IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Comparative Performance Evaluation of Intrusion Detection Based on Machine Learning in In-Vehicle Controller Area Network Bus

TAREK MOULAHI[1], SALAH ZIDI[2], ABDULATIF ALABDULATIF[3], AND MOHAMMED ATIQUZZAMAN.[4]

[1]Department of Information Technology, College of Computer, Qassim University, Buraydah, KSA, and FSTSB, Kairouan University, Tunisia. (e-mail: t.moulahi@qu.edu.sa)
[2]Department of Management Information System, College of Business and Economics, Qassim University, Buraydah, KSA, and Hatem Bettaher Laboratory (IRESCOMATH) in Gabes University, Tunisia (e-mail: s.zidi@qu.edu.sa)
[3]Department of Computer science, College of Computer, Qassim University, Buraydah, KSA (e-mail: ab.alabdulatif@qu.edu.sa)
[4]School of Computer Science, University of Oklahoma, Norman, OK 73019-6151, USA (atiq@ou.edu)

ABSTRACT

Communication between the nodes in a vehicle is performed using many protocols. The most common of these is known as the Controller Area Network (CAN). The functionality of the CAN protocol is based on sending messages from one node to all others throughout a bus. Messages are sent without either source or destination addresses. Consequently, it is simple for an attacker to inject malicious messages. This may lead to some nodes malfunctioning or to total system failure, which can affect the safety of the driver as well as the vehicle. Detecting intrusions is a challenging problem when using a CAN bus protocol for in-vehicle communication. Most existing work focuses on the physical aspects without taking into consideration the data itself. Machine Learning (ML) tools, especially classification techniques, have been widely used to address similar problems. In this paper, we use and compare several ML techniques to deal with the problem of detecting intrusions in in-vehicle communication. An experimental study is performed using a real dataset extracted from a KIA Soul car. Compared to previous work, which focuses on detecting intrusions based on the physical aspect, in this paper, data analysis and statistical learning techniques are applied. Furthermore, the paper provides a comparative study of the most common ML techniques. The results show that the techniques proposed in this paper outperform other techniques that have been used previously.

INDEX TERMS

CAN Bus; Data Classification; Intrusion Detection System; In-Vehicle Communication; Machine Learning.

I. INTRODUCTION

Recently, a considerable amount of research has focused on vehicle communication technology for smart vehicles, Vehicular Ad hoc Networks (VANET) [1], [2], and Intelligent Transportation Systems (ITS). Vehicles are necessary in many daily activities, and they are becoming more electronically equipped and are on longer simple mechanical machines. Electronic Control Units (ECUs) are used in vehicles to monitor and control various components. ECUs are connected through buses managed by several protocols [3] [4]. A vehicle bus is an intravehicular communication network that does not have a host computer. A bus is used to link a set of ECUs to simplify the task of exchanging messages as well as diagnostics. Intravehicular networks have many advantages [5], including (1) reducing the cable budget, which is the third most costly system after the engine and the chassis; (2) minimizing the packaging space by using fewer connections for more electrical and electronic features, thereby reducing vehicle size; (3) meeting higher bandwidth demands that can manage the large number of ECUs, with some vehicles containing up to 70 ECUs with 2500 internal signals [5]; and (4) making communication more reliable because bus-based communication is more robust than the traditional point-to-point communication in older vehicles.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

Currently, the most widely used protocols for in-vehicle communication are [3]:

- Local Interconnection Networks (LIN),
- Controller Area Networks (CAN),
- FlexRay,
- Ethernet,
- Media-Oriented Systems Transport.

All these protocols are based on bus communication, with each one having its own advantages and weakness. Among these protocols, we have chosen the CAN bus protocol, developed by Bosch in 1985 [6]. This protocol is used in the majority of vehicles today. Approximately 500 million CAN chips are used in vehicles [5]. In addition, a recent study predicted that the CAN bus will maintain its popularity for the next decade [5]. The CAN bus is the leading technology because it is less costly than other protocols, the maximum bit rate for high-speed CAN is $1Mbit/s$ by specification, and its acceptable fault tolerance behavior is better than the other intra-vehicle communication protocols mentioned earlier.

Despite its advantages, CAN bus suffers from many vulnerabilities. The main problem is that a CAN lacks any kind of security mechanism because this was not considered in its design [7]. Attacks on a CAN bus can come from outside, particularly from the On-Board Diagnostics (OBD) [8], or from other wireless interfaces, such as cellular links, Wi-Fi, and Bluetooth [5], [9]. Figure. 1 illustrates a combination of attack types, attack surfaces, and vulnerable assets.

The first type of attack includes frame falsifying, sniffing, and relay attacks, which can be addressed by encryption and improving authentication. The second type includes impersonation, Denial of Service (DoS), and fuzzy attacks, which must be handled by developing an Intrusion Detection System (IDS) to distinguish between normal behavior and an attack.

Most of the previous research dealing with security problems in the CAN protocol have concentrated on physical aspects by, for example, limiting physical access or using cryptography to protect CAN transmission [10]. However, there is still a need to achieve better IDS. Indeed, limited physical access will affect the effectiveness of transmission in CAN bus. Cryptography is not always suitable with such a lightweight system. This will be discussed in detail in the section on related work.

Over the last decade, Artificial Intelligence (AI) tools have produced interesting and effective results when solving complex problems that resemble ours, such as automatic system diagnostics and identification [11], fault detection in wireless sensor networks [12], [13], [14], [15], [16], and certain security problems in other fields. Thus, ML techniques, as the most interesting approach in the field of AI, can be very effective for the detection of intrusions. Three ML models can be used for prediction purposes: (1) the regression model, (2) the classification model, and (3) the clustering model. For real-time or predictive intrusion detection, the classification-based or clustering-based models are applied, the former for a supervised problem and the latter for a non-supervised problem.

In this paper, a comparative study is conducted of intrusion detection systems based on different ML models. For that, Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), and MultiLayer Perceptron (MLP) have been used to improve other models applied recently to the same dataset. Unlike previous studies, we undertake the detection of three types of attack on the KIA Soul dataset as one of the comparison criteria; the attacks are: DoS, impersonation, and fuzzy attacks.

A. MOTIVATION

Security was not considered when bus-based CANs were designed in the 1980s [9]. However, most modern vehicles use bus-based CANs, which is a non-secure network that can be hacked by injecting faulty messages. Consequently, attacks can cause accidents that could result in injury or death. This makes the protection of a CAN bus-based network a high priority in order to ensure the safety of drivers and passengers. While previous research works have used ML models to deal with this challenging problem, they appear to be inadequate and can be improved by using other ML models. This motivates us to explore the capabilities of other advanced ML techniques, such as SVM, DT, RF, and MLP to overcome the current security issue concerning in-vehicle CAN buses.

