such circumstances, the output values provided by the final layer are used to alter each hidden layer inside the network [22], [26], [27].

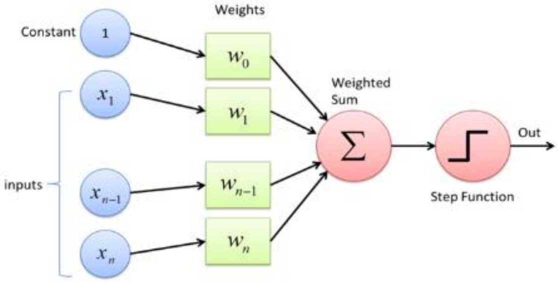

Figure 3. FNN algorithm

2) Support Vector Machine

Support Vector Machines (SVM) are not new but still helpful for classification because they do not overfit and perform well in many circumstances.

In n-dimensional space, the goal is to find a hyperplane that connects the data points to their possible classes. The hyperplane should be placed as close to the data points as possible. Support Vectors are the data points with the shortest distance to the hyperplane. Because of their proximity, they have a more significant impact on the exact location of the hyperplane than other data points [23], [25]. Figure 4 shows the support vector machine algorithm.

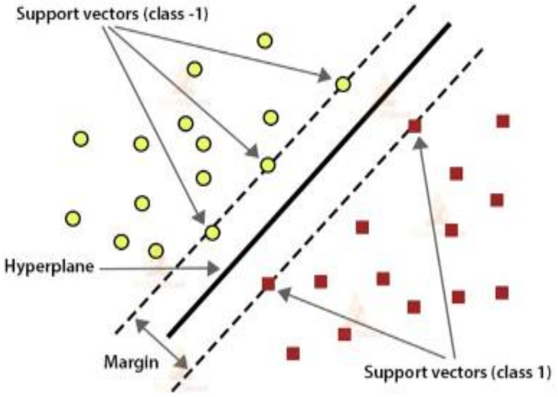

Figure 4. Support vector machine algorithm

Due to the kernel that turns the input data space into a higher-dimensional space SVMs are also known as kernelized SVMs. The most popular kernel functions available are Linear Function, Polynomial Function, Radial Basis Function (RBF), and Sigmoid Function.

IV. Result and Discussion

In this section, we present the results of several experiments carried out.

Table 1 presents the samples of the raw dataset before the data cleaning.

Table 1.
Raw dataset before the data cleaning

Timestamp, Arbitration ID, DLC, Data, Class
1597707827.052591,260,8,06 25 05 30 FF CF 71 55,Normal
1597707827.05398,329,8,4A C5 7E 8C 31 2D 01 10,Normal
1597707827.05467,38D,8,00 00 49 00 90 7F FE 01,Normal
1597707827.054904,420,8,50 1E 00 C8 FC 4F 6A 00,Normal
1597707827.05514,421,8,FE 07 00 FF E3 7F 00 52,Normal
1597707827.055378,153,8,20 80 10 FF 00 FF A0 4E,Normal
1597707827.055615,220,8,12 24 7C 00 00 6D 10 F6,Normal
1597707827.055853,340,8,00 00 00 24 36 01 25 30,Normal
1597707827.056095,389,8,00 00 00 20 00 00 C2 00,Normal
1597707827.056338,47F,8,00 EC FF FA 00 78 00 36,Normal
1597707827.0598,130,8,E4 80 AC 80 00 00 07 95,Normal
1597707827.060041,140,8,6C 80 00 6E 20 00 07 E8,Normal
1597707827.060268,251,8,59 04 DA 4F 00 45 B8 7F,Normal
1597707827.060471,2B0,6,67 00 00 07 CA 5B,Normal
1597707827.060639,164,4,00 08 1C FA,Normal
1597707827.060887,356,8,00 00 00 80 1F 00 00 00,Normal

The dataset represented in Table 1 contains non-suitable data for analysis. Therefore, the first step of data analysis, which consists of data transformation, is represented in Table 2.

