International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

A Hybrid Cluster Based Intelligent IDS with Deep Belief Network to Improve the Security over Wireless Sensor Network

Dr. Priyanka R.[1], Teena K. B.[2], Rashmi T. V.[3], Dr. Reshma J.[4], Dr. Tejashwini Nagaraj[5], Tejaswini N.[6]

Submitted: 22/12/2023 **Revised**: 28/01/2024 **Accepted**: 08/02/2024

Abstract: Numerous inexpensive, compact devices compose a Wireless Sensor Network (WSN). They're usually readily available to some types of attacks due to their location, which is not well protected. A large number of researchers are focusing on WSN security at the moment. This kind of network is characterized by vulnerable characteristics, such as the ability to organize oneself without a stable infrastructure and open-air transmission. To train variables for the probability-based feature vectors, a Deep Neural Network (DNN) framework that is derived from international vehicle network packets shall be applied. The detector is capable of detecting any malicious attack on the vehicle since DNN gives each category a chance to distinguish between attacks and regular packets. Intrusion Detection Systems (IDS), can help to identify and stop security attacks on vehicles. The study proposes a mechanism for enhancing the security of WSNs based on Hybrid Clusters and Intelligent Intrusion Detection Systems with Deep Belief Networks (HCIIDS-DBN). It can provide a protection system for intrusions and an analysis of vehicle attacks in real time. They are designed based on their respective attack probability and ability, to the sensor node, sink, or cluster head. The proposed HCIIDS-DBN is composed of modules designed to detect anomalies and dereliction. The objective is to increase detection rates and decrease false positive incidences by detecting anomalies and abuse. Finally, the detected data are integrated and the various types of vehicle communication attacks are reported using the Decision Support System (DSS). The results of the experiment show that the proposed method may respond to the attack in real-time with a much detection of higher ratio in the Controller Area Network (CAN) bus.

Keywords*: Wireless Sensor Network; Hybrid Cluster Intelligent IDS; Deep Belief Network; Deep Learning; In-Vehicle security; Performance measures*

1. Introduction

The development of Sensor Nodes (SNs) and microelectronics has brought about miniaturized, cheap sensors that are available with features such as sensing, processing, or interaction at low power [1]. As a consequence, there is an increasing interest in research on issues related to WSN. A WSN is a network that is not infrastructure and is created by the widespread deployment of SNs. However, due to the limited power of sensors, it is only possible to reduce energy consumption using multiple communications among sensors [2]. WSN's primary function is to gather and store relevant data about a given environment, such as healthcare, military, business, or environmental protection. After detection of the target or surrounding area, SNs use wireless communications to transmit information to the sink. The information shall then be examined to determine the present status of the target. As a result of the hardware architecture, though, WSNs are limited in their resource capacity, including low processing power, less memory, and lower energy [3].

In Figures 1 and 2, respectively, dual of the most popular topologies of WSNs are the Cluster-based Wireless Sensor Network (CWSN) and the Flat-based Wireless Sensor Network (FWSN) are depicted. However, multi-hop communication is generating a lot of information and increasing the energy used in FWSNs such as SPIN [4]. CWSN is the most widely exploited network structure for WSN. Each SN in the CWSN is divided into clusters, and each cluster is run by an elected cluster leader, who is responsible for the operation of the cluster. CH should compile data from any SNs that have been identified on a specific target. Several protocols, including LEACH and APTEEN, have been proposed for

[1]*Assistant Professor, Department of Information Science & Engineering, Cambridge Institute of Technology, Bangalore, Affiliated to VTU, Belgaum, India**
priyanka.89.r@gmail.com
[2]*Assistant Professor, Department of Information Science & Engineering, East Point College of Engineering and Technology, Bangalore, India*
teena.k@eastpoint.ac.in
[3]*Assistant Professor, Department of Computer Science & Engineering, East Point College of Engineering and Technology, Bangalore India,*
rashmitv.harsha@gmail.com
[4]*Associate Professor, Department of Information Science & Engineering, Dayananda Sagar College of Engineering, Bangalore, India*
reshma-ise@dayanandasagar.edu
[5]*Associate Professor, Department of Computer Science & Engineering, Sai Vidya Institute of Technology, Bangalore, India*
tejashwini.n@gmail.com
[6]*Assistant Professor, Department of CS&E, JSS Science and Technology University, Mysore India*
tejaswinin@jssstuniv.in
**Corresponding author: priyanka.89.r@gmail.com*

the CWSN [5]. WSNs are the target of a wide range of attacks because they're composed of several inexpensive, small pieces of equipment that typically deploy into an unguarded area. In the case of using a WSN in combat, adversaries attack and destroy SNs [6]. As a result, consideration must be given to WSN security. A preventive technique shall be applied to deal with well-known attacks. Based on the characteristics of an attack, it develops a comparable defense strategy. Nevertheless, precautionary measures are at risk of being attacked in large numbers. Consequently, attacks have to be identified. The IDS is often utilized for identifying packets on the network and ascertaining whether they occur attackers. Moreover, IDS could assist in the development of a prevention system if it studies the acquired characteristics of attacks [7].

Integration with several computing devices, known as Electronic Control Units (ECUs), has recently made significant advances in automotive systems [8]. The ECU of an automobile is used for monitoring and controlling a subsystem that improves energy efficiency and reduces vibration and noise. The ECU replaces the traditional mechanical control components. Computer equipment for Vehicle-to Infrastructure (V2I) and Vehicle-to-Vehicle (V2V), which facilitates both intra and inter-vehicle communications, has become increasingly necessary in recent years [9]. The use of vehicle communications can be applied to many suitable transport schemes. It is proposed that this communication be used to learn driving behaviour, such as the speed and fuel consumption of each vehicle. A unique communication system for the transmission of messages has been created. Fuel efficiency shall be considered when setting the speeds or distances of connected vehicles [10]. Moreover, wireless communication can facilitate the cooperation of platoons and thus improve traffic flow. The most recent developments in realistic cooperative driving are highlighted by the best-performing results from the Grand Cooperative Driving Challenge (GCDC). Consequently, the vehicle's processing components have a great deal more power. Different communication protocols have been established to facilitate the exchange of information [11]. The most straightforward communication protocol for connecting sensors and actuators to ECUs is CAN, which is also the de facto model for in-vehicle network communication. New automotive applications are encouraged to be developed through the adoption of CAN [12].