B. PAPER GOALS

The main objectives of the paper are as follows:

- To develop an intrusion detection-based ML for an in-vehicle controller area network bus by applying various ML techniques in the context of in-vehicle CAN bus networks as an IDS.
- To conduct a comparative performance evaluation of applied ML for intrusion detection in an in-vehicle CAN bus using a set of classifiers on a real dataset that includes messages transmitted using a CAN bus extracted from a KIA Soul car [6].
- To detect both the intrusion and the attack type: DoS, impersonation or fuzzy attack.

To the best of our knowledge, this is the first time that RF, DT, SVM, and MLP have been applied to the KIA Soul dataset. The results of our experimental study show that RF outperforms not only SVM and DT, but also the other classifiers , including Hierarchical Temporal Memory (HTM), Recurrent Neural Networks (RNN), and Hidden Markov Models (HMM), previously used in the same context.

The rest of this paper is organized as follows: Section II examines the related work. In Section III, a review of the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

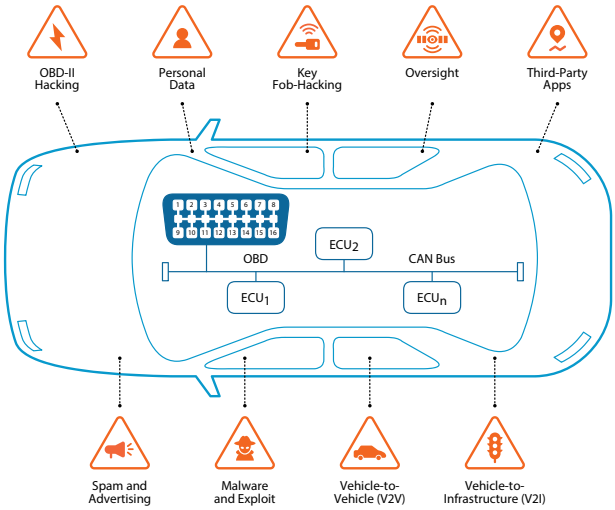

FIGURE 1: Modern cars are exposed to various types of attacks on the CAN bus from external devices connected to the car, particularly from OBD.

classifiers used for intrusion detection is given. The experimental study and a discussion of the results are presented in Section IV. Section V concludes the paper.

II. RELATED WORK

Protecting communication inside vehicles is very important since it affects the safety of vehicles as well as that of their drivers and passengers. Achieving this task by means of the CAN protocol is challenging due to the shortcomings of CANs, which are vulnerable to many types of attack, including DoS, impersonation, and fuzzy attacks. This makes the development of an IDS for this type of network an attractive problem for the research community. Indeed, much research has been undertaken to deal with this problem. In the following, we discuss the most relevant research investigating IDS for intra-vehicle communication.

In [6], the authors proposed using an analysis of the offset ratio and the time interval between the request and the response; i.e., working on a remote frame and data frame to create an IDS. Analysis of the response performance of ECUs helps to determine whether a behavior is an attack (i.e., intrusion detection) or a normal behavior. The authors considered three types of attacks: DoS, fuzzy, and impersonation attacks in CAN-based networks. Some results showed that this approach is very encouraging. However, a metric-like accuracy of attack detection is not given to determine whether or not the proposed approach achieved the best detection performance.

Groza and Murvay [8] proposed a bloom filtering-based IDS. A bloom filter is a probabilistic structure for testing whether an item belongs in a set. There are no false negatives with this filter, providing a 100% recall rate. The authors used this filtering method based on frame identifiers and part of the data fields to test frame periodicity, as it facilitates the detection of frame modification attacks or possible replays. Although the authors tested this approach on a CAN bus, it can also be used with other types of in-vehicle communication. The disadvantage of this approach is that the authors did not compare it with other methods. Furthermore, they included an important overload on ECU, which could affect their time response.

Tariq et al. [17] used RNNs and heuristics to detect attacks, employing the same dataset as [6] used in their study. The detection dealt with three types of attacks: DoS, replay, and fuzzy attacks. The authors used both neural networks and network traffic signatures. The accuracy of intrusion detection was high; however, these authors did not did not propose a technique for dealing with unseen attacks.

Neural networks are also used for intrusion detection in CANs in [18]. This study reported good results despite several weaknesses. For example, the detection of replay attacks was not adequate due to the high degree of similarity between genuine frames and injected frames, which makes the time stamp very useful in this case. Globally, the use of neural networks as IDS in CANs is promising and provides satisfactory results while still providing CAN bus communication safety.

A Deep Neural Network (DNN) was used in a novel technique for intrusion detection in CANs [19]. The authors used deep learning techniques to distinguish between normal behavior and attacks. The comparison between DNN-based IDS and standard neural networks shows that a DNN is better in terms of improving detection accuracy with a real-time response.

Wu et al. [20] proposed a novel intrusion detection method based on the information entropy method. This approach uses sliding windows with a fixed number of messages. The authors show that the optimization of the decision con-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

ditions and the enhancement of the sliding windows help to improve intrusion detection accuracy while decreasing the false positive rate. Furthermore, the effectiveness of the proposed method was demonstrated in an experimental study providing real-time responses to intrusion with important detection precision. Despite promising results, the authors did not consider the impact that the vehicle's operation state had on information entropy.

Wang et al. [21] used the benefits of hierarchical temporal memory (HTM) to define a distributed anomaly IDS in a CAN-based, in-vehicle network. The proposed technique predicts data flow depending on previous state learning in real time. Through an experimental study, the authors showed that HTM outperforms other detection models based on neural networks and HMMs in terms of detection accuracy.

A practical security architecture for a CAN-FD (which is designed to deal with the CAN bandwidth limitations)-based network is defined in [22]. The effectiveness of the proposed architecture was tested on three kinds of microcontrollers. This technique could be considered for use in vehicles manufactured in the future.

Despite the fact that a considerable amount of research has focused on developing an IDS in CAN-based networks, there is still a need to produce better systems. Most of the previous work has examined the behavior of exchanged frames or uses the data in the frames only superficially. In addition, traditional classification techniques have not been used. The aim of this paper is to mine the data within the exchanged frames deeply and take advantage of the benefits of different classifier methods to create a smart IDS for CANs that is able to detect attacks in real time in order to protect vehicles, drivers and passengers.

III. CLASSIFICATION MODELS FOR INTRUSION DETECTION SYSTEMS

We have applied three ML techniques for intrusion detection. Because intrusion detection is a supervised classification problem, we can use a known dataset containing labeled data. The four approaches tested to solve this problem are SVM, DT, RF, and MLP.

In this section, the problem statement is outlined. Next, the four classification techniques used and the evaluation criteria are defined. Finally, the experimental results are given.

A. PROBLEM STATEMENT

Many research studies have dealt with the problem of intrusion detection using experimental approaches and published datasets [6]. In this study, a set of classification techniques is used for intrusion detection in the same dataset. The dataset contains three types of attacks: DoS, fuzzy, and impersonation attacks. This dataset was created by injecting messages through the OBD-II port in real CAN traffic belonging to a KIA Soul car.

