FAST CORRELATION DEEP BELIEF AND RAPHSON GRADIENT BOOSTING ENSEMBLE CLASSIFIER FOR DOS ATTACK DETECTION IN WSN

P. Nagarajan[*1] & Dr. S.Veni[2]

[*1]Research Scholar, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore -641024, India

[2]Professor, Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore -641024, India

Abstract:

In today's contemporary world, the utilization of technology is inevitable and the swift advancements in the Internet and communication fields have emerged to diversify the Wireless Sensor Network (WSN) technologies. Enormous devices collect several sensory data for a wide range of fields and applications. However, WSN has been evinced to be susceptible to security lapses, integrated with their limited resources and voluminous of data generated instigate a crucial security concern. In this context, the objective remains in designing significant Denial of Service (DoS) attack detection by applying a salient machine learning abstraction known as ensemble learning in order to improve detection performance. The proposed method is called, Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) for DoS attack detection in WSN. However, most accessible datasets consist of multiclass output data with instable distributions, that remain to be the major pitfalls for attack detection accuracy reduction. Therefore, first Min-Max Normalization-based Preprocessing algorithm is designed with the normalization function that fix as instabilities identified in the raw dataset. Second, with the normalized network samples as input, pertinent features are extracted by means of Fast Correlation-based Deep Belief Network Feature Extraction algorithm. Finally, with the extracted features, by applying Raphson Gradient Boosting Ensemble Classifier algorithm, the detection and classification of four kinds of DoS attacks in WSN have detected. Moreover, the WSN-DS Dataset was utilized to examine efficiency of FCDB-RGEC. Results illustrate significant improvement with attack detection time, false alarm, recall, as well as precision, as compared to existing methods.

Keywords: Wireless Sensor Network, Ensemble Learning, Min-Max Normalization, Fast Correlation, Deep Belief Network, Raphson Gradient Boosting.

DOI: 10.24297/j.cims.2023.6.20

1. Introduction

The application of web-based services and transactions has considerably shoot over the past few decade owing to the evolution of both the electronic and communication devices. With the enormous advantages of employing web-based services and transactions, the security of important information acquired from end users is jeopardized. At the initial stage, attackers acquire the crucial web-based applications consisting of delicate and valuable data like, personal information of the user, their financial aspects, health care, and so on. Upon successfully acquiring of the data, the worthwhile resources are said to be exploited, hence threatening the confidentiality, accessibility and organizations' reliability as a whole.

In order to identify as well as defend against XSS attacks, novel fusion of ML as well as DL frameworks with great amount of accuracy and efficiency, called, hybrid stacking ensemble was proposed in [1]. On the basis of this representation, a novel idea for combining stacking ensemble called, hybrid stacking was proposed. With this design of hybrid detection method attacks were said to be identified in a robust manner. Also the defense acquired the dominance of URL encoding with mapping using dictionary with the objective of enhancing prediction accuracy, converge training process. Despite improvement observed in prediction accuracy, convergence time and the false alarm during the attack detection was not focused.

STLGBM-DDS integrating LightGBM ML, data balancing as well as attribute chosen method were proposed [2] called distinctive ensemble DoS Intrusion Detection System (DDS). The method was designed by Synthetic Minority Oversampling as well as Tomek-Links sampling approach with the objective of minimizing the influences of data imbalance termed STL. Moreover, during preprocessing, information Gain Ratio has utilized. Here, information adjusting as well as attribute chosen influences was also investigated. The overall accuracy was said to be improved. Though accuracy was concentrated, but the major factor influencing data imbalance, i.e., false alarm rate was not concentrated.

In today's competitive environment, it is highly impossible to think life in the absence of internet. Despite its several advantages, different crimes have mushroomed over the internet. Amongst them, one is DoS attack occur when service or machine or network are said to be unavailable to its intended users.

In [3], anomalies and unseen attacks are recognized in Double-Layered mixture method. Here, by generating PCA variables, the most prevailing features of numerous attack classes were learning which increase difference as of with assault category. With this rare attacks were detected in a significant manner. DL techniques for analyzing attacks were investigated in [4]. In detecting network invasion, several works have employed both ML as well as DL algorithms. A novel intrusion recognition mechanism on the basis of reinforcement learning with the objective of extended periods without model updates was proposed in [5]. With this type of design both

the false positive and false negative were said to be reduced. However, the time factor was not focused.

Detection of attacks in WSN and its various counter measures were discussed in [6]. In [7], machine learning techniques like, Logistic Regression and Nave Bayes were applied to differentiate between normal scenarios and attack types. Yet another systematic review of data availability, different types of risks involved in designing WSN was discussed in [8].

With the transition of secure network framework getting open to everyone, the network becomes more flexible, ubiquitous and also empirical. These evolutions have proliferated the deployment of next-generation Internet jargons, like, designing of cloud environment, IoT, and so on. However, with WSN architecture, the potential of a DoS attack brought on by focused dominance becomes more obvious.

DL was developed by trustworthy routing assault identification in [9]. Moreover, during both Low-Power as well as Lossy Networks, adversarial training model was also presented with determining planned attack. With these two distinct mechanisms applied aided in attaining a reliable learning, therefore significantly reducing the learning time. However, the classification accuracy was not focused. With this objective ensemble of algorithm via voting classifier was designed in [10].

Using this ensemble classifier model the classification accuracy was found to be significantly good. Although this ensemble classifier model maintains high classification accuracy levels, the attack detection time complexity may gets increased with the increase in the train process. To obtain maximum precision as well as recall need to update the model though minimizing attack detection time. In this work, Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) for DoS attack detection in WSN is proposed. Three methods on the WSN-DS dataset performance are examined. Contribution is listed below.

- This work utilizes Raphson Gradient Boosting Ensemble Classifier model to forecast malevolent traffic, DoS attack detection in WSN. An efficient method is designed via four dissimilar ensemble variants proposed to defend DoS attack in WSN.
- Min-Max Normalization-based Preprocessing with Fast Correlation-based Deep Belief Network Feature Extraction has integrated in proposed method. Highly correlated features are executed by feature extraction via correlation function and extracting abstract features via contrastive divergence deep belief network. Further to lessen the attack detection time Fast Correlation-based Deep Belief Network Feature Extraction algorithm is designed. Additionally, Raphson Gradient Boosting Ensemble Classifier algorithm has utilized for classifier optimization results.
- Experiments of proposed DoS attack method are validated with NS3 simulator.

- Outcomes of DoS attack detection method achieve effectual as well as precise categorization by other techniques with precision, recall, false alarm rate as well as attack time.

Structure of the article has prearranged. Contextualizes machine, deep and ensemble approaches for DoS attack detection and reviews the related works are discussed in Section 2. Section 3 introduces our novel WSN-DS dataset and describes our proposed method Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) for DoS attack detection in WSN. Section 4 provides with the experimental sections and makes comparative analysis with the aid of tables and graphical representations. Section 5 presents the final remarks to conclude this paper.

2. Related works

Intrusion detections based on deep learning have attained immense reception from the research community for their potentiality in handling contemporary security systems both in small and in large-scale networks. In spite of their significant evolution during ML as well as traditional DNN method failed to remember concept as trained at fresher information points.

In [11] several machine learning methods were employed and their results were integrated for measuring denial of service attack. Owing to the repeated changes observed in distribution of data distribution, DNN models lack in both precision as well as FPR aspect. Which learning as well as computing alteration, eight-stage statistics as well as ML approach was proposed in [11]. With this not only the accuracy was found to be improved but also reduced the false positive rate significantly.

Yet another ensemble learning employing the advantages of several learning mechanisms was presented in [13]. With this the ease in computation burden was said to be reduced with improved accuracy rate. However, classification involving several classes was not focused. An ensemble-based intrusion detection system to focus both on the binary and multi-class classification scenarios was presented in [14].

An intrusion detection system for distributed DDoS on the basis of big data technology was proposed in [15]. Due to the advancement of network both in terms of rapid development of associated devices, network attacks are becoming flexible as well. Upon comparison with the conventional detection techniques, machine learning is considered as a unique and tensile method for detecting network intrusions.

In [16], the issues of anomaly detection in the conventional network and also in the next generation network were discussed in detail and also a detailed review on ML execution was presented. For detecting intrusions and also preventive mechanisms were investigated in [17], a detailed machine as well as DL review.

In [18], a DNN was investigated to emerge a pliable and efficient intrusion detection mechanism with the purpose of detecting and classifying unexpected and unforeseeable cyber-attacks. In [19] a holistic comparative analysis of intrusion detection employing machine learning was presented. Yet another stacked generalization ensemble technique was proposed in [20]. With this type of ensemble superior predictions were said to be ensured.

Related works done with ensemble classification, network has susceptible for DoS attacks like, gray hole, black hole, flooding as well as scheduling assaults respectively and authors have preferred ML as well as DL techniques. Also, feature selection motionless is inevitable as of raw network dataset. Four DoS attacks category were not recognized by little amount of features. Therefore, with suitable correlation potentialities, we have adopted Raphson Gradient Boosting Ensemble Classifier with the Fast Correlation-based Deep Belief Network Feature Extraction for precise detection accuracy with minimum false alarm rate. The elaborate description of the proposed method is given in the following sections.

