



## Article

# Detection and Identification of Malicious Cyber-Attacks in Connected and Automated Vehicles' Real-Time Sensors

Elvin Eziamma <sup>1,\*</sup> , Farooq Awin <sup>1,2</sup> , Sabbir Ahmed <sup>3</sup> , Luz Marina Santos-Jaimes <sup>4</sup> ,  
Akinyemi Pelumi <sup>1,5</sup>  and Danilo Corral-De-Witt <sup>1,6</sup> 

<sup>1</sup> Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON N9B 3P4, Canada; awin@uwindsor.ca (F.A.); akinyemp@uwindsor.ca or akinyemiakinpelumi@gmail.com (A.P.); corraldd@uwindsor.ca (D.C.-D.-W.)

<sup>2</sup> Electrical and Electronic Department, University of Tripoli, Tripoli 13555, Libya

<sup>3</sup> Former Visiting Researcher, Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON N9B 3P4, Canada; Sabbir1009@yahoo.com

<sup>4</sup> Departamento de Ingeniería de Sistemas, Universidad de Pamplona, Pamplona 31009, Colombia; lsantos@unipamplona.edu.co

<sup>5</sup> Ibom International Center for Research and Scholarship Incorporated, Windsor, ON N8R1A2, Canada

<sup>6</sup> Departamento de Eléctrica Electrónica, Universidad de las Fuerzas Armadas ESPE, Sangolquí 171103, Ecuador

\* Correspondence: eziamma@uwindsor.ca; Tel.: +1-416-294-2079

Received: 29 September 2020; Accepted: 30 October 2020; Published: 4 November 2020



**Abstract:** Connected and automated vehicles (CAVs) as a part of Intelligent Transportation Systems (ITS) are projected to revolutionise the transportation industry, primarily by allowing real-time and seamless information exchange of vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I). However, these connectivity and automation are expected to offer vast numbers of benefits, new challenges in terms of safety, security and privacy also emerge. CAVs continue to rely heavily on their sensor readings, the input obtained from other vehicles and the road side units to inspect roadways. Consequently, anomalous reading of sensors triggered by malicious cyber attacks may lead to fatal consequences. Hence, like all other safety-critical applications, in CAVs also, reliable and secure information dissemination is of utmost importance. As a result, real time detection of anomaly along with identifying the source is a pre-requisite for mass deployment of CAVs. Motivated by this safety concerns in CAVs, we develop an efficient anomaly detection method through the combination of Bayesian deep learning (BDL) with discrete wavelet transform (DWT) to improve the safety and security in CAVs. In particular, DWT is used to smooth sensor reading of a CAV and then feed the data to a BDL module for analysis of the detection and identification of anomalous sensor behavior/data points caused by either malicious cyber attacks or faulty vehicle sensors. Our numerical experiments show that the proposed method demonstrates significant improvement in detection anomalies in terms of accuracy, sensitivity, precision, and F1-score evaluation metrics. For these metrics, the proposed method shows an average performance gain of 7.95%, 9% 8.77% and 7.33%, respectively when compared with Convolutional Neural Network (CNN-1D), and when compared with BDL, the corresponding numbers are 5%, 7.9% 7.54% and 4.1% respectively.

**Keywords:** connected and automated vehicles; discrete wavelet transform; intelligent transportation system; Bayesian deep learning; convolution neural network

## 1. Introduction

Intelligent transportation (IT) is an emerging technology where a large number of vehicles can collect, process and communicate information to make collaborative decisions without direct human intervention [1], through Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) communications. Taking into account the enormous benefits expected from vehicular communications and the number of vehicles (millions worldwide), it is evident that vehicular communications will possibly become the most important mobile ad hoc network realization. Transportation has tremendous impacts on both the economic and human assets in this era. From all indications, statistics show that as of 2018, in US alone traffic accident amounts to a total of 36,473 deaths and traffic congestion resulted in the economic cost of about 115 billion US Dollars [2]. This impending situation calls for meaningful enhancement of transportation safety and efficiency. A promising approach to this end is to integrate transportation systems with information technology in an intelligent fashion, where V2Vs communication networks have been envisioned as an indispensable component.

Significant efforts have been made by researchers to investigate many aspects of vehicular communication [3]. The United States' Federal Communication Commission (FCC) has allocated 75 MHz bandwidth for these applications, usually known as Dedicated Short Range Communications (DSRCs) and this same initiative has been expected in different parts of the world. The DSRC standards IEEE 1609.1—IEEE 1609.4 was designed to provide active safety and enhance driver experience [4]. These IEEE standards provide a clear cut definition of physical (PHY) and Medium Access Control (MAC) layer specifications for DSRC and security standards [3]. The DSRC extends to Society of Automotive Engineers (SAE) standards, such as J2735 which defines the protocols and application layers for Vehicular Ad hoc Networks (VANETs). Vehicles are known to generate 15 different messages, well defined by SAE J2735. Among those messages is the basic safety messages (BSMs) which communicate important information about the state of the road network, such as acceleration, heading, global position systems (GPS) data, speed and braking system [5]. BSMs are created at the rate of 10 messages per second (m/s) and contained the required status information so that vehicles can be aware of their environment and prepare to take action in the event of emergency situation.

The bid for adequate and reliable incorporation of the BSMs in some mission critical, safety and emergency for improvement in transportation safety and traffic optimization, have raised formidable research challenges. One of those challenges is security which has so far received limited attention. For instance, any disruption to this network can potentially have deadly consequences [6]. That is why disseminated information must be trusted (i.e., correct), and anomaly free. Currently, all the safety critical information, such as BSMs must be signed to authenticate sender and to verify integrity. However, an authenticated device might be compromised, tampered by malicious users and passed the information among the Connected and Automated Vehicles (CAVs) as well as the centralized server in the cloud. Even worse, malicious users can group to coordinate their efforts to attack the network to inflict a safety related accidents. As of today, identifying malicious user in these networks is an active research area.

Anomalous behaviour in sensors could emerge in different forms and depictions. Several taxonomies of network attacks are discussed in the literature. In line with the summary of intrusion or attacks taxonomy on automated vehicles, the most dangerous attack is false injection attack. In this paper, three types of anomalous sensor behaviour resulting from both sensor faults and injection attacks are considered. An anomalous sensor behaviour is represented as follows, according to the literature [7,8]:

1. Instant I: This form of anomaly is simulated as a Gaussian random variable.
2. Bias B: The anomaly is simulated by adding a temporarily offset to the observation, which is different when compared with the normal sensor reading.
3. Gradual Drift G: This type of anomaly is simulated by linearly adding set of values in decreasing/increasing order to the base sensor values.

Along with literature, focus on detecting and identifying anomalous activity induced by either cyber attacks or faulty sensors, resulting in 'instant', 'bias' and 'gradual drift.'

## Contributions

The contribution of the paper is summarized as follows:

1. We develop anomaly detection approach through combining Bayesian deep learning (BDL), with a well established filter techniques , discrete wavelet transform (DWT), applied to time series BSMs data obtained from multiple sensors.
2. Extensive experimental evaluations are carried out to investigate the effects of anomaly type, magnitude, duration in single and multiple anomaly scenario (unseen anomaly) in real world BSMs dataset.
3. We investigate the sensitivity and distribution of the selected anomalous BSMs sensor values used in the experiment with or without DWT.

The rest of the paper is organised as follows—Section 2 provides a short overview of the relevant works in progress in the field of anomaly detection and identification in CAVs. Section 3 demonstrates out the different misbehavior scenarios and their respective alert types. Section 4 shows out the mechanisms used in the detection and identification of anomalous behaviors associated with cyber-attacks in CAVs networks. In Section 5, presents and discusses the result of our analysis of the methods on anomalous CAVs readings. Finally, in Section 6, concludes the paper.

## 2. Related Work

In this section presents the important works and addresses the related research studies to identify the gaps that need to be taken into consideration for the proposed study question. The recent research on anomaly detection have created a large amount of literature over the past few years, as it is challenging topic in many disciplines, including but not limited to automotive [9], environmental engineering and wireless networks [8]. Methods of anomaly detection are used in a range of applications including systems for fault detection, diagnosis, monitoring, and intrusion detection [8]. In some scenarios where the cause of an anomaly can be detected easily, effective reconfiguration control steps can be taken to prevent or reduce potential loss. A variety of methods have been developed in recent years to detect anomalous behavior, and/or to identify the source of anomaly [7]. For example, in the field of CAVs, current studies conducted, demonstrate the vulnerability of CAV sensors to cyber-attacks or faults, for example, speed, acceleration, and position sensors. Sensor behavior with effect to anomaly can be as a result of either sensor failure or malicious cyber-attack. CAVs have many internal and external cyber-attack surfaces from which adversaries nodes can act on and exploit [10–12]. Different methods for detection of anomalies include observer-based, parity relations and parameter calculation methods have been well explored in the literature [7]. Observer-based (quantitative model-based) detection of faults is among the most popular approach for faults detection. This approach is based on residual sequence derived from using mathematical model and (adaptive) threshold. The work in Reference [7] conceived a novel observer-based approach with comprehensive architecture that combines the adaptive extended Kalman Filter (AEKF) with a moving vehicle model in detection of faults/malicious activities in CAVs network. The authors stated that the proposed model could detect different types of anomalies effectively. However, this approach, is that it is critically affected by uncertainty in noise processing and it is very sensible to the corruption of outliers [13]. In addition, Kalman Filter based strategy has some computational burden [14]. The work in Reference [15] shows that bogus message intrusion and map network attack are two of the most dangerous possible attacks on CAVs. For example, fake messages may be communicated by the infrastructure, that is, Road Side Unit (RSU) , or a nearby vehicle, for example, Wireless Access in Vehicular Environment (WAVE) service advertisement, BSMs, which may in effect generate incorrect, and potentially dangerous responses, for example, bogus braking. The fake message communication may put CAV passengers and other road users in life-threatening conditions. The authors of Reference [16], developed anomaly detection mechanism using entropy-based approaches to detect anomalies in in-vehicle networks. The entropy-based

approach has been well studied in the literature. This approach relies primarily on the similarity between characteristics. However, due to the variation of traffic in CAV systems, the Entropy detection technique is vulnerable to a high rate of false positives [17]. Moreover, research in Reference [8] developed a methodology that can seamlessly detect anomalies and their sources in real time. They developed an anomaly detection mechanism by combining deep learning method, in particular convolutional neural network (CNN) with Kalman Filter- $X^2$  mechanism to detect and identify anomalous sensor readings in CAV system. However, the second phase of the analysis of the model may not perform well when subjected to false data injection attacks derived statistically, due to the independence of statistics characteristics nature of Kalman Filter- $X^2$  [18]. Furthermore, Kalman Filter is known for being computational intensive [14]. The research in Reference [19] created the VANET positional attacks by using conventional attack methods and developed a dataset called Veremi (i.e., Vehicle Comparison Misbehavior). The authors developed a detection methodology termed Maat3, which is a detection and fusion framework based on subjective logic. The mechanism is deployed on false position attacks. Though subjective logic utilizes probabilistic models with an explicit notion of uncertainty, reputation and computation here depends on the structure of trust framework and often involves the discarding of information [20]. Again, this method employs a traditional trust technique with predefined threshold that does not perform well in practice in real time scenario, such as in vehicular networks [21]. These are severe setbacks. Detection of misbehavior in Reference [22] includes deployment of smart protection system to secure the external communication of self-driving cars. The smart system is capable of detecting both grey hole and rushing attack using Intrusion Detection Based Systems-Based (IDS-based) support vector Machine (SVM) and feed-forward neural networks (FFNN). The authors of the work found out that SVM seems to be more reliable. However, this technique is its reliant on the selection of kernel and complex computation in optimization process [23,24], and they considered only single attack scenario.

The work in Reference [25] addresses the problem of cyber-tracking for a platoon that moves in a cohesive form along a single lane, and subjected with different kinds of cyber-threats. The authors proposed a cooperative mechanism that leverage an adaptive synchronization on the basis of control algorithm that incorporates distributed mitigation mechanism of adversary information. The cooperative mechanism in form of closed loop stability is analytically demonstrated using the Lyapunov-Krasovskii theory. A major drawback of this methodology comes as a result of the analytical methods which are not scalable in real time scenario [26]. In Reference [27], the authors suggested an intrusion detection approach for user-oriented V2V to protect the network from access denial, integrity aim, and false warning generation using Greenshield's model. To evaluate the trustworthiness of vehicles's behavior, a series of identification rules related to each attack is utilized. In fact, a vehicle behaviour evaluation technique is established to determine a vehicle's level of trustworthiness. However, this method might not be scalable in high dense network where huge amount of information is involved, using analytical means [21]. The authors of Reference [28] proposed a novel trust method using logistic regression to identify events and malicious vehicles. In this context, the nodes iteratively learn about the environment from received messages and then update the value of their neighbours' trust. A drawback of this model is the complex iterations which may likely result in detection latency. This paper addresses and discusses the drawbacks relative to the cited works above. Moreover, the paper proposes a data-driven anomaly detection mechanism for CAV systems that combines DWT, which is a well denoising technique to smooth the CAV sensor values and Bayesian deep learning (BDL) to learn the normal vehicular behavior, with the aim of identifying anomalous behaviors. The proposed approach is robust with the aid of DWT and the automatic relevant determination (ARD) mechanism of BDL, and shows an adequate care of the instances of noise/outliers that can cause the detection function to exceed the threshold, assuming that the proposed methodology is devoid of complexity based on the fact that BDL operates with optimized weight as a result of addition of prior to weight of neural network (NN). In addition, single and multiple anomalies are considered to access the reliability and robustness of our approach in a realistic network setting.

