An Efficient Intrusion Detection and Prevention System for DDOS Attack in WSN Using SS-LSACNN and TCSLR

Vikash Kumar Singh[1,*], Durga Sivashankar[2], Kishlay Kundan[3] and Sushmita Kumari[4]

[1]*Consultant Tech Lead, Societe Generale, India*
[2]*Technical Lead, Siemens Healthineers, India*
[3]*Department of Information Technology, National Institute of Technology (NIT), Patna, India*
[4]*Software Engineer, Computer Science, Jayotividyapeeth Women University, Jaipur, India*
E-mail: vikashsd2@yahoo.com
[*]*Corresponding Author*

Received 26 November 2022; Accepted 29 March 2023;
Publication 05 December 2023

Abstract

Sensor Nodes (SNs) are utilized by Wireless Sensor Networks (WSNs) to recognize their environment; in addition, the WSN delivers data from sensing nodes to the sink. The WSNs are exposed to several security threats owing to the broadcast performance of transmission along with the increase in the growth of application regions. Countermeasures like Intrusion Detection and Prevention Systems (IDPS) should be adopted to overcome the aforementioned attacks. By implementing these systems, several intrusions can be detected in WSN; also, WSN can be prevented from various security attacks. Therefore, identifying the general attack that influences the SNs mentioned as Distributed Denial of Service (DDoS) attack and recuperating the data utilizing Soft Swish (SS)-Linear Scaling-centered Adam Convolution

Journal of Cyber Security and Mobility, Vol. 13_1, 135–160.
doi: 10.13052/jcsm2245-1439.1315

Neural Network (SS-LSACNN) along with Two's Compliment Shift Reverse (TCSLR) operation are the intentions of this work. Firstly, for extracting the vital features, the data gathered as of the dataset are utilized. After that, the extracted features are pre-processed. It is then utilized for attack detection. The null features and the redundant data are removed in preprocessing. By employing the Correlation Coefficient-centered Synthetic Minority Over-sampling Technique (CC-SMOTE) methodology, data separation regarding classes and data balancing was performed to prevent the imbalance issue. Subsequently, to provide the preprocessed data for attack detection, the Numeralization and feature scaling are executed. After that, by utilizing Chebyshev Distance (CD)-centric K-Means Algorithm (KMA), the real-time SNs are initialized as well as clustered. The data gathered as of the SNs are utilized for attack detection following the clustering phase. Following the detection phase, the data being attacked are amassed in the log file; similarly, the non-attacked data are inputted into the prevention phase. Next, the experiential analysis is carried out for examining the proposed system's efficacy. The outcomes revealed that the proposed model exhibits 98.15% accuracy, 97.59% sensitivity, 95.72% specificity, and 95.48% F-measure, which displays the proposed model's efficacy.

Keywords: Distributed denial of service (DDoS) attack, soft swish linear scaling based adam convolution neural network (SS-LSACNN), two's Compliment shift left reverse (TCSLR), correlation coefficient based SMOTE (CC-SMOTE).

1 Introduction

Since WSNs have inherent characteristics of being easy to deploy and are low cost, it is employed in various fields of technology [1]. Recently, a broad series of applications have been developed by WSNs that play a main role in the existing research domain [2]. Even though WSN's unfinished resources face severe problems, it has become the majority preferred networking solution. Owing to a lack of tamper resistance, the SNs are extremely sensitive to a few attacks [3]. The major challenges in WSN are security, trust, along with routing. Without considering the back hole along with DoS in the network, data should be transmitted safely [4]. Sybil attacks, routing attacks, and Denial of Service (DoS) are a few of the various sorts of attacks on WSNs [5]. A vital class of DDoS attacks, which functions on the application layer together with the network layer, is continuously produced by hackers [6]. This is attained by the utilization of attacked server resources and unauthorized