Table 2.
Data transformation

	Arbitration ID	DLC	Data	Class
0	1136	8	1531506466388726889	Normal
1	304	8	324330091670671314	Normal
2	304	8	16464932089003771944	Normal
3	903	8	17595060653156141568	Normal
4	913	8	569	Normal
...	...	...	...	...
2937981	320	8	14446985127634404666	Normal
2937982	1151	8	30117796746100767	Normal
2937983	897	8	9275232701815638789	Normal
2937984	870	7	11399797360299265	Normal
2937985	916	8	63059195164041212	Normal
	[2937986 rows x 4 columns]			

To make process better the analysis, after data transformation, the data values were the same range scale. This process is data normalization. The normalized data is represented in Table 3.

Table 3.
Data normalization

	Arbitration ID	DLC	Data
0	0.263094	0.833333	2.139152e-04
1	0.544093	1.000000	6.445305e-07
2	0.259683	1.000000	8.156812e-10
3	0.000000	1.000000	0.000000e+00
4	1.000000	1.000000	3.096252e-01
...	...	...	...
616405	0.000000	1.000000	0.000000e+00
616406	0.267844	1.000000	1.714715e-01
616407	0.000000	1.000000	0.000000e+00
616408	0.263094	0.833333	1.741932e-04
616409	0.000000	1.000000	0.000000e+00

After training and testing of the different model, the following confusion matrix was obtained.

A. Confusion Matrix for FNN

Table 4 shows that the FNN misses classified attacks. The model only classified the flooding as an attack and considered others as false positives data.

Table 4.
Confusion matrix for FNN

	Normal	Fuzzing	Flooding	Replay	Spoofing
Normal	643394	30522	0	0	0
Fuzzing	18244	0	0	0	0
Flooding	0	0	30898	0	0
Replay	9034	489	0	0	0
Spoofing	1584	0	0	0	0

With an accuracy of 0.9190596394, the classification report is given in Table 5.

Table 5.
Classification report for FNN

	Precision	Recall	f1-score
Normal	0.957067	0.954709	0.955887
Fuzzing	0	0	0
Flooding	1	1	1
Replay	0	0	0
Spoofing	0	0	0

B. Confusion Matrix for SVM

Four different models were used for the classification of the data (Tables 6-12).

1) Linear Kernel

Table 6.
Confusion matrix for SVM linear kernel

	Normal	Fuzzing	Flooding	Replay	Spoofing
Normal	24944	0	0	0	0
Fuzzing	294	11456	35	2348	217
Flooding	18	6466	58	14952	3450
Replay	7	1586	19	4933	1042
Spoofing	0	0	0	0	1187

This model could classify the attacks. However, some of them were misclassified as false positives and false negatives in different classes.

With an accuracy of 0.5831644113296445, the classification report is the following:

Table 7.
Classification report for linear kernel

	Precision	Recall	f1-score
Normal	0.987373	1	0.993646
Fuzzing	0.587246	0.798328	0.676709
Flooding	0.517857	0.002325	0.00463
Replay	0.221877	0.650191	0.330852
Spoofing	0.201323	1	0.335169

2) Confusion Matrix for Polynomial Kernel

Table 8.
Confusion of polynomial kernel

	Normal	Fuzzing	Flooding	Replay	Spoofing
Normal	24944	0	0	0	0
Fuzzing	30	11728	811	1745	36
Flooding	8	4983	2067	13131	4755
Replay	5	1464	351	4301	1466
Spoofing	0	0	0	0	1187

The polynomial kernel was also able to the different attacks better than the linear kernel. Meanwhile, some data are false positives data and false negatives in different classes.

With an accuracy of 0.6057497397688051, the classification report is the following:

Table 9.
Classification report of polynomial kernel

	Precision	Recall	f1-score
Normal	0.998279	1	0.999139
Fuzzing	0.645282	0.817282	0.721168
Flooding	0.640136	0.082866	0.146736
Replay	0.224279	0.566891	0.321402
Spoofing	0.159457	1	0.275055

3) Confusion Matrix of Radial Basis Kernel

Table 10.
Confusion matrix of radial basis kernel

	Normal	Fuzzing	Flooding	Replay	Spoofing
Normal	24944	0	0	0	0
Fuzzing	117	11985	650	1465	133
Flooding	18	4991	2734	15130	2071
Replay	7	1419	639	4897	625
Spoofing	0	0	0	0	1187

The radial basis was able to categorize some attacks with false positives and false negatives in different classes.