The data is prepared as shown in Table 1, describing the list of features.

The results of applying RF, SVM, and DT will be compared with those of the latest research studies [21] investigating the same dataset. Three types of attacks are treated:

- **DoS attack**
  This attack occurs when messages with high priority are injected into the CAN bus. The aim of this attack is to occupy the bus with packets carrying identifiers with high priority.
  This attack is done by the injection of packet 0x000 CAN ID in a short cycle inside the traffic.

- **Impersonation attack**
  This attack occurs when an attacker creates an impersonating node for answering remote frames. Thus, data frames will be broadcast periodically by the impersonating node to respond as a target node for remote frames. This attack is performed by inserting packets coming from impersonating node, with an arbitration ID = "0x164".

- **Fuzzy attack**
  This attack occurs when packets of randomly-spoofed identifiers with arbitrary data are injected by an adversary. Consequently, many functional packets will be received by all nodes, which may result in unintended vehicle responses. Hence, fuzzy attacks can completely prevent any bus communication or the transmission of certain frames by launching an attack on the CAN bus, as happens in a DoS attack. To conduct a fuzzy attack, packets are injected with spoofed random CAN ID and DATA values. Most known DoS attacks on CANs do not merely delay legitimate frame transmission, but completely prevent any bus communication or the transmission of certain frames.

B. SUPPORT VECTOR MACHINES (SVM) CLASSIFIER

SVM [23], [24], [25] is a statistical learning technique. It consists of a determination of decision boundaries. It is a supervised classification technique that uses a set of labeled examples and is based on the calculation of a learning model that can be generalized. As shown in Figure. 2, SVMs can efficiently perform non-linear as well as linear classification. For the non-linear model, this technique uses kernel functions.

C. DECISION TREES (DT) CLASSIFIER

A DT is a decision support tool based on the representation of the choices in the graphical form of a tree with the different classification decisions placed in sheets [26]. This technique uses a hierarchical representation of the data structure in the form of decision sequences (tests) for the result-prediction class. Each observation, which must be assigned to a class, is described by a set of variables that are tested in the tree nodes. Tests are performed in internal nodes, and decisions are made in leaf nodes.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

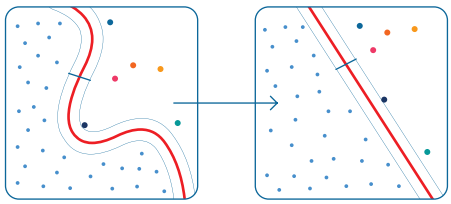

FIGURE 2: The SVM classification model is a supervised technique aiming to find a decision boundary based on a set of labeled samples by calculating a learning model that can be generalized.

To explain the principle of this tool, we consider the classification problem. Each element x of the database is represented by a multidimensional vector (x_1, x_2, ... x_n) corresponding to the set of descriptive variables of the point. Each internal node of the tree corresponds to a test performed on one of the variables xi. Once the tree has been built, the classification of a new candidate is done by going down the tree, from the root to one of the leaves (which encodes the decision or class). At each level of the descent, we pass an intermediate node where a variable xi is tested to decide which path (or subtree) to choose to continue the descent.

To build the tree, all the learning base points are placed in the root node. One of the variables describing the points is the class of the point (the "ground truth"); this variable is called the "target variable". The target variable can be categorical (classification problem) or a real value (regression problem). Each node is cut (split operation), giving rise to several descending nodes. An element of the learning base located in a node will be found in only one of its descendants.

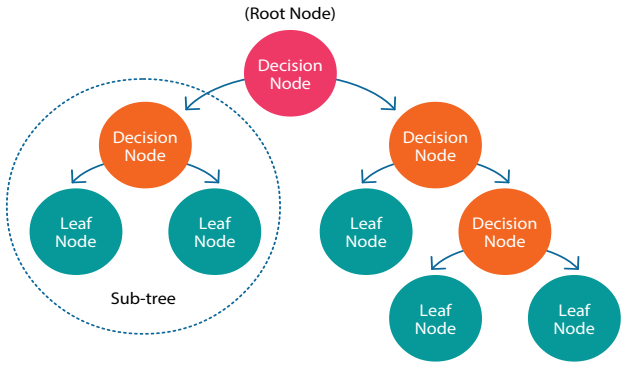

FIGURE 3: DT is a decision tool based on the representation of the choices in the graphical form of a tree with the different classification decisions, where all the learning base points are placed in the root node.

- The tree is built by the recursive partitioning (see Figure. 5) of each node according to the attribute value tested in each iteration (top-down induction). The optimized criterion is the homogeneity of the descendants compared to the target variable. The variable that is tested in a node will be the one that maximizes this homogeneity.
- The process stops when the elements of a node have the same value as the target variable (homogeneity).

D. RANDOM FOREST (RF) CLASSIFIER

Figure. 4 shows how RF is used for intrusion detection. RF involves the creation of multiple decision trees and determining the class of each DT [27]. The final class is defined using majority voting.

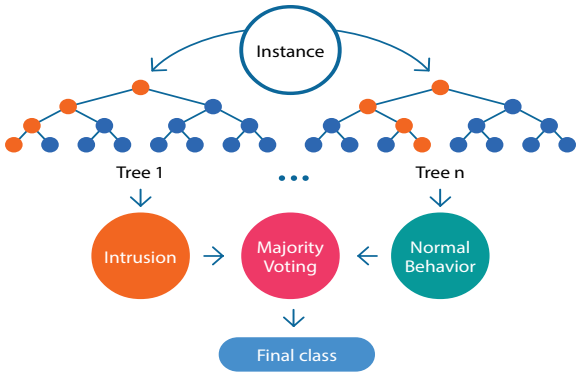

FIGURE 4: RF decision is a classification method based on the construction of multiple DTs in the training phase, after which the final decision is based on a majority voting system among the trees.

RF uses bootstrap aggregation applied to a learning tree. It operates on a training set, for example, $X = x_1, x_2, ..., x_n$, having $Y = y_1, y_2, ..., y_n$ as responses. RF is executed by looping B times. In each iteration, it chooses a sample with changes n training examples X_b, Y_b from X, Y. Next, RF trains a classification tree f_b on X_b, Y_b. Finally, after finishing the loop, a majority vote is applied to determine the right class.

If C_b is the class prediction of the b^{th} RD tree, the final class will be:

$$\overset{B}{rf} = majorityVoting\{\hat{C}_b\}_1^B \quad (1)$$

E. MULTILAYER PERCEPTRON

The MultiLayer Perceptron (MLP) is a neural network learning approach. It is a feedforward learning algorithm with several layers of nodes, including an input layer, an output layer, and some hidden layers. This supervised learning technique uses a nonlinear activation function in each neuron. By applying back propagation training, MLP is able to solve several multidimensional classification problems. It can distinguish

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

non-linearly separable data. Given its large number of layers, it can be considered as a type of deep learning technique.