utilization of different divisions of the global network. Authorized users receive DoS owing to such destructive influences and such sorts of attacks are named Distributed Denial of Service (DDoS) [7]. A server is overloaded with traffic by a DoS attack; thereby shutting it down. A DoS attack in which several computers or machines flood a targeted resource is termed a Distributed DoS (DDoS) attack. All DDoS is equal to DoS but not all DoS is equal to DDoS. The functionality of the network may be troubled partially or entirely by these attacks and also from the battery, it may take unnecessary power consumption; also, decreases the life span of SNs. Saving wireless networks from these sorts of attacks is significant [8]. For detecting the nodes' suspicious behaviour inside WNS, Intrusion Detection Systems (IDS) could be utilized [9]. In WSNs, most of the research fields of security include secure routing, key management, and prevention [10]. Here, by the utilization of the SS-LSACNN along with the TCSLR technique, efficient IDPS for DDoS attacks is employed. The proposed work's major contributions are,

- To handle the imbalanced data, CC-SMOTE techniques are utilized.
- To efficiently classify DDoS attacks in WSN, the SS-LSACNN technique is utilized.
- To cluster the node, CD-KMA techniques are utilized.
- To safeguard the data from intruders, TCSLR techniques are implemented.

The balance part is arranged as: the existing IDPS is surveyed in Section 2; the proposed technique is illustrated in Section 3; the proposed system's performance is evaluated in Section 4; lastly, Section 5 winds up the work.

2 Related Work

P.J. BeslinPajila et al. [11] intended to discover the DDoS attack rapidly and to recover sensors utilizing the Fuzzy Logic (FL) approach. To find the availability of a DDoS in a node, type 1 FL was utilized by the Fuzzy Based DDoS Detection together with Recovery (FBDR) mechanism. Likewise, for recovering DDoS attacks, fuzzy-type 2 was utilized. On the basis of identifying the DDoS and recovering the DDoS attacks, the type-1 along with type-2 rules were executed greatly; also, it helped to decrease the energy usage of every node and enhanced the network's lifespan. The experimental outcomes proved that the FBDR technique worked better when compared with different correlated techniques. Nevertheless, outcomes are not generated correctly by the FL utilized by this technique.

Wenjie Zhang et al. [12] developed hierarchical IDS that cluster the nodes in WSN in accordance with their functions. Moreover, in this system, the Kernel Extreme Learning Machine's (KELMs) classification algorithm along with Mercer Property to synthesize Multi-Kernel (MK) functions was utilized to increase the detection accurateness of WSN IDS's anomalous behaviour and decrease the false alarm rate. By testing as well as applying the MK function, this technique recognized the optimal linear combination; in addition, it built an MKELM for WSN IDS. The experimental outcomes illustrated that the technique not only warrants increased detection accurateness but also spectacularly decreases the detection duration; thus, it was matched for resource-constrained WSNs. However, energy utilization was high.

Shuai Jiang et al. [13] presented SLGBM, an IDS for WSNs. To decrease the data dimension on the feature space of the actual traffic data, the Sequence Backward Selection (SBS) algorithm was utilized; as a consequence, the computation overhead is decreased. After that, to identify the network's various attacks, a Light GBM algorithm was utilized. The simulation outcomes grounded on the WSN-DS dataset illustrated that in Grayhole, Normal, Flooding, Blackhole, along with Scheduling (Time Division Multiple Access (TDMA)) attack detections, the F-measure of SLGBM was evidently higher than the current typical detection techniques. However, the time taken was high for data processing.

M. Premkumar and T.V.P. Sundararajan [14] employed a lightweight DoS detection scheme to detect along with isolate the attacks by utilizing Deep Learning-centric Defense Mechanism (DLDM). For the effective detection of DoS attacks like flooding, exhaustion, homing, and jamming, the algorithm was utilized. Here, Extensive simulation experiments were done. The simulation outcomes exhibited that a higher detection rate, throughput, packet delivery ratio, as well as accuracy, were attained by the method. However, if the attacker deploys a combination of attacks, the system might be compromised.