### 3. Misbehavior Scenarios and Alert Types

Attacks in CAVs can alter the safety related BSMs features, which eventually can result to false emergency alert in the network. For example, a CAVs system might detect certain road characteristics or danger conditions (influencing vehicle driving) and trigger alert to other CAVs. The false alerts as a result of BSMs features manipulations in real time are discussed below.

#### 3.1. Emergency Electronic Brake Light (EEBL)

Anomaly in vehicular speed and lateral acceleration ( $A_x$ ) can result to EEBL false notification. An adversary CAV node  $V_i$  with manipulated speed  $V_i'$ , is capable of introducing any false reference position. This is a passive attack and can cause damage such as rear-end collision. As depicted in Figure 1, an attacker  $V_i$ , raises an alert and sends its false coordinate location  $(x_i', y_i')$  marked with red dotted square and velocity  $v_i'$  across the network to prevent anyone from detecting its false warning.

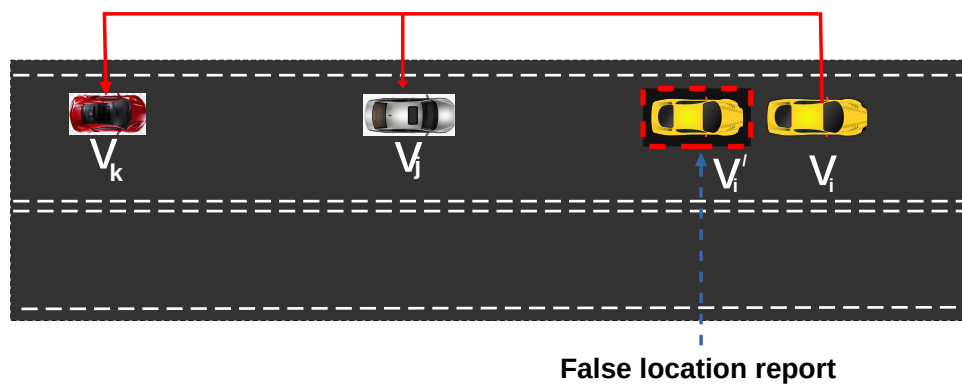


Figure 1. False Emergency ELectric Brake Light (EEBL) alert.

#### 3.2. Change of Lane (CoL)

Change of lane alert (CoL) increases vehicle safety in a dense driving area. This helps to prevent fatal accidents that can occur when a car unexpectedly switches its current path on a roadway. Here, an attacker  $V_i$  in one lane raises a false CoL warning to the next  $V_j$  in another lane with the bid of moving ahead of it. This is an intentional attack in which CAV tries to obtain space in one lane and move ahead of CAV in another lane. This is illustrated in Figure 2, at time  $t$  with velocity  $v_i'$ , attacker  $V_i$  sends a false CoL warning to other CAVs for lane switching with current false location denoted as  $(x_i', y_i')$  instead of its actual position  $(x_i, y_i)$ , as shown in Figure 2. The short distance between  $V_i$  and  $V_k$  prohibits  $V_i$  from changing lane, however, the false reported location makes the inter-vehicle gap between  $V_i$  and  $V_k$  appear too wide for  $V_i$  to change lane immediately with coordinates  $(x_i, y_i) < (x_i', y_i') < (x_j, y_j)$ .

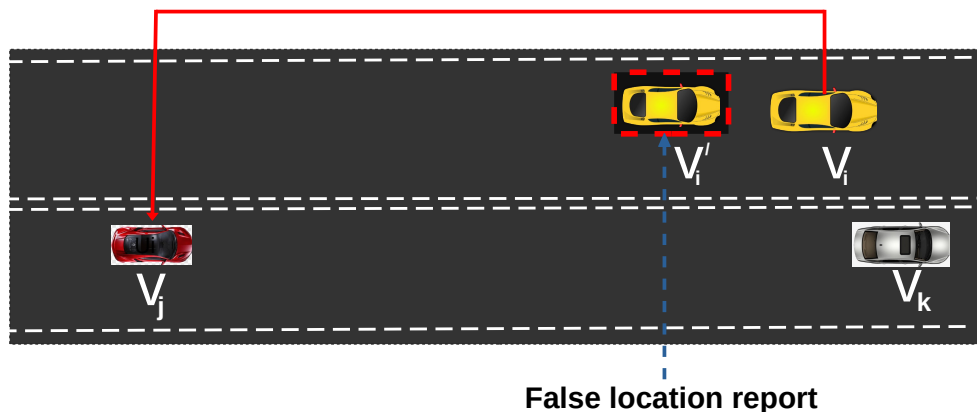


Figure 2. Change of Lane Attack Scenario.

### 3.3. Path Deviation Alert (PDA)

For a straight road, it is estimated that the lateral acceleration should be 0, since the radius of curvature (RoC) is 0. If the road has a RoC that is not negative, then the lateral acceleration  $A_x$  is related with RoC and velocity  $v$  by:

$$A_x = RoC \times v^2, \quad (1)$$

where RoC is  $\frac{1}{Radius(R)}$ .

As shown in Figure 3, the attacker node  $V_i$  at position  $(x_i, y_i)$  can communicate to vehicle  $V_j$  with RoC value of 0.  $V_j$  receives a falsified message and adjusts its speed and keeps heading in a straight line with the assumption that  $V_i$  is at position  $(x_i', y_i')$ , marked with red dotted square. Undoubtedly,  $V_j$  will deviate from the lane if it goes by the information given by attacker node  $V_i$ . The node  $V_j$  needs to take into consideration the tangential speed needed to ply the curved road. Such a false alert will invariably leads to accident or crashing of vehicle.

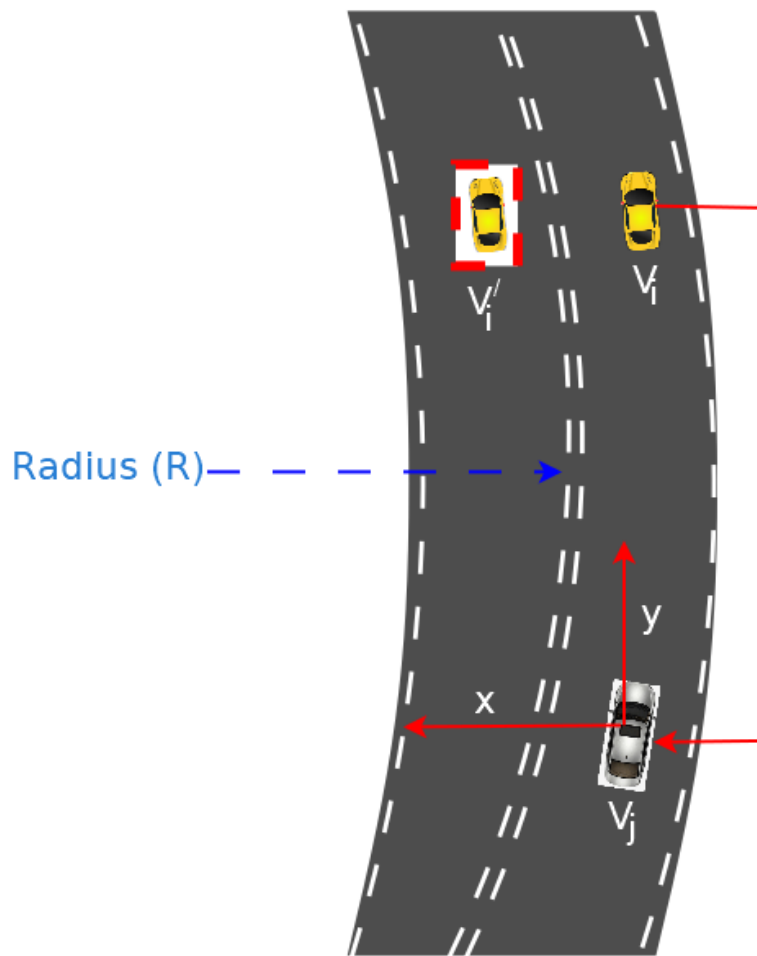


Figure 3. Path Deviation Attacker Scenario.

## 4. Methods

This section discusses the mechanisms used in the detection and identification of anomalous behaviors associated with cyber attacks in CAVs networks. Those approaches are namely, Bayesian deep learning (BDL), convolutional neural networks (CNN-1D). The proposed approach comprises the combination of DWT and BDL. In general, the BSMs data gathered from sensors are the inputs to the detection mechanisms, reading the same or highly correlated physical quantities. Based on the data, for example, a few milliseconds (ms) of each time step, outputs are generated as to whether anomalous behaviour is present, and as such, which sensor reading is anomalous.



#### 4.1. Convolutional Neural Network (CNN)

The receptive field (RF) study provides the theoretical basis of the CNN 's local perception. CNN consists of layer data, hidden layer, and output layer. The hidden layer includes convolution layer, pooling layer, activation layer and fully connected layer. As the core of CNN, the convolution layer, which is prompted by the RF, and computes the convolution of the data from input layer with filters or kernels to extract high-level spatial characteristics. The main function of the pool layer is down-sampling to reduce the number of features. Convolution operations cannot only improve the original features of the data, but also reduces the data's noise [29].

In this paper, we use 1D convolution method in our analysis. The convolution of the input sequence  $x$ , at a time  $t$  is represented as:

$$y_k = f \left( \sum_{i=1}^W (w_i \times x_{t-k+1}) + b_t \right), \quad (2)$$

where  $y_k$  is the output feature at time  $t$ ,  $f(x)$  is a nonlinear activation function,  $w_i$  ( $i = 1, 2, \dots, m$ ) is the filter or kernel of length  $W$  and  $b_t$  is a given offset vector at time  $t$ .

The outstanding effect of extracting spatial features at CNN makes it applicable to time series. Usually, the kernel of convolution is two dimensions. To apply CNN for real-time detection and identification of anomalous sensor behaviour, a fixed-width sliding window, is inserted on input data from all sensors measuring the same quantity, either directly, where conversions or combination with other sensors may be necessary to infer the quantity. The new observations are gathered from sensors at each epoch, and from the sliding window shifts that includes the latest observations. Thus, the input to the CNN during a CAV trip is a series of continuous feed of raw sensor data. Therefore, the CNN allows for a holistic view of multiple sensors simultaneously, by incorporating information from other sensors over time to help detect and identify anomalous values. Since the goal is to detect and identify anomalous behaviour, training of the model for each sensor is implemented using the labelled sensor readings. That is, if there is an anomalous behaviour relating to a given sensor the response variable is either 0 or 1. Different models are trained to detect anomalous behaviour for each sensor, a logical OR operator on the outcomes of the model decides whether or not an anomalous behaviour is observed among all sensors. The architecture in Figure 4 is adopted in the course of the experiment in this work to optimized anomaly detection and identification efficiency on a validation set. This architecture involves three max-pooling and convolution layers. To train the CNN model, a random dropout rate of 0.1 and a batch size of 128 are employed. Additionally, batch normalization and rectified linear unit (ReLU) activation functions are used in layers as shown in Figure 4, and Adam optimizer for TensorFlow is implemented in Python for the binary cross entropy minimization.

The following parameters are implemented for the Adam optimization; learning rate,  $\alpha = 0.001$  fuzz factor  $\epsilon = 10^{-8}$  and  $\beta_1 = 0.9, \beta_2 = 0.99$ . Furthermore, to reduce the chance of over-fitting, early stopping is introduced to track validation set accuracy with a duration of 200 epochs.