3. Methodology

In WSN, the security designs are indispensable characteristics of WSNs that must be conveyed to keep away from security trade-off of any type. There has been an ever-increasing implementation of WSNs in breaching security environments. Routing is considered to be one of the platforms for malicious users to interrupt the network. Hence, it becomes mandatory in implementing corrective actions to secure the network from attacks. DoS are considered as one of the most recurrent attacks in WSNs. In this work, for finding as well as classifying DoS attacks, Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) in WSN method has introduced. Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) method work flow is demonstrated in Figure 1.

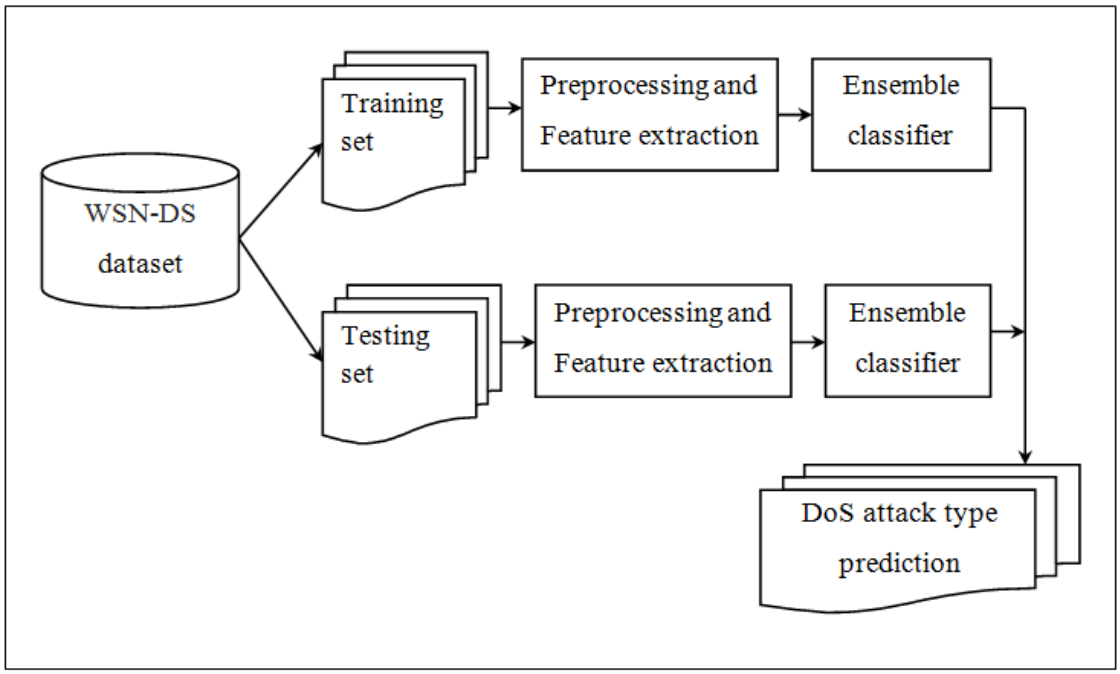

Figure 1 Workflow of Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) for DoS attack detection in WSN

WSN-DS dataset has further divide in ratio between training as well as testing of 80:20, respectively, as given in above figure. First, Min-Max Normalization-based Preprocessing model is applied to the raw dataset with the purpose of eliminating and fix instabilities identified in the data. Second, Fast Correlation-based Deep Belief Network Feature Extraction is performed with the normalized features as input that first is subjected to correlation function and then abstracts pertinent features for further attack detection process. Finally, with Raphson Gradient Boosting Ensemble Classifier model, detection and classification of four DoS attacks in WSN, namely, Blackhole, Grayhole, Flooding, as well as Scheduling attacks have made.

a. WSN-DS Dataset description

In this work, with the objective of detecting DoS attack in WSN, the WSN-DS dataset [21] is constructed and acquired from the sent and received data packets. After deep study 19 attributes or features are obtained for detecting DoS attack in WSN. These 19 features or attributes are provided in the following table.

Table 1 Details of WSN-DS dataset

S. No	Features or attributes	Description
1	Node ID	Sensor node unique ID
2	Time	Current sensor node simulation time
3	Is CH	Flag to differentiate between CH node (1 – CH, 0-Normal)
4	WHO CH	CH ID in the current round
5	Distance to CH	Distance between node and its CH in the current round
6	ADV_CH sends	Number of advertise CH's broadcast messages sent
7	ADV_CH receives	Number of advertise CH messages received
8	Join_REQ send	Number join request messages sent by nodes to the CH
9	Join_REQ_receive	Number of join request messages received by CH from nodes
10	ADV_SCH send	Number of advertise TDMA schedule broadcast messages sent to nodes
11	ADV_SCH receive	Number of TDMA schedule messages received from CHs
12	Rank	Order of this node within TDMA schedule
13	Data sent	Number of data packets sent from sensor to its CH
14	Data received	Number of data packets received from CH
15	Data sent to BS	Number of data packets sent to the BS
16	Distance CH to BS	Distance between the CH and the BS
17	Send code	Cluster sending code
18	Expanded energy	Amount of energy consumed in the previous round
19	Attack type	Node attack type with class of five possible values, namely, Blackhole, Grayhole, Flooding, and Scheduling, in addition to normal, if the node is not an attacker.

With the utilization of above features, via proposed method, four DoS attack category are implemented.

b. Min-Max Normalization-based Preprocessing model

The backdrop of DoS attack detection in WSN consist of distinct forms of features like, continuous, discrete, and noise with differing resolution and domains, WSN-DS utilized. The data has to be processed to make it suitable for our DoS attack detection in WSN and also to remove noise and fix instabilities identified in the data. This is owing to the reason that missing values depend on the individual features, wherein certain features possess zero as missing value, whereas certain other features possess zero as within the bounds of its value. For the sake of circumventing difficulties, the preprocessing model using normalization is proposed in our work.

The uncertain feature scales of data in divergent proportions will impact the DoS attack detection results in WSN. Hence, the data has to be normalized to remove the proportion

impact between criterions. Therefore, all the feature scales of data is set to '[0,1]', other hand the attack type label. The min-max normalization function is given as below.

$$NF = Nor(F_i^N) = \frac{F_i^N - Min(F_i^N)}{Max(F_i^N) - Min(F_i^N)} \quad (1)$$

From the above equation (1), 'F_i^N' specifies the values before normalization, '$Nor(F_i^N)$' denotes the value after normalization with '$Min(F_i^N)$' and '$Max(F_i^N)$' denoting higher as well as lower value of network sample data. In training as well as testing procedure, proportion among training as well as testing information ascertain sampling information individually utilized. Information has been split in 80:20 during training as well as testing. To obtain the initial energy of sensor node in WSN deployment, the energy value is mathematically formulated as given below.

$$CE_n = P_m * T_m \quad (2)$$

Then, with the above current energy value, residual energy in WSN deployment has mathematically represented as given below.

$$E_{res} = E_{ini}[SN] - E_{cons}[SN] \quad (3)$$

From the above equation (3), the residual energy 'E_{res}' is obtained, '$E_{ini}[SN]$' are initial energy of sensor node as well as energy being consumed by the sensor node '$E_{cons}[SN]$' respectively. With the obtained residual energy, cluster head 'CH' and cluster member 'CM' nodes are obtained. This is mathematically represented as given below.

$$CH = \sum_{i=1}^{n} Max[E_{res}(SN_i)] \quad (4)$$

From the above equation (4), sensor nodes possessing maximum residual energy '$Max[E_{res}(SN_i)]$', for a particular network size denote the cluster head 'CH' node, whereas the other nodes become the cluster member nodes 'CM' respectively. Min-Max Normalization - basis of preprocessing algorithm as follows.

Input: Dataset 'DS', Features '$F = \{F_1, F_2, \dots, F_m\}$', Network Samples '$S = \{S_1, S_2, \dots, S_n\}$'
Output: Noise removed processed data
1: **Initialize** '$m = 19$', '$n = 3{,}74{,}669$', power '$P_m = 1.0$', time '$T_m = 0.5ms$', network size '$500m * 500m$', learning rate '$\eta = 0.1$' 2: **Begin** 3: **For** each Dataset 'DS' with Features 'F' and Network Samples 'S' 4: Perform min-max normalization function as given in (1) 5: Evaluate current energy of sensor node as given in (2) 6: Evaluate residual energy of sensor node as given in (3)

7: Obtain cluster head and cluster member nodes by evaluating the function as given in (4)
8: **Return** cluster head 'CH' and cluster member 'CM' nodes
9: **End for**
10: **End**

Algorithm 1 Min-Max Normalization-based Preprocessing

As given in the above algorithm, with the objective of removing noisy data and fixing instabilities identified in the raw dataset, the data are processed to make it suitable for our DoS attack detection in WSN. By removing noisy data and fixing instabilities, the time and overhead incurred during further DoS attack detection in WSN is said to be reduced. First, in dataset features, min-max normalization function has utilized. With this function get all the scaled data in the ranges '0,1'. Next, with this scaled data and residual energy as base, differentiation between cluster head node and cluster member node are made for further processing.

c. Fast Correlation-based Deep Belief Network Feature Extraction

Feature extraction has discarding features which were redundant. This is due to the reason that not every data characteristic makes sense. Feature extraction is a way with which subset of significant features can be selected for DoS attack detection in WSN. Moreover, by extracting the pertinent feature, training time and accuracy will be improved. In our work, Fast Correlation-based Deep Belief Network Feature Extraction model is applied to the normalized features. Figure 2 shows the structure of Fast Correlation-based Deep Belief Network Feature Extraction model.