With an accuracy of 0.6265682353585712, the classification report is the following:

Table 11.
Classification report of radial basis kernel

	Precision	Recall	f1-score
Normal	0.994339	1	0.997162
Fuzzing	0.651536	0.835192	0.73202
Flooding	0.679592	0.109606	0.188767
Replay	0.227852	0.645446	0.336807
Spoofing	0.295568	1	0.456275

4) Confusion Matrix of Sigmoid Kernel

Table 12.
Confusion matrix of sigmoid kernel

	Normal	Fuzzing	Flooding	Replay	Spoofing
Normal	24944	0	0	0	0
Fuzzing	0	5003	22	9290	35
Flooding	0	14238	18	9668	1020
Replay	0	4088	7	2960	532
Spoofing	0	583	0	604	0

The sigmoid kernel perfectly classified the normal data and classified attacks accordingly, but some were considered false negatives in different classes.

With an accuracy of 0.4509532679559524, the classification report is shown in Table 13:

Table 13.
Classification report of sigmoid kernel

	Precision	Recall	f1-score
Normal	1	1	1
Fuzzing	0.209225	0.348641	0.261513
Flooding	0.382979	0.000722	0.001441
Replay	0.131427	0.390141	0.196619
Spoofing	0	0	0

[24] stated the accuracy paradox in that the accuracy itself is not sufficient to evaluate the performance of a model, especially in unbalanced data. Therefore, precision, recall, and F1-score are mostly favored. Furthermore, the business importance matters since having a model that detects what is supposed to detect are key in selecting a model.

Based on the result previously obtained, the Radial basis kernel provides the most satisfying output in a sense that even though the accuracy is not the highest, the precision and recall metrics show that one can rely on the Radial basis kernel of SVM to perform the detection of attacks in the CAN Bus.

The previous intrusion detection methods may be effective only for specific threat models that have already been considered in the design stages. To cope with the problem, adopting machine learning-based IDS is recommended, mainly for conventional communication networks. The idea is to capture underlying statistical features of data and use them to detect any malicious attack. Intrusion detection methods using support vector machine and FNN are highlighted here for classifying attack types. These anomaly-based techniques intervene to correctly or mistakenly identify any deviation captured out of the range of the normal profile. It is important to have the complete normal profile, so the system does not suffer from high false positives. One of the key advantages of anomaly detection using a support vector machine and the FNN is identifying new and previously unknown attacks.

V. Conclusion

Intrusion detection is an interesting area of research that helps both researchers and businesses understand the security aspect of computer and network systems. However, there is no ideal way of assessing the performance of a model for a classification problem, but different metrics give valuable insights into how a classification model performs.

This work proposed a means to detect attacks in a car network automatically. Considering the business importance, the FNN model used in this work is not acceptable because it provides a reliable classification. Meanwhile, the SVM, with its different kernels, was able to give an attractive result. Furthermore, the Radial basis kernel presented the most relevant result even though its accuracy is not the highest. This is to say that the Radial basis kernel can be selected as the best model because it provides suitable output compared to other developed by looking at the precision and recall metrics. After all, the accuracy alone does not provide a good insight into the model's performance.

Since the proposed model has not yet been provided in the literature in “Car Hacking: Attack and Defense Challenge,” another model such as the Short-Term Long Memory can still be developed to benchmark the proposed model.

REFERENCES

[1] JABBAR, W.A., WEI, C.W., AZMI, N.A.A.M., and HAIRONNAZLI, N.A. (2021) An IoT Raspberry Pi-based parking management system for smart campus. *Internet of Things*, 14, 100387. DOI: 10.1016/j.iot.2021.100387.

[2] BISWAS, S., TATCHIKOU, R., and DION, F. (2006) Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety. *IEEE Communications Magazine*, 44 (1), pp. 74-82. DOI: 10.1109/MCOM.2006.1580935.