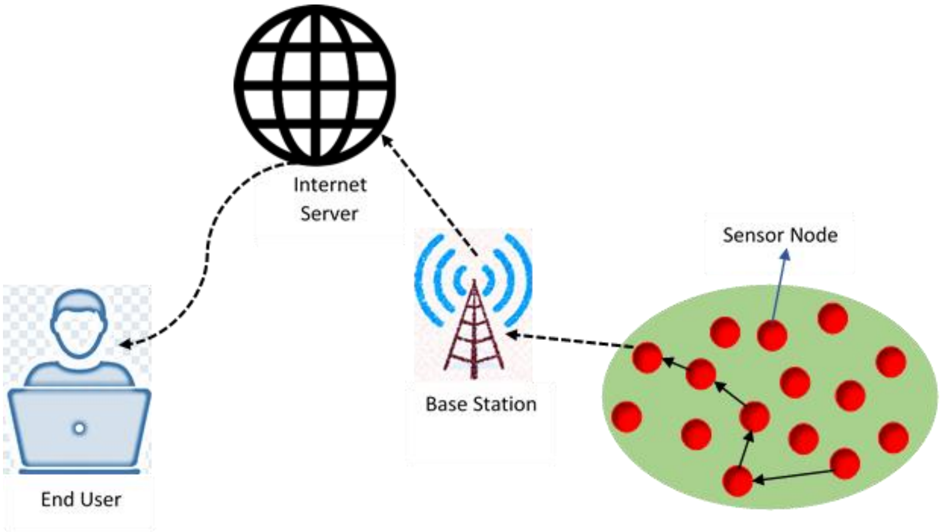


Fig 1: Flat WSN

The protection of drivers should be ensured by the confidentiality of information. Unfortunately, there are several safety vulnerabilities in the automotive network and networking capacity is growing with serious security concerns. ECUs can't determine the sender of the broadcast messages, but they may receive any message from another ECU-to-ECU on the same bus [13]. It illustrates the potential for misunderstanding of critical components to protect drivers' safety caused by attempted attacks, such as data manipulation and packet injection. Several studies were performed on safety issues associated with intra and intervehicular communications. A significant focus is placed on the effectiveness and ease of use of IDS sensors [14]. An intrusion detection method is proposed by applying a series of different attack patterns that have been arranged in the dataset. Developed a description Strategy to compare the performance of the standard system in practice with selected patterns. To identify an undesirable intrusion, this detection technique relies upon numerous sensors engineered to deal with some attack situations. Safe procedures that conform to the standard criteria are proposed [15].

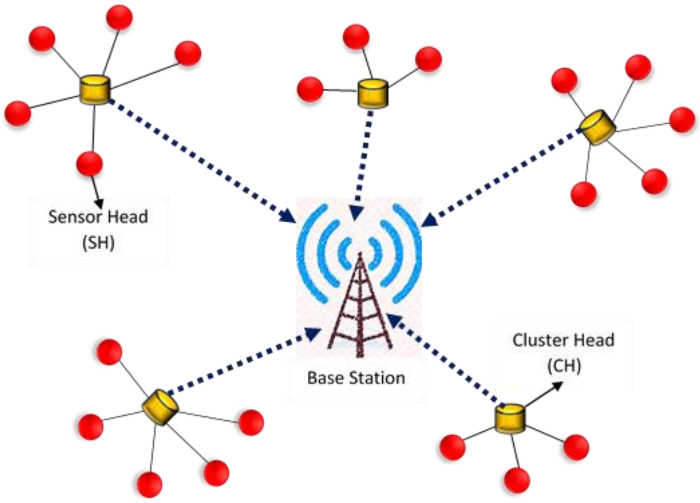
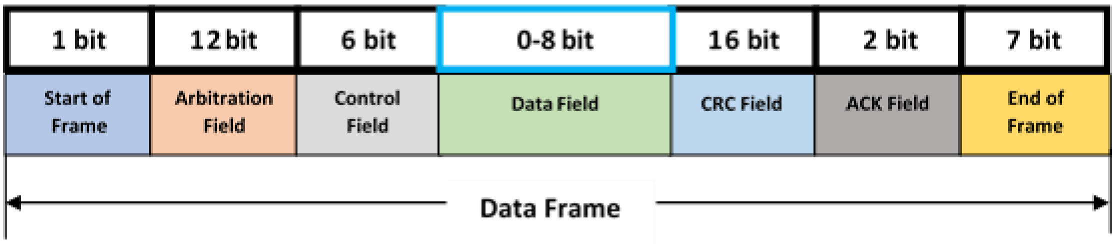


Fig 2: Cluster Based WSN

WSN has become more and more important as an area of study in recent years. Due to their various characteristics such as low power consumption, energy limitation, radio frequency usage, and so on, mobile sensor networks may be subject to many security risks. In case of an attacker's presence in the network, encryption, which is generally regarded as one of the primary security lines, shall no longer be effective [16]. The identification and suppression of both external and internal threats are facilitated by IDS, sometimes referred to as the 2nd line of defense. Direct installation of IDSs that are built for ad hoc networks or wired in a WSN is not possible. This necessitates the development of a special detection system for cellular sensor networks, which takes account of their constraints [17].

2. Related Works

With a maximum communication speed of 1 Mbps, CAN is intended to be used for high-speed, half-duplex broadcast bus networks in the rail network. The packet is obtained by a receiver ECU, which recognizes the ID of the sender [14]. The unique ID number of the sender ECU may be provided in the package. As a result, the CAN packet doesn’t exclude an explicit destination field. The syntax of the CAN data packet can be found in Figure 3. There is an arbitration field including 11 bits of identification, each with a distinct ECU. There are 2 features to the arbitration field: 1) allowing each ECU to filter out a strange message, and 2) ranking messages according to their ID in descending order [15]. The area of data may contain as many as 8 bits of information that can be sent in the message. The angle of the on-and-off status and the steering wheel of the display panel elements are examples of this type of information. The data field size is set in a control field. Any mistakes in the data packet are found using the Cyclic Redundancy Check (CRC) field. It is confirmed that valid CAN packets were received by the acknowledgment field [16].



1 bit	12 bit	6 bit	0-8 bit	16 bit	2 bit	7 bit
Start of Frame	Arbitration Field	Control Field	Data Field	CRC Field	ACK Field	End of Frame

Fig 3: CAN packet syntax

Intensive research is being undertaken into intrusion detection techniques to strengthen the defense of conventional networks against malicious attacks. Numerous intrusion detection techniques have been developed in the literature using machine learning techniques, presuming that attack packet patterns are different from regular packet patterns. The Artificial Neural Networks (ANN) and Support Vector Machines (SVM) codes use a radio frequency encoding technique which can be used to determine the characteristics of packets [17]. As the abovementioned work relies on supervised Machine Learning techniques, several labeled data sets are required to be used during training. In the detection of network intrusions, for example by Self-Organised Maps (SOM) an autonomous machine learning algorithm is used in contrast to conventional methods [18]. In Figure 4, we can see a model of an IDS architecture based on machine learning. The monitoring unit is usually able to determine the category of incoming packets following feature extraction. The profiling unit contains features that can be manually trained from the Internet. In case of discovery by the Monitoring unit of the latest type of attack, the profiling unit may update its database for future packets [19].

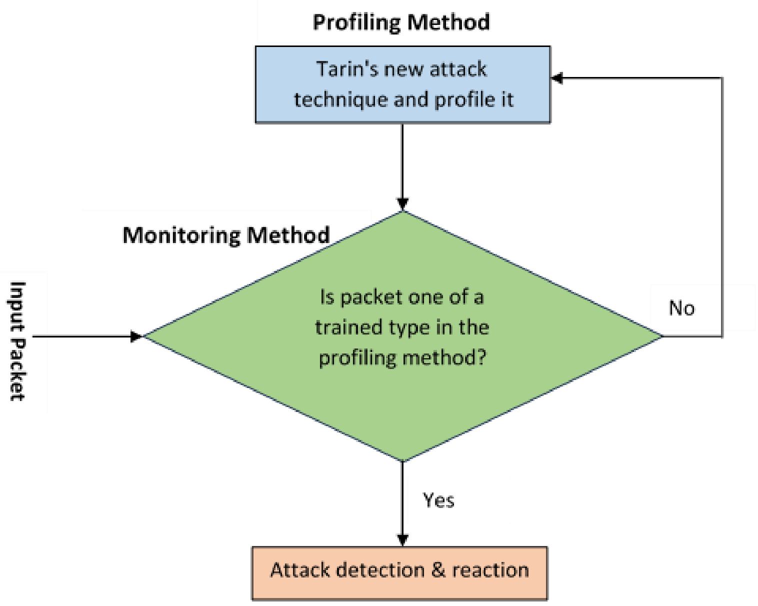

Profiling Method
Tarin's new attack technique and profile it
Monitoring Method
Input Packet
Is packet one of a trained type in the profiling method?
No
Yes
Attack detection & reaction

Fig 4: IDS based on machine learning method architecture

Deep learning is a machine learning approach, which takes advantage of the architecture with several hierarchical levels for non-linear processing. The two types of architecture that could be distinguished based on how they are used are generative deep architecture and discriminative deep architecture. The DNN or deep structure can include many hidden layers from an ANN structure [20]. It is proposed to build a cluster-based hierarchical IDS. Using this method, the authors installed an IDS agent (core defense) on every cluster head. The agent has 3 modules: Decision Making, a supervised learning component, and an anomaly detection module based on the rules [21]. To achieve this detection method, a large number of calculations needed for cluster heads may shorten the network's lifetime. It proposed a lightweight, hybrid IDS that is integrated into sensor networks based on the proposed model. IDS use cluster protocols for the creation of a hierarchy network and offer an intrusion architecture derived from abuse mechanisms and anomaly models [22]. The IDS agent is composed of two detection modules, global and local, in their system. The authors use their model in a procedure of cooperation between the two agents where they are both located within one node, to detect an attack more accurately. However, this system is a disadvantage in that it overloads the node's memory due to an excessive number of signatures [23].