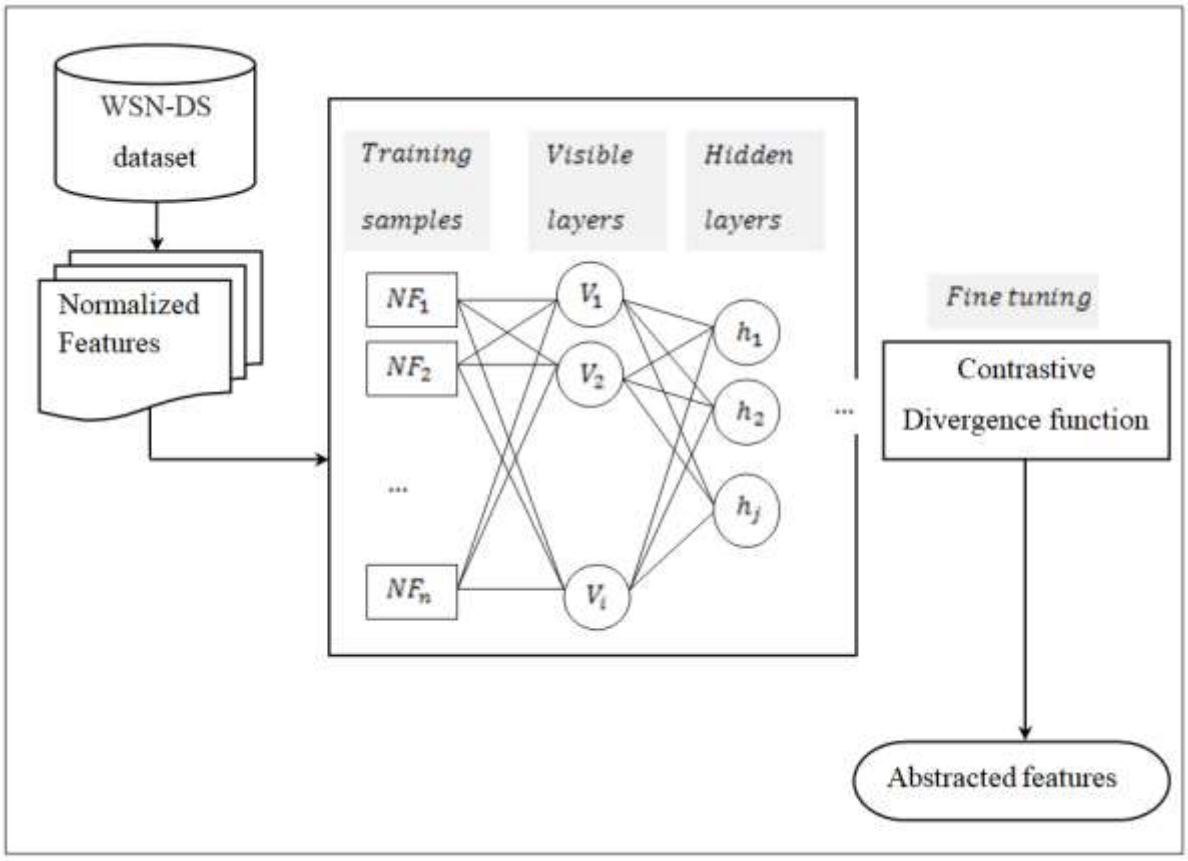

Figure 2 Structure of Fast Correlation-based Deep Belief Network Feature Extraction

As illustrated in the above figure, with the normalized features provided as input, the network training samples are subjected to Deep Belief Network via visible and hidden layer. The hidden layer output are then subjected or fine tuned via contrastive divergence function to finally obtain the abstracted features. Let us consider the normalized features 'NF' in the dataset 'DS' that are initially subjected to fast correlation functions that in turn create a correlation matrix. A correlation matrix is first created due to the reason that certain features in the dataset are correlated with certain others and vice versa. Hence, highly correlated features are obtained by dividing the covariance of two feature value and then multiplying the divided value by standard deviation of each feature value, as given below.

$$CC = \frac{Cov(NF,RF)}{[SD(NF)-SD(RF)]} \tag{5}$$

From the above equation (5), initially, the highly correlated coefficient 'CC' values are obtained based on the covariance 'Cov' of normalized feature 'NF' and random feature 'RF' and then dividing it with the standard deviation 'SD' of normalized feature 'NF' and random feature 'RF' respectively. Next, with the highly correlated coefficient 'CC' resultant values that actual feature extraction process is carried out. On the basis of DoS attack characteristics, in this work, correlation coefficient-based deep belief network model is employed in extracting pertinent traffic features.

The correlation coefficient-based deep belief network is represented in the form of series of Boltzmann machine modules that are stacked together. The front layer here is represented as the visible layer (i.e., the highly correlated normalized features) of the next hidden layer and the input of the next hidden layer. By considering it as a bipartite graph, with one being the visible layer (that is the data input layer consisting of highly correlated normalized features), and the other one is the hidden layer.

Though connections are said to be established between all visible layer and hidden layer, but no connection is said to exist between hidden layers. The correlation coefficient-based deep belief network is employed as feature extraction because of energy factor employed in our work. Then, via correlating energy with all configurations of variables or highly correlated normalized features, dependence among visible as well as hidden units, 'v' as well as '$\boldsymbol{h}$' are evaluated. Also, energy function acquires minimum values as two correlated normalized feature values are compatible whereas acquires maximum values as '$\boldsymbol{h}$' has fewer well-matched by 'v'. It is then formulated as given below.

$$E(v,h) = CC[NF]\left[\sum_{i=1}^{v}\sum_{j=1}^{h} w_{ij}v_i h_j - \sum_{i=1}^{v} a_i v_i - \sum_{j=1}^{h} b_i h_j\right] \quad (6)$$

From the above equation (6) the energy function for the corresponding highly correlated normalized features between visible and hidden units '$E(v,h)$' is obtained based on the bias of visible unit '$a_i v_i$', bias of hidden unit '$b_i h_j$', weight 'w_{ij}' among visible layer'i' as well as hidden layer 'j', binary states '$v_i h_j$' respectively. Through energy function, cooperative probability distribution of a set of visible vector'v' and a hidden vector '$\boldsymbol{h}$' is established as given below.

$$prob(v,h) = \frac{1}{NF} e^{-E(v,h)}, where NF = \sum_{v,h} e^{-E(v,h)} \quad (7)$$

From the above equation (7), 'NF' being the normalization factor represents the aggregates over all probable sets of visible vector 'v' as well as hidden vector '$\boldsymbol{h}$' respectively. Finally, the abstract features extracted by the correlation coefficient-based deep belief network are arrived at by updating the weight 'w_{ij}' using Contrastive Divergence function as given below.

$$w_{ij} = \eta\left(\left[v_i, h_j\right]BR, \left[v_i, h_j\right]AR\right) \quad (8)$$

From the above equation (8), 'η' represents the learning rate, '$\left[v_i, h_j\right]BR$' product before reconstruction and '$\left[v_i, h_j\right]AR$' expected abstracted features after reconstruction. The expected abstracted features after reconstruction becomes the final features extracted for further attack detection process. The pseudo code representation of Fast Correlation-based Deep Belief Network Feature Extraction is given below.

Input: Dataset 'DS', Features '$F = \{F_1, F_2, \dots, F_m\}$', Network Samples '$S = \{S_1, S_2, \dots, S_n\}$'
Output: Robust and pertinent feature extraction 'FE'
1: **Initialize** '$m = 19$', '$n = 3{,}74{,}669$', learning rate '$\eta = 0.1$', Normalized Features 'NF'
2: **Initialize**cluster head 'CH' and cluster member 'CM' nodes
3: **Begin**
4: **For** each Dataset 'DS' with Normalized Features 'NF', Network Samples 'S'
5: Evaluate highly correlated features as given in (5)
6: Measure energy function between visible and hidden units as given in (6)
7: Evaluate cooperative probability distribution for a set of visible vector and hidden vector as given in (7)
8: Measure weight using Contrastive Divergence function as given in (8)
9: **Return** abstracted features '$FE = [v_i, h_j]AR$''
10: **End for**
11: **End**

Algorithm 2 Fast Correlation-based Deep Belief Network Feature Extraction [features extracted are 15, Node ID, Time, Is CH, Current Energy, Energy Consumption, ADV_CH send, ADV_CH received, Join_REQ send, Join_REQ received, ADV_SCH send, ADV_SCH received, Data sent, Data received, Data sent to BS, Attack Type]

As given in the above algorithm, with the normalized features and network samples provided as input are first subjected to correlation function for producing highly correlated features. Second, with the highly correlated features as input, energy function is evaluated between visible and hidden units. Next, cooperative probability distributions were modeled and finally, are fine-tuned by means of Contrastive Divergence function. The product before reconstruction are eliminated whereas expected abstracted features after reconstruction are retained that is said to be the extracted features. As the next process, classification is performed using the Raphson Gradient Boosting Ensemble Classifier model.

d. Raphson Gradient Boosting Ensemble Classifier

Supervised learning techniques execute the job of searching via a hypothesis to identify an appropriate hypothesis that will build fine attack detection results. Though the hypothesis results produce hypotheses that well-suited for attack detection, it may be very difficult to find a good one. On the other hand, ensembles integrate numerous hypotheses to generate a robust and optimal hypothesis, where generate multiple hypotheses utilizing the same base learner. In this work, Raphson Gradient Boosting Ensemble Classifier is employed for intrusion detection at

different thresholds. Figure 3 shows the structure of Raphson Gradient Boosting Ensemble Classifier model.