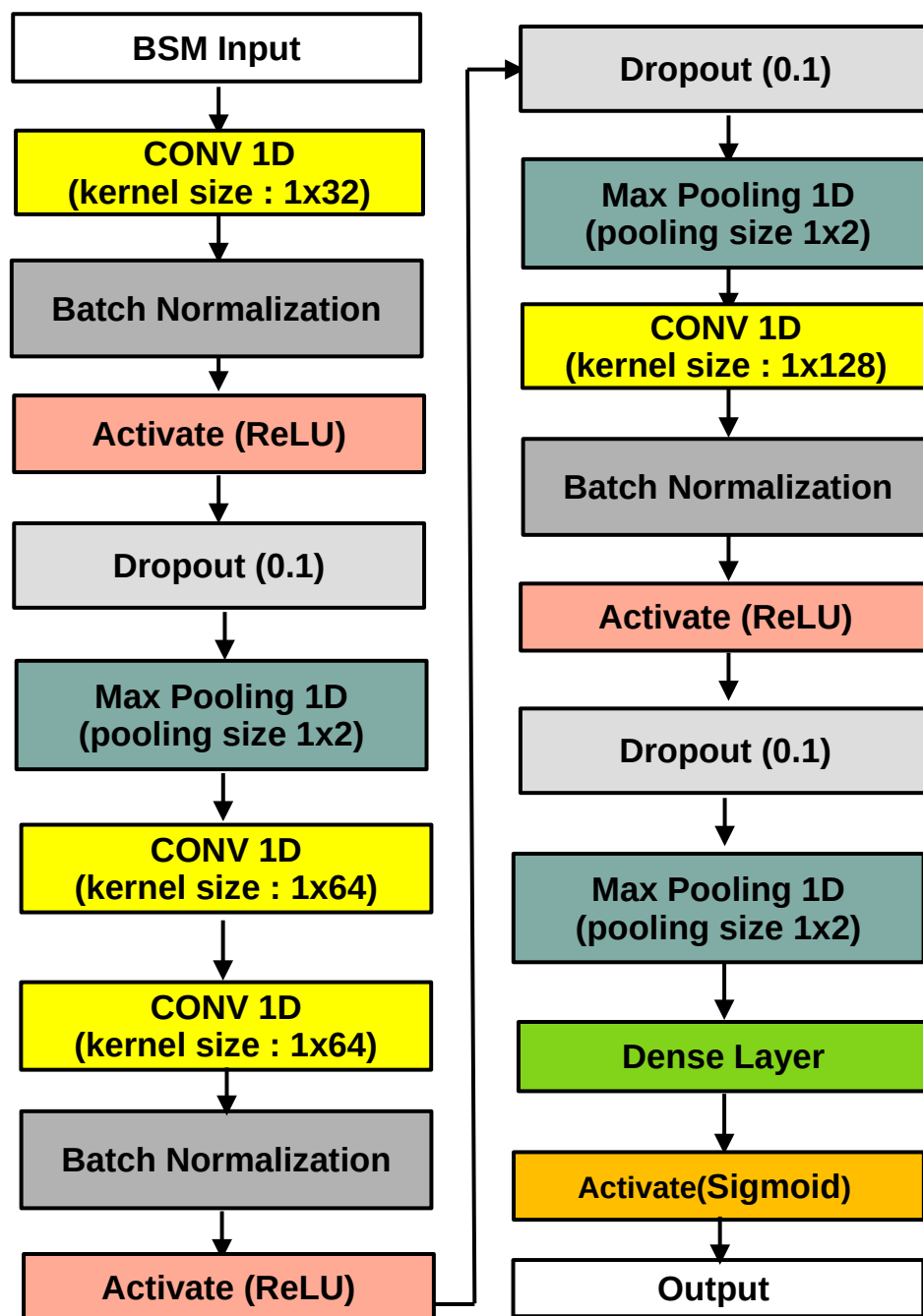


Figure 4. 1-Dimensional Convolutional Neural Network Architecture.

#### 4.1.1. Classification Criterion of CNN Algorithm

Consider a time series  $X = (x^1, x^2, \dots, x^n)^T = (x_1, x_2, \dots, x_t) \in \mathbb{R}^{n \times t}$ , given that  $t$  is the time stamp of each value and  $n$  remains the number of features. Where  $x_t = (x_t^1, x_t^2, \dots, x_t^n) \in \mathbb{R}^n$  is denoted as an input vector at time  $t$  time series anomaly detection is characterized by two problems: (1) Figuring all the anomaly points; (2) Labelling all the target series.

The anomaly detection mechanism can be transformed into a problem of classification with the use of sliding windows. The sliding window mechanism could fragment the entire multi-variable series into continuously shorter sequences and get the two-dimensional data-set.



$$D = (d_1, d_1, \dots, d_{t-T+1}) = ((x_1, \dots, x_T); (x_1, \dots, x_T), \dots, x_1, \dots, x_T) \in \mathbb{R}^{T \times (t-T+1)}, \quad (3)$$

where  $T$  is the sliding window length and  $d_i$  is known as the anomaly series if it has some anomaly value from the series of origin.

Hence the problem of detecting anomalies for time series  $x$  is transformed to label each input vector for  $D$ . Classification approach is used to solve the problem of anomaly detection and training data-set with labels is used to train NN. The target can be shown as:  $\max(\text{Prob}(y_k = 0 | d_j), \text{Prob}(y_k = 1 | d_j))$ , where  $y_k = 1$  denotes honest message and  $y_k = 0$  denotes anomalous message.

#### 4.2. Discrete Wavelet Transform (DWT)

Wavelet transform actually decomposes the data of the time series in both time and frequency domain, even though the data is non-stationary. Wavelet transform has achieved numerous successful applications in engineering fields such as signal processing and image processing. The basic idea of a classic wavelet denoising model is shown below:

$$s(n) = f(n) + \varepsilon(t), \quad (4)$$

where  $s(n)$  is known to be the observed signal (noisy signal),  $f(n)$  is the real signal and  $\varepsilon(t)$  remains the Gaussian white noise. The essence of denoising the wavelet is to filter out the  $\varepsilon(t)$  as much as possible.

The theoretical basis of the wavelet threshold denoising method based on Mallat's theory assumes that the low-frequency approximation part and high frequency information portion of a signal can be fully reconstructed [30]. Suppose an original sensor reading denoted by  $s(n)$ , is given by:

$$s(n) = \sum_{k \in z} c_{j,k} \varphi_{j,k}(n) + \sum_{i=1}^j \sum_{k=z} d_{i,k} \Psi_{i,k}(n), \quad (5)$$

where  $z$  is an integer,  $c_{j,k}$  is the approximate coefficient,  $\varphi_{j,k}$  is the scaling function, while  $j$  is the decomposition level,  $\Psi_{i,k}(n)$  is the wavelet basis function and  $d_{i,k}$  is the detailed coefficient.  $c_{j,k}$  contains information on the low frequency of the original discrete signal  $s(n)$ , which is stated as follows

$$c_{j,k} = \langle s(n), \varphi_{j,k}(n) \rangle, \quad (6)$$

where  $\langle s(n), \varphi_{j,k}(n) \rangle$  denotes the orthogonal relationship between  $s(n)$  and  $\varphi_{j,k}(n)$ . The notation  $d_{i,k}$  has the original discrete signal's high frequency information, which is defined as follows:

$$d_{i,k} = \langle s(n), \Psi_{j,k}(n) \rangle \quad (7)$$

where  $s(n)$  and  $\Psi_{j,k}$  are orthogonal to each other.

The wavelet threshold denoising approach uses its main profile to be the low-frequency part of the signal, while the high-frequency part represents its details. The details of each level has its own noise information after decomposition of the wavelet. Threshold function tune the description coefficients of each level  $d_{i,k}$  and is computed with the approximate coefficients of the last level. The denoising wavelet threshold process is shown in Figure 5.

In Figure 5,  $s(n)$  is the original noisy signal, while  $m_{j,k}$  is the wavelet coefficient as a result of wavelet decomposition of the  $s(n)$ ,  $m_{j,k}$  is obtained from the combination of the approximate coefficient  $c_{j,k}$  and the detailed coefficient  $d_{i,k}$ ,  $v_{j,k}$  remains the estimated wavelet coefficient after the denoising threshold, and  $\hat{f}(n)$  is the estimated  $s(n)$  derived from the reconstruction of  $v_{j,k}$ .



Figure 5. Wavelet Threshold Denoising Mechanism.

#### 4.3. Connected and Automated Vehicles (CAVs) Data Characteristics and Anomaly Model

The data for this analysis are derived from the Safety Pilot Model Deployment (SPMD) program research data exchange (RDE) database [31]. The program was implemented with the primary aim of presenting CAVs in real-world environments, with focus on communication systems such as V2V and V2I communications. The system data was generated from high frequency data (gathered every 100 ms) over a span of two years for more than 2,500 vehicles. The data characteristics derived from the SPMD data-set used in this analysis includes speed ( $s$ ) (sensor 1),  $A_x$  (sensor 2), and  $RoC$  (sensor 3).

As there is no public data-set available for CAVs that includes anomalous behavior in sensor measurements due to attacks and ground truths, the experimental data-sets are generated by simulations. In particular, three types of anomalous behaviors are considered including instant, bias and gradual drift. In the sensors experiment, we presume anomalous values exist independently due to either attack or faults. Therefore, we assume that no more than one anomalous behavior will begin in any epoch at the same time, which is in fact quite unlikely provided that sensors are usually accurate, and that attacks will occur independently.

For our experiment, we generate various anomalous data with different attack rates denoted as  $\alpha \in \{3\%, 10\%, 50\%\}$ , where either exactly one type of attack or all the three types are equally likely to adversely affect each of the three sensors. Explicitly, we sample a uniform random variable of  $\mathcal{U}(0, c)$  where  $c \in 1, 2, 3$  at each time epoch (every 100 ms), in the CAV trip to decide whether anomaly exists, and if so, another uniform random variable of  $\mathcal{U}(0, c)$  is used to decide the sensor that is affected. Based on the experiments, we randomly sample from a set of one or three types of anomalies, with uniform or normal distributions. The simulation is prepared in Python and the generated attacks are added to the base value of the sensor.

Algorithm 1 provides the pseudocode depicting the random generation of anomalies.

---

**Algorithm 1:** Connected and Automated Vehicles (CAV) Cyber Attack Generation Process.

---

```

1   $\alpha$ : rate of anomaly
2  m: number of sensors
3  D: highest anomaly duration
4  t: time epoch
5  for  $t \in T$  do
6    for  $i \in \{1, 2, \dots, m\}$  do
7      if no trace of anomaly occurs at time  $t$  for the  $i$ th sensor then
8        if  $\mathcal{U}(0, c) \leq \alpha$  then
9           $d \leftarrow \text{randi}(D)$ 
10         switch (Choose anomaly type with probability distribution  $f_\omega$ )
11           case Instant:
12             Inject 'instant' anomaly type with parameter  $c_1$ 
13           case Bias:
14             Inject 'bias' anomaly type with parameter  $c_2$  and duration  $d$ 
15           case Gradual Drift:
16             Inject 'Gradual drift' anomaly type with parameter  $c_3$  and duration  $d$ 
17         end switch
18       end if
19     end if
20   end for
21 end for
  
```

---

#### 4.3.1. DWT Pre-Analysis of the Data

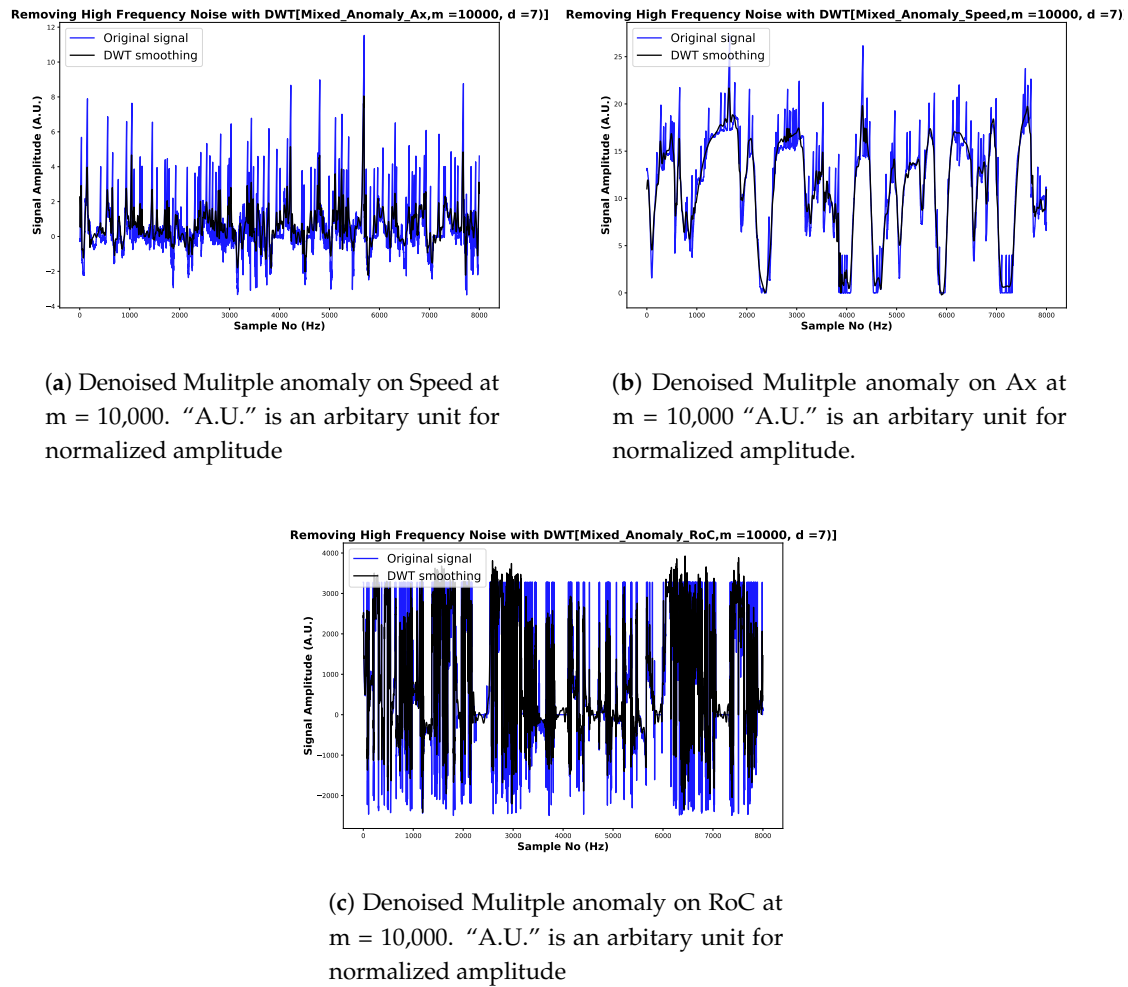
BSMs variables are usually in the form of time series format, and successive time series values are not necessarily independent in real time but are strongly correlated. This makes it difficult to establish successful feature selection strategies for the function that operate directly on time series data. To mitigate the problem, DWT which comes with the flavor of both a feature extractor and de-noising techniques as detailed in Section 4.2, can be applied to time series data to transform it from time domain to frequency domain.