[3] PARK, T. J., HAN, C.S., and LEE, S.H. (2005) Development of the electronic control unit for the rack-actuating steer-by-wire using the hardware-in-the-loop simulation system. *Mechatronics*, 15 (8), pp. 899-918. DOI: 10.1016/j.mechatronics.2005.05.002.

[4] BARBOSA, M., RATCLIFF, K., and FARSI, M. (1999) An overview of Controller Area Network. *Computing and Control Engineering*, 10 (3), pp. 113-120. DOI: 10.1049/cce:19990304.

[5] Ge, I., AHMAD, X., HAN, Q. L., WANG, J., and ZHANG, X.M. (2021) Dynamic event-triggered scheduling and control for vehicle active suspension over controller area network. *Mechanical Systems and Signal Processing*, 152, 107481. DOI: 10.1016/j.ymssp.2020.107481.

[6] TARIQ, S., LEE, S., KIM, H.K., and WOO, S.S. (2020) CAN-ADF: The controller area network attack detection framework. *Computer Security*, 94, 101857. DOI: 10.1016/j.cose.2020.101857.

[7] IEHIRA, K., INOUE, H., and ISHIDA, K. (2018) Spoofing attack using bus-off attacks against a specific ECU of the CAN bus. In *15th IEEE Annual Consumer Communications and Networking Conference*, 2018-January, pp. 1-4. DOI: 10.1109/CCNC.2018.8319180.

[8] MERCO, R., BIRON, Z.A., and PISU, P. (2018) Replay Attack Detection in a Platoon of Connected Vehicles with Cooperative Adaptive Cruise Control. In *Proceedings of the American Control Conference*. 2018-June, pp. 5582-5587, DOI: 10.23919/ACC.2018.8431538.

[9] JEONG, S., JEON, B., CHUNG, B., and KIM, H.K. (2021) Convolutional neural network-based intrusion detection system for AVTP streams in automotive Ethernet-based networks. *Vehicular Communications*, 29, 100338. DOI: 10.1016/j.vehcom.2021.100338.

[10] LEE, H., CHOI, K., CHUNG, K., KIM, J., and YIM, K. (2015) Fuzzing CAN packets into automobiles. In *Proceedings of the International Conference on Advanced Information Networking and Applications*, 2015-April, pp. 817-821. DOI: 10.1109/AINA.2015.274.

[11] WYGLINSKI, A.M., HUANG, X., PADIR, T., LAI, L., EISENBARTH, T.R., and VENKATASUBRAMANIAN, K. (2013) Security of autonomous systems employing embedded computing and sensors. *IEEE Micro*, 33 (1), pp. 80-86. DOI: 10.1109/MM.2013.18.

[12] AMOOZADEH M., RAGHURAMU, A., CHUAH, C., GHOSAL, ZHANG, D.H.M., ROWE, J., and LEVITT K. (2015) Security vulnerabilities of connected vehicle streams and their impact on cooperative driving. *IEEE Communications Magazine*, 53 (6), pp. 126-132. DOI: 10.1109/MCOM.2015.7120028.

[13] LIU, J., YAN, C., and XU, W. (2016) *Can You Trust Autonomous Vehicles: Contactless Attacks against Sensors of Self-Driving Vehicles*. Las Vegas: DEF CON 24. DOI: 10.1145/1235.

[14] NILSSON, D.K., LARSON, U.E., PICASSO, F., and JONSSON, E. (2009) A first simulation of attacks in the automotive network communications protocol flexRay. In *Advances in Soft Computing*, 53, pp. 84–91. DOI: 10.1007/978-3-540-88181-0_11.

[15] CHANDRASEKARAN, S., RAMACHANDRAN, K.I., ADARSH, S., and PURANIK, A.K. (2020) Avoidance of

Replay attack in CAN protocol using Authenticated Encryption. In *11th International Conference on Computing, Communication and Networking Technologies*. Kharagpur, India. DOI: 10.1109/ICCCNT49239.2020.9225529.