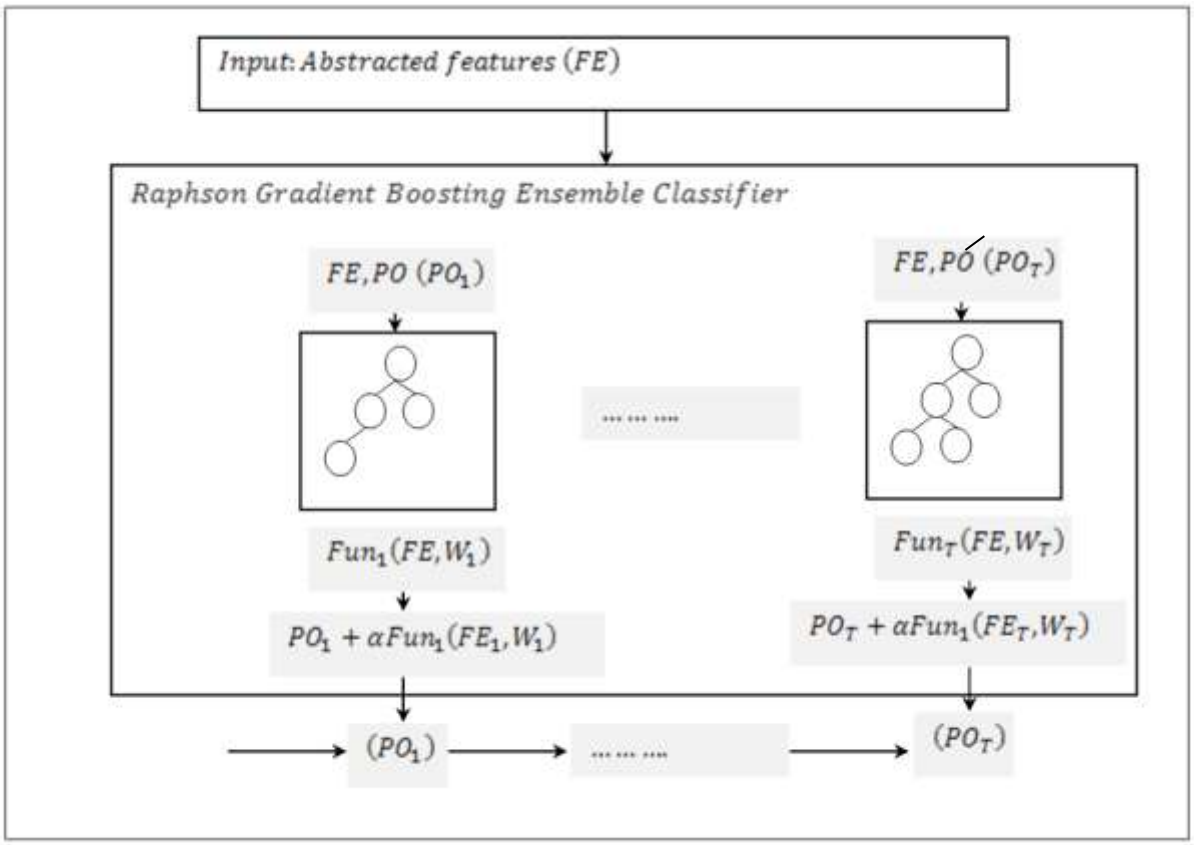

Figure 3 Structure of Raphson Gradient Boosting Ensemble Classifier

As illustrated in the above figure, the Raphson Gradient Boosting Ensemble Classifier is a type of boosting algorithm that boost weak learners in DoS attack detection. The trees in Raphson Gradient Boosting create a new tree by taking into consideration the previous DoS attack detection results for given input network samples of the tree and therefore maximizing the attack detection gain. The Raphson Gradient Boosting Ensemble Classifier process is an iterative model that add a new tree that fine tunes prior tree issues. Followed by which this entire process is said to be integrated with the preceding trees to generate final prediction results. The prediction value is mathematically stated as given below.

$$PO_T(FE) = PO_{T-1}(FE) + \alpha * Fun_T(FE, W_T) \quad (9)$$

From the above equation (9), prediction output of 'Th' iteration is '$PO_T(FE)$', with a learning factor being 'α' and function 'Fun_T' to fine tune the 'Th' iteration weight 'W_T'. On the basis of the optimal loss, the Raphson Gradient Boosting Ensemble Classifier model obtains a leaf node, also includes tree with learning novel function in all iteration '$Fun_T(FE, W_T)$', finally obtaining the predicted value by including related scores of each tree. On the other hand, training objective

function of Raphson Gradient Boosting Ensemble Classifier model (i.e., DoS attack detection in WSN) provided with input network samples 'S' consists of two parts, namely, the training error and regularization as given below.

$$FE(PO_T) = \sum_{i=1}^{n} Loss(PO_i, AO_i) + \sum_{t=1}^{T} RT(Fun_T) \quad (10)$$

From the above equation (10), '$Loss(PO_i, AO_i)$' is utilized in employing the difference between the predicted output 'PO_i' and actual output 'AO_i' with which the loss function '$Loss$' is measured along with the weak learners regularization term '$RT(Fun_T)$' respectively. Finally, to avoid over-fitting, gradients and hessians are measured for each abstracted features as given below.

$$G_m(FE_i) = \frac{\partial Loss(PO_i, Fun(FE_i))}{\partial Fun(FE_i)} \quad (11)$$

$$H_m(FE_i) = \frac{\partial^2 Loss(PO_i, Fun(FE_i))}{\partial Fun(FE_i)^2} \quad (12)$$

From the above equations (11) and (12), with the purpose of looking in to the matter that if the test error is higher than the training error over-fitting aspects employing gradients (i.e., first order derivatives) '$G_m(FE_i)$' and hessians (i.e., second order derivatives) '$H_m(FE_i)$' are obtained. With the above derivative function results, the classified results, i.e., DoS attack detections are made. The pseudo code representation of Raphson Gradient Boosting Ensemble Classifier is given below.

Input: Dataset 'DS',Network Samples '$S = \{S_1, S_2, \dots, S_n\}$'
Output: False alarm-minimized precise DoS attack detection
1: **Initialize** Normalized Features 'NF', abstracted features 'FE', base station 'BS' 2: **Initialize** cluster head 'CH' and cluster member 'CM' nodes 3: **Initialize** threshold advertisement message '$TADV_{CH} = 5$' 4: **Begin** 5: **For** each dataset 'DS' with Normalized Features 'NF', abstracted features 'FE' and Network Samples 'S' 6: Obtain the prediction value as given in (9) 7: Train the objective function as given in (10) 8: **For** each weak learners 9: Formulate first order and second order derivatives as given in (11) and (12) 10: **End for** 11: **For** each 'CH' broadcast advertisement message '$TADV_{CH} = ADV\_CH$' 12: Cluster member 'CM' nodesjoins 'CH' 13: Cluster head 'CH' creates 'ADV_SCH_send' 14: Cluster member 'CM_i' sends data packets 'DP' to 'CH' **//Black hole attack** 15: **If** 'CH' drops all data packets 'DP' 16: **Then** attack identified with black hole

17: **Else** go to step 32
18: **End if**
//Gray hole attack
19: **If** 'CH' drops data packets 'DP' randomly
20: **Then** attack identified with gray hole
21: **Else** go to step 32
22: **End if**
//Flooding attack
23: **If** broadcast advertisement message '$ADV_{CH} > TADV_{CH}$'
24: **Then** attack identified with flooding
25: **Else** go to step 32
26: **End if**
//Scheduling attack
27: **If** 'CH' sends same 'ADV_SCH_send' to all the cluster member 'CM' nodes
28: **Then** attack identified with scheduling
29: **Else** go to step 32
30: **End if**
31: **End for**
32: Sends aggregated data packets 'DP' to base station 'BS'
33: **End for**
3: **End**

Algorithm 3 Raphson Gradient Boosting Ensemble Classifier

As given in the above algorithm, two different tasks are performed, namely, ensemble and with the ensemble results, perform classification for detecting attacks. With the network samples and abstracted features obtained as input, first, the prediction value is derived. Followed by which the objective function is designed with which the over-fitting issues via first order derivative and second order derivative are formulated. Next, for each cluster head that broadcast advertisement messages to the base station, accordingly, cluster member joins the corresponding cluster. Followed by which the cluster memory sends data packets to cluster head, if the cluster head drops all the data packets, then, black hole attack is said to be detected, and on contrary if the cluster head drops the data packets arbitrarily, then gray hole attack is said to be detected. Next, if the cluster head broadcast large numbers of advertisement messages than the threshold with the purpose of grabbing the energy of cluster members those which consumes large amount of energy in deciding to join which cluster head, then, the attack is said to be flooding attack. Finally, for each cluster member nodes, in case if the cluster head transmits same TDMA schedule, resulting in data packet collision resulting in data packet loss or else called as scheduling attack.

4. Experimental setup

In this section, experimental analysis of the Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) is presented. With aid of WSN-DS dataset (https://www.kaggle.com/datasets/bassamkasasbeh1/wsnds), FCDB-RGEC method has been discussed by Hybrid stacking ensemble [1] and STLGBM-DDS [2]. Simulations are performed in NS3. To ensure fair comparison between proposed FCDB-RGEC method and existing Hybrid stacking ensemble [1] and STLGBM-DDS [2] similar network samples are employed for evaluating different parameters like, precision, recall, false alarm rate and attack detection time for different iterations. In this work, three standard classification performance parameters are applied to extensively estimate our proposed method.

a. Qualitative analysis

In this section the qualitative analysis of FCDB-RGEC method is discussed in detail. With the WSN-DS dataset obtained as input, 8network samples are used for simulation as given below in table 2.