According to these hybrid models. We present a proposal for an efficient light sensor network IDS in this paper. This research aims to explore and apply a new intrusion detection model in an environment of wireless sensor clusters that incorporates the advantages of both anomaly-relying models, which exhibit superior performance compared with conventional hybrid models proposed in the literature, and those derived from signature models [24].

3. Proposed System

A common scenario for which an attacker will attack the CAN bus within a vehicle with unauthorized packets of information shall be considered in the recommended IDS. In-vehicle networks can be accessed via cellular data links such as 3G, 4G, or WIFI and the OBD diagnostics tools which are integrated with a driver's wireless device.

3.1 Data Collection

In total, around 200,000 packets are created in the simulation. To avoid over-fitting issues, researchers divided the packets into 70% learning and 30% testing data. To confuse the system, a proportion of packets are edited and inserted in an attack scenario. To avoid a breakdown of the in-vehicle network, it is important to note that assault packets are injected at certain intervals shown in Figure 5. Figure 6 displays the CAN data packets for experimental control ECUs, including an identifier and a field of data. To avoid over-fitting issues, researchers divided the packets into 70% learning and 30% testing data. To confuse the system, a proportion of packets are edited and inserted in an attack scenario. To avoid a breakdown of the in-vehicle network, it is important to note that assault packets are injected at certain intervals.

3.2 Proposed technique

The proposed HCIIDS-DBN is capable of detecting an attack by observing CAN communications packets on the bus, as shown in Figure 7. Our architecture consists of two main stages: the detection and training stage, as shown in Figure 8, an existing IDS based on machine learning. It is being carried out offline in this phase because of the long duration of training. There's a binary label on every CAN training packet in supervised learning, meaning it's either an attack or a standard packet. As a result, it is expected that the information on the label will be communicated using matching features. The adoption of a DNN framework for feature training allows us to gain weight parameters at edges linking nodes. The detection phase is also depicted in Figure 8.

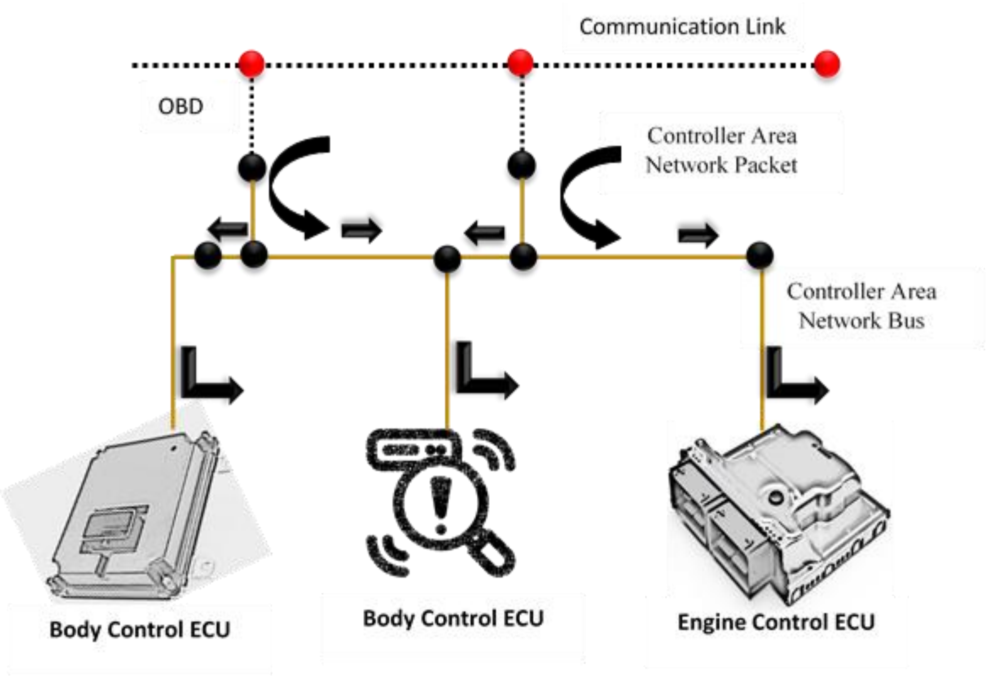


Fig 5: Simulation configuration

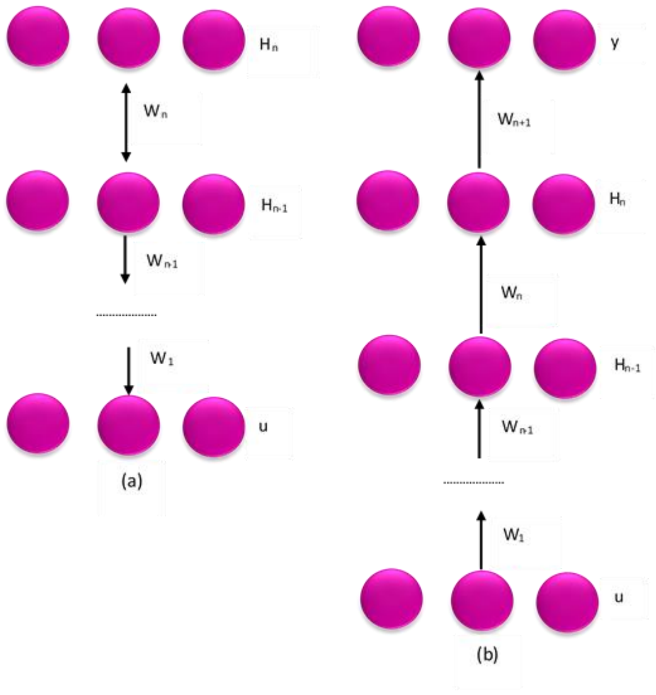


Fig 6: (a) Top-down with n hidden layers DBN; (b) Bottom-up with n hidden layers DNN with Pretrained layers

The learning structure would be created of supervised learning, as the HCIIDS-DBN framework gives rise to an unobserved learning mechanism in Figure 6 (a). For this purpose, a discriminative deep learning structure is created by appending the last classification layer to the top level of the HCIIDS-DBN framework and including label information. The altered structure is presented in Figure 6 (b) a deep feed forward structure of the ANN. The correction of parameters is performed with a gradient descent approach within the deep feed forward ANN structure as soon as they have been initialized for weights.