K. Lakshmi Narayanan et al. [15] introduced an ML-centered Naive Bayes Classifier for detecting and an Enhanced Code-centered Round Trip Time-centered model to avert those '2' important attacks. With the aid of an authorized code, the black hole attack was analyzed utilizing this technique. To discover the data delay time for reaching the destination, the wormhole attack along with the fake destination attack was prohibited by trip time. Though the method reduced the communication overhead, it is limited to a lower number of transmissions.

Ademola P. Abidoye and Boniface Kabaso [16] presented a methodology for detecting along with eradicating DoS attacks in WSNs. The detection methodology was grounded on utilizing cluster heads that were observing the traffic intensity in clusters. For the cluster heads' selection, a unique model was utilized. The experimental outcomes exhibited that a higher true-positive detection rate, lower false-positive detection rate, energy-efficient, as well as high data reliability were attained by the scheme. However, the model didn't meet the resource constraints of WSN.

E. Suryaprabha and N. M. Saravana Kumar [17] suggested an optimized energy-centered constraint DoS attack detection algorithm, that is, the OBES approach for handling DoS attacks that studies the network traffic and handles the intruders. In the NS2, the implementation was done. The outcomes displayed that the OBES algorithm exhibited better performance when analogized with DoS attack detection with the energy constraint algorithm. Poor scalability was encompassed in this method in handling attacks for a large type of user.

Mukaram Safaldin [18] implemented improved IDS by utilizing the amended binary Grey Wolf Optimizer with a Support Vector Machine (GWOSVM-IDS). To identify the best number of wolves, 3 wolves, 5 wolves, along with 7 wolves were utilized by the GWOSVM-IDS. Via decreasing false alarm rates, the model aimed to elevate the detection accuracy as well as detection rate in the WSN environment. Also, for minimizing the processing time, the number of features resulting from the IDSs was diminished. The outcomes exhibited that the other comparative algorithms were overwhelmed by the GWOSVM-IDS with seven wolves. This approach was not scalable and failed to detect the attackers when the network size is very large.

S. Balaji and T. Sasilatha [19] presented a routing approach to coordinate the system's characteristics for DoS attack detection. To schedule the SNs in such a route to the point where network traffic was monitored by the node, this methodology was employed. By this procedure, the attack's source was identified by the model, which also attained DoS provision to the users. Grounded on the implementation of the routing approach, the throughput, packet loss ratio, and packet delivery outcomes were simulated and analyzed in the network. This approach had a high data transmission delay.

Somnath Sinha and Aditi Paul [20] designed a robust along with efficient AIDS, which utilized fuzzy and neural network (NN) centered tools. To independently monitor the local nodes' behavior and identify whether a node was trusted, distrusted, or an enemy, the system was installed in each node. The false alarms engendered owing to the fuzzy logic applied in the

initial step were filtered by the use of a trained NN. A 100% true positive with 0% false positive was exhibited by the outcome. However, handling a large amount of intruded data was difficult.

3 Proposed Intrusion Detection And Prevention System

The WSN has significant usage in numerous fields; thus, they are turning into a vital part of the research. Sensors, which might be hundreds in number, are included by the WSNs. These sensors function jointly regarding the local decision process. WSNs are highly optimistic in the region of their application since they are scalable along with infrastructure-less. However, malicious nodes, which are regarded as security threats, might get into the system. WSNs are prone to several security threats, namely snooping, sinkhole, tampering Sybil, clone, wormhole, and spoofing. (a) Higher bandwidth demand, (b) higher energy consumption, (c) quality of service (QoS) provisioning, (d) data processing together with compressing methodologies, and (e) cross-layer design are the challenges in WSN. The system's overall performance along with security can get influenced by these attacks. Thus, identifying along with preventing attacks on WSN is highly significant. Hence, this paper proposes an SSLS-based Adam CNN Classifier to detect and an efficient TCSLR method to prevent DDoS attacks for WSNs. The incoming data was classified by the proposed model as an attack/intrusion or not; if it's an attack, then the methodology tries to secure the non-attacked data from the intruders. Using the SS-LSACNN technique, the DDoS attack can be analyzed with the aid of extracted features. Likewise, the DDoS attack can be prevented by securing the data from several security attacks using the structure of the TCSLR system. (a) The intrusion detection training phase, (b) intrusion detection testing phase, and (c) intrusion prevention phase are the '3' phases encompassed in the methodology. Figure 1 displays the proposed model's block diagram.