Table 2 Network samples from WSN-DS dataset

ID	Time	IsCH	WhoCH	Dist to CH	ADV_S	ADV_R	JOIN_S	JOIN_R	SCH_S	SCH_R	Rank	Data_S	Data_R	Data_Sent_To_BS	Dist_CH_To_BS	Send_Code	Expanded_Energy	Attack_Type
101007	50	0	101010	26.75033	0	4	1	0	0	1	21	41	0	0	0	3	0.06662	Normal
101008	50	0	101044	63.66485	0	4	1	0	0	1	17	38	0	0	0	4	0.06649	Normal
112029	603	1	112029	0	1	5	0	31	1	0	0	0	1239	21	150.3168	0	2.29263	Grayhole
116073	803	1	116073	0	1	5	0	44	1	0	0	0	1276	13	96.57363	0	1.36872	Grayhole
111029	553	1	111029	0	1	5	0	36	1	0	0	0	1260	0	0	0	0.00722	Blackhole
114065	703	1	114065	0	1	3	0	40	1	0	0	0	1280	0	0	0	0.0073	Blackhole
101096	53	1	101096	0	7	0	0	90	1	0	0	0	1350	15	121.695	0	2.25865	Flooding
102001	103	1	102001	0	6	14	0	51	1	0	0	0	150	0	0	0	0.00743	Flooding

After applying min-max normalization function, the above table 2 is said to undergo preprocessing and is listed as given below in table 3.

Table 3 Network samples normalized values

Time	Is CH	Who CH	Dist to CH	ADV_S	ADV_R	JOIN_S	JOIN_R	SCH_S	SCH_R	Rank	Data_S	Data_R	Data_Sent_To_BS	Dist_CH_To_BS	Send_Code	Expanded_Energy
-0.964	-1.620	-0.953	0.668	-0.773	-0.250	-1.620	-1.264	-1.620	1.620	1.834	1.701	-1.281	-0.701	-0.707	1.294	-0.664
-0.964	-1.620	-0.948	2.263	-0.773	-0.250	-1.620	-1.264	-1.620	1.620	1.383	1.537	-1.281	-0.701	-0.707	1.903	-0.664
0.722	0.540	0.718	-0.488	-0.409	0.000	-0.540	-0.190	0.540	-0.540	-0.536	-0.540	0.656	1.702	1.599	-0.533	1.469
1.328	0.540	1.332	-0.488	-0.409	0.000	-0.540	0.260	0.540	-0.540	-0.536	-0.540	0.714	0.786	0.775	-0.533	0.584
0.670	0.540	0.567	-0.488	-0.409	0.000	-0.540	-0.017	0.540	-0.540	-0.536	-0.540	0.689	-0.701	-0.707	-0.533	-0.721
1.025	0.540	1.027	-0.488	-0.409	-0.500	-0.540	0.121	0.540	-0.540	-0.536	-0.540	0.720	-0.701	-0.707	-0.533	-0.721
-0.945	0.540	-0.940	-0.488	1.774	-1.250	-0.540	1.853	0.540	-0.540	-0.536	-0.540	0.830	1.015	1.160	-0.533	1.436
-0.793	0.540	-0.803	-0.488	1.410	2.250	-0.540	0.502	0.540	-0.540	-0.536	-0.540	-1.047	-0.701	-0.707	-0.533	-0.720

The figure 4 consider 10 sensor nodes '$S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}$' for deployment as given below.

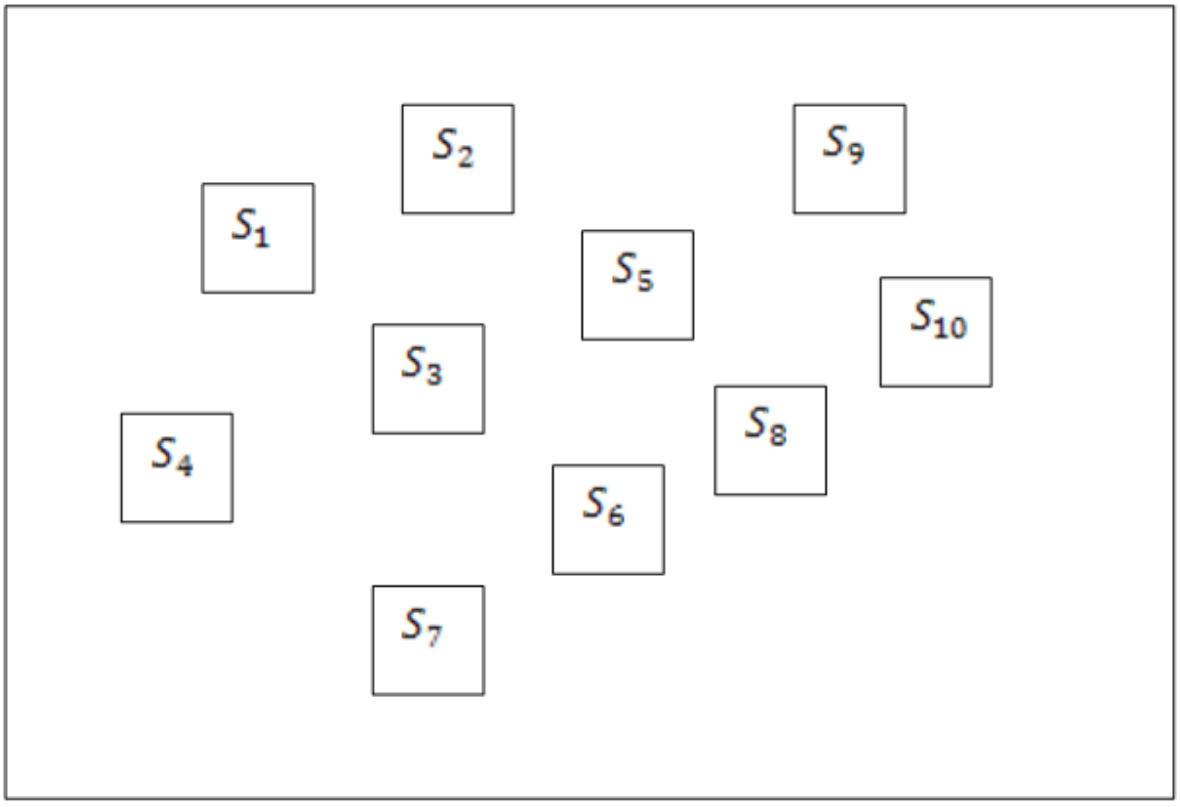

Figure 4 Sample WSN configurations

With the above WSN configuration considered for simulation, let us first achieve initial energy of sensor node with considering the power is 1.0 and the time as 0.5ms, then value of energy is obtained as given below.

$$CE_n = 1.0 * 0.5 = 0.5\text{J}$$

From the above equation results, initial energy in deployment is assumed to be '0.5J'. Residual energy is measured based on the difference between initial energy level and consumed energy of both sensor nodes. The residual energy outcomes are listed in table 4 as given below.

Table 4 Residual energy evaluation

Initial energy	Consumed energy	Residual energy
$\boldsymbol{E_{ini}[S_1] = 0.5}$	$E_{cons}[S_1] = 0.22$	$E_{res}[S_1] = 0.28$
$\boldsymbol{E_{ini}[S_2] = 0.5}$	$E_{cons}[S_2] = 0.15$	$E_{res}[S_2] = 0.35$
$\boldsymbol{E_{ini}[S_3] = 0.5}$	$E_{cons}[S_3] = 0.13$	$E_{res}[S_3] = 0.37$
$\boldsymbol{E_{ini}[S_4] = 0.5}$	$E_{cons}[S_4] = 0.28$	$E_{res}[S_4] = 0.22$
$\boldsymbol{E_{ini}[S_5] = 0.5}$	$E_{cons}[S_5] = 0.17$	$E_{res}[S_5] = 0.33$
$\boldsymbol{E_{ini}[S_6] = 0.5}$	$E_{cons}[S_6] = 0.19$	$E_{res}[S_6] = 0.31$
$\boldsymbol{E_{ini}[S_7] = 0.5}$	$E_{cons}[S_7] = 0.23$	$E_{res}[S_7] = 0.27$
$\boldsymbol{E_{ini}[S_8] = 0.5}$	$E_{cons}[S_8] = 0.21$	$E_{res}[S_8] = 0.29$
$\boldsymbol{E_{ini}[S_9] = 0.5}$	$E_{cons}[S_9] = 0.35$	$E_{res}[S_9] = 0.15$
$\boldsymbol{E_{ini}[S_{10}] = 0.5}$	$E_{cons}[S_{10}] = 0.09$	$E_{res}[S_{10}] = 0.41$

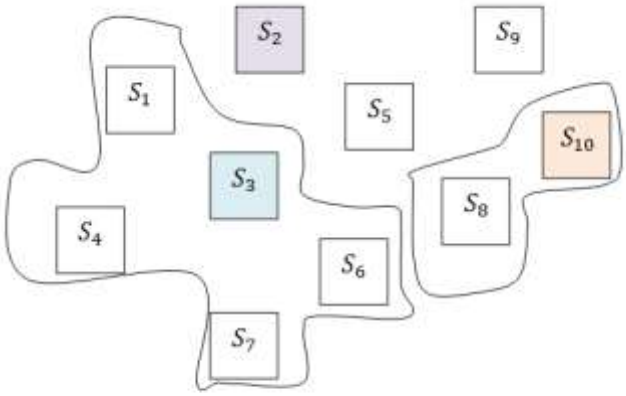

Figure 5 Cluster head and cluster member formulation

Then, figure 5 and table 5 exemplifies cluster head and cluster member are formulated based on the residual energy.