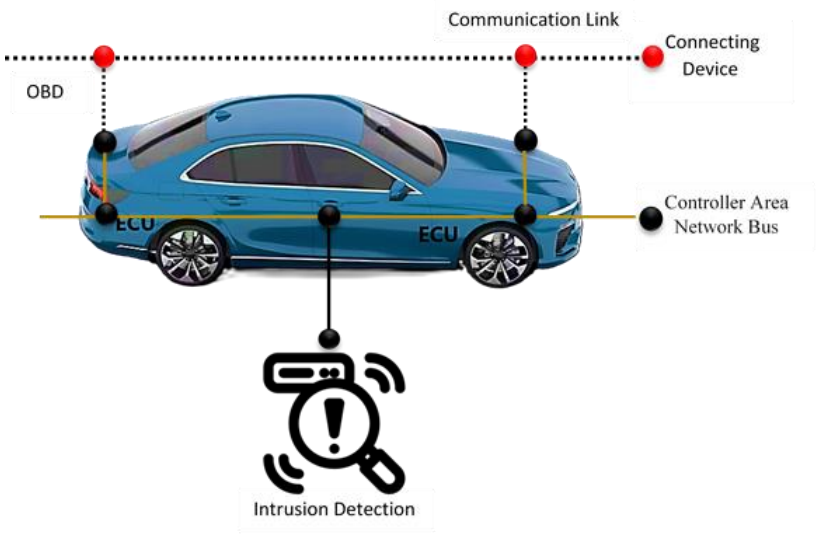


Fig 7: Attack scenario in the connected car

A method to detect intrusions built into the CWSN is proposed in this study. In the case of sinks, CHs, and SNs, three separate IDS are established based on a variety of capacities and attack probability. It is proposed that an IDS be misused for SN, a HCIIDS-DBN for CH, and an HCIIDS-DBN for the sink. For HCIIDS-DBN to be trained in the new attacks that HCIIDS-DBN has identified and classified, a feedback system operates at each sink and on CH.

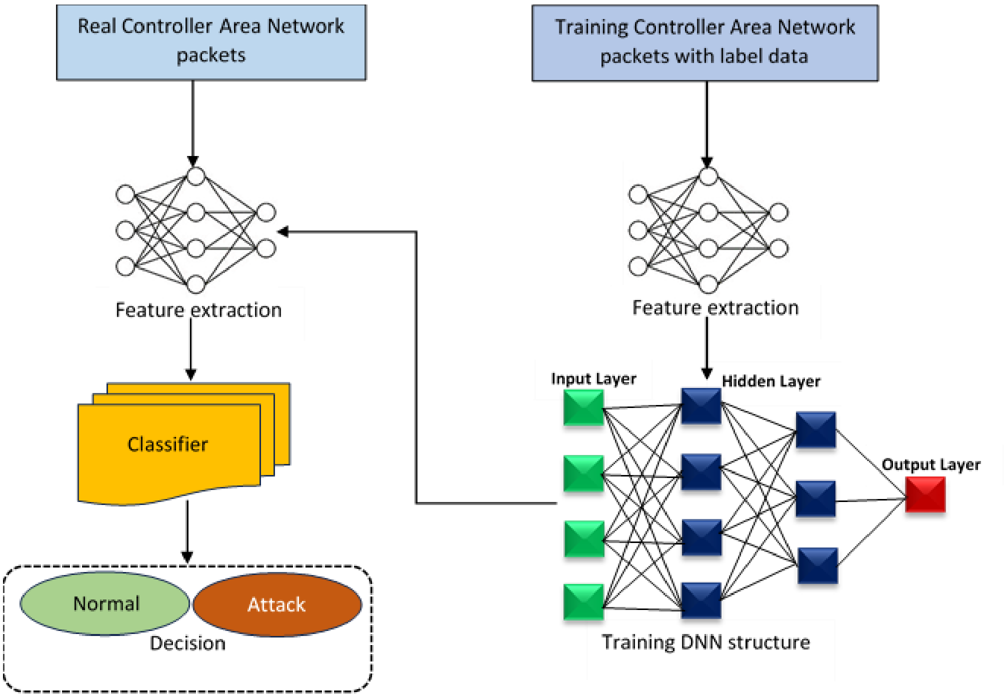


Fig 8: Overview architecture of the proposed HCIIDS-DBN

An HCIIDS-DBN for CWSN is created overall. In real-time, it could offer effective defense against invaders and analysis of attacks. Application of techniques that need greater computational power and more energy could not be achieved due to limited resources for general supernovae. An immense source of the sink, which links anomaly and abuse detection, is used in this research to design an HCIIDS-DBN. The proposed HCIIDS-DBN is capable of learning in real-time and adding new classes if attacked by unexpected threats, alongside having a low false positive rate and a very high detection rate. The HCIIDS-DBN identified in this analysis are composed of 4 models, as shown in Figure 8. The DSS will combine data from both sensors to determine the nature and extent of any breach before sending this information back to management for further action shown in Figure 9.

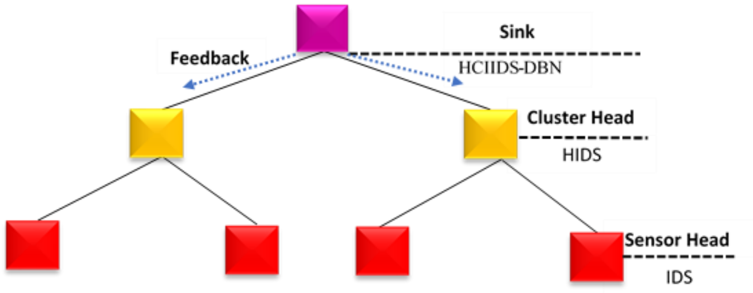


Fig 9: Proposed Architecture

3.3 Anomaly detection model

The anomaly detection model operates the same way as a filter. To be further detected, the misuse detection model will receive abnormal packets. If the present behavior deviates from the normal behavior model, the system will classify the packet as abnormal, since the anomaly detection is based on that model. Consequently, it is of concern that anomaly detections may classify regular communications as abnormal and thereby lead to errors in classification. However, it rarely defines abnormal communication as normal. Many packets on CWSN should be recognized, a majority of which are common. The anomaly detection method will be used as the starting filter and screening packet. In the event of an abnormal packet being detected, a misuse detection module shall be used to detect additional suspicious packets. An anomaly detection module has detected an intrusion to compare the present behavior with the expected behaviour as shown in Figure 10. If the existing behavior is contrary to normal behavior patterns, it would not be appropriate for the system to be classified as an anomaly. To be able to monitor CWSN packet status, packets have to create a regular pattern of behavior. In these cases, the necessary rules are defined by professionals and anomaly detection modules have been developed in this study with a Rule-Based Analysis method.

The process of construction may be separated into the following three steps:

Step 1: The packets delivered from CH's neighbor are the ones that go through the sink in CWSN. Therefore, the analysis of all prior packets that have communicated to the tank will be carried out and each packet type shall be classified as either normal or irregular.

Step 2: Choose a feature. Search for the essential characteristics assigned to a particular package so that it can be differentiated into normal and irregular packages.

Step 3: The development of guidelines for the detection of anomalies. Attributes chosen the specifications of a standard pack are used to develop these rules.

Any packet that runs into the sink when all CHs communicate with each other shall be filtered by an anomaly detection module in CWSN to see if any abnormal packets are present. As the misuse detection module relies on many well-known attack behavior models, it is appropriate to design model foundations based on those behaviors. Since most intruder detection techniques guarantee performance based on training data, this study uses BPN with a supervised learning method. When BPN learns an appropriate relationship between input and output variables, its weight is increased accordingly. This can reduce the error of inferences to a great extent, to achieve an accurate level of precision. Therefore, BPN can achieve perfect accuracy by HCIIDS-DBN based on extensive training. This study uses three layers of HCIIDS-DBN which consist of an input layer, the concealed layer, and the output layer. Figure 11 depicts the abuse detection model's structure. The input vector we use is an anomalous packet, which has been identified by the anomaly detection module. The maximum number of processors in the input layer shall be defined by the characteristics selected for packets. Moreover, to get a hidden layer's number of processing units, input and output layer unit values shall be averaged. After examination, eight prevalent assaults were identified in the CWSN: Sinkhole, Select Forward, Hello Floods, Sybil Attack, Acknowledgment Spoofing, Denial of Service, and Spoofed/Altered/Replayed Routing Information.