[16] NOURELDEEN, P., AZER, M.A., REFAAT, A., and ALAM, M. (2018) Replay attack on lightweight CAN authentication protocol. In *12th International Conference on Computer Engineering and Systems*, 2018-January, pp. 600-606. DOI: 10.1109/ICCES.2017.8275376.

[17] HOPPE, T., KILTZ, S., and DITTMANN, J. (2008) Security threats to automotive CAN Networks – Practical examples and selected short-term countermeasures, In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5219, pp. 235-248. DOI: 10.1007/978-3-540-87698-4_21.

[18] LARSON, U.E., NILSSON, D.K., and JONSSON, E. (2008) An approach to specification-based attack detection for in-vehicle networks. In *IEEE Intelligent Vehicles Symposium, Proceedings*, pp. 220-225. DOI: 10.1109/IVS.2008.4621263.

[19] FOWLER, D.S., CHEAH, M., SHAIKH, S.A., and BRYANS, J. (2017) Towards a Testbed for Automotive Cybersecurity. In *Proceedings of the 10th IEEE International Conference on Software Testing, Verification and Validation*, pp. 540-541. DOI: 10.1109/ICST.2017.62.

[20] MARTINELLI, F., MERCALDO, F., NARDONE, V., and SANTONE, A. (2017) Car hacking identification through fuzzy logic algorithms. DOI: 10.1109/FUZZ-IEEE.2017.8015464.

[21] FOWLER, D.S., BRYANS, J., CHEAH, M., WOODERSON, P., and SHAIKH, S.A. (2019) A Method for Constructing Automotive Cybersecurity Tests, a CAN Fuzz Testing Example. in *Proceedings of the 19th IEEE International Conference on Software Quality, Reliability and Security*, pp. 1-8. DOI: 10.1109/QRS-C.2019.00015.

[22] RAZAVI S. and TOLSON, B.A. (2011) A New Formulation for Feedforward Neural Networks. *IEEE Transactions on Neural Networks*, 22 (10), pp. 1588-1598. DOI: 10.1109/TNN.2011.2163169.

[23] DERIS, A.M., ZAIN, A.M., and SALLEHUDDIN, R. (2011) Overview of Support Vector Machine in Modeling Machining Performances. *Procedia Engineering*, 24, pp. 308-312. DOI: 10.1016/j.proeng.2011.11.2647.

[24] UDDIN, M.F. (2019) Addressing Accuracy Paradox Using Enhanched Weighted Performance Metric in Machine Learning. In *2019 Sixth HCT Information Technology Trends*, pp. 319-324. DOI: 10.1109/ITT48889.2019.9075071.

[25] JOHN, S.N., ANELE, C., KENNEDY, O.O., OLAJIDE F., and KENNEDY, C.G. (2016) Real-time Fraud Detection in the Banking Sector Using Data Mining Techniques/Algorithm. *2016 International Conference on Computational Science and Computational Intelligence*, pp. 1186-1191. DOI: 10.1109/CSCI.2016.0224.

[26] OKESOLA, O.J., OKOKPUJIE, K.O., ADEWALE, A.A., JOHN, S.N., and OMORUYI, O. (2017) An Improved Bank Credit Scoring Model: A Naïve Bayesian Approach. *2017 International Conference on Computational Science and Computational Intelligence*, pp. 228-233. DOI: 10.1109/CSCI.2017.36.

[27] OKOKPUJIE, K. & APEH, S. (2020) Predictive Modeling of Trait-Aging Invariant Face Recognition System Using Machine Learning. In: KIM K., KIM H.Y. (eds) *Information Science and Applications. Lecture Notes in Electrical Engineering*, 621. Singapore: Springer. https://doi.org/10.1007/978-981-15-1465-4_43

参考文:

[1] JABBAR, W. A., WEI, C.W., AZMI, N.A.A.M. 和 HAIRONNAZLI, N.A. (2021) 基於物聯網樹莓派的智慧校園停車管理系統。物聯網, 14, 100387. DOI: 10.1016/j.iot.2021.100387。