3.1 Intrusion Detection Phase

A monitoring system that detects doubtful activities along with produces alerts when they are detected is termed IDS. Security operations have been developed to remediate the threat centered on these alerts. Here, regarding the data's features, the attacked and non-attacked data are recognized by employing the SS-LSACNN. The CIC DoS dataset is utilized to train the SS-LSACNN. From this dataset, 80% of data is utilized for training; in addition, for testing, 20% of data together with the real-time sensor data

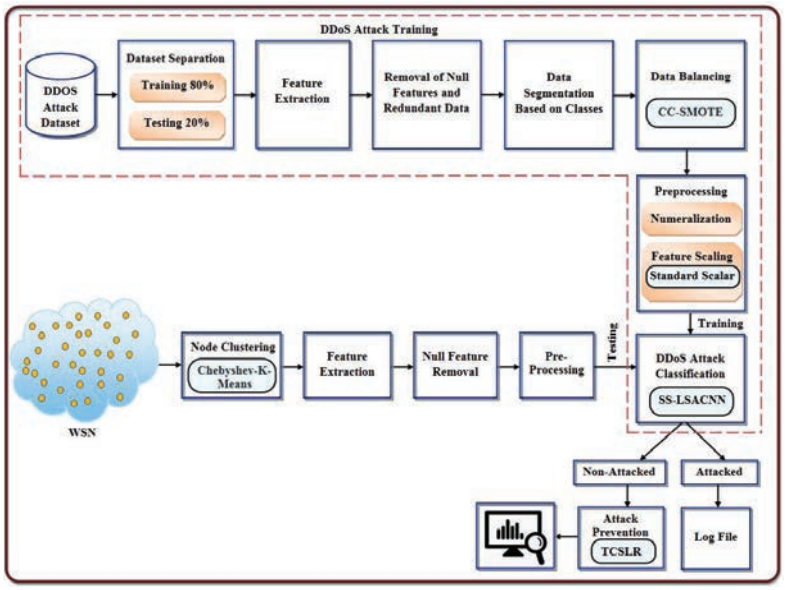


Figure 1 Block diagram of the proposed methodology.

are utilized. (i) Feature extraction, (ii) null features along with redundant data removal, (iii) data balancing, and (iv) pre-processing are the stages undergone by the dataset to identify the data status precisely. The training data is formulated as,

$$d_m = \{d_1, d_2, d_3, \ldots, d_M\} \tag{1}$$

Where, the data collected as of the dataset is identified as d_m, the M number of data is proffered as d_M.

3.1.1 Feature extraction

The process of converting raw data into numerical features that can be processed whilst protecting the information in the original data set is named feature extraction. It helps to lessen the number of redundant data from the data set, which augments the speed of models. The significant features are extracted after collecting the data d_m. Timestamp, Flowid, Source IP address, Source port, Destination IP address, total forwarded packets, et cetera are the features being extracted here. The features being extracted are notated as,

$$\Psi_n = \{\Psi_1, \Psi_2, \Psi_3, \ldots, \Psi_N\} \tag{2}$$

Where, the number of extracted features is symbolized as Ψ_n, and the N^{th} number of features is notated as Ψ_N.

3.1.2 Preprocessing

In the IDS, the most vital step is pre-processing. It is a technique to perform a series of operations to transform or change data in which the actual data is converted into a useful format that improves accuracy and reliability. Removal of Null features, Data balancing, Numeralization, and Feature scaling are the important steps utilizing the features extracted in the preceding step Ψ_n for pre-processing.