Consequently, there is no balance in the data on training. On the other hand, HCIIDS-DBN will ignore anomalous packets owing to their low occurrence rate. To prevent this, after training data have been processed in the anomaly detection module, the abnormal packet that has been taken for training shall be segregated. To verify the malicious behaviour of the target that has been detected using anomaly detection, a set of attacks characterized by predefined rule signatures and anomalies based on SVM is used in this proposed approach. Cluster-based architectures incorporate detection technology to prolong a network's lifetime. This is accomplished by appointing one recognized node as the CH, which transmits packets (aggregated data) from the nodes to the BS rather than the node's collected data being sent to the base station that is located in a faraway place.

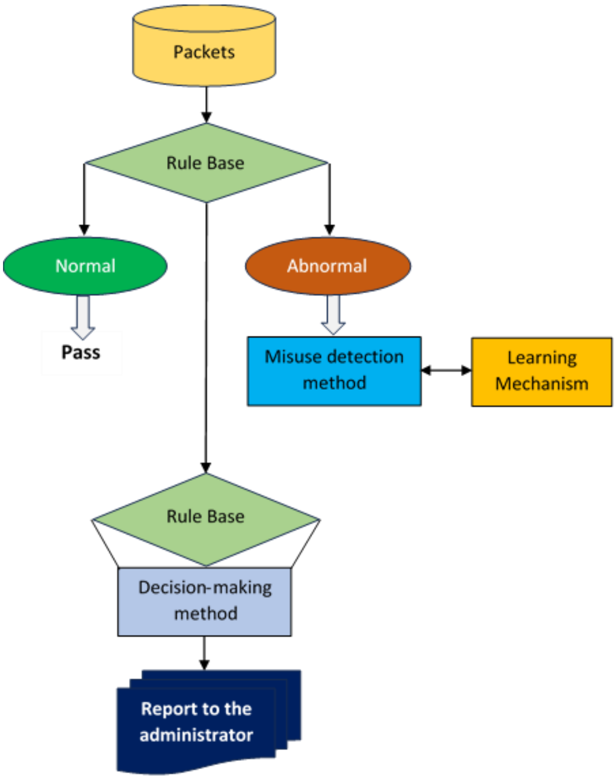


Fig 10: HCIIDS-DBN System architecture

CHs are sensors for local base stations, and groups have a specific probability of becoming CHs at any time. A single cluster is composed of all nodes in this architecture, which are geographically distributed throughout the network. The CH is aimed at prolonging the network's life and reducing its energy consumption. Consideration will be given to the circumstances of the bit symbol symbols in a data packet. Specifically, we select the CAN syntax's 64-bit locations (equivalent to 8 bytes) in the DATA field, and we examine the bit-symbol statistical distributions. A mathematical representation of the data vector PO2 R64 is possible. Any bit locations in the DATA field can be used to generate a characteristic. However, it is possible to reduce the dimension of certain semantic elements in a related syntactic element.

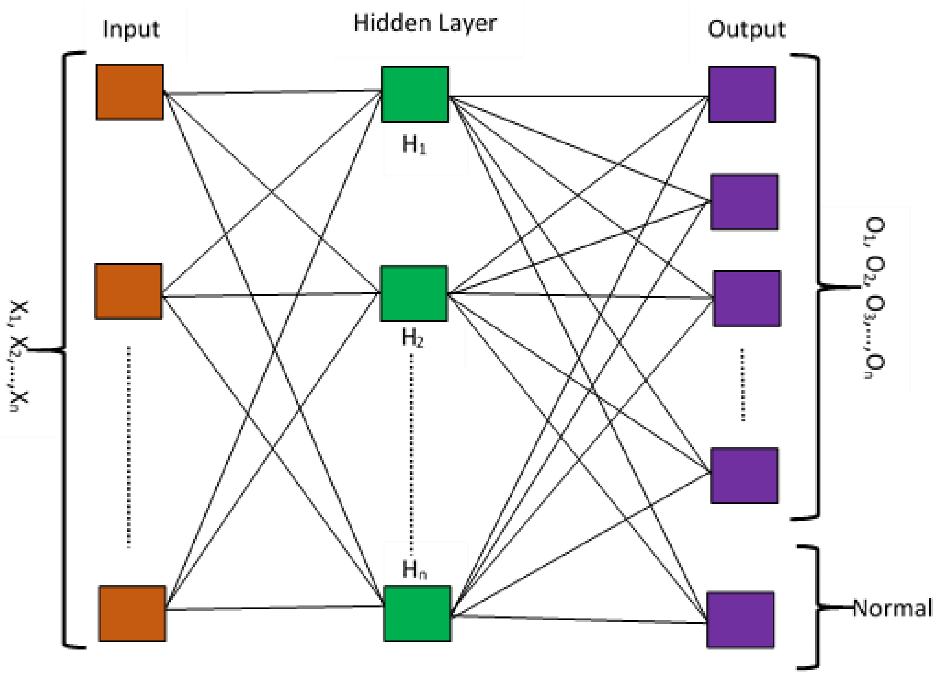


Fig 11: HCIIDS-DBN misuse detection method

4 Results and Discussions

4.1 Performance measures

The computer used in this study is an AMD AthlonTM 64 X2 Dual Core Processor 5000+ 2.59GHz and has 2048MB of RAM running Windows XP Professional. The NN tool, which is part of MATLAB 7.1, allows the BPN method to be trained. The experiment's effectiveness is assessed using the following Equations: (1)–(3), which determine the accuracy and False Positive Rate (FPR), and Detection Rate (DR).

$$Accuracy = \frac{No.of\ detected\ attacks}{.of\ attacks} \times 100\% \quad (1)$$

$$False\ Positive\ Rate = \frac{No.of\ misclassified\ connections}{No.\ of\ normal\ connections} \times 100\% \quad (2)$$

$$Detection\ Rate = \frac{No.of\ correct\ classified\ connections}{No.of\ connections} \times 100\% \quad (3)$$

Table 1: Categorization detail

Attacks Category	No. of the amount of sample/ correct detection	DR (%)
Normal	2511/2573	98.95
Probe	1672/2155	77.42
DoS	9742/9896	98.54
U2R	5/27	15.78
R2L	56/569	9.92

Table 2: Threshold accuracy

Value of threshold	Amount of sample/ Correct outuput	DR (%)
0.9	14612/5328	93.55
0.95	14589/1527	94.07
0.98	14576/1556	94.46
0.99	14467/1510	96.15
0.997	12024/1223	98.39

The results of the test show that the accuracy is 91.26%, the FP is 2.06%, and the DR is 90.96%, as shown in Table 1. Examining each assault class displayed in Table 2 and each efficiency, it is evident that Probe, U2R, and R2L have poor DR figures—even 9.77% R2L's and 15.38% U2R's. It is due to the lack of training data for R2L or U2R, which makes it difficult to detect them. In addition, if BPN were incapable of identifying several new attacks that are present in these classes it would have misclassified those attacks. This means that our models need to be continually updated during learning to maintain a large data rate for Integrated Data Streams shown in Table 3.