[2] BISWAS, S.、TATCHIKOU, R. 和 DION, F. (2006) 用於提高公路交通安全的

車對車無線通信協議。電氣和電子工程師學會通信雜誌，44 (1)，第 74-82 頁。 DOI：10.1109/MCOM.2006.1580935。

[3] PARK, T. J., HAN, C.S. 和 LEE, S.H. (2005) 使用硬件在環仿真係統開髮用於齒條驅動線控轉向的電子控制單元。機電一體化，15 (8)，第 899-918 頁。 DOI：10.1016/j.mechatronics.2005.05.002。

[4] BARBOSA, M.、RATCLIFF, K. 和 FARSI, M. (1999) 控制器局域網概述。計算與控制工程，10 (3)，第 113-120 頁。DOI：10.1049/cce:19990304。

[5] Ge, I., AHMAD, X., HAN, Q. L., WANG, J., 和 ZHANG, X.M. (2021) 控制器局域網上車輛主動懸架的動態事件觸發調度和控制。機械系統和信號處理，152, 107481。DOI：10.1016/j.ymssp.2020.107481。

[6] TARIQ, S.、LEE, S.、KIM, H.K. 和 WOO, S.S. (2020) 控制器局域網：控制器局域網攻擊檢測框架。計算機安全，94，101857。DOI：10.1016/j.cose.2020.101857。

[7] IEHIRA, K.、INOUE, H. 和 ISHIDA, K. (2018) 使用總線關閉攻擊對能够總線的特定電子控制單元進行欺騙攻擊。在第 15 屆電氣和電子工程師學會年度消費者通信和網絡會議上，2018 年 1 月，第 1-4 頁。DOI：10.1109/CCNC.2018.8319180。

[8] MERCO, R.、BIRON, Z.A. 和 PISU, P. (2018) 在具有協同自適應巡航控制的聯網車輛中重放攻擊檢測。在美國控制會議論文集。2018 年 6 月，第 5582-5587 頁，DOI：10.23919/ACC.2018.8431538。

[9] JEONG, S.、JEON, B.、CHUNG, B. 和 KIM, H.K. (2021) 基於卷積神經網絡的汽車以太網網絡中飛行器技術計劃流的入侵檢測系統。車輛通信，29, 100338。DOI：10.1016/j.vehcom.2021.100338。

[10] LEE, H., CHOI, K., CHUNG, K., KIM, J. 和 YIM, K. (2015) 將 控制器局域網 數據包模糊到汽車中。高級信息網絡和應用國際會議論文集，2015 年 4 月，第 817-821 頁。 DOI：10.1109/AINA.2015.274。

[11] WYGLINSKI, A.M.、HUANG, X.、PADIR, T.、LAI, L.、EISENBARTH, T.R. 和 VENKATASUBRAMANIAN, K. (2013) 採用嵌入式計算和傳感器的自主系統的安全性。電氣和電子工程師學會微，33 (1)，第 80-86 頁。 DOI：10.1109/MM.2013.18。

[12] AMOOZADEH M.、RAGHURAMU, A.、CHUAH, C.、GHOSAL、ZHANG, D.H.M.、ROWE, J. 和 LEVITT K. (2015) 聯網車輛流的安全漏洞及其對協同駕駛的影響。電氣和電子工程師學會通信雜誌，53 (6)，第 126-132 頁。DOI：10.1109/MCOM.2015.7120028。

[13] LIU, J., YAN, C., 和 XU, W. (2016) 你能相信自動駕駛汽車嗎：對自動駕駛汽車傳感器的非接觸式攻擊。拉斯維加斯：防御准備狀態 24。DOI：10.1145/1235。

[14] NILSSON, D.K.、LARSON, U.E.、PICASSO, F. 和 JONSSON, E. (2009) 第一次模擬汽車網絡通信協議柔性射線中的攻擊。軟計算進展，53，第 84-91 頁。DOI：10.1007/978-3-540-88181-0_11。

[15] CHANDRASEKARAN, S.、RAMACHANDRAN, K.I.、ADARSH, S. 和 PURANIK, A.K. (2020) 使用身份驗證加密避免 控制器局域網 協議中的重放攻擊。在第 11 屆計算、通信和網絡技術國際會