In this section, we mainly focus on the denoising of the BSMs attributes with DWT. The BSMs data have some disturbances as a result of measurement noise and estimate error, which may cause problems in the training phase of the anomaly detection mechanism. In addition, the inbuilt noise in the system can induce outlier effect which can affect the performance of the attack detection mechanisms. The details of mechanism of operation of DWT is examined on the selected BSMs sensors and the simulation for the experiment is carried out with Python.

To make them smoother and accelerate the convergence of the loss function during the training process, the input data is denoised. Table 1 demonstrates the anomalous and denoised plots of BSMs attributes. From the simulation carried out at  $d = 3$  in various network sizes,  $m$  in Table 1, indicates finer coefficients of the noise measurement with decrease in standard deviation as indicated in Table 1, for instant anomaly case. Essentially, in Table 1, we equally conduct analysis on Bias anomalous BSMs, we observed a smoother noise coefficient with DWT. The results in Table 1 of the different simulation carried out on different network size  $m$ , with  $d = 3$ , presents a noise amplitude value reduction as shown in Table 1. A similar simulation is carried out on Gradual Drift anomaly on the BSMs as shown in Table 1, the result shows that the magnitude of the noise level of Gradual Drift anomaly is smaller as compared with Instant and Bias anomaly. We note that the DWT, in which in this context, Daubechies 12 (dB12), consistently maintained a finer coefficient that shows closer representation of the original feature of BSMs. Lastly, analysis of the noise distribution on the mixed anomaly data is carried out on network size  $m = 10,000$ , and  $d = 7$ . From the observations in Figure 6, we find out a significant decrease in the amplitude of the noise distributions of the three BSMs attributes selected in this paper, namely Speed, Ax and RoC. The values of the noise amplitude decreases by  $\sigma = 0.731$  for Figure 6a,  $\sigma = 0.437$  of Figure 6b and  $\sigma = 335.858$  for Figure 6c, respectively.

**Table 1.** Selected basic safety messages (BSMs) variables Descriptive Statistics.

Instant Anomaly (Network Size m)	$\mu$ (Anomaly)	$\sigma$ (Anomaly)	$\mu$ (Wavelet dB(12))	$\sigma$ (Wavelet dB(12))
2000	10.307401	6.170231	10.300251	5.7634621
4000	10.307407	6.170261	10.300258	5.763563
6000	10.307414	6.170258	10.300262	5.7635907
8000	10.307417	6.1702566	10.300263	5.7635937
10,000	10.307415	6.170257	10.300264	5.7635903
Bias Anomaly (Network Size m)	$\mu$ (anomaly)	$\sigma$ (anomaly)	$\mu$ (wavelet dB(12))	$\sigma$ (wavelet dB(12))
2000	0.5829332	1.5610949	0.5813745	1.1946311
4000	0.53638434	1.581967	0.53705674	1.0952997
6000	0.6300388	1.7301117	0.6321792	1.0724595
8000	0.64839834	1.6975222	0.65079004	1.0431184
10,000	0.7417457	1.7784712	0.74250895	1.0478663
Gradual Drift Anomaly (Network Size m)	$\mu$ (anomaly)	$\sigma$ (anomaly)	$\mu$ (wavelet dB(12))	$\sigma$ (wavelet dB(12))
2000	0.07693122	0.910007	0.076246604	0.4898912
4000	0.07699456	0.9098382	0.0763265	0.49061427
6000	0.07699457	0.908578	0.07632571	0.49055964
8000	0.07699094	0.9098535	0.07632209	0.49055964
10,000	0.07699085	0.9098535	0.07632202	0.49055642



**Figure 6.** Denoised Multiple anomaly on Speed, Ax and radius of curvature (RoC) sensor readings.

#### 4.4. Bayesian Deep Learning (BDL)

Imposing of BDL framework is required to overcome the challenges in a NN. BDL incorporates as illustrated in Figure 7, the transformation of NN from point to probabilistic estimation is achieved by first establishing series of functional transformation in different correlated layers. The mathematical representation is stated below:

$$y_k(x, w) = h \left( \sum_{j=1}^H w_{kj}^{(2)} g \left( \sum_{i=1}^D w_{ji}^{(1)} x_i + w_{j0}^{(1)} \right) + w_{k0}^{(2)} \right), \quad (8)$$

where  $y_k$  is taken to be the  $k$ th output of the NN,  $x$  is the vector of the variable  $D$  for the input layer, while  $w$  remains the combination of the adaptive weight parameters  $w_{ji}^{(1)}$  and  $w_{kj}^{(2)}$ , and the bias  $w_{j0}^{(1)}$  and  $w_{k0}^{(2)}$ , while  $H$  is the number of units in the hidden layer. From the traditional approach, the variable  $\theta$  from the training samples is estimated by possible minimization of the error function [32,33]

$$E = E_D + E_W = \frac{1}{2} \sum_{n=1}^N \sum_{k=1}^{N_0} \{y_k(x^n; w) - c_k^n\}^2 + \frac{\alpha}{2} \sum_{i=1}^W |w_i^2|$$

where  $y_k$  denotes the  $k$ th NN output with respect to  $x^n$ , of the  $n$ th training input data;  $c_k^n$  remains the  $n$ th target of the output training data,  $N$  is the corresponding input and output pairs in the target data set;  $N_o$  is denoted as the number of output variables; while  $W$  is the number of parameters in  $w$  and  $\alpha$  is the regularization parameter. The variable  $E_D$  and  $E_\theta$  remain the error between the data and the approximation with respect to NN and decay regularization. To present the NN model within the Bayesian context, the learning processing is to be interpreted as probabilistic ally. Bayesian phase achieves this probabilistic nature of NN by adding strong distribution and uncertainty on the weights in the network. The uncertainty in the weight of the network model enhances the practical framework in automatic calculation of error associated with the predictions when dealing with data of unknown targets. This also leverages the system to learn from a small amount of evidence [34] when Information sparsity is experienced in a given network. Generally, data need to be pre-processed reduces the complexity. A suitable network architecture is then selected, and the model probability is defined as:

$$p(w | \mathcal{D}, \alpha, \beta, M) = \frac{p(\mathcal{D} | w, \beta, M) p(\alpha, \beta | M)}{p(\mathcal{D} | M)} \quad (9)$$

where  $w$  is the adaptive weight parameter,  $\mathcal{D}$  is the data,  $\mathcal{H}$  denotes the Bayesian model class that specifies the form of the likelihood function and the prior probability distribution, and  $\alpha, \beta$  are the regularization parameters. At this stage, network training starts with the optimization of the input and output data by maximising the posterior likelihood of the model specific by  $w$ . At the end of the preparation, the degree of understanding and generalisation is considered adequate, the iterative network optimization process is stopped and predictions can be made using the training network. Bayes' Theorem can be extended as seen below to select the appropriate values for the hyper-parameters:

$$p(\alpha, \beta | \mathcal{D}, M) = \frac{p(\mathcal{D} | \alpha, \beta, M) p(\alpha, \beta | M)}{p(\mathcal{D} | M)} \quad (10)$$

The hyper-parameters  $\alpha$  and  $\beta$  are assumed to be known. Initial values for  $\alpha$  and  $\beta$  are chosen as seen in Figure 3, and the associated values of  $w$  are obtained by maximising their posterior likelihood. Then, using the following relationship, the hyper-parameters are re-estimated where their MAP values are based on uniform prior values for  $\alpha$  and  $\beta$ , and the estimate of these values maximise evidence  $p(\mathcal{D} | \alpha, \beta, \mathcal{H})$  in (10). The estimated values of  $\alpha$  and  $\beta$  are represented below:

$$\alpha' = \frac{\gamma}{2E_D} \quad (11)$$

$$\beta' = \frac{N - \gamma}{2E_D} \quad (12)$$

The  $\gamma$  parameter calculates the approximate number of parameters whose values, rather than the prior, are controlled by the data, that is, the number of parameters that are well determined.

Bayesian approach achieves the correct solution by allowing objective comparison between different models. The most probable model class within a set of classes  $M$  of  $N_m$  (no of candidates) is obtained in Bayesian sample selection by applying the Bayes Theorem as follows:

$$p(M_j | \mathcal{D}, \mathcal{M}) \propto p(\mathcal{D} | M_j) p(M_j | \mathcal{M}) \quad (13)$$

The factor  $p(M_j | \mathcal{D}, \mathcal{M})$  is known as the evidence provided by data  $\mathcal{D}$  for the model class  $M_j$ . The user's judgement on the initial plausibility of each NN model is expressed by the prior probability  $p(M_j | \mathcal{D}, \mathcal{M})$  over the set of model classes  $M_j$  for  $j = 1, \dots, N_m$  where:

$$\sum_{j=1}^{N_m} p(M_j | \mathcal{M}) = 1 \quad (14)$$

The last problem to be discussed when deciding the optimum architecture is the relative value of each input variable is the Automatic Relevance Determination (ARD). Using real-system data, distinguishing the important variables from the redundant ones may be difficult. In the Bayesian framework, the ARD method proposed in Reference [35] can address this problem.