Table 3: Simulation packets of CAN

ID	Data Field	ECU Target
10F	$02\ \alpha_0\beta_0\ A0\ B2\ \alpha_1\beta_1\ \alpha_2\beta_2\ \alpha_3\beta_3\ \alpha_4\beta_4$	Engine
24F	$33\ \alpha_0\beta_0\ B1\ A3\ \alpha_1\beta_1\ \alpha_2\beta_2\ \alpha_3\beta_3\ \alpha_4\beta_4$	Body Control
400	$00\ \alpha_0\beta_0\ EF\ 10\ \alpha_1\beta_1\ \alpha_2\beta_2\ \alpha_3\beta_3\ \alpha_4\beta_4$	Display Panel

As seen in Figure 12, the value data indicates the value of the mode, such as the speed or the wheel angle, and the mode data denotes the command state of an ECU, such as controlling wheels. While noise may distort the values information, for a brief period mode data remain unchanged. The value data is only utilized as part of the training phase for the proposed method. During the detection stage, the use of mode data shall be demonstrated.

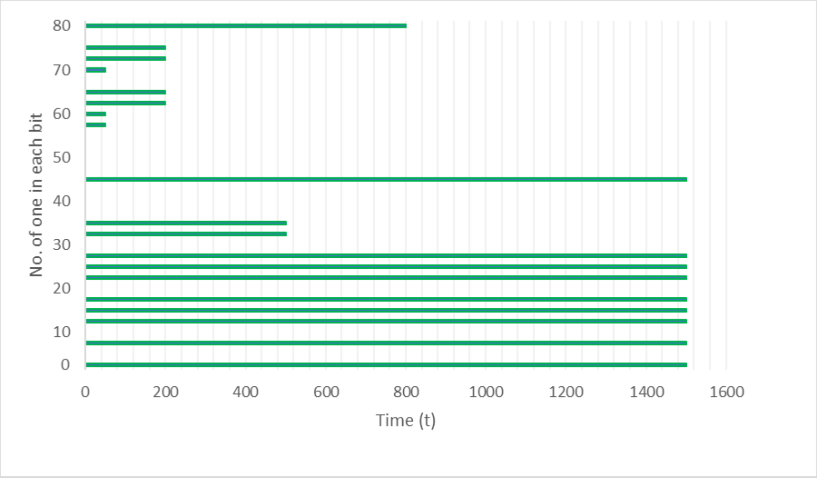


Fig 12: The instances of the bit-symbol "1" at time t in the 8-byte Data field, which contains both value and mode data

$$C(Wi) = \frac{1}{K}\sum_k C(Wi; u^k, j^k) + \frac{\lambda}{2}\sum_n^N \sum_x^{m_l} \sum_y^{m_{l+1}} \left(Wi_{yx}^{xt}\right)^2$$ (4)

Where i- Neural Network depth, m_l – nodes in ith layer, and $Wi_{yx}^{xt} \in Wi$ - weight edge between node y and node x. Optimal parameter set represented as Wi* is shown in Equation (5).

$$Wi^* = \arg\min_{Wi} C(Wi) \quad (5)$$

$$Wi_{yx}^{xt} = Wi_{yx}^{xt-1} + \xi \frac{\partial}{\partial Wi_{yx}^{xt-1}} C(Wi) \quad (6)$$

The class of the test CAN packet is expected to be detected during the detection phase. To calculate an output, the set of features taken from the test CAN packet and a set of trained weighting variables are used, just as they were used for training. The classification algorithm returns a logistic value of 0 or 1 so that it can determine whether the sample is an attack packet or not. An ECU may be able to draw on many attack situations, and its weight vectors can be adapted for each situation. To determine the situation to enable an appropriate training set to be applied, mode information shall be considered in the proposed strategy. To this end, template matching shall be provided in the proposed method shown in Figure 13. A template containing information on the mode shall be drawn up by reference to data and training samples applied for the specific situation. In Figure 14, find a template-matching example that uses the YellowfinColor template. As demonstrated, if the template matches a CAN packet and training sample that must be assessed, the detector will use matched trained parameters derived from value data.

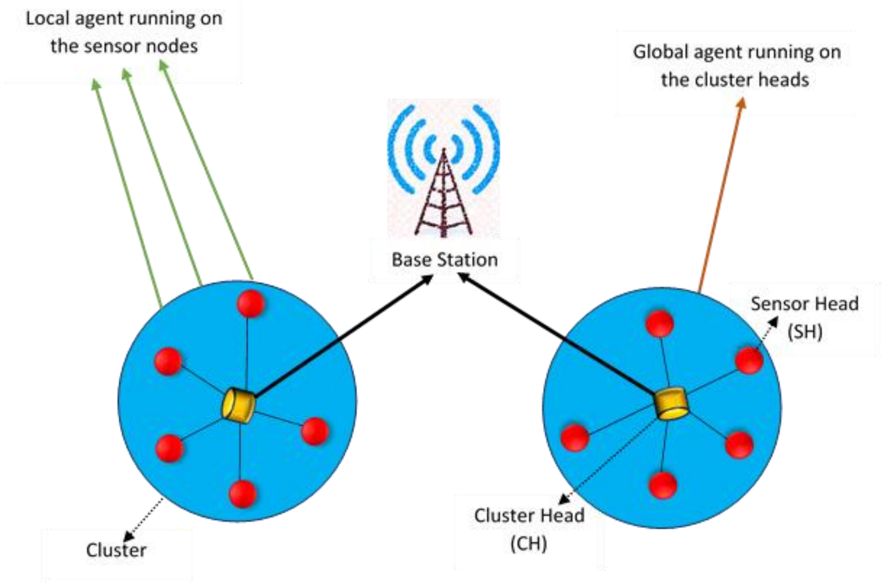

Local agent running on the sensor nodes
Global agent running on the cluster heads
Base Station
Sensor Head (SH)
Cluster
Cluster Head (CH)

Fig 13: Strategy location of HCIIDS-DBN

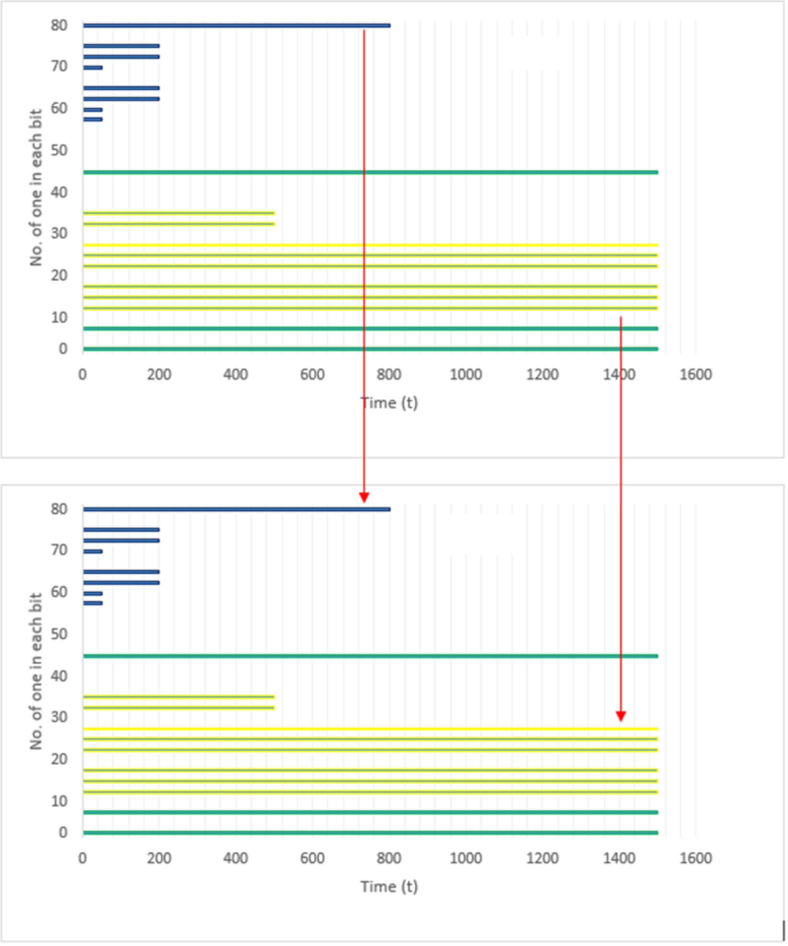


Fig 14: Determining the appropriate training parameters using the template matching method

According to its energy, CH will be chosen dynamically. The CH election procedure is announced by the BS, and the CHs determine the residual energy using the formula Vi(t) = [Initial – Re(t)] / s, in which Initial denotes the initial energy, Re(t) is the residual energy, and s is the CH selection round number eight. The BS shall compute the mean deviation and average, based on the values obtained. The CH election process of nodes is announced by CH. The message regarding the transfer of power will be sent by Old CH. Alert messages from a new CH are received by sensor nodes. The BS oversees CH authentication, whereas CH is in charge of the other cluster members' authentication. The individual agents are operated only where necessary due to the limited resources and battery life.