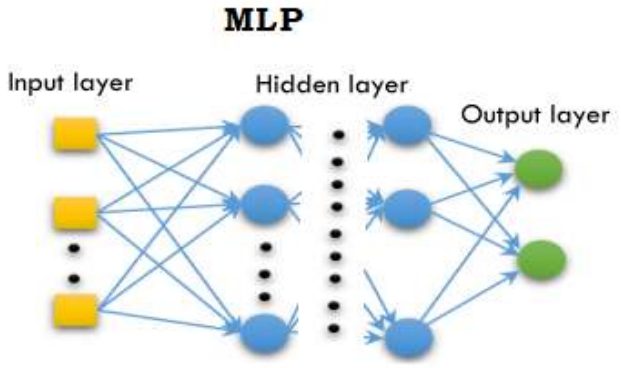

FIGURE 5: A simple illustration of MLP as a neural network learning approach

IV. PROPOSED MODEL AND EXPERIMENTAL STUDY

This section describes the evaluation criteria, which are followed by the results obtained by using ML as an IDS.

A. APPLIED MODEL

The overall architecture of the applied model is depicted in Figure. 6, including the details of the model workflow. The KIA Soul dataset CAN bus has been extracted from a shared repository. Then, the process of labelling is performed by executing prepossessing according to the dataset description given in [6]. Then, a set of ML tools is applied using Python. Finally, the results are presented according to attack types. Furthermore, an overall comparison is made with other ML models executed in other works with the same dataset.

B. EVALUATION CRITERIA

In this paragraph, we present the list of criteria that have been used to evaluate the RF results:

Precision, which is defined by the following equation (2):

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

Recall, which is defined by the following equation (3):

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

The $f1{-}score$ combines the precision and the recall given by the equation (4):

$$f1{-}score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

Finally, accuracy is the most significant parameter representing the success of a classification method, as follows (5):

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP} \tag{5}$$

Where:

- TP: True positive: True intrusion that is detected correctly,
- TN: True negative: True intrusion that is not detected,
- FP: False positive: Normal behavior that is considered an attack,
- FN: False negative: Normal behavior that is not considered an attack.

C. DATASET

We have used a dataset which includes DoS, fuzzy and impersonation attacks. This dataset was constructed by logging CAN traffic via the OBD-II port from a real vehicle while message injection attacks were launched. The in-vehicle data was extracted from KIA SOUL.

- **DoS Attack:** Injecting messages of '0x000' CAN ID in a short cycle.
- **Fuzzy Attack:** Injecting messages of spoofed random CAN ID and DATA values.
- **Impersonation Attack:** Injecting messages of Impersonating node, arbitration ID = '0x164'.

This dataset with 47519 examples contains 2201 DoS attacks, 313 fuzzy attacks, 824 impersonation attacks. all the 16 features are presented in the table1.

D. RESULTS AND DISCUSSION

Table I gives the feature list describing the prepared dataset [6], which includes three types of attacks: DoS, impersonation, and fuzzy attacks. A Python program was executed on a machine with 8GB RAM and an i7 processor. In the following, two comparisons are made. The first comparison is based on attack type, and the second is an overall comparison with well-known methods.

TABLE 1: Feature a list of the vectors in the KIA Soul dataset, which contains essential information about the frames transmitted in the CAN bus .

Feature	Significance and description
$time$	Time stamp
$time_{remote}$	Last remote frame time stamp
id	Frame id
$id1$	Previous frame id
$id2$	Id of previous of previous frame
$id3$	Id of previous of previous of previous frame
rtr	If the frame is a remote frame or not (1 or 0)
dlc	Size of data filed in the frame (0:8)
$d0$	First byte of data
$d1$	Second byte of data
$d2$	Third byte of data
$d3$	Fourth byte of data
$d4$	Fifth byte of data
$d5$	Sixth byte of data
$d6$	Seventh byte of data
$d7$	Eighth byte of data

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

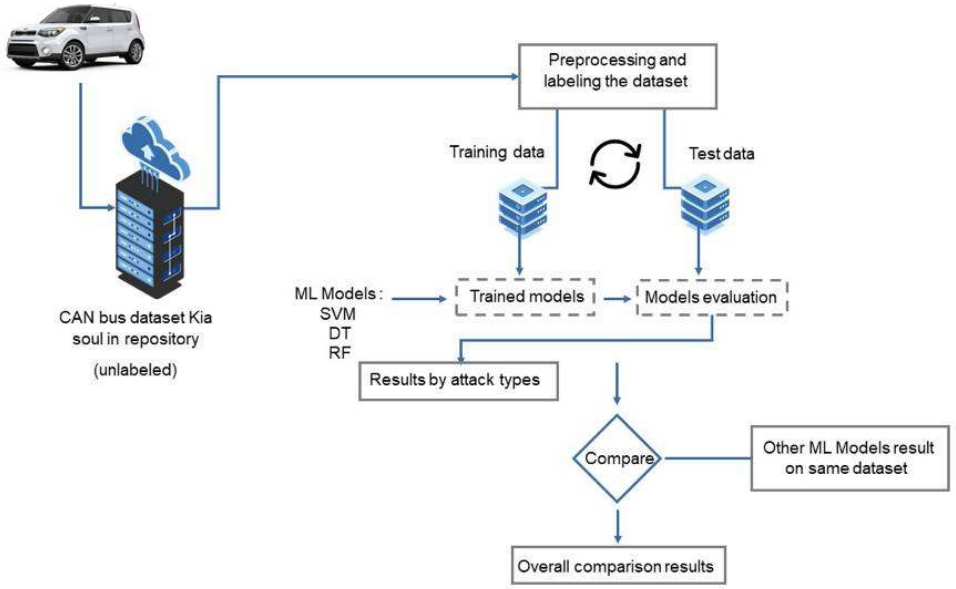

FIGURE 6: Overall architecture of the proposed model. The KIA Soul dataset is extracted from a shared repository, and then a prepossessing labelling is performed. Then, advanced ML algorithms, SVM, DT, and RF, are applied for intrusion detection, and the various ML models are compared.

1) Comparison based on attack type

As mentioned previously, we consider three type of attacks: DoS, impersonation, and fuzzy attacks. In Table II, a comparison based on attack type is given.

Figures. 7, 9 and 8 show classifier results in terms of precision, recall, and f1-score for impersonation, DoS and fuzzy attacks, respectively. We found that the best result for the four classifiers is linked to detecting impersonation attacks. Meanwhile, the detection of fuzzy attacks is very low. SVM shows the worst performance with fuzzy attacks.