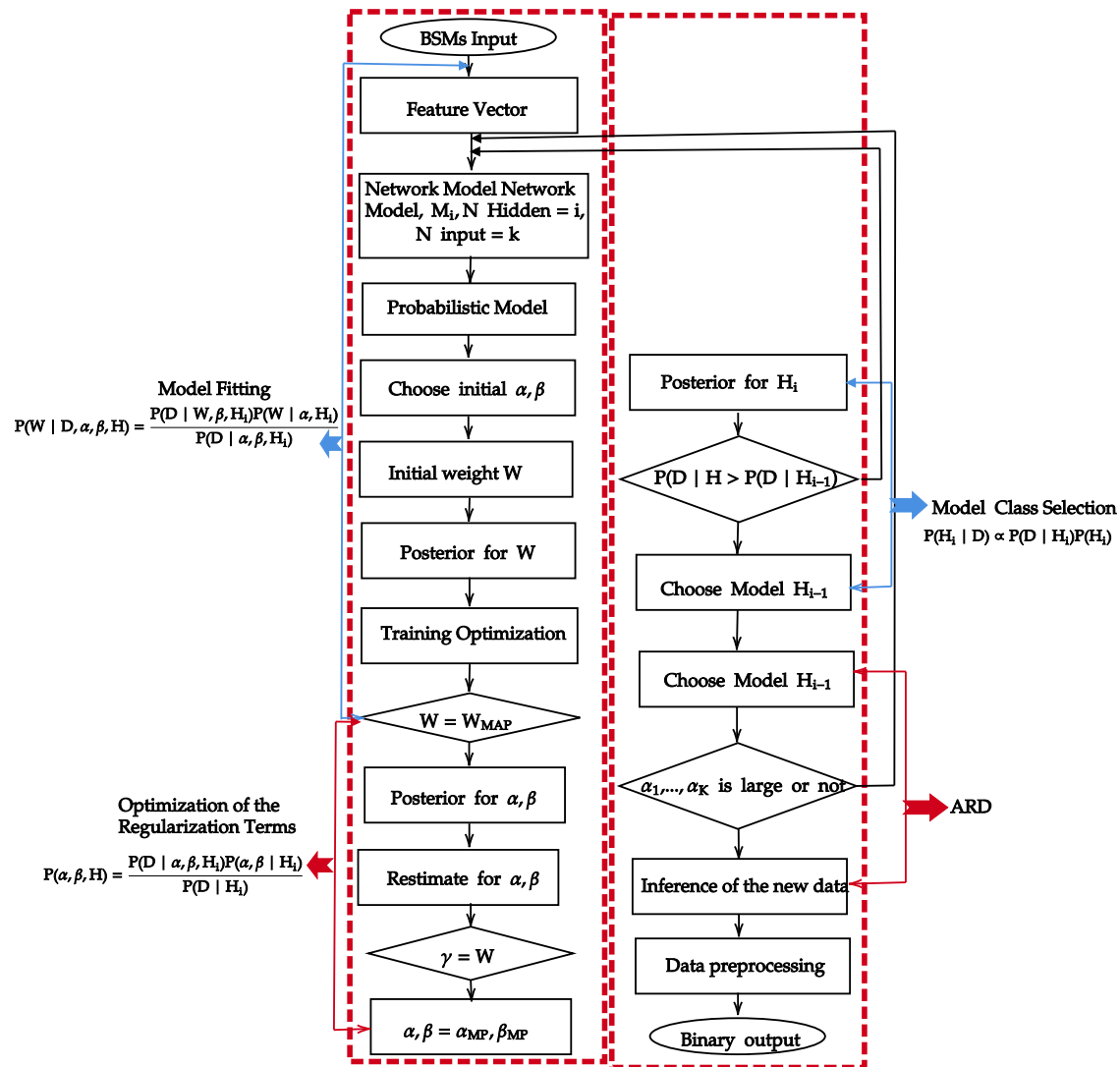


Figure 7. Bayesian Hierarchical Framework for Neural Network.

By using this method, each input variable is associated with a separate hyper-parameter  $\alpha$  which represents the inverse variance of that input parameter's prior distribution. In this way, each hyper-parameter clearly reflects the importance of an input: a small value implies that a large weight parameter value is permitted and the resulting input is important; on the contrary, a large weight parameter value  $\alpha$  is allowed and the associated weights parameter is confined to zero, and thus the corresponding input is less relevant [35,36].

Therefore, the ARD enables inference to be applied in which the  $\alpha$  hyper-parameters are chosen by minimizing evidence for the class of model identified by these hyper-parameters. If the model architecture is established, the importance of each input is assessed. If any hyper-parameters are very high, the relevant input will be removed from the model and the equilibrium design will be re-estimated for the new implementation.



The BDL architecture is assumed to comprise of 4 hidden layers, with 20 nodes and 1 bias in the first layer and the rest of the layers have 10 nodes and 1 bias each. To train the BDL mechanism, ReLU activation function is used, coupled with Adam optimizer with default learning rate to minimize the validation binary cross entropy loss.

#### 4.5. The Proposed Method

To improve the detection and identification efficiency of the BDL algorithm, a new framework is proposed (DWT-BDL) based on reliance of DWT and BDL as shown in Figure 8. Prior to feeding the data into the BDL detection algorithm, DWT is added to the BSMs sensory information for denoising (as explained in Section 4.2). In particular, the noisy sensory BSMs data is decomposed by transforming it into an orthogonal domain and processing operations on the resulting coefficients followed. Eventually, through reconstruction process, the sensory input is transformed back to original state.

The denoised reconstructed BSMs sensory input is fed into the BDL algorithm for further examination and anomaly detection as explained in Section 4.4. This stage of anomaly detection is achieved by first splitting the data into training and testing data-set. The proposed approach is trained to develop a prediction on the training data-set. while the testing data-set is fed to the algorithm for prediction test.

##### 4.5.1. Classification Criterion of the Proposed Approach

This section provides a detailed description of the anomaly detection of the proposed approach in CAVs network and Figure 8 illustrates the process. The value  $\vec{x}$  is a vector of BSMs sensory input of  $D$  variables, while  $c_k$  and  $M$  are the relevant class output (ground truth) to be estimated by the proposed method and the model respectively.

In other words,  $\vec{x}$  is the piece of evidence to be predicted. The variable  $c_k$  is assigned with values 0 or 1. Where 0 or 1, represent the malicious (mal) and honest (hon) information, sent by the nodes respectively. Mathematically,  $\vec{c}_k \in \{mal, hon\} \equiv \vec{c}_k \in \{0, 1\}$  considering a binary classification. The mathematical representation of the classification process can be expressed using Bayes Theorem as follows:

$$p(C = \vec{c}_k | M_j, D = \vec{x}_i) = \frac{p(M_j | C = \vec{c}_k, D = \vec{x}_i) p(C = \vec{c}_k | D = \vec{x}_i)}{p(M_j | D = \vec{x}_i)} \quad (15)$$

By application of total probability theorem, (15) can be expressed as follows:

$$p(C = \vec{c}_k | M_j, D = \vec{x}_i) = \frac{P(M_j | C = \vec{c}_k, D = \vec{x}_i) p(C = \vec{c}_k | D = \vec{x}_i)}{\sum_{c \in (mal, hon)} [p(M_j | C = c_k, D = \vec{x}_i) p(C = c_k | D = \vec{x}_i)]} \quad (16)$$

For possible expression of mathematical simplicity, it is assumed that the individual reports remain independent [37]. From conditional probability of honest and malicious information in vehicular networks, the following mathematical expression is deduced:

$$p(mal | \vec{x}) + p(hon | \vec{x}) = 1 \quad (17)$$

It can be deduced from (17), that  $\vec{X}$  is malicious, when  $\vec{X} = 1 - p(hon | \vec{x})$ . Threshold selection for the categorization of the information into malicious and honest classes is adaptively done as detailed in Section 4.4.

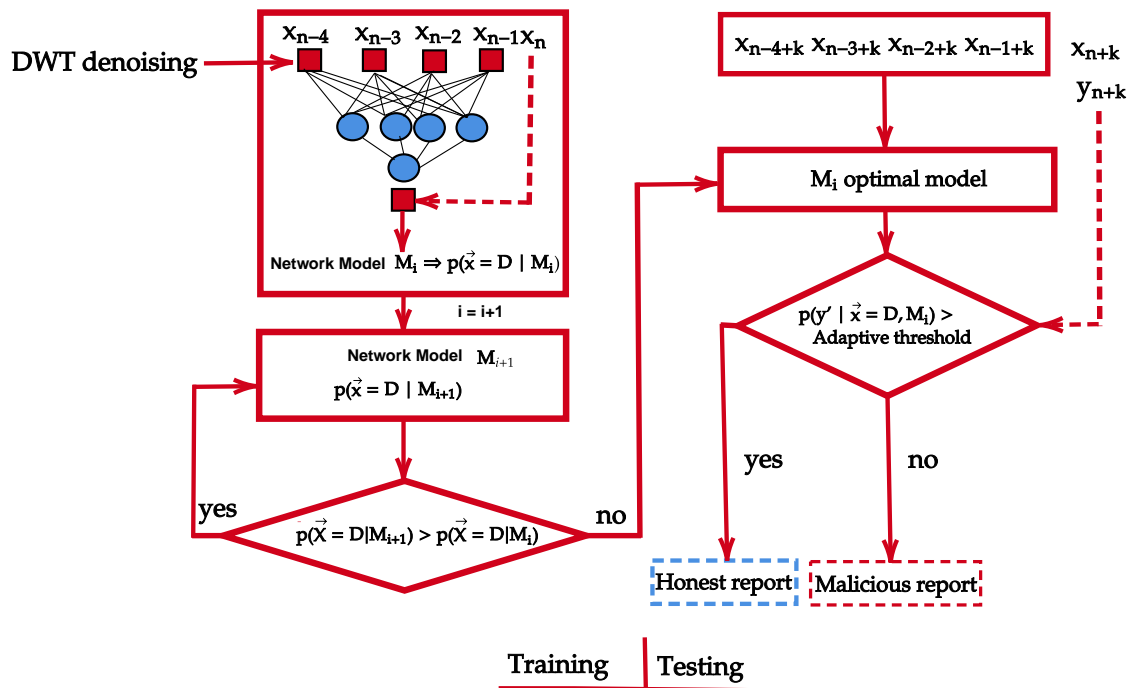


Figure 8. Proposed Approach Framework.

## 5. Results and Discussion

This section shows the results of the analysis of anomaly detection mechanisms in CAVs network. Comparing the results of the individual mechanisms are made to highlight their respective capabilities. The detection performances of the different approaches are taken when trained and tested for a specific category of anomaly or in the presence of all the anomaly types.

In this study analysis, the three anomaly types are simulated from the CAVs data-set with varying anomaly durations, magnitudes of network density and the anomaly rate  $\alpha$  to draw insight on the performance strength from the use of CNN, BDL, and the proposed approach (DWT-BDL) in detecting/identifying anomalous sensor behaviors in real-time. The simulation is carried out with Python libraries namely: TensorFlow and TensorFlow Probability. The training/validation/testing split of 60%, 20%, 20%, are used on every given sample. The validation and training sets are used to tune the parameters of the selected detection mechanisms and different test sets are used to assess and measure the performance of each of the detection/identification mechanisms. For a higher degree of confidence level, each simulation is repeated 15 times with a different seeds. Each experimental result is the average over the number of repeated simulations

### 5.1. Mechanisms Under Single Anomaly System

The different detection mechanisms are applied on the modeled anomalies types and their performance evaluations are taken accordingly. Different datasets are generated, each with a specific type of anomaly, with the anomaly incidence rate  $\alpha$ , and duration  $d$  set to values of 3% and 3 respectively. Furthermore, in all the experiments we consider a varying network densities  $m_i \in \{2000, 4000, 6000, 8000, 10000\}$ .

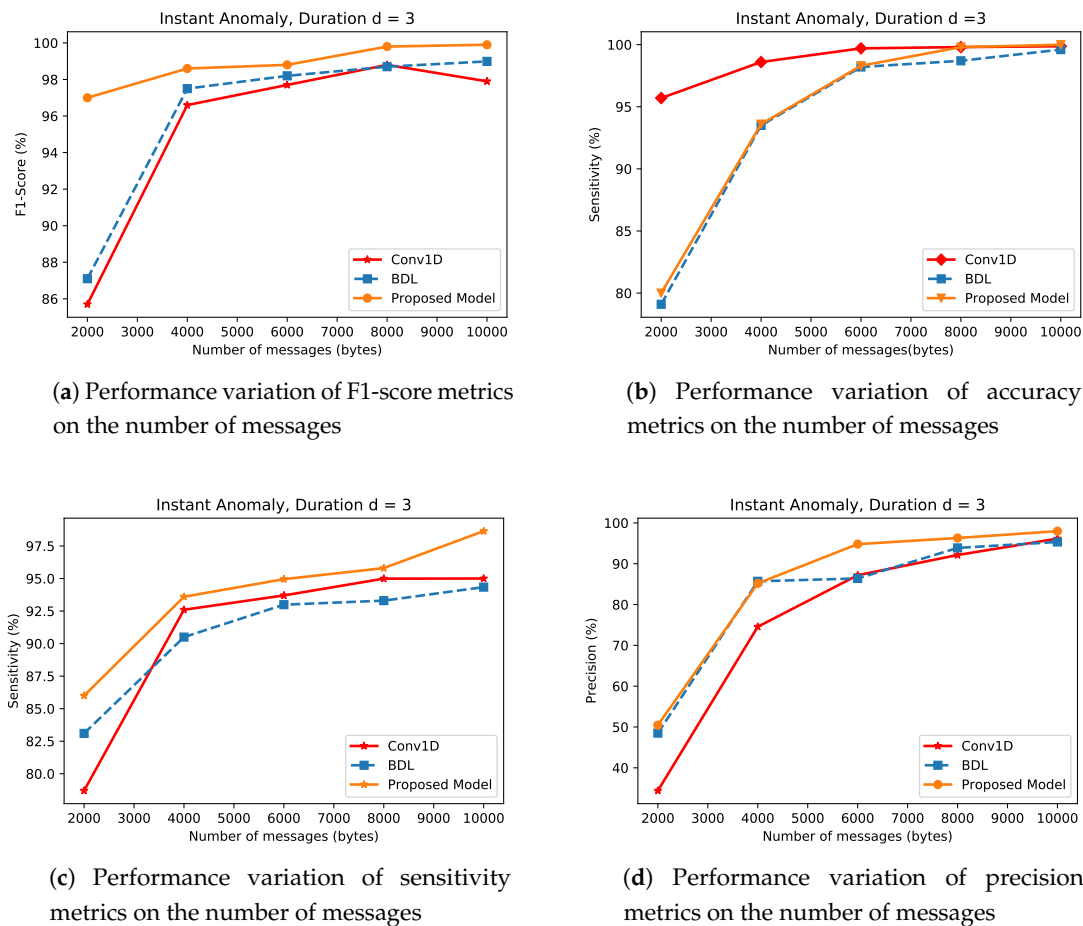
#### 5.1.1. Impact of Network Density on Anomaly Detection

##### (i) Instant Anomaly

In this section, the impacts of the magnitude of network density  $m$ , on the performance of CNN, BDL and the proposed approach (DWT-BDL) are evaluated. Figure 9 indicates that at a low network density CNN, BDL and the proposed approach show poor performance.