Local agent: The monitoring of sensor data that is sent or received shall be performed by the local agent module. In its internal database, the node is keeping a close eye on attacks of specific hostile network nodes. When the network was first created, sensor nodes didn't even know that a rogue node existed. After the WSNs have been deployed, their signature database will be built up. A CH makes this item in a malicious node database and distributes it to all the nodes.

Global agent: The global agent is monitoring the communications between the neighboring nodes. Due to the broadcasting nature of wireless networks, it can receive all packets passing through the radio range of the node. It must know the details of its nearest nodes for a global agent to monitor packets. We shall use the preconfigured rules and a Local Monitoring System to keep an eye on these packets. Each detector node shall generate and communicate to the Community Health Authorities every potential security breach detected at one of its neighboring nodes. When the alert has been received, CHs use the X threshold to determine if a node is suspicious.

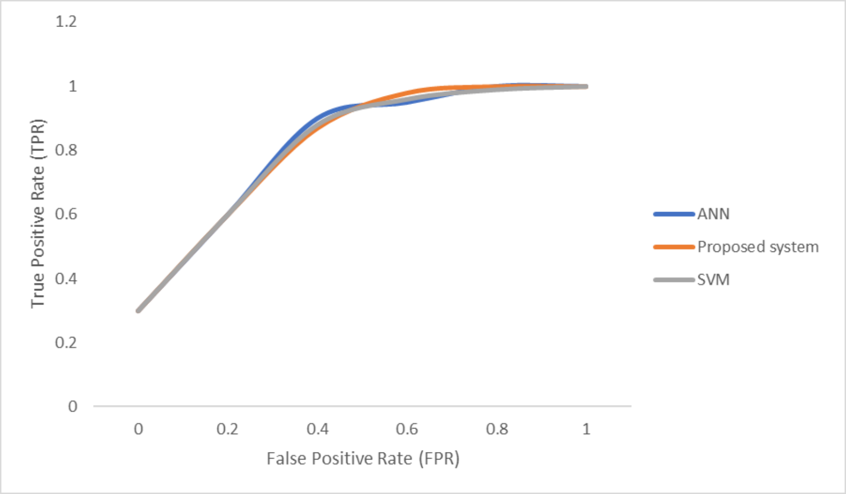
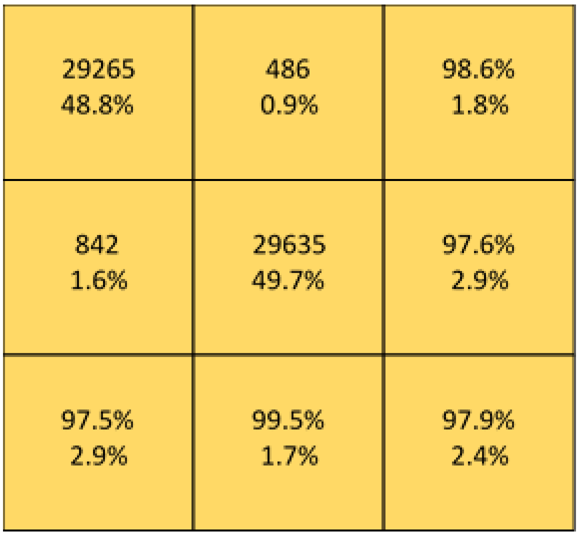


Fig 15: Evaluation of the performance of intrusion detection with ROC curves



29265 48.8%	486 0.9%	98.6% 1.8%
842 1.6%	29635 49.7%	97.6% 2.9%
97.5% 2.9%	99.5% 1.7%	97.9% 2.4%

Fig 16: Confusion Matrix Outcomes

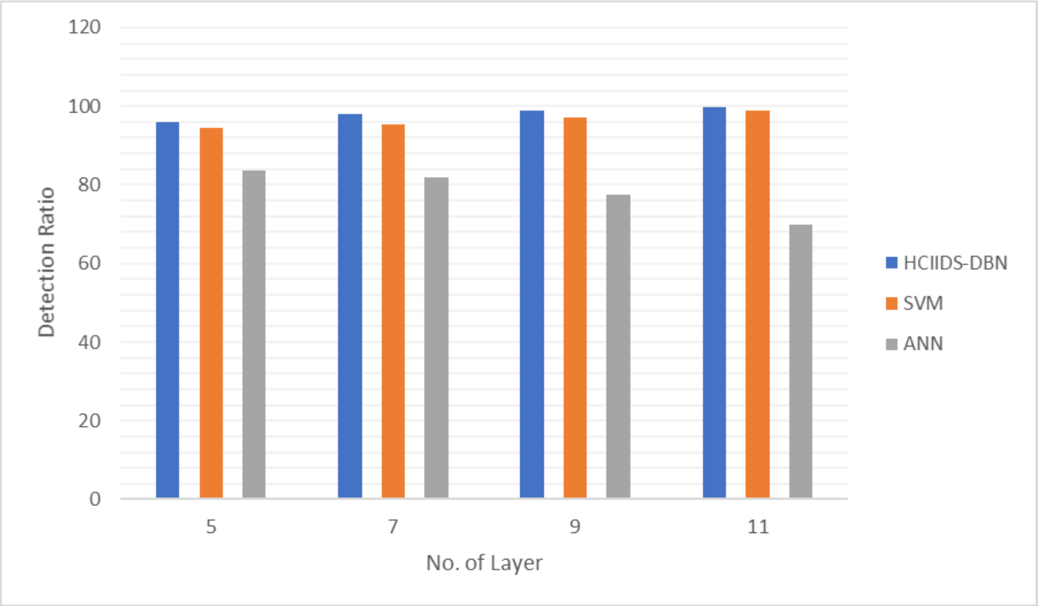


Fig 17: The HCIIDS-DBN performance based on no. of layer

A ROC curve that plots more points at the top, leftmost corner is superior for detection. In Figure 15 the ROC curves for the proposed method and an ANN which were used in the tests demonstrated that the proposed technique has a detection ratio superior to conventional methods. When the FPR is less than 1-2%, the detection ratio exceeds 99%. To evaluate the performance of quantitative detection, coherence matrices are shown in Figure 16. The proposed technique's performance offers a noticeably high detection rate. False positives make up just 1.6 percent of errors, while 2.8% are false negatives. This is an overall accuracy of approximately 97.8%. Two differences in the recommended DL structure utilizing the DNN framework are compared between several layers of the conventional feedforward artificial neural network and the intrusion detection efficiency. The recommended approach outperforms the feed-forward ANN in terms of detection effectiveness $\frac{(RA+RN)}{2}$ of two scenarios, as illustrated in Figure 17. The vanishing gradient issue with the ANN structure results in inconsistent results as the number of layers increases.