***Removal of Null Features and Redundant Data*:** The system's performance is affected harmfully by the existence of null features; thus, removing null values is a crucial step. Multiple copies of the same data in the database may bring about data redundancy. Data abnormalities along with corruption might be caused by data redundancy; thus, they should be taken away. When a similar piece of data is present in multiple places, data redundancy takes place. Wasteful data redundancy and positive data redundancy are the '2' kinds of data redundancy. By removing redundant data, the storage can be made efficient [21]. The data obtained after the removal of null features together with redundant data are partitioned regarding the classes. Data Partitioning is the methodology of distributing data across multiple (i) tables, (ii) disks, or (iii) sites to ameliorate processing performance or augment database manageability. Horizontal, vertical, and functional partitioning are the different ways to partition the data. Merely '2' classes are included in the proposed approach. Hence, the data are partitioned under those '2' classes. It is formulated as,

$$
\begin{aligned}
{}^{rem(NF,RD)}\Psi_{n(Ben,Mal)} = \{ & {}^{rem(NF,RD)}\Psi_{1(Ben,Mal)}, \\
& {}^{rem(NF,RD)}\Psi_{2(Ben,Mal)}, \ldots, \\
& {}^{rem(NF,RD)}\Psi_{N(Ben,Mal)}\}
\end{aligned} \tag{3}
$$

Where, following the removal of null features along with redundant data, the output data being partitioned underneath the Benign and Malicious classes is defined as ${}^{rem(NF,RD)}\Psi_{n(Ben,Mal)}$.

***Data Balancing*:** By employing the CC-SMOTE methodology, the data being separated ${}^{rem(NF,RD)}\Psi_{n(Ben,Mal)}$ are balanced to address the data imbalance problem. Unequal distribution of target classes in the dataset causes data imbalance issues. To handle imbalanced data, SMOTE, which is an

oversampling methodology, is utilized. SMOTE is an oversampling methodology where synthetic samples are generated for the minority class [22]. It concentrates on the feature space to engender new instances with the aid of interpolation betwixt the positive instances that lie together. The data imbalance is addressed here by augmenting the data instances in which the new instances are equivalent to the actual data of minority classes. By creating synthetic data from its closest neighbour in terms of actual features, the data can be balanced. K-no. of nearest neighbours is selected randomly in the existent SMOTE methodology. It causes data balancing to be unbalanced at every single iteration. Regarding the nearest neighbour nodes, the CCs of current along with preceding iteration outputs are selected for data balancing to overcome the aforementioned issues. To filter the K-nearest neighbours of every single data sample of minority classes, the CC is wielded. It is proffered as,

$$\Psi_{NN} = \frac{\chi^{t}\Psi_{\min(i)},^{t-1}\Psi_{\min(j)}}{\chi^{t}\Psi_{\min(i)}\chi^{t-1}\Psi_{\min(j)}} \tag{4}$$

Subsequently, the synthetic data is presented as,

$$^{\min(Ben,Mal)}\Psi_{n(syn)} = \Psi_{\min(i)} + (\Psi_{NN} - \Psi_{\min(i)}) + \beta \tag{5}$$

Where, the K-nearest neighbours filtered by the CC betwixt the samples $(^{t}\Psi_{\min(i)},^{t-1}\Psi_{\min(j)})$ of the minority class is specified as Ψ_{NN}, the sample covariance is signified as $\chi^{t}\Psi_{\min(i)},^{t-1}\Psi_{\min(j)}$, the sample standard deviations are defined as $\chi^{t}\Psi_{\min(i)}$, $\chi^{t-1}\Psi_{\min(j)}$, the current iteration is proffered as t, the minority class's data sample is indicated as $\Psi_{\min(i)}$, the generated synthetic data is illustrated as $^{\min(Ben,Mal)}\Psi_{n(syn)}$. The data can be exhibited as balanced data $\Psi_{n(Bal)}$ after identifying the new feature vectors. It is then inputted into the pre-processing phase.