This is very much observed in the state-of-the-art approaches compared to the proposed approach. For instance at,  $m = 2000$ , CNN and BDL have the performance values of 85.70% and 85.00% and 85.0% and 87.1% in F1-score as plotted in Figure 9a and sensitivity metrics as plotted in Figure 9c, while the proposed approach improves over CNN and BDL in the same metric values with performance gains of 4.3% and 2.7% compared to CNN and 2.9% and 1.9% compared to BDL respectively. Similarly, Figure 9d illustrates the superiority of the proposed approach. However, the CNN approach in this scenario maintained a lead performance in some cases of  $m$  in the experiment as shown in Figure 9b.

However, at high values of  $m$ , the overall strength of the detection approaches used in this analysis systematically improves. The proposed approach demonstrates a superior performance over BDL and CNN in all the performance metrics except at some accuracy values as shown in Figure 9b, where CNN has a better performance. However, the proposed approach outperforms CNN, BDL in the rest of the metrics in all the cases of  $m$ . For instance, at density  $m = 10,000$ , the sensitivity of the proposed approach shows performance gains of 5.0% and 4.0% when compared to BDL and CNN, respectively. This superior performance trend of the proposed approach is replicated in all the performance metrics as seen in Figure 9.



**Figure 9.** Detection Performance of Instance Anomaly type for Convolutional Neural Network (CNN), Bayesian Deep Learning (BDL) and the Proposed Method.

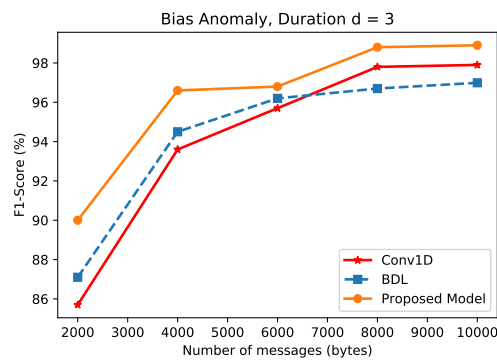
It can be generalized that as the value of  $m$  decreases, the overall detection approaches systematically deteriorates. This clearly indicates the relevance of  $m$  in anomaly detection system. The consistent superior performances of the proposed approach, relatively in all the metrics as a result of combining the performance of the BDL and DWT. The proposed

approach utilizes the decomposition and denoising qualities of DWT, coupled with the robust BDL mechanism in optimal decision making.

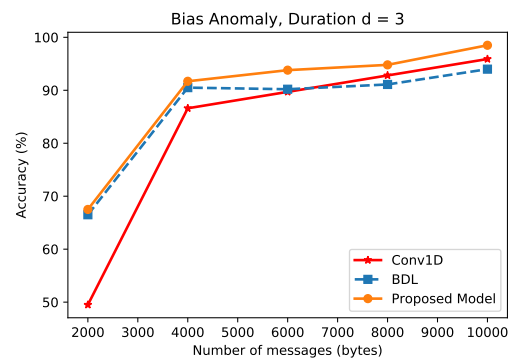
(ii) **Bias Anomaly**

Figure 10 presents the bias anomaly type results for BDL, CNN and the proposed mechanism. As demonstrated in the experiments, at a very small magnitude of  $m$ , the BDL approach performs better than the CNN, while the proposed approach outperforms both BDL and CNN mechanisms in all the metrics in small network density scenario. For example, BDL's accuracy as shown in Figure 10b, by approximately 5.2% higher than CNN's, while the proposed approach has performance gain of 1.2% and 5.2% over BDL and CNN respectively, with  $m = 2000$  samples drawn from  $\mathcal{U}(0, 1)$ .

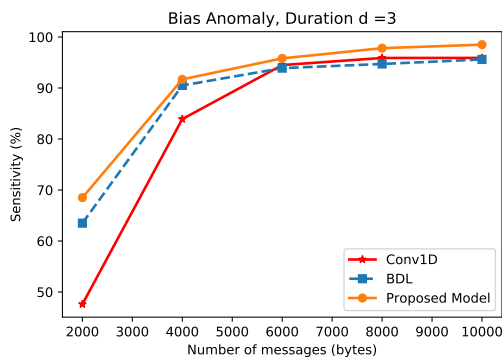
At  $m = 10,000$  in the simulation, the efficiency of these approaches increase as the magnitude of network density increases. Detection mechanisms in this anomaly/attack case show a similar behavior as shown in instant anomaly case with the distribution drawn from a fixed random variable  $\mathcal{U}(0, 1)$  and duration  $d = 3$ . The proposed approach shows improvement in sensitivity metrics as illustrated in Figure 10c, more than BDL and CNN by values of 4.5% and 2.6%, respectively. Similarly, the performance evaluation on bias anomaly as demonstrated in Figure 10a,d support the superiority of the proposed approach over BDL and CNN.



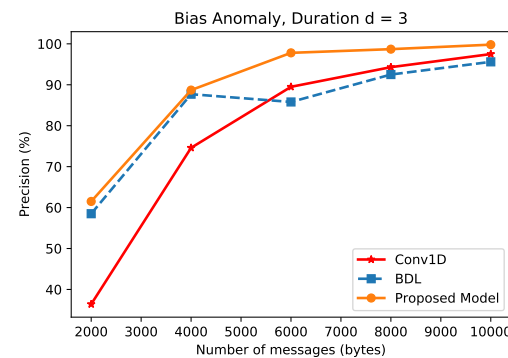
(a) Performance variation of F1-score metrics on the number of messages



(b) Performance variation of accuracy metrics on the number of messages



(c) Performance variation of sensitivity metrics on the number of messages



(d) Performance variation of precision metrics on the number of messages

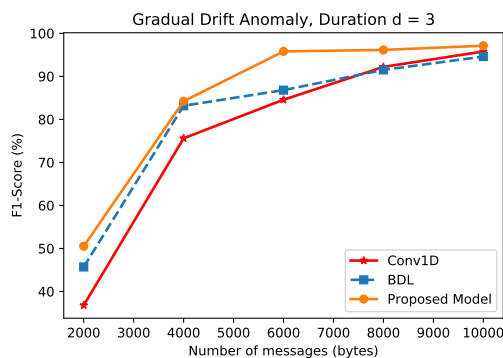
**Figure 10.** Detection Performance of Bias Anomaly type for CNN, BDL and the Proposed Method.

(iii) **Gradual Drift Anomaly**

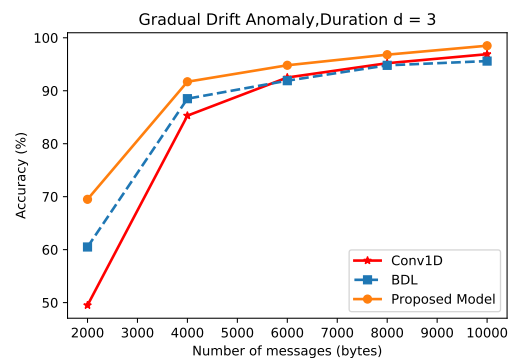
Figure 11 shows the results of the BDL, CNN, and proposed approach for gradual drift anomaly type detection. This type of anomaly involves a gradual rise in sensor values making

it difficult to identify and discern the onset of anomaly from normal sensor values. In general, for a small magnitude of network densities, BDL outperforms the CNN detection performance. For instance, in Figure 11a,d using BDL approach, at network density of 2000, the F1-score and precision respectively, increase by approximately 2.0% and 14.10% when compared to CNN mechanism. However, at high magnitude of network density, CNN consistently outperforms BDL in all the performance across the experiments. For instance, at  $m = 10,000$ , the F1-score and precision metrics of CNN, improve by 2.4% and 3% over BDL approach.

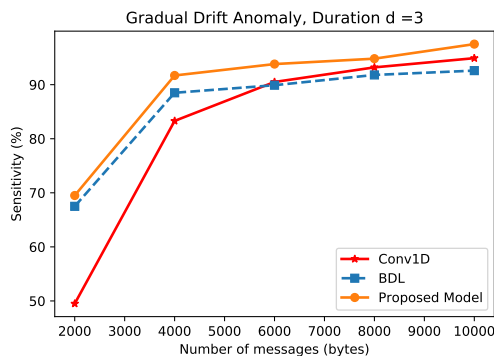
Considering the anomaly detection performance of the proposed approach for gradual drift anomaly, the following is noted. The experiments indicate that the proposed approach provides a significant improvement in low and high density networks scenarios when compared to CNN and BDL. For instance, at low density network, the proposed approach in respect to F1-score, precision and sensitivity, as plotted in Figure 11a,c,d has performance gains of 9%, 16% and 20%, respectively, over CNN approach and 6.95%, 1.95% and 2% compared to BDL. At a high value of  $m$ , the detection performance of the various approaches used in this context increase across all the metrics. Furthermore, it is shown that in general, the proposed approach outperforms CNN and BDL. For instance, experiment carried out on network density of 10,000, again shows that F1-score, precision and sensitivity approach are increased by 2.4%, 1.8% and 2.6% over the CNN mechanism, and by 2.4%, 2.68% and 4.9% over BDL approach. Moreover, the proposed approach improves upon the detection performance of both DWT and BDL respectively.



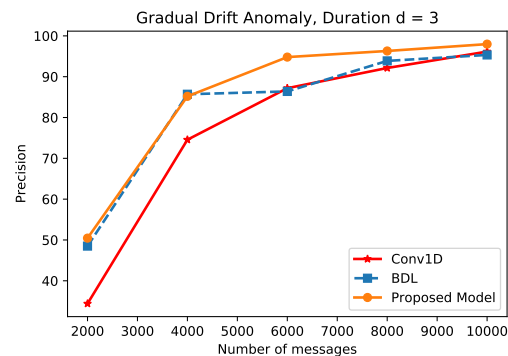
(a) Performance variation of F1-score metrics on the number of messages



(b) Performance variation of accuracy metrics on the number of messages



(c) Performance variation of sensitivity metrics on the number of messages



(d) Performance variation of precision metrics on the number of messages

**Figure 11.** Detection Performance of Gradual Drift Anomaly type for CNN, BDL and the Proposed Method.

## Discussions of the Mechanisms Under Single Anomaly

As seen in the experiments of single anomaly types, the performances of the detection approaches significantly increase in consonant with the magnitude of anomaly duration  $d$ , network density  $m$  and the considered distributions,  $c \times \mathcal{N}(0, 0.01)$ ,  $\mathcal{U}(0, c)$ , and  $\text{linspace}(0, c)$ . Intuitively, the larger the distribution, the larger the deviation from the true values of the normal sensors behaviors, thus the greater the effectiveness of the approaches in detecting the anomalies.

In addition, regarding the high performance of the detection mechanisms with longer duration, is simply because longer duration extends the detection mechanisms time to accumulate a practical body of knowledge about the behaviors and anomaly impacts in a given environment.

In all, the single attack system as shown in Section 5.1, the detection mechanisms can generalize and correctly classify previous unseen observation with similar distribution, (test set) throughout the experiment by training on representative training sets. However, in practice, CAVs anomaly detection methods can experience instance of anomalies for which the mechanisms are not trained specifically. Details of the incidents of unseen observations as expressed in the multiple attack scenarios are discussed in the next section below.

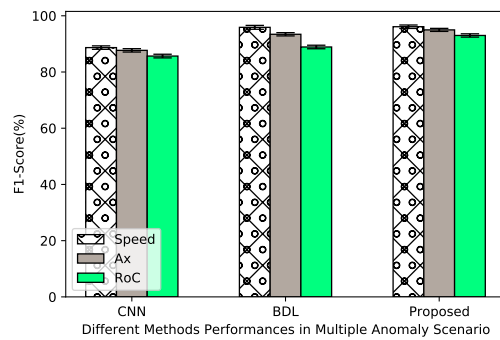
### 5.2. Mechanisms Under Multiple Anomaly System

This section addresses the effectiveness and reliability of a specific detection mechanism used in multiple anomaly scenarios. To obtain a more solid insight on the robustness of detection mechanisms in producing practical performance, the anomaly/attack detection approaches are exposed to multiple attack/anomaly scenarios, with real-time attribute of varying anomalous behaviors. In this context, the instant, bias and gradual drift anomalies are all present in the test data-set and modeled with  $100,000 \times \mathcal{N}(0, 7)$ ,  $\mathcal{U}(0, 3)$ ,  $\text{linspace}(0, 3)$  and  $d = 7$ . Investigation is carried out on the generalisation of the detection mechanisms on the unobserved multiple anomaly scenarios, having been trained on one of the anomaly types. We only consider the impacts of the percentage of anomaly rate  $\alpha$  on the three different sensors in the experimental settings. The evaluation performances of the detection mechanisms are carried out with two anomaly rates (at  $\alpha = 10\%$  and  $\alpha = 50\%$ ) scenarios.