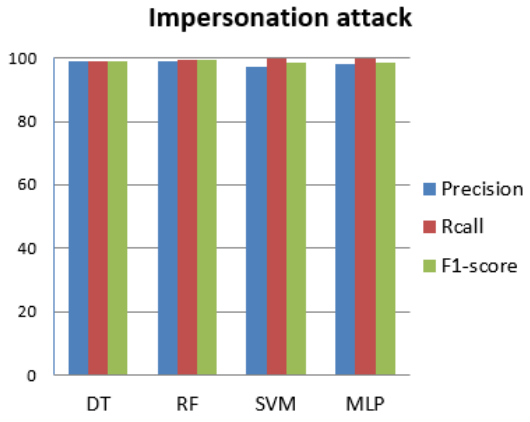

FIGURE 7: Impersonation attack detection results: DT, RF, MLP, and SVM show good performances. The good results can be explained by the high support for impersonation attacks in the dataset.

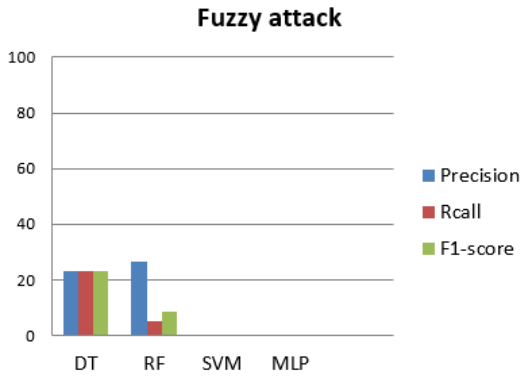

FIGURE 8: Fuzzy attack detection results: DT and RF show weak results, while SVM and MLP are not detecting this type of attack.

As we can see, the results are poor for fuzzy and DoS attacks. This can be explained by the insufficient number of examples of these attacks in the dataset.

As evident, RF outperforms DT, MLP and SVM with impersonation or fuzzy attacks. However, DT performs slightly better than RF and far from SVM and MLP. The worst performance is given by SVM and MLP with fuzzy attacks. The best performance is given with impersonation attacks due to the support included in the dataset 11046. Meanwhile, the worst performance of the three classifiers is with fuzzy attacks, which is explained by the low support.

DT performs better than the other methods when DoS attacks occur. SVM has the worst performance with fuzzy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

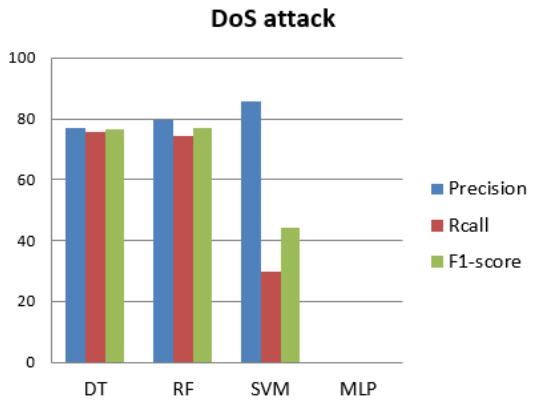

FIGURE 9: DoS attack detection results. SVM has the highest precision. In terms of combined precision, recall, and F1-score, DT and RF outperform SVM. MLP is not detecting DoS attacks.

attacks and worst performance compared to DT and RF. For fuzzy attack detection, SVM shows the worst results as it detects nothing. In addition, the detection of this attack by DT and RF is relatively weak. This fact can be explained by the low support of this attack in the dataset.

TABLE 2: Comparison based on attack type: RF outperforms DT in terms of fuzzy and impersonation attack.

	Attack	Precision	Recall	F1-score	Support
	Normal	1.000	1.000	1.000	550
DT	Fuzzy	0.233	0.230	0.232	78
	DoS	0.768	0.757	0.762	206
	Impers.	0.990	0.990	0.990	11046
	Accuracy : 0.981				11880
RF	Fuzzy	0.266	0.051	0.086	78
	DoS	0.796	0.742	0.768	206
	Impers.	0.988	0.995	0.992	11046
	Accuracy: 0.985				11880
SVM	Fuzzy	0.000	0.000	0.000	78
	DoS	0.859	0.296	0.440	206
	Impers.	0.972	0.998	0.985	11046
	Accuracy: 0.972				11880
MLP	Fuzzy	0.000	0.000	0.000	78
	DoS	0.000	0.000	0.000	206
	Impers.	0.979	1.000	0.987	11046
	Accuracy: 0.961				11880

2) Overall comparison

In this subsection, RF, DT, MLP, and SVM results will be compared to with those of obtained using three other techniques: HTM, RNN, and HMM. The results of these three methods are directly taken from [21], where they were obtained from the same dataset.

Table III shows the accuracy results for the RF, SVM, MLP and DT techniques. The table contains the values for accuracy, precision, recall, training time, and testing time for the four classifiers (SVM, RF, MLP, and DT) used to detect intrusion. Figure IV-D2 shows a comparison of the precision achieved by the best-known ML techniques (SVM, RF, DT, MLP, RNN, HTM, and HMM). It is clear that the precision of RF, SVM, MLP and DT is better than that of RNN and HMM, but slightly worse than HTM.

Additionally, for each attack, we used a specific database that contains only some examples of the aforementioned attacks in normal cases. The results confirm the explanations of the previous results. It is clear that the attacks that have fewer examples in the base are the least recognized by the learning techniques. Indeed, the learning rates have been improved since the total number of examples in each base (per attack) has decreased. So, with only two classes, the recognition improves.

TABLE 3: Overall comparison of RF, DT, SVM, and MLP performance results using Python and executed on an i7 PC with 8GB of RAM.

Classifier	RF	SVM	DT	MLP
Precision (%)	98.5269	97.2895	98.1902	95.2800
Recall (%)	98.1214	96.5583	98.1782	97.6100
Accuracy (%)	98.5269	97.2895	98.1902	97.6100
Training Time (s)	460.627	460.383	460.719	460.710
Testing Time (s)	14.933	14.919	14.935	14.925

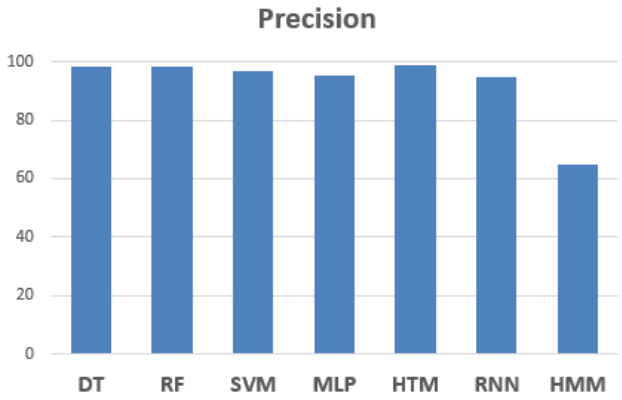

FIGURE 10: A comparison of the precision achieved by the techniques used (RF, DT, MLP, and SVM) and methods previously used with the same dataset (HTM, RNN, and HMM).

Figure. IV-D2 shows the recall factors for the seven methods. The RF, SVM, MLP, and DT classifiers outperform the other techniques (RNN, HMM, and HTM).