***Numeralization*:** Here, the data in non-numerical format are transmuted into a numerical format. The system processes the numerical values; thus, the conversion is performed by assigning certain particular values to the variables.

***Feature Scaling*:** It is a technique wielded to normalize the range of independent variables or features of data. It significantly boosts model performance. Here, to enhance the data model, the feature vectors are normalized to range betwixt 0 to 1 or -1 to 0. In this, the standard scalar model [23] is utilized as,

$$\Psi_{n(Nor)} = \frac{\Psi_{n(Bal)} - \mu_{\Psi_{n(Bal)}}}{\sigma_{\Psi_{n(Bal)}}} \tag{6}$$

Where, the mean and standard deviation are represented as $\mu_{\Psi_{n(Bal)}}$, $\sigma_{\Psi_{n(Bal)}}$, the normalized data is depicted as $\Psi_{n(Nor)}$.

3.1.3 Classification

Here, to identify the DDoS attack in the WSN, the pre-processed data $\Psi_{n(Nor)}$ are wielded by the IDS; then, the data is classified into attacked along with non-attacked types. The SS-LSACNN is utilized for classification. Further, by [24–27] in this work, the Cauchy problem, which is an initial value problem or a boundary value problem in CNN has been computed and resolved, which leads to low computational complexity and shows the model's better performance. The loss function is augmented in the previous CNN by choosing the weights randomly; thus, lowering the network's training speed. To overcome this issue, the weight values are updated by utilizing an Adaptive Moment Estimation (ADAM) optimizer, and a soft swish activation function is wielded in the softmax layer that augments the network's training speed. The data is headed by every single layer as follows,

***Convolution layer*:** The convolution layer allows the merging of two sets of information. Here, betwixt '2' sets of data like the input matrix and convolution filter termed weights, the convolution operation is executed. The outcomes are appended to engender the output feature map following the convolution.

***Pooling layer*:** Here, the feature map's size acquired as of the CL is mitigated by performing a pooling operation. This is achieved by conducting the max-pooling operation that chooses the maximum element from the feature map's region covered by the filter. Therefore, the output subsequent to the max-pooling layer would be a feature map encompassing the earlier feature map's most outstanding features.

***Fully connected layer*:** The CL along with the PL's output is inputted to the FCL. Via flattening, the PL's output is transmuted into a single vector since the FCL functions merely on data of 1D volume. Lastly, the probabilities of the input being in a particular class are produced by the softmax layer using the score value of each class as,

$$\phi_{sftm}^{lr} = \wp(\Omega_{wt}\Psi_{flat\ FM(n)} + bs) \tag{7}$$

$$\wp = \frac{x \cdot \exp(x)}{e^x \sum \exp(x)} \quad where,\ x = \Omega_{wt}\Psi_{flat\ FM(n)} + bs \tag{8}$$

Where, the classification output is signified as ϕ^{lr}_{sftm}, the weight vector is notated as Ω_{wt}, the flattened data is defined as $\Psi_{flat\ FM(n)}$, the layer's bias value is illustrated as bs, and the SS activation function is illustrated as $\wp$.

The output loss is determined after getting the output. The output is represented as the final output if the loss is extremely small. The optimum point is missed by the network if the loss occurred is high; thus, mitigating the learning rate. Consequently, the network parameters must be optimized. Therefore, the ADAM optimizer, which trains the network by updating the optimum weight values is employed here.

The learning rates are computed by the ADAM, which is an optimization algorithm. It then utilizes the gradients' first along with second moments to detect the optimal solutions. Regarding the fixed decay rates, which influence the algorithm's searching capacity, the moments of velocity were gauged in the conventional ADAM approach. Thus, the values are updated as of the particular range fixed by the linear scaling methodology that sequentially scales the database by adding resources that correlate to the augmented throughput. For the weights, the uncentered moments are determined as,

$$R^{u+1}_{\Omega_{wt}} = \lambda_1 R^{u}_{\Omega_{wt}} + (1-\lambda_1) h_u \tag{9}$$

$$T^{u+1}_{\Omega_{wt}} = \lambda_2 T^{u}_{\Omega_{wt}} + (1-\lambda_2) h^2_u \tag{10}$$