Table 5 Formation of cluster head and cluster member

S. No	Cluster head 'CH'	Cluster member 'CM'
1	S_2	S_5, S_9
2	S_3	S_1, S_4, S_6, S_7
3	S_{10}	S_8

According to highly correlated features (i.e., the sum of the values being zero [0], Is CH, Who CH, ADV_R, JOIN_S, SCH_S, SCH_R, DATA_R are selected) and Contrastive Divergence function (i.e., the sum of the values being closer to 0.002, ADV_S, JOIN_R, DATA_S, DATA_SENT_TO_BS are selected). Hence, the features selected are listed in table 6.

Table 6 Features extracted

ID	Is CH	Who CH	ADV_S	ADV_R	JOIN_S	JOIN_R	SCH_S	SCH_R	Data_S	Data_R	Data_Sent_To_BS	Attack_Type
101007	-1.620	-0.953	-0.773	-0.250	1.620	-1.264	-1.620	1.620	1.701	-1.281	-0.701	Normal
101008	-1.620	-0.948	-0.773	-0.250	1.620	-1.264	-1.620	1.620	1.537	-1.281	-0.701	Normal
112029	0.540	0.718	-0.409	0.000	-0.540	-0.190	0.540	-0.540	-0.540	0.656	1.702	Grayhole
116073	0.540	1.332	-0.409	0.000	-0.540	0.260	0.540	-0.540	-0.540	0.714	0.786	Grayhole
111029	0.540	0.567	-0.409	0.000	-0.540	-0.017	0.540	-0.540	-0.540	0.689	-0.701	Blackhole
114065	0.540	1.027	-0.409	-0.500	-0.540	0.121	0.540	-0.540	-0.540	0.720	-0.701	Blackhole
101096	0.540	-0.940	1.774	-1.250	-0.540	1.853	0.540	-0.540	-0.540	0.830	1.015	Flooding
102001	0.540	-0.803	1.410	2.250	-0.540	0.502	0.540	-0.540	-0.540	-1.047	-0.701	Flooding

Finally, the DoS attack detection is hypothesized as given below.

- First, with the data sent (1.701, 1.537) is greater than data received (-1.281, -1.281) by the IDs (101007, 101008) both of them are considered as normal.
- Higher advertisement of CH being sent (i.e., 1.774 and 1.410), the IDs (101096, 102001) are considered as flooding attack.
- In a similar manner, all the data packets were dropped (i.e., SCH_R being -0.540 and Data_Sent_To_BS being -0.701) hence the IDs (111029, 114065) are considered as blackhole attack.
- In case of the IDs (112029, 116073), the data packets received were (0.656, 0.714) and the data packets sent were (0.540, 0.540), with the drop rate being (0.116 and 0.174), hence considered to be grayhole attack.

b. Quantitative analysis

In this section, the quantitative analysis of Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) for DoS attack detection in WSN is validated in terms of four metrics, namely, DoS attack detection time, precision, recall and false alarm rate. To perform fair comparison similar numbers of network samples are utilized for validation using the three methods, FCDB-RGEC, Hybrid stacking ensemble [1] and STLGBM-DDS [2] respectively.

i. Case analysis of DoS attack detection time

The time consumed during detecting DoS attack has first significant metric. To be more specific, time needed to find different types of DoS attacks are defined as DoS attack detection time. Lower time, more significant the method is said to be because earlier the time consumed in detecting the DoS attack earlier remedial actions can be taken. The attack detection time is computed in milliseconds (ms) and it as given below.

$$DoSAD_{time} = \sum_{i=1}^{n} S_i * Time\ [AD] \quad (13)$$

Where (13), '$DoSAD_{time}$' denotes DoS attack detection time, network samples are 'S_i' as well as '$Time\ [AD]$' is time utilized for attack detection. $DoSAD_{time}$ taken by various classifiers is given in Table 7.

Table 7 DoS attack detection time taken by different methods

Network samples	DoS Attack Detection time (ms)		
	FCDB-RGEC	Hybrid stacking ensemble	STLGBM-DDS
50	17.5	24	31.5
100	25.35	38.55	45.35
150	31	48.35	60.25
200	38.25	55.35	85.35
250	42.55	65.25	100.15
300	50	80	125.35
350	53.15	95.35	155.35
400	70	105.25	170.25
450	85.35	125.35	205.35
500	105.25	140	215.25

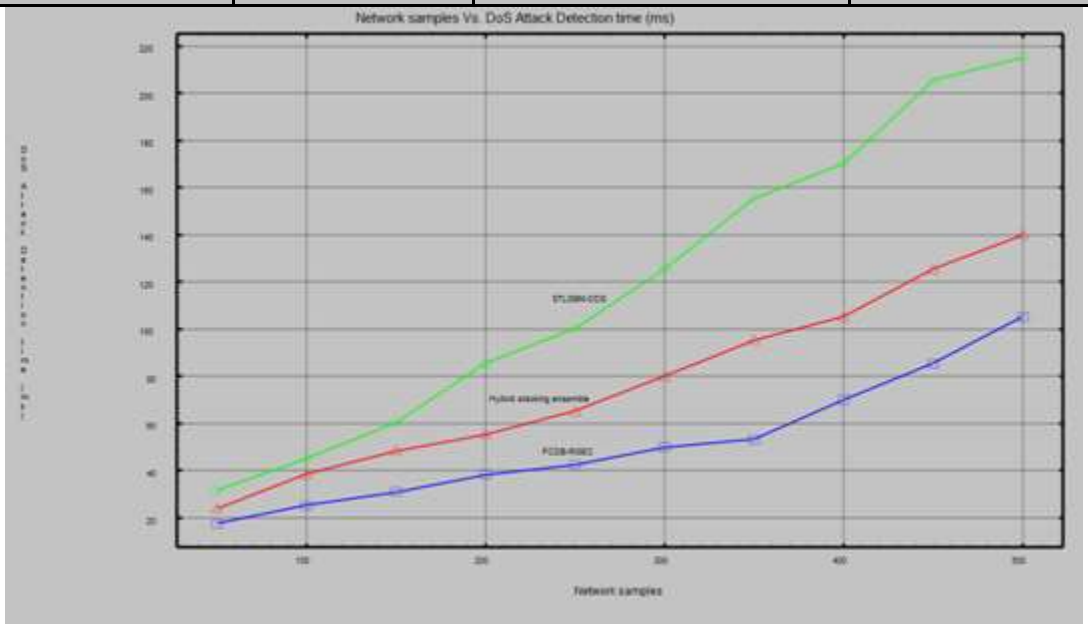

Figure 6 Comparative analysis of DoS attack detection time using FCDB-RGEC, Hybrid stacking ensemble [1] and STLGBM-DDS [2]

Figure 6 given above illustrates the graphical portrayal of DoS attack detection time using the three methods, FCDB-RGEC, Hybrid stacking ensemble [1] and STLGBM-DDS [2]. From the figure, during simulation period, time has enhanced also network samples are improved. This is due to the reason that with larger number of network samples found in WSN, large amount of

time have consumed during sensing, then, increases DoS assault time also. So the time has direct proportionality to network samples. However, with simulations performed with 50 network samples, construct a reliable DoS attack detection system, the time consumed in detecting correct type of DoS attack for a particular network sample being '0.35ms', the overall attack detection time using FCDB-RGEC was 17.5ms, the time consumed in detecting correct type of attack for a particular network sample being '0.48ms', the overall attack detection time using [1] was 17.5ms, the time consumed in detecting correct attack for a particular network sample being '0.63ms', the overall attack detection time using [2] was 24ms and finally observed to be 1625ms using [2]. Time in detecting different types of attacks outcome using FCDB-RGEC method is observed to be better than when compared to [1] and [2]. The improvement is due to the application of Min-Max Normalization-based Preprocessing algorithm in FCDB-RGEC method. By applying this algorithm, not only the noisy data were removed but also fixed instability therefore minimizing the overhead. With the overhead being reduced, the attack involved are said to be detected at an early stage. With this function, pertinent and essential features were extracted, therefore reducing the DoS attack detection time using FCDB-RGEC method by 33% as well as 54% than [1], [2].

ii. Precision

Significance DoS attack detection of precision rate has second metric. Precision represents the fraction of correctly classified and detected DoS attacks. The precision rate is represented as given below.

$$P = \frac{TP}{TP+FP} \tag{14}$$

From the above equation (14), the precision rate 'P', is measured, 'TP' is true positive (properly classified DoS attacks) as well as 'FP' is false positive (wrongly identified DoS attacks) respectively. The precision observed by various classifiers is given in Table 8.