The most important comparison is that of accuracy. Figure IV-D2 shows that the four classifiers used in this study, RF, SVM, MLP, and DT, outperform other techniques. RF exceeds HTM by $1:3\%$, RNN by $12:2\%$, and almost doubles the performance of HMM. DT also outperforms other techniques by the same rate, while SVM exceeds HTM by 1.2%, RNN 12.1%, and also almost doubles HMM.

In the next part, we present the confusion matrix of all techniques in Figure. 13, 14, 15 and 16 . The different results of each attack show that the number of attacks can influence the learning results. It can even be a determinant above a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

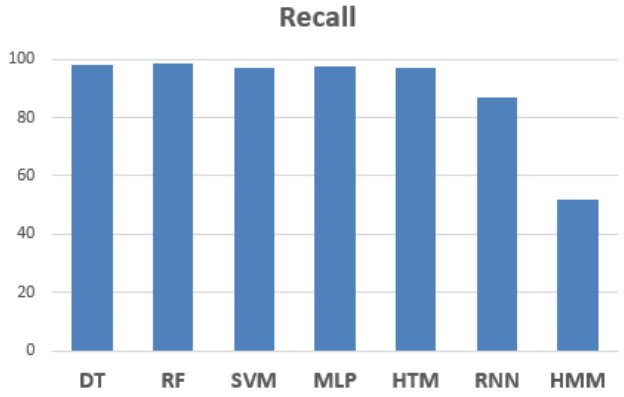

FIGURE 11: Recall comparison of the techniques used (RF, DT, MLP, and SVM) and the methods previously used on same dataset (HTM, RNN, and HMM).

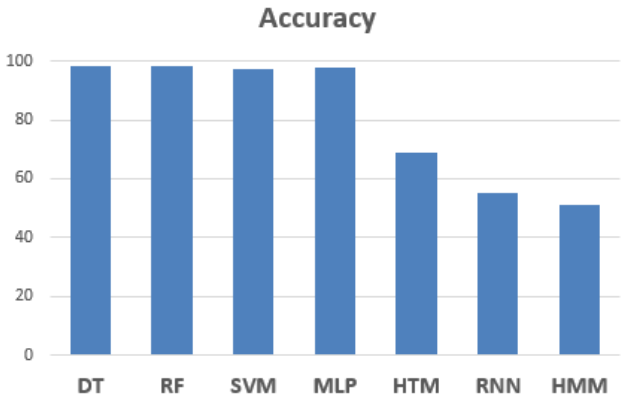

FIGURE 12: Accuracy comparison between the techniques used (RF, DT, MLP and SVM) and methods previously used on the same dataset (HTM, RNN, and HMM).

certain number. This is logical, as any learning model can be generalized only on the basis of a certain number of examples. This reminds us of the overfitting and underfitting problems.

Another type of comparison between the performance of different techniques can be made according to the percentage difference, as represented generally by equation 6:

$$PD = 100 \times \frac{|\Delta V|}{\frac{\sum V}{2}} \quad (6)$$

In our case general equation 7 can be used as follows:

$$PD\ (x,y) = 100 \times \frac{|x-y|}{\frac{x+y}{2}} \quad (7)$$

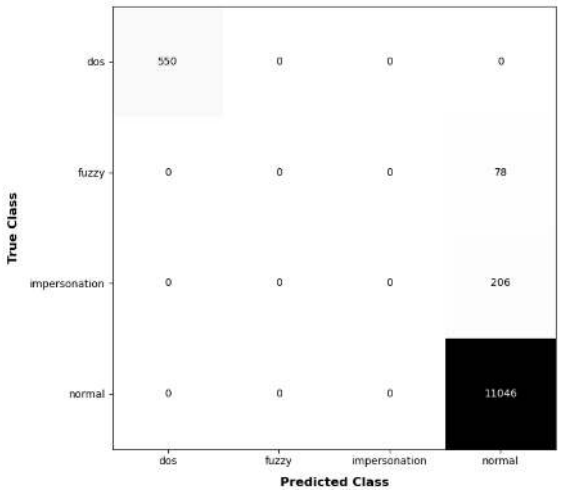

FIGURE 13: MLP confusion matrix

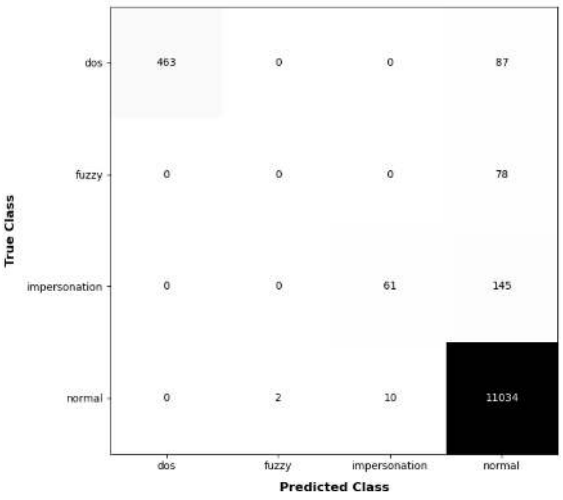

FIGURE 14: SVM confusion matrix

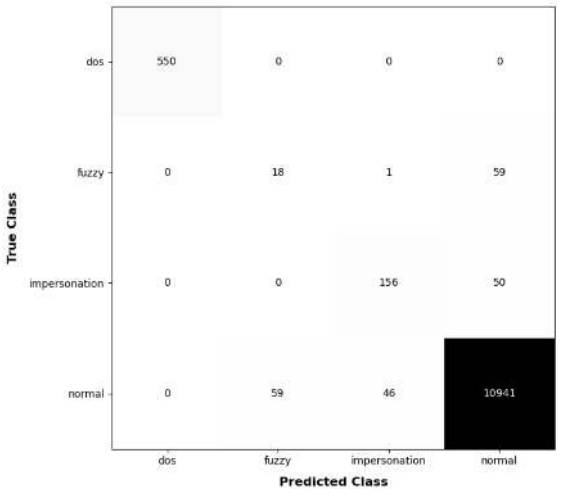

FIGURE 15: DT confusion matrix

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3095962, IEEE Access

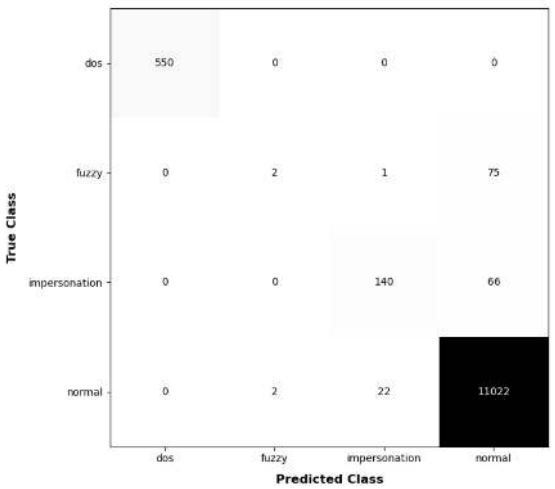

FIGURE 16: RF confusion matrix

TABLE 4: Percentage distance-based comparison between RF and HTM, RNN, HMM, SVM, DT and, MLP.