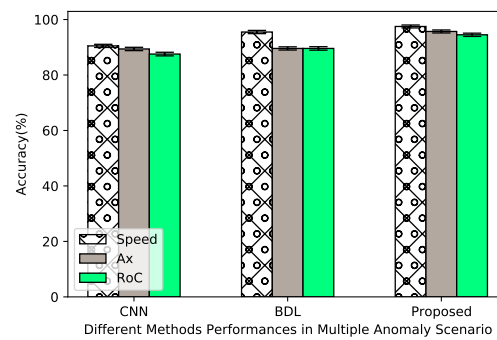
To ensure statistical relevance of the performance of the detection mechanisms, the simulations are run multiple times at least 20 times to provide a confidence interval (CIs) for CNN and credible interval denoted in this context as (CRIs) for the proposed mechanism and BDL respectively. Specifically, we present the mean performance along with the 95% CI and CRIs.

Figure 12 depicts the performance of the detection mechanisms used in this study, for the multiple anomaly scenario with  $\alpha = 50\%$ . The value of  $\alpha$  is set high to capture the behaviors and detection capabilities of the selected mechanisms for threat that can pose considerable risk to the operation of the CAVs system. Figure 12a shows that the detection performances of all the metrics vary across the sensors, in all the experiments. In particular, for sensor 3 (RoC), it is observed that the performance values of the approaches are worse when compared with the other two sensors, Ax and speed that appear to show much smoother reading over time, under the same anomaly scenario. This is partly being attributed to the tremendous variation of consecutive RoC readings. The distribution of the RoC values in Figure 12a, validates the performance degradation of the mechanisms when subjected to unobserved anomalous RoC sensor values. However, the proposed approach demonstrates a lead performance in all the scenarios with the values of  $4.92 \pm 0.009$  and  $6.95 \pm 0.009$  over BDL and CNN, considering the worst case scenario of RoC sensor analysis. This significant improvement in performance is also replicated in Figure 12b–d.

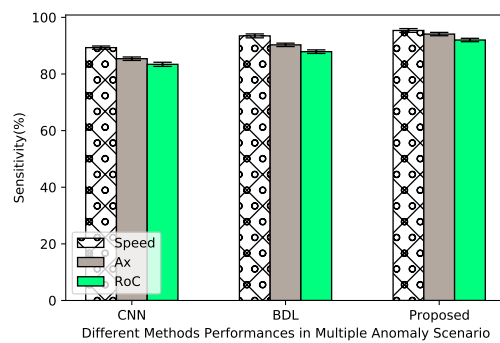




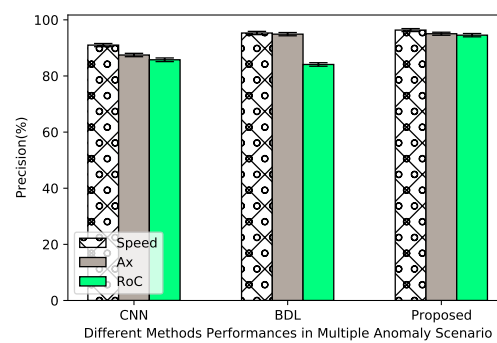
(a) Performance Variation of F1-Score Metrics of the Different Methods on the Three Sensors (Speed, Ax, and RoC)



(b) Performance Variation of Accuracy Metrics of the Different Methods on the Three Sensors (Speed, Ax, and RoC)



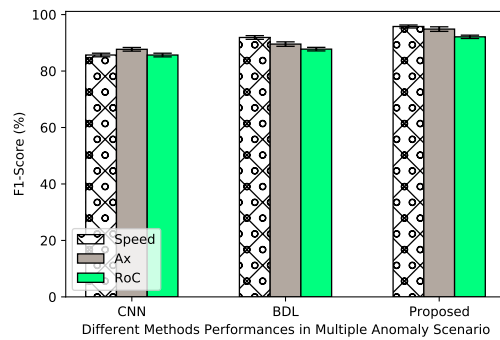
(c) Performance Variation of Sensitivity Metrics of the Different Methods on the Three Sensors (Speed, Ax, and RoC)



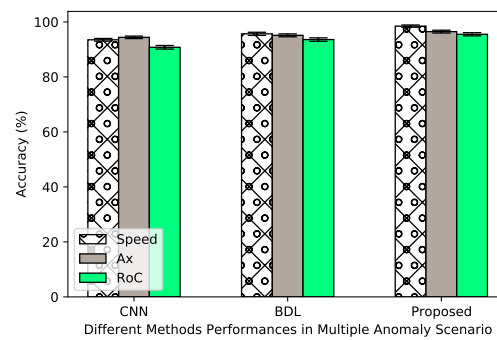
(d) Performance Variation of Precision Metrics of the Different Methods on the Three Sensors (Speed, Ax, and RoC)

**Figure 12.** Detection performance and 95% confidence intervals (CIs) and CRIs across 15 to 20 different executions for all three methods, at anomaly rate  $\alpha = 50\%$  and in the presence of all the types of anomalies.

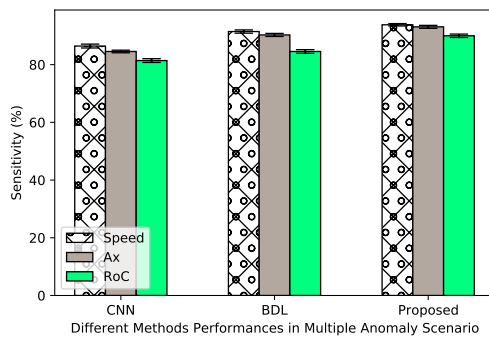
A similar experiment is conducted to investigate the behaviors of CAVs system and detection mechanisms at low value of  $\alpha = 10\%$  and  $d = 7$ , with all the types of anomaly present. As shown in Figure 13 the detection performances of the selected mechanisms vary among various sensors, again detection mechanisms show the lowest performance values when applied to anomalous RoC sensor values. This may equally be attributed to the variation in input read sequence of RoC sensor readings. At a value of  $\alpha$  set to 10%, the approaches tend to show a better classification accuracy but generally indicate poor performances on other metrics, especially on Figure 13a when compared to  $\alpha = 50\%$  as shown in Figure 12a. The observation is in compliance with intuition, as lower value of  $\alpha$  makes the anomaly more elusive and thus more difficult to spot. The high detection accuracy as shown in Figure 13b in this case may be as a result of the imbalance nature of the BSMs samples, classification in this context appears to favor the more representative class [38]. Accuracy metrics may not be a much appropriate performance metric for imbalance data. The proposed approach presents significant performances in the values of precision and sensitivity metrics when compared to BDL and CNN as shown in Figure 13c and 13d respectively. Our focus mainly is on F1-score which provide more insights on the strength of detection mechanisms in event of imbalance class [39] scenario. From the obtained F1-score results in Figure 13a, BDL outperforms CNN, while the proposed approach provides significant improvement of performance by values of  $4.36 \pm 0.010$  and  $6.45 \pm 0.010$  compared to the performance of BDL and CNN mechanisms.



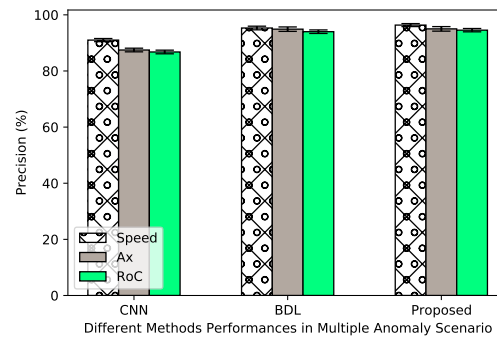
(a) Performance Variation of F1-Score Metrics of the Various Methods on the Three Sensors (Speed, Ax, and RoC)



(b) Performance Variation of Accuracy Metrics of Different Methods on the Three Sensors (Speed, Ax, and RoC)



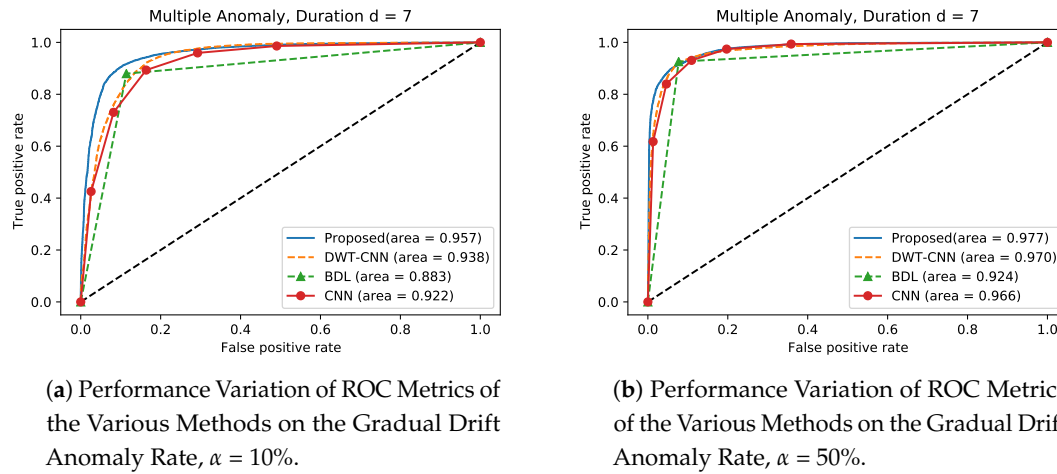
(c) Performance Variation of Sensitivity Metrics of the Various Methods on the Three Sensors (Speed, Ax, and RoC)



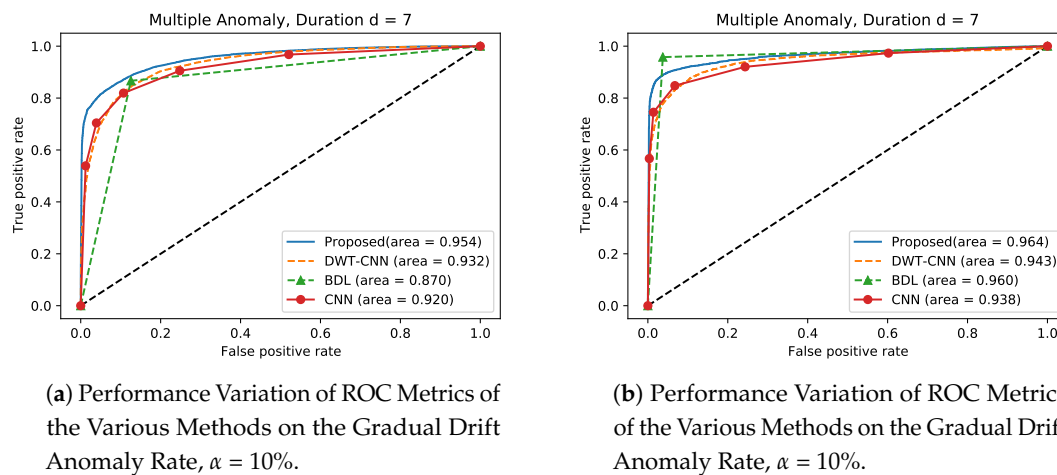
(d) Performance Variation of Precision Metrics of the Various Methods on the Three Sensors (Speed, Ax, and RoC)

**Figure 13.** Detection performance and 95% CIs and CRIs across 15 to 20 different executions for all three methods, at anomaly rate  $\alpha = 10\%$  in the presence of all the types of anomalies.

Furthermore, in the experimental setting, we compute the Area Under the Curve (AUC) of the Receiver Operating Characteristics (ROC) curve to validate the performance and reliability of our proposed model on gradual drift anomaly at  $\alpha = 10$  and  $\alpha = 50$  respectively. The ROC curve is a graphical plot tool that shows the binary classifier's diagnostic potential as its discrimination threshold is varied and this result denotes the plot of true positive rate (sensitivity) as against false positive rate (1-specificity). In the simulation setting in this scenario, we train the methods on Bias and Instant anomaly and then carry out detection process of these trained methods on gradual drift anomaly to validate and generalize the performance of the proposed approach. From Figures 14 and 15 respectively, we observe that the proposed approach shows superior performances over the existing approaches. For instance, in Figure 14a at  $\alpha = 10\%$ , the proposed approach has a performance gain of 1.9%, 7.4% and 3.5 over DWT- CNN, BDL and CNN, respectively. At the same time, Figure 14b shows that when  $\alpha = 50\%$ , the proposed approach is improved by values of 2.2%, 8.4% and 3.4, respectively, when compared with over DWT- CNN, BDL and CNN, respectively. Similarly, Figure 15a indicates that the proposed method demonstrates a significant improvement by values of 0.7%, 5.3% and 1.1 compared with the performance of DWT- CNN, BDL and CNN respectively. Finally, results in Figure 15b also indicates that the propose approach provides significant improvement in performance by values of 2.1%, 0.4% and 2.6 over DWT- CNN, BDL and CNN, respectively.