議上。印度卡拉格布爾。DOI：10.1109/ICCCNT49239.2020.9225529。

[16] NOURELDEEN, P.、AZER, M.A.、REFAAT, A. 和 ALAM, M. (2018) 對輕量級 控制器局域網 身份驗證協議的重放攻擊。在第 12 屆計算機工程與系統國際會議上，2018 年 1 月，第 600-606 頁。 DOI：10.1109/ICCES.2017.8275376。

[17] HOPPE, T. 、 KILTZ, S. 和 DITTMANN, J. (2008) 汽車 控制器局域網 網絡的安全威脅 – 實踐示例和選定的短期對策，在計算機科學講義中（包括子系列講義在人工生物信息學中的智能和講義），5219，第 235-248 頁。DOI：10.1007/978-3-540-87698-4_21。

[18] LARSON, U.E.、NILSSON, D.K. 和 JONSSON, E. (2008) 一種基於規範的車載網絡攻擊檢測方法。在電氣和電子工程師學會智能汽車研討會，論文集，，第 220-225 頁。DOI：10.1109/IVS.2008.4621263。

[19] FOWLER, D.S. 、 CHEAH, M. 、 SHAIKH, S.A. 和 BRYANS, J. (2017) 邁向汽車網絡安全試驗台。在第 10 屆電氣和電子工程師學會軟件測試、驗證和驗證國際會議論文集，第 540-541 頁。DOI：10.1109/ICST.2017.62。

[20] MARTINELLI, F.、MERCALDO, F.、NARDONE, V. 和 SANTONE, A. (2017) 通過模糊邏輯算法進行汽車黑客識別。DOI：10.1109/FUZZ-電氣和電子工程師學會.2017.8015464。

[21] FOWLER, D.S. 、 RYANS, J.B 、 CHEAH, M. 、 WOODERSON, P. 和 SHAIKH, S.A. (2019) 一種構建汽車網絡安全測試的方法，一個 控制器局域網 模糊測試示例。在第 19 屆電氣和電子工程師學會軟件質量、可靠性和安全性國際會議論文集，第 1-8 頁。DOI：10.1109/QRS-C.2019.00015。

[22] RAZAVI S. 和 TOLSON, B.A. (2011) 前饋神經網絡的新公式。電氣和電子工程師學會神經網絡彙刊，22 (10)，第 1588-1598 頁。DOI：10.1109/TNN.2011.2163169。

[23] DERIS, A.M. 、 ZAIN, A.M. 和 SALLEHUDDIN, R. (2011) 支持向量機建模加工性能概述。繼續工程，24，第 308-312 頁。DOI：10.1016/j.proeng.2011.11.2647。

[24] UDDIN, M.F. (2019) 在機器學習中使用增強的加權性能指標解決準確性悖論。2019 年第六屆信息技術趨勢，第 319-324 頁。DOI：10.1109/ITT48889.2019.9075071。

[25] JOHN, S.N.、ANELE, C.、KENNEDY, O.O.、OLAJIDE F. 和 KENNEDY, C.G. (2016) 使用數據挖掘技術/算法的銀行業實時欺詐檢測。2016 年計算科學與計算智能國際會議，第 1186-1191 頁。DOI：10.1109/CSCI.2016.0224。

[26] OKESOLA, O.J.、OKOKPUJIE, K.O.、ADEWALE, A.A. 、 JOHN, S.N. 和 OMORUYI, O. (2017) 改進的銀行信用評分模型：樸素貝葉斯方法。2017 年計算科學與計算智能國際會議，第 228-233 頁。DOI：10.1109/CSCI.2017.36。

[27] OKOKPUJIE, K. 和 APEH S. (2020) 使用機器學習的特徵老化不變人臉識別系統的預測建模。在：KIM K.，KIM H.Y。(編輯) 信息科學與應用。電氣工程講義，

621。新加坡：施普林格。https://doi.org/10.1007/978-981-15-1465-4_43