Percentage distance (%)	Precision	Recall	Accuracy
RF to HTM	0.891	35.250	1.561
RF to RNN	3.217	70.204	*10.869*
RF to HMM	41.401	89.673	50.359
RF to SVM	1.934	1.121	1.114
RF to DT	0.058	0.070	0.070
RF to MLP	2.937	0.522	0.522

Table 4 describes the percentage distance between RF and HTM, RNN, HMM, SVM, MLP, and DT. When the other methods outperform RF, the values are underlined. Bold values represent outperformance of RF by more than 35% compared to the other classifiers, while italic values indicate outperformance of RF by more than 10%. Ordinary values indicate outperformance of RF by less than 10%. In terms of accuracy, the RF classifier outperforms HTM slightly, and it also outperforms RNN by 10.68%. Notably, RF performs better than HMM by more than 50%. This table clearly shows that the RF classifier is more suitable for intrusion detection for CAN-based, in-vehicle networks.

SVM, DT, MLP, and RF achieve better results than RNN because statistical learning techniques are often more efficient when applied to multidimensional problems. In our intrusion detection problem, the input data dimension is 16. The most difficult phase for the statistical learning technique is parameterization, and optimal parameters are crucial to the success of this approach. We thoroughly explored the research space before closing the training phase. This yielded results comparable to the neural network techniques.

We noticed a few disadvantages of the SVM technique including the long training and testing time required. It takes almost 100 times longer than the others techniques (MLP, DT and RF) to train and to test.

Parameterization is also difficult for statistical learning techniques, especially for nonlinear learning. For example, it is difficult to find optimal parameters for the kernel function.

We also applied cross-validation. Figure 16 shows the accuracy rates of all the various executions (cv = 5) for each learning approach.

```
-----Random Forest------
......testing......
accuracy:  [0.15867003 0.79555976 0.91677189 0.69412879 0.39103441]
------------------------
-----Neural Network------
......testing......
accuracy:  [0.9760101  0.97611532 0.9760101  0.9760101  0.97611281]
------------------------
-----Decision Tree------
......testing......
accuracy:  [0.97590488 0.93297559 0.90088384 0.91245791 0.42628644]
------------------------
-----SVM------
......testing......
accuracy:  [0.95991162 0.9704335  0.96738215 0.97180135 0.9567505 ]
```

FIGURE 17: Cross validation results

V. CONCLUSION AND FUTURE WORK

This paper addresses an important problem: malicious intrusion in communications in vehicles using the CAN bus protocol. By examining the previous research in this area, we found that most of the previous studies have examined the behavior of exchanged frames or only superficially used the data contained in the frame without thoroughly considering the data itself. In addition, these studies do not use traditional classification techniques. For these reasons, in this study, we have proposed the use of the RF, SVM, MLP, and DT classifiers to distinguish between normal and malicious communications. The results of the experimental study performed with our dataset indicate that these four machine learning tools outperform the other techniques (HTM, RNN, HMM) in terms of accuracy.

In future work, we intend to apply non-supervised classification techniques to demonstrate the detection performance using several unknown or new intrusions. This will necessitate the application of deep learning techniques to large intrusion datasets.

REFERENCES

[1] R. Hajlaoui, H. Guyennet, and T. Moulahi, "A survey on heuristic-based routing methods in vehicular ad-hoc network: Technical challenges and future trends," IEEE Sensors Journal, vol. 16, no. 17, pp. 6782–6792, 2016.
[2] A. Mchergui, T. Moulahi, B. Alaya, and S. Nasri, "A survey and comparative study of QoS aware broadcasting techniques in VANET," Telecommunication Systems, vol. 66, no. 2, pp. 253–281, 2017.
[3] J. Liu, S. Zhang, W. Sun, and Y. Shi, "In-vehicle network attacks and countermeasures: Challenges and future directions," IEEE Network, vol. 31, no. 5, pp. 50–58, 2017.
[4] L. Ran, W. Junfeng, W. Haiying, and L. Gechen, "Design method of CAN bus network communication structure for electric vehicle," in International Forum on Strategic Technology 2010, pp. 326–329, 2010.