Table 8 Precision observed by different methods

Network samples	Precision		
	FCDB-RGEC	Hybrid stacking ensemble	STLGBM-DDS
50	0.91	0.82	0.77
100	0.91	0.82	0.77
150	0.9	0.82	0.75
200	0.88	0.78	0.73
250	0.87	0.77	0.71
300	0.85	0.75	0.68
350	0.83	0.73	0.64
400	0.81	0.71	0.62
450	0.79	0.68	0.6
500	0.77	0.65	0.6

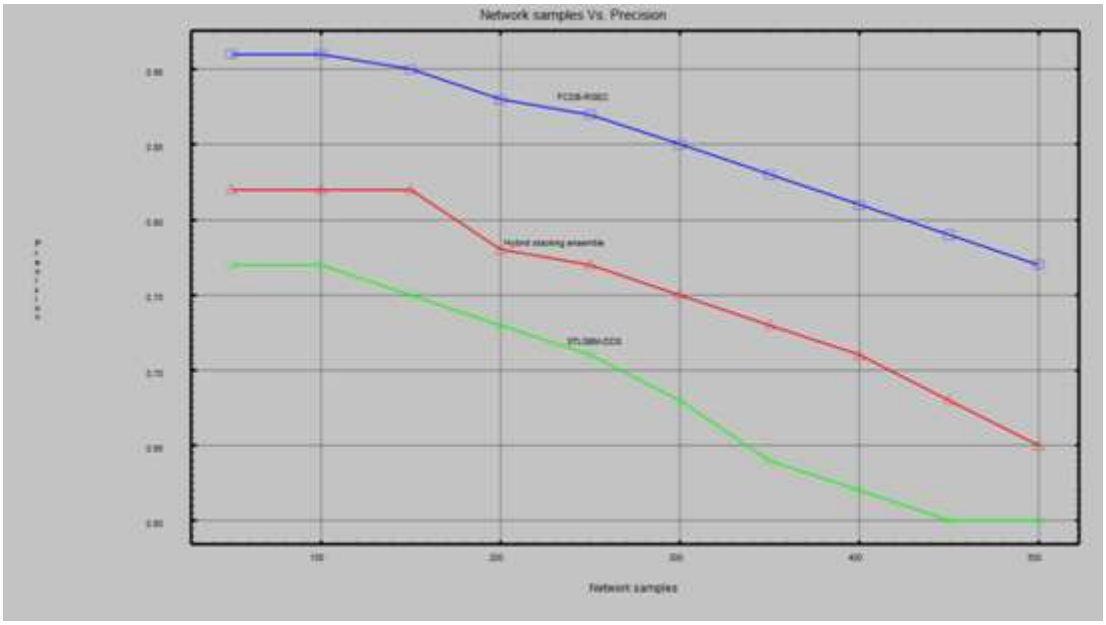

Figure 7 Comparative analysis of Precision usingFCDB-RGEC, Hybrid stacking ensemble [1] and STLGBM-DDS [2]

Figure 7 given above shows the graphical representation of the proposed FCDB-RGEC, existing Hybrid stacking ensemble [1] and STLGBM-DDS [2] on WSN_DS dataset in terms of precision rate. Vertical axis indicates measure of precision rate as well as network samples taken horizontal axis in the above figure. The network samples are defined as the ID of the corresponding traffic instances and employed for experimental purpose so that DoS types of attack detection can made in terms of precision. The reported result from figure shows that the proposed FCDB-RGEC method outperforms other methods, [1] and [2] compared from 13%, 24% in term of precision rate. This is evident from the simulation with 50 network samples involved in DoS attack detection system (where 45 samples were found to be correctly classified into their corresponding DoS attacks) and '41' number of network samples were correctly detected by the network using FCDB-RGEC method, '37' number of network samples were detected by the network using [1] and '35' number of network samples were detected by the network using [2]. It is because FCDB-RGEC method utilizes highly correlated features by dividing the covariance of two feature value and then multiplying the divided value by standard deviation of each feature value to obtain highly correlated coefficient features. This in turn eliminates the values of each feature within an explicit range and correctly classifying into their corresponding types of DoS attacks, therefore improving the precision rate using FCDB-RGEC method upon comparison with [1] and [2] respectively.

iii. Case analysis of recall

Recall rate represents the ratio that the DoS attacks are correctly detected. The recall rate is measured as given below.

$$R = \frac{TP}{TP+FN} \tag{15}$$

From the above equation (15), the recall rate 'R' is measured, 'FN' are false negative (improperly recognized usual traffic flows) respectively. The recall measured using (15) by three different methods is given in Table 9.

Table 9 Recall observed by different methods

Network samples	Recall		
	FCDB-RGEC	Hybrid stacking ensemble	STLGBM-DDS
50	0.85	0.8	0.75
100	0.849	0.795	0.735
150	0.842	0.785	0.7
200	0.84	0.77	0.7
250	0.835	0.75	0.685
300	0.82	0.735	0.655
350	0.81	0.7	0.635
400	0.805	0.685	0.615
450	0.795	0.635	0.595
500	0.78	0.6	0.555

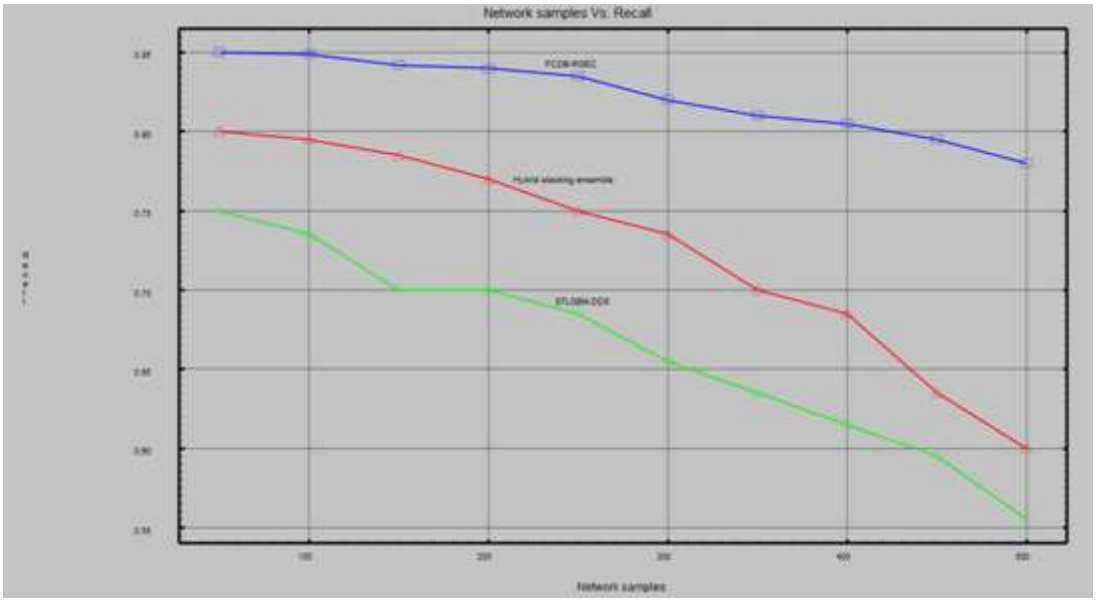

Figure 8 Comparative analysis of Recall using FCDB-RGEC, Hybrid stacking ensemble [1] and STLGBM-DDS [2]

Figure 8 given above shows the figurative representation of recall rate. With horizontal axis representing the network samples, vertical axis denotes the recall rate obtained by utilizing true positive instances and false negative instances. Here, a small dip was observed using all the three methods. Simulations performed with 50 network samples, saw a false negative rate of 6, 8 and 10 using the three methods, Recall using FCDB-RGEC, Hybrid stacking ensemble [1] and

STLGBM-DDS [2]. With this simulation, the recall when applied with FCDB-RGEC method was found to be comparatively lesser than [1] and [2]. The improvement in recall rate was found due to the application of Fast Correlation-based Deep Belief Network Feature Extraction algorithm. By applying this algorithm, first the processed network sample instances were subjected to correlation for generating highly correlated features. Next, with the highly correlated features energy function between visible and hidden units were obtained. Finally, the weights were fine-tuned by means of Contrastive Divergence function. With this, the number of incorrectly classified normal traffic flows to be attack types were reduced, therefore improving the recall rate using FCDB-RGEC method by 14% compared to [1] and 25% compared to [2].

iv. Case analysis of false alarm rate

Finally, false alarm rate is discussed in this section. False Alarm Rate represents the ratio that the normal traffic flows are detected as the DoS attacks. The false alarm rate is measured as given below.

$$FAR = \frac{FP}{FP+TN} \tag{16}$$

From the above equation (16), the false alarm rate 'FAR' is measured based on the false positive 'FP'and the true negative 'TN' rate respectively. Finally, table 10 given below lists the false alarm rate values.