Table 4: Time complexity

Layer's	Testing(s)	Testing	
		Classification(ms)	Feature extraction(μs)
5	5.11	3.06	9.5
7	7.28	3.28	9.6
9	9.85	4.18	9.8
11	10.98	4.77	9.8

Furthermore, Table 4 shows the complexity of identification times which is different according to the number of concealed layers. The measurement time required to examine each, and every packet sent over a network is reflected in the evaluation time and the training time is reflected in the measurement time required to train the structure of DNN during the training stage. It is best to conduct training in the offline environment as it involves time complexity ranging from 4 to 11 seconds. Nevertheless, the testing time complexity for packet inspection only takes 8–9 μs to process each packet's features and 2–5 ms to classify the packets—a time complexity that can be used for real-time applications.

5. Conclusion

The three distinct IDSs of the sink, SH, and CN are engineered based on the diverse threats and likelihoods they face. An HCIIDS-DBN with learning capabilities is proposed for the sink; when the sink is subjected to unknown attacks, it not only studies and improves new classes through learning mechanisms in actual time, but also lowers the threat of assault in the system. It aims at effective detection of attacks and the prevention of waste of resources. By contrast, when HCIIDS-DBN is updating its attack class, it will use a feedback mechanism from the CH to the sink. In the case of SNs, an IDS for misuse is proposed. A quick and easy way to manage SN is being developed to save resources and avoid overwork for security. For the protection of the in-vehicle network, they proposed a powerful HCIIDS-DBN. DNN allows each class to distinguish between legitimate and malicious packets so that a vehicle security system will be able to detect an attack of some kind. In addition, we have proposed a new vector graphic feature that is capable of effectively serving as both training and testing equipment. They consist of information received from network packets on mode and value. The findings showed that the proposed methodology could react to an attack promptly and have a significant reliable identification ratio of about 99.8% when computational complexity related to multiple layers is very low.

References

[1] Bediya, A. K., & Kumar, R. (2023). A novel intrusion detection system for internet of things network security. In *Research Anthology on Convergence of Blockchain, Internet of Things, and Security* (pp. 330-348). IGI Global.

[2] He, K., Kim, D. D., & Asghar, M. R. (2023). Adversarial machine learning for network intrusion detection systems: a comprehensive survey. *IEEE Communications Surveys & Tutorials*.

[3] Abdulganiyu, O. H., Ait Tchakoucht, T., & Saheed, Y. K. (2023). A systematic literature review for network intrusion detection system (IDS). *International Journal of Information Security*, 1-38.

[4] Yi, L., Yin, M., & Darbandi, M. (2023). A deep and systematic review of the intrusion detection systems in the fog environment. *Transactions on Emerging Telecommunications Technologies, 34*(1), e4632.

[5] Sivanantham, S., Mohanraj, V., Suresh, Y., & Senthilkumar, J. (2023). Association Rule Mining Frequent-Pattern-Based Intrusion Detection in Network. *Computer Systems Science & Engineering, 44*(2).

[6] Talukder, M. A., Hasan, K. F., Islam, M. M., Uddin, M. A., Akhter, A., Yousuf, M. A., ... &

Moni, M. A. (2023). A dependable hybrid machine learning model for network intrusion detection. *Journal of Information Security and Applications*, *72*, 103405.

[7] Sharma, B., Sharma, L., Lal, C., & Roy, S. (2023). Anomaly based network intrusion detection for IoT attacks using deep learning technique. *Computers and Electrical Engineering*, *107*, 108626.

[8] Mohy-eddine, M., Guezzaz, A., Benkirane, S., & Azrour, M. (2023). An efficient network intrusion detection model for IoT security using K-NN classifier and feature selection. *Multimedia Tools and Applications*, 1-19.

[9] Hnamte, V., & Hussain, J. (2023). DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system. *Telematics and Informatics Reports*, *10*, 100053.

[10] Okey, O. D., Melgarejo, D. C., Saadi, M., Rosa, R. L., Kleinschmidt, J. H., & Rodríguez, D. Z. (2023). Transfer learning approach to IDS on cloud IoT devices using optimized CNN. *IEEE Access*, *11*, 1023-1038.

[11] Ennaji, S., El Akkad, N., & Haddouch, K. (2023). i-2NIDS Novel Intelligent Intrusion Detection Approach for a Strong Network Security. *International Journal of Information Security and Privacy (IJISP)*, *17*(1), 1-17.

[12] Lilhore, U. K., Manoharan, P., Simaiya, S., Alroobaea, R., Alsafyani, M., Baqasah, A. M., ... & Raahemifar, K. (2023). HIDM: Hybrid Intrusion Detection Model for Industry 4.0 Networks Using an Optimized CNN-LSTM with Transfer Learning. *Sensors*, *23*(18), 7856.

[13] Cui, J., Sun, H., Zhong, H., Zhang, J., Wei, L., Bolodurina, I., & He, D. (2023). Collaborative Intrusion Detection System for SDVN: A Fairness Federated Deep Learning Approach. *IEEE Transactions on Parallel and Distributed Systems*.

[14] Du, J., Yang, K., Hu, Y., & Jiang, L. (2023). Nids-cnnlstm: Network intrusion detection classification model based on deep learning. *IEEE Access*, *11*, 24808-24821.

[15] Shaorong, W., & Guiling, L. (2023). Research on campus network security protection system framework based on cloud data and intrusion detection algorithm. *Soft Computing*, 1-10.

[16] Sousa, B., Magaia, N., & Silva, S. (2023). An Intelligent Intrusion Detection System for 5G-Enabled Internet of Vehicles. *Electronics*, *12*(8), 1757.

[17] Huang, Y., & Ma, M. (2023). Ill-ids: An incremental lifetime learning ids for vanets. *Computers & Security*, *124*, 102992.

[18] Sood, K., Nosouhi, M. R., Nguyen, D. D. N., Jiang, F., Chowdhury, M., & Doss, R. (2023). Intrusion detection scheme with dimensionality reduction in next generation networks. *IEEE Transactions on Information Forensics and Security*, *18*, 965-979.

[19] de Carvalho Bertoli, G., Junior, L. A. P., Saotome, O., & dos Santos, A. L. (2023). Generalizing intrusion detection for heterogeneous networks: A stacked-unsupervised federated learning approach. *Computers & Security*, *127*, 103106.

[20] Putri, A. A., Agustina, C., Fauzan, H., Saputra, M. R. E., Erdiansyah, M., & Wardani, P. S. (2023). Network security implementation with snort-based intrusion detection system using windows 10. *JComce-Journal of Computer Science*, *1*(1).

[21] Logeswari, G., Bose, S., & Anitha, T. (2023). An intrusion detection system for sdn using machine learning. *Intelligent Automation & Soft Computing*, *35*(1), 867-880.

[22] Kadry, H., Farouk, A., Zanaty, E. A., & Reyad, O. (2023). Intrusion detection model using optimized quantum neural network and elliptical curve cryptography for data security. *Alexandria Engineering Journal*, *71*, 491-500.

[23] Ghanbarzadeh, R., Hosseinalipour, A., & Ghaffari, A. (2023). A novel network intrusion detection method based on metaheuristic optimisation algorithms. *Journal of ambient intelligence and humanized computing*, *14*(6), 7575-7592.

[24] Wang, R. X., Wang, Y., & Dai, L. (2023, March). Intrusion detection in network security. In *Second Guangdong-Hong Kong-Macao Greater Bay Area Artificial Intelligence and Big Data Forum (AIBDF 2022)* (Vol. 12593, pp. 366-371). SPIE.