[5] W. Zeng, M. A. S. Khalid, and S. Chowdhury, "In-vehicle networks outlook: Achievements and challenges," IEEE Commun. Surv. Tutorials, vol. 18, no. 3, pp. 1552–1571, 2016.
[6] H. Lee, S. H. Jeong, and H. K. Kim, "Otids: A novel intrusion detection system for in-vehicle network by using remote frame," in 2017 15th Annual Conference on Privacy, Security and Trust (PST), pp. 57–5709, 2017.
[7] M. Bozdal, M. Samie, S. Aslam, and I. Jennions, "Evaluation of CAN bus security challenges," Sensors, vol. 20, no. 8, p. 2364, 2020.
[8] B. Groza and P. Murvay, "Efficient intrusion detection with bloom filtering in controller area networks," IEEE Trans. Information Forensics and Security, vol. 14, no. 4, pp. 1037–1051, 2019.
[9] M. Bozdal, M. Samie, and I. Jennions, "A survey on can bus protocol: Attacks, challenges, and potential solutions," in 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE), pp. 201–205, 2018.
[10] 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 Trans. Intell. Transp. Syst., vol. 16, no. 2, pp. 993–1006, 2015.
[11] L. Sellami, S. Zidi, and K. Abderrahim, "Self-adaptative multi-kernel algorithm for switched linear systems identification," IJMIC, vol. 31, no. 1, pp. 103–111, 2019.
[12] S. Mahfoudhi, M. Frehat, and T. Moulahi, "Enhancing cloud of things performance by avoiding unnecessary data through artificial intelligence tools," in 15th International Wireless Communications & Mobile Computing Conference, IWCMC 2019, Tangier, Morocco, June 24-28, 2019, pp. 1463–1467, 2019.
[13] M. Panda, B. S. Gouda, and T. Panigrahi, "Fault diagnosis in wireless sensor networks using a neural network constructed by deep learning technique," in Nature Inspired Computing for Wireless Sensor Networks, pp. 77–101, Springer, 2020.
[14] M. Emperuman and S. Chandrasekaran, "Hybrid continuous density hmm-based ensemble neural networks for sensor fault detection and classification in wireless sensor network," Sensors, vol. 20, no. 3, p. 745, 2020.
[15] M. Kordestani, M. F. Samadi, M. Saif, and K. Khorasani, "A new fault diagnosis of multifunctional spoiler system using integrated artificial neural network and discrete wavelet transform methods," IEEE Sensors Journal, vol. 18, no. 12, pp. 4990–5001, 2018.
[16] D. P. Kumar, T. Amgoth, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," Information Fusion, vol. 49, pp. 1–25, 2019.
[17] S. Tariq, S. Lee, H. K. Kim, and S. S. Woo, "Detecting in-vehicle CAN message attacks using heuristics and rnns," in Information and Operational Technology Security Systems - First International Workshop, IOSec 2018, CIPSEC Project, Heraklion, Crete, Greece, September 13, 2018, Revised Selected Papers (A. P. Fournaris, K. Lampropoulos, and E. Marín-Tordera, eds.), vol. 11398 of Lecture Notes in Computer Science, pp. 39–45, Springer, 2018.
[18] C. Jichici, B. Groza, and P. Murvay, "Examining the use of neural networks for intrusion detection in controller area networks," in Innovative Security Solutions for Information Technology and Communications - 11th International Conference, SecITC 2018, Bucharest, Romania, November 8-9, 2018, Revised Selected Papers (J. Lanet and C. Toma, eds.), vol. 11359 of Lecture Notes in Computer Science, pp. 109–125, Springer, 2018.
[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, 2016.
[20] W. Wu, Y. Huang, R. Kurachi, G. Zeng, G. Xie, R. Li, and K. Li, "Sliding window optimized information entropy analysis method for intrusion detection on in-vehicle networks," IEEE Access, vol. 6, pp. 45233–45245, 2018.
[21] C. Wang, Z. Zhao, L. Gong, L. Zhu, Z. Liu, and X. Cheng, "A distributed anomaly detection system for in-vehicle network using HTM," IEEE Access, vol. 6, pp. 9091–9098, 2018.
[22] S. Woo, H. J. Jo, I. Kim, and D. H. Lee, "A practical security architecture for in-vehicle CAN-FD," IEEE Trans. Intell. Transp. Syst., vol. 17, no. 8, pp. 2248–2261, 2016.
[23] S. Zidi, T. Moulahi, and B. Alaya, "Fault detection in wireless sensor networks through svm classifier," IEEE Sensors Journal, vol. 18, no. 1, pp. 340–347, 2017.
[24] T.-K. Dao, T.-T. Nguyen, J.-S. Pan, Y. Qiao, and Q.-A. Lai, "Identification failure data for cluster heads aggregation in wsn based on improving classification of svm," IEEE Access, vol. 8, pp. 61070–61084, 2020.
[25] F. Qu, Q. Jiang, G. Jin, Y. Wei, and Z. Zhang, "Mud pulse signal demodulation based on support vector machines and particle swarm optimization," Journal of Petroleum Science and Engineering, p. 107432, 2020.
[26] S. R. Safavian and D. A. Landgrebe, "A survey of decision tree classifier methodology," IEEE Trans. Syst. Man Cybern., vol. 21, no. 3, pp. 660–674, 1991.
[27] G. Biau and E. Scornet, "A random forest guided tour," Test, vol. 25, no. 2, pp. 197–227, 2016.

TAREK MOULAHI received his PhD degree in March 6th 2015, done simultaneously in the University of Franche-Comté (Besançon), France, and Sfax National School of Engineering, Tunisia. Currently, he is an Assistant Professor in both the Mathematics and Computer Science department, and the Faculty of Science and Technology of Sidi Bouzid (FSTSB), University of Kairouan, Tunisia, and in the department of Information Technology, College of Computer, Qassim University (KSA). His research interests include Wireless Sensor Networks, Vehicular Ad hoc Networks (VANET) and Internet of Things (IoT). He received the 2019 IEEE Sensors Council Sensors Journal Runner-Up Award for Best Paper.

SALAH ZIDI received his PhD in the University of Lille in France. His PhD was received in July 2007 and it was focused on regulation and reconfiguration of multimodal transportation systems. Currently, Dr. Zidi is an assistant professor in the MIS department in the College of Business and Economics, Qassim University (KSA) and Associate Professor, University of Gabes, Tunisia. He received his HDR degree in 2017 from the University of Lille1 (France). His research interests include: optimization, artificial intelligence, machine learning, feature extraction and data analysis for automation systems and complex systems. He received the 2019 IEEE Sensors Council Sensors Journal Runner-Up Award for Best Paper.

ABDULATIF ALABDULATIF is an assistant professor at the School of Computer Science IT, Qassim University, Saudi Arabia. He completed his Ph.D. degree in Computer Science from RMIT University, Australia in 2018. He received his B.Sc. degree in Computer Science from Qassim University, Saudi Arabia in 2008 and his M.Sc. degree in Computer Science from RMIT University, Australia in 2013. His research interests include applied cryptography, cloud computing, data mining and remote healthcare.

MOHAMMED ATIQUZZAMAN (Senior Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering and electronics from the University of Manchester, U.K., in 1984 and 1987, respectively. He currently holds the Edith J. Kinney Gaylord Presidential Professorship with the School of Computer Science, University of Oklahoma, USA. His research has been funded by the National Science Foundation, National Aeronautics and Space Administration, U.S. Air Force, Cisco, and Honeywell. He has coauthored Performance of TCP/IP Over ATM Networks and has authored over 300 refereed publications. His current research interests are in the areas of transport protocols, wireless and mobile networks, ad hoc networks, satellite networks, power-aware networking, and optical communications. He Co-Chaired the IEEE High Performance Switching and Routing Symposium, in 2003 and 2011, the IEEE GLOBECOM and ICC, in 2006, 2007, 2009, 2010, 2012, and 2014, the IEEE VTC, in 2013, and the SPIE Quality of Service Over Next Generation Data Networks conferences, from 2001 to 2003. He was the Panels Co-Chair of INFOCOM'05. He has been on the program committee of many conferences, such as INFOCOM, GLOBECOM, ICCCN, ICCIT, Local Computer Networks. He serves on the review panels at the National Science Foundation. He was the Chair of the IEEE Communication Society Technical Committee on Communications Switching and Routing. He received the IEEE Communication Society's Fred W. Ellersick Prize and the NASA Group Achievement Award for outstanding work to further NASA Glenn Research Center's efforts in the area of the Advanced Communications/Air Traffic Management's Fiber Optic Signal Distribution for Aeronautical Communications Project. He received the 2018 Satellite and Space Communications Technical Recognition Award for valuable contributions to the Satellite and Space Communications Scientific Community from the IEEE, and the 2017 Distinguished Technical Achievement Award from the IEEE Communications Society in recognition of outstanding technical contributions and services in the area of communications switching and routing. He is the Editor in Chief of the Journal of Networks and Computer Applications, the Founding Editor in Chief of Vehicular Communications, and serves served on the editorial boards of many journals, including the IEEE Communications Magazine, the IEEE Journal on Selected Areas in Communications, the IEEE Transactions on Mobile Computing, Real Time Imaging Journal, the Journal of Sensor Networks, and the International Journal of Communication Systems.