Table 10 False alarm rate observed by different methods

Network samples	False alarm rate		
	FCDB-RGEC	Hybrid stacking ensemble	STLGBM-DDS
50	0.12	0.17	0.22
100	0.14	0.195	0.235
150	0.17	0.205	0.25
200	0.1755	0.218	0.265
250	0.185	0.235	0.273
300	0.193	0.25	0.295
350	0.205	0.263	0.305
400	0.215	0.285	0.315
450	0.225	0.305	0.335
500	0.24	0.315	0.35

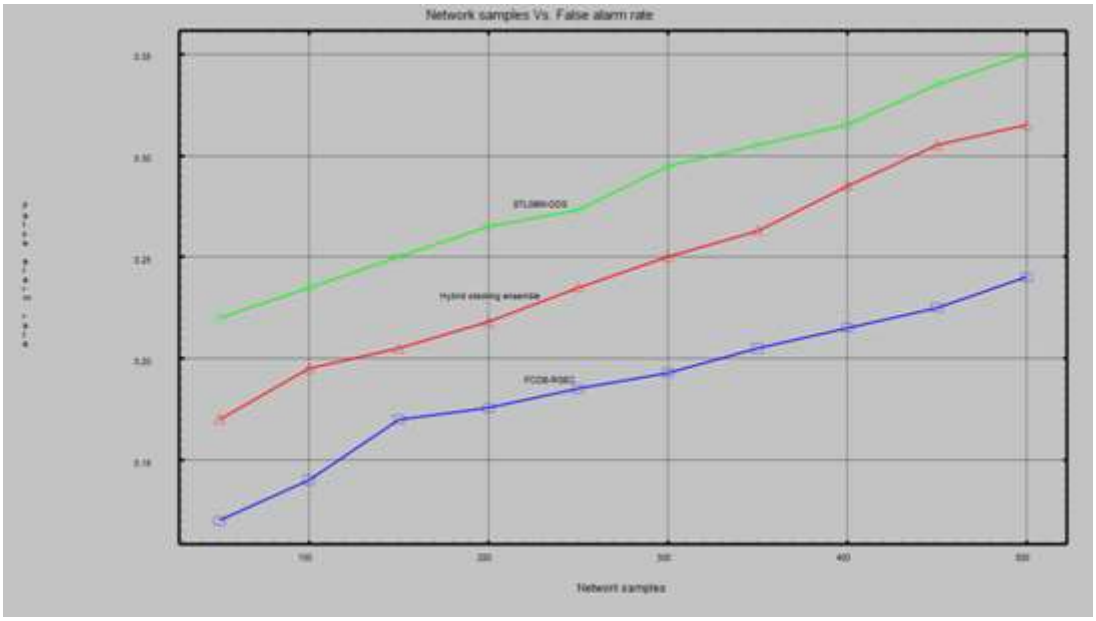

Figure 9 Comparative analysis of false alarm rate usingFCDB-RGEC, Hybrid stacking ensemble [1] and STLGBM-DDS [2]

Finally, figure 9 given above illustrates the influence of false alarm rate for different numbers of network samples ranging between 50 and 500. False alarm rate is found to be directly proportional to the network samples provided as simulation in figure. This is owing to the reason that with distinct network samples IDs provided as input over different time instances, an increasing trend is said to be observed when measuring the false alarm rate also. However, with the simulations performed for 50 network samples and the network samples false predicted as attack types using FCDB-RGEC method was found to be 5, 7 [1] and [9] for an average of 40 positive samples, the overall false alarm rate were observed to be '0.12', '0.17' and '0.22' respectively. From these results it is inferred that the false alarm rate is lesser using FCDB-RGEC when compared to [1]and [3]. The result behind the minimization of false alarm rate using FCDB-RGEC method was due to the application of is due to the application of Raphson Gradient Boosting Ensemble Classifier algorithm. By applying this algorithm, the weak learner results were ensemble using gradients and hessians that in turn not only ensured overfitting but also minimized the falsification of attack types. Also, by including training error and regularization function, fine tuning of weight was made, therefore reducing the false alarm rate using FCDB-RGEC method by 23% compared to [1] and 35% compared to [2].

5. Conclusion

In many DoS attack detection systems, similarity function at a fine grained fashion are specifically employed in differentiating among attacks. Upon comparison to most of the prevailing DoS attack detection methods, a novel Fast Correlation Deep Belief and Raphson Gradient Ensemble Classifier (FCDB-RGEC) method using ensemble classifier based on network samples is proposed to improve the detection accuracy in addition to minimizing the time and false alarm rate. The main innovation of our method is obtaining a measure for different types

of DoS attacks in WSN by proposing Fast Correlation-based Deep Belief Network Feature Extraction algorithm. Specifically, with the processed network sample instances, highly correlated features are first obtained. Here, the correlated features are said to be fast as it not only obtained highly correlated features but also subjected to visible and hidden layer with highly correlated normalized features or correlating energy for all organization of variables that in turn generates fast correlated features. Second, the Raphson Gradient Boosting Ensemble Classifier is presented to provide detection of DoS attacks in WSN via ensemble classifiers and generates attack detection results. In addition, along with the experiments, an empirical evaluation of our FCDB-RGEC method with the aid of discussion was performed to compare to the traditional and state-of-the-art methods using the WSN-DS dataset. The observed numerical results have confirmed that the proposed FCDB-RGEC method outperforms well by achieving a higher attack detection accuracy, precision, recall and minimum false alarm compared other approaches.

References

1. Muralitharan Krishnan, Yongdo Lim, Seethalakshmi Perumal, Gayathri Palanisamy, "Detection and defending the XSS attack using novel hybrid stacking ensemble learning-based DNN approach", Digital Communications and Networks, Elsevier, Sep 2022 [Hybrid stacking ensemble]
2. Murat Dener, Samed Ali, Abdullah Orman, "STLGBM-DDS: An Efficient Data Balanced DoS Detection System for Wireless Sensor Networks on Big Data Environment", IEEE Access, Sep 2022
3. TreepopWisanwanichthan and Mason Thammawichai, "A Double-Layered Hybrid Approach for NetworkIntrusion Detection System Using CombinedNaive Bayes and SVM", IEEE Access, Oct 2021
4. Meenakshi Mittal, Krishan Kumar, Sunny Behal, "Deep learning approaches for detecting DDoS attacks: a systematicreview", Soft Computing, Springer, Nov 2021
5. Roger R. dos Santos, Eduardo K. Viegas, Altair O. Santin, Vinicius V. Cogo, "Reinforcement Learning for Intrusion Detection:More Model Longness and Fewer Updates", IEEE Transactions on Network and Service Management, Sep 2022
6. Ademola P. Abidoye, Ibidun C. Obagbuwa, "DDoS attacks in WSNs: detection andcountermeasures", The Institution of Engineering and Technology, IET Wireless Sensor Systems, Jan 2018
7. Kimmi Kumari and M. Mrunalini, "Detecting Denial of Service attacks usingmachine learning algorithms", Journal of Big Data, Sep 2022
8. Frank Cremer, Barry Sheehan, Michael Fortmann, Arash N. Kia, Martin Mullins, Finbarr Murphy, Stefan Materne, "Cyber risk and cybersecurity: a systematic review of dataavailability", The Geneva Papers on Risk and Insurance, Springer, Feb 2022

9. Sharmistha Nayak, Nurzaman Ahmed, Sudip Misra, "Deep Learning-Based Reliable Routing Attack Detection Mechanism forIndustrial Internet of Things", Ad Hoc Networks, Elsevier, Aug 2021
10. Varun Sharma, Dr. R.K. Bathla, "Ddos Attack Prediction Using Voting Classifer", Elementary Education Online, Jul 2021
11. Mona Esmaeili,Seyedamiryousef Hosseini Goki,Behnam HajipourKhireMasjidi,Mahdi Sameh,Hamid Gharagozlou,and Amin Salih Mohammed, "ML-DDoSnet: IoT Intrusion Detection Based on Denial-of-ServiceAttacks Using Machine Learning Methods and NSL-KDD", Wireless Communications and Mobile Computing, Wiley, Aug 2022
12. Sai Prasath, KamalakantaSethi, Dinesh Mohanty, PadmalochanBera, SubhransuRanjan Samantaray, "Analysis of Continual Learning Models forIntrusion Detection System", IEEE Access, Nov 2022
13. Joffrey L. Leevy, John Hancock, Taghi M. Khoshgoftaar and Jared M. Peterson, "IoT information theft prediction usingensemble feature selection", Journal of Big Data, Nov 2022
14. Adeel Abbas, Muazzam A. Khan, Shahid Latif, Maria Ajaz, Awais Aziz Shah, Jawad Ahmad, "A New Ensemble-Based Intrusion Detection System for Internet ofThings", Arabian Journal for Science and Engineering, Springer, Aug 2021
15. Ying Gao, HongruiWu, BinjieSong, YaqiaJin, XiongwenLuo, Xing Zeng, "A Distributed Network Intrusion DetectionSystem for Distributed Denial of Service Attacksin Vehicular Ad Hoc Network", IEEE Access, Oct 2019
16. Song Wang, Juan Fernando Balarezo, SithamparanathanKandeepan, AkramAl-Hourani, Karina Gomez Chavez, BenhaminRubinstein, "Machine Learning in Network AnomalyDetection: A Survey", IEEE Access, Nov 2021
17. P. L. S. Jayalaxmi, Rahul Saha, Gulshan Kumar, Mauro Conti, Tai-HoonKim, "Machine and Deep Learning Solutions forIntrusion Detection and Preventionin IoTs: A Survey", IEEE Access, Oct 2022
18. R. Vinayakumar, MamounAlazab, K. P. Soman, PrabaharanPoornachandran, Ameer Al-Nemrat, SitalakshmiVenkatraman, "Deep Learning Approach for Intelligent IntrusionDetection System", IEEE Access, Apr 2019
19. Tarek Moulahi, Salah Zidi, AbdulatifAlabdulatif, Mohammed Atiquzzaman, "Comparative Performance Evaluation of IntrusionDetection Based on Machine Learning inIn-Vehicle Controller Area Network Bus", IEEE Access, Jul 2021
20. Smitha Rajagopal, Poornima PandurangaKundapur,and KatiganereSiddaramappaHareesha, "A Stacking Ensemble for Network Intrusion Detection UsingHeterogeneous Datasets", Security and Communication Networks, Wiley, Jan 2020.