**Figure 14.** Performance Variation of the Various Methods trained on Instance Anomaly, on Gradual Drift Anomaly at  $\alpha = 10\%$  and  $\alpha = 10\%$ .



**Figure 15.** Performance Variation of the Various Methods trained on Bias Anomaly, on Gradual Drift Anomaly at  $\alpha = 50\%$  and  $\alpha = 50\%$ .

### Discussion of the Mechanisms Under Multiple Anomaly

The purpose of this experiment is to establish a likely situation of multiple anomaly/attack which depicts real-time scenarios. The proposed approach demonstrates a reasonable detection performance as a result of the Bayesian prior probability to establish synergies/fusion between heterogeneous information and classification of out-of-distribution instances as unknown, for impressive detection of unknown attacks/anomalies in a system [40,41].

## 6. Conclusions

From CAVs framework standpoint, the proposed mechanism is developed by combining DWT and BDL. The DWT applies wavelet transform to decompose the sequence of measurements of the anomalous input vectors (BSMs sensor readings) and to denoise them before being fed into BDL for further analysis. The simulation results show the effectiveness of the proposed approach in real-time detection/identification of anomalous sensor values in CAVs setting. In particular, the experiment shows the ability of the proposed approach in adapting and capturing of anomalous behaviours in the unstable CAVs network states (density, duration and anomaly rate) and hence providing good

performance. More specifically, simulation results shows the ability of the proposed approach to detect and identify anomalous sensor values in real time with high precision, accuracy, F1-score and sensitivity by using BDL empowered by DWT on raw sensor data. Moreover, the simulation results show the performance gain of the proposed approach in comparison with the baseline mechanisms, with a significant difference.

The anomalous values of sensors used in the experiments are simulated, along with previous literature researches, primarily because this form of data is not yet readily available. Additionally, the tests are limited to on-board sensors due to the lack of data on CAVs.

As a conclusion, manufactures and policy-makers will benefit from the findings in this study on the importance of providing redundant information for a particular parameter such as curvature radius and speed of a vehicle. Consequently, more redundant sensors and systems for collecting information may be introduced and considered in CAVs to improve their resilience against anomalous sensor values. The proposed mechanisms presented in this paper are also intended to be extended to various sources of CAVs/M2M networks to improve their safety.

**Author Contributions:** E.E. conceived and designed the experiments as well as analyzing the data generated from the experiments. F.A., S.A., L.M.S.-J., A.P., and D.C.-D.-W. reviewed and discussed the manuscript. All authors have read and agreed to the published version of the manuscript

**Funding:** This research received no external funding.

**Acknowledgments:** Authors thank to the WiCIP Lab for providing all the equipment necessary to collect the information which is analyzed and discussed in this manuscript.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Ahmed, S.; Tepe, K. Entropy-Based Recommendation Trust Model for Machine to Machine Communications. In *Ad Hoc Networks*; Springer: Berlin, Germany, 2017; pp. 297–305.
2. Grace, N.; Thornton, P.; Johnson, J.; Blythe, K.; Oxley, C.; Merrefield, C.; Bartinique, I.; Morin, D.; Zhang, R.; Johnson-Moffet, L.; et al. *Volpe Center Annual Accomplishments: Advancing Transportation Innovation for the Public Good-January 2018*; Technical Report; National Transportation Systems Center (US): Cambridge, MA, USA, 2018.
3. den Hartog, J.; Zannone, N. Security and privacy for innovative automotive applications: A survey. *Comput. Commun.* **2018**, *132*, 17–41.
4. Ahmed, S. Trust Establishment and Management in Adhoc Networks. Ph.D. Thesis, University of Windsor, Windsor, ON, Canada, 16 September 2016.
5. Liu, J.; Khattak, A.J. Delivering improved alerts, warnings, and control assistance using basic safety messages transmitted between connected vehicles. *Transp. Res. Part C Emerg. Technol.* **2016**, *68*, 83–100.
6. Cai, R.; Zhang, Z.; Tung, A.K.; Dai, C.; Hao, Z. A general framework of hierarchical clustering and its applications. *Inf. Sci.* **2014**, *272*, 29–48.
7. Wang, Y.; Masoud, N.; Khojandi, A. Real-Time Sensor Anomaly Detection and Recovery in Connected Automated Vehicle Sensors. *IEEE Trans. Intell. Transp. Syst.* **2020**, doi:10.1109/TITS.2020.2970295.
8. van Wyk, F.; Wang, Y.; Khojandi, A.; Masoud, N. Real-time sensor anomaly detection and identification in automated vehicles. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 1264–1276.
9. Ahmad, S.; Lavin, A.; Purdy, S.; Agha, Z. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing* **2017**, *262*, 134–147.
10. Petit, J.; Feiri, M.; Kargl, F. Spoofed data detection in vanets using dynamic thresholds. In Proceedings of the 2011 IEEE Vehicular Networking Conference (VNC), Amsterdam, The Netherlands, 14–16 November 2011; pp. 25–32.
11. Checkoway, S.; McCoy, D.; Kantor, B.; Anderson, D.; Shacham, H.; Savage, S.; Koscher, K.; Czeskis, A.; Roesner, F.; Kohno, T.; et al. Comprehensive experimental analyses of automotive attack surfaces. In Proceedings of the USENIX Security Symposium, San Francisco, CA, USA, 8–12 August 2011; Volume 4, pp. 447–462.

12. Weimerskirch, A.; Gaynier, R. An Overview of Automotive Cybersecurity: Challenges and Solution Approaches. In Proceedings of the 5th International Workshop on Trustworthy Embedded Devices co-located with CCS 2015, University of Michigan, Ann Arbor, MI, USA, 16 September 2015; p. 53.
13. Salahshoor, K.; Mosallaei, M.; Bayat, M. Centralized and decentralized process and sensor fault monitoring using data fusion based on adaptive extended Kalman filter algorithm. *Measurement* **2008**, *41*, 1059–1076.
14. Sewak, M.; Singh, S. IoT and distributed machine learning powered optimal state recommender solution. In Proceedings of the 2016 International Conference on Internet of Things and Applications (IOTA), Pune, India, 22–24 January 2016; pp. 101–106.
15. Petit, J.; Shladover, S.E. Potential cyberattacks on automated vehicles. *IEEE Trans. Intell. Transp. Syst.* **2014**, *16*, 546–556.
16. Müter, M.; Asaj, N. Entropy-based anomaly detection for in-vehicle networks. In Proceedings of the 2011 IEEE Intelligent Vehicles Symposium (IV), Baden-Baden, Germany, 5–9 June 2011; pp. 1110–1115.
17. Marchetti, M.; Stabili, D.; Guido, A.; Colajanni, M. Evaluation of anomaly detection for in-vehicle networks through information-theoretic algorithms. In Proceedings of the 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a Better Tomorrow (RTSI), Bologna, Italy, 7–9 September 2016; pp. 1–6.
18. Ding, D.; Han, Q.L.; Xiang, Y.; Ge, X.; Zhang, X.M. A survey on security control and attack detection for industrial cyber-physical systems. *Neurocomputing* **2018**, *275*, 1674–1683.
19. van der Heijden, R.W.; Lukaseder, T.; Kargl, F. Veremi: A dataset for comparable evaluation of misbehavior detection in vanets. In Proceedings of the International Conference on Security and Privacy in Communication Systems, Singapore, 8–10 August 2018; pp. 318–337.
20. Škorić, B.; de Hoogh, S.J.; Zannone, N. Flow-based reputation with uncertainty: Evidence-based subjective logic. *Int. J. Inf. Secur.* **2016**, *15*, 381–402.
21. Ezizama, E.; Tepe, K.; Balador, A.; Nwizege, K.S.; Jaimes, L.M. Malicious Node Detection in Vehicular Ad-Hoc Network Using Machine Learning and Deep Learning. In Proceedings of the 2018 IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, UAE, 21 February 2018; pp. 1–6.
22. Godsmark, P.; Kirk, B.; Gill, V.; Flemming, B. *Automated Vehicles: The Coming of the Next Disruptive Technology*; The Van Horne Institute: Calgary, AB, Canada, 22 January 2016.
23. Yang, Y.; Feng, Q.; Sun, Y.L.; Dai, Y. RepTrap: A novel attack on feedback-based reputation systems. In Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, Istanbul, Turkey, 22 September 2008; p. 8.
24. Ozay, M.; Esnaola, I.; Vural, F.T.Y.; Kulkarni, S.R.; Poor, H.V. Machine learning methods for attack detection in the smart grid. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *27*, 1773–1786.
25. Petrillo, A.; Pescape, A.; Santini, S. A secure adaptive control for cooperative driving of autonomous connected vehicles in the presence of heterogeneous communication delays and cyberattacks. *IEEE Trans. Cybern.* **2020**, doi:10.1109/TCYB.2019.2962601.
26. Liu, X.; Datta, A.; Lim, E.P. *Computational Trust Models and Machine Learning*; CRC Press: Boca Raton, FL, USA, 2014.
27. Zaidi, K.; Milojevic, M.B.; Rakocevic, V.; Nallanathan, A.; Rajarajan, M. Host-based intrusion detection for vanets: A statistical approach to rogue node detection. *IEEE Trans. Veh. Technol.* **2015**, *65*, 6703–6714.
28. Ahmed, S.; Al-Rubeai, S.; Tepe, K. Novel Trust Framework for Vehicular Networks. *IEEE Trans. Veh. Technol.* **2017**, *66*, 9498–9511.
29. Zhang, H.; Huang, L.; Wu, C.Q.; Li, Z. An Effective Convolutional Neural Network Based on SMOTE and Gaussian Mixture Model for Intrusion Detection in Imbalanced Dataset. *Comput. Netw.* **2020**, *177*, 107315.
30. Li, M.; Wang, Z.; Luo, J.; Liu, Y.; Cai, S. Wavelet denoising of vehicle platform vibration signal based on threshold neural network. *Shock Vib.* **2017**, *2017*, 7962828.
31. Bezzina, D.; Sayer, J. *Safety Pilot Model Deployment: Test Conductor Team Report*; USDOT Report No. DOT HS 812 171; United State Department of Transportation: Washington, DC, DC, USA, June 2015.
32. Bishop, C.M. *Neural Networks for Pattern Recognition*; Oxford University Press: Oxford, UK, 1995.
33. Arangio, S.; Beck, J. Bayesian neural networks for bridge integrity assessment. *Struct. Control Health Monit.* **2012**, *19*, 3–21.
34. Gal, Y. *Uncertainty in Deep Learning*; University of Cambridge: Cambridge, UK, 2016.

35. Ghahramani, Z. A history of bayesian neural networks. In Proceedings of the NIPS Workshop on Bayesian Deep Learning, Barcelona, Spain, 10 December 2016.
36. Rodrigo, H.S. Bayesian Artificial Neural Networks in Health and Cybersecurity. Ph.D. Thesis, University of South Florida, Tempa, FL, USA, 2017.
37. Raya, M.; Papadimitratos, P.; Gligor, V.D.; Hubaux, J.P. On data-centric trust establishment in ephemeral ad hoc networks. In Proceedings of the IEEE INFOCOM 2008-The 27th Conference on Computer Communications, Phoenix, AZ, USA, 13–18 April 2008; pp. 1238–1246.
38. Goh, S.T. Machine Learning Approaches to Challenging Problems: Interpretable Imbalanced Classification, Interpretable Density Estimation, and Causal Inference. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2018.
39. Koyejo, O.O.; Natarajan, N.; Ravikumar, P.K.; Dhillon, I.S. Consistent binary classification with generalized performance metrics. In Proceedings of the Neural Information Processing Systems Conference (NIPS 2014), Montréal, QC, Canada, December 2014; pp. 2744–2752.
40. Silvestro, D.; Andermann, T. Prior choice affects ability of Bayesian neural networks to identify unknowns. *arXiv* **2020**, arXiv:2005.04987.
41. Maier, A.; Lorch, B.; Riess, C. Toward Reliable Models for Authenticating Multimedia Content: Detecting Resampling Artifacts With Bayesian Neural Networks. *arXiv* **2020**, arXiv:2007.14132.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).