Where, the uncentered moments are mentioned as $R^{u}_{\Omega_{wt}}$, $T^{u}_{\Omega_{wt}}$, the exponential decay rates of the moment estimates are illustrated as λ_1, λ_2, the current training iteration is denoted as u, the gradient vector is indicated as $h_u \cdot \lambda_1, \lambda_2$ are updated as,

$$(\lambda_1, \lambda_2)_{LS} = \frac{\lambda - \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}} \tag{11}$$

The weight values are altered and the weights are updated after estimating the uncentered moments as,

$$\Omega^{u}_{opt(wt)} = \Omega^{u-1}_{(wt)} - \partial \frac{\overline{R}^{u+1}_{\Omega_{wt}}}{\sqrt{\overline{T}^{u+1}_{\Omega_{wt}}} + \tau} \tag{12}$$

$$loss = \frac{\sum(\psi - \hat{\psi})^2}{n} \tag{13}$$

Where, (ψ) represents the target output and $(\hat{\psi})$ represents the predicted output. Here, after modifying the weights, the first and second momentums obtained are described as $\overline{R}^{u+1}_{\Omega_{wt}} = \frac{R^{u}_{\Omega_{wt}}}{1-\lambda^{u}_1}$, $\overline{T}^{u+1}_{\Omega_{wt}} = \frac{T^{u}_{\Omega_{wt}}}{1-\lambda^{u}_2}$, a small scalar to

prevent division by zero is exhibited as τ, the optimal weight value is depicted as $\Omega^{u}_{opt(wt)}$, and the learning rate is termed as ∂. Thus, the loss function is mitigated by optimizing along with updating the weight values. Attacked data $\psi_{att(n)}$ and non-attacked data $\psi_{non-att(n)}$ are the '2' classes encompassed in the classification outcome. Here, the attacked data is amassed in the log file; similarly, the non-attacked data is inputted into the data prevention phase.

3.2 Wireless Sensor Network

To gather real-time data and to test with an attack detection system, the wireless SNs are initialized following the training procedure. The set of SNs SR_n initialized is proffered as,

$$SR_n = \{SR_1, SR_2, \ldots SR_N\} \tag{14}$$

Where, the number of SNs initialized is indicated as SR_n.

3.2.1 Clustering

The SNs are created as clusters since the SNs and base stations are located in the WSN. The process of partitioning the network region into a number of clusters is known as clustering. The CD-centric KMA is utilized for clustering. The data is separated into the required number of clusters by utilizing the K-Means, which is an unsupervised learning algorithm. Until finding the optimal centroid, the centroid and iterates are computed. Nevertheless, the accurate clusters are produced by the Euclidian distance utilized in the prevailing KMA because Euclidean distance calculates the distance between two real-valued vectors. Therefore, the CD is employed here for clustering. Following are the steps involved in clustering.

Step 1: By choosing the l number of centroids randomly, the number of clusters is estimated. The centroids are signified as,

$$\delta_l = \{\delta_1, \delta_2, \delta_3, \ldots, \delta_L\} \tag{15}$$

Where, the l number of centroids selected randomly is defined as δ_l.

Step 2: To form the clusters, every single data point is assigned to the nearest centroid. The distance betwixt every single data point and the centroid is computed to assign data points. It is computed as,

$$r = \max |(\delta_l - SR_n), (\delta_{l+1} - SR_{n+1})| \tag{16}$$

Where, the CD is illustrated as r.

Input: *Sensor Nodes* SR_n
Output: *number of clusters* C_l

Begin
Initialize sensor nodes SR_n
Determine number of clusters
Randomly select number of centroids
$$\delta_l = \{\delta_1, \delta_2, \delta_3, ..., \delta_L\}$$
For each remaining data
Compute distance to each centre
$$r = \max |(\delta_l - SR_n), (\delta_{l+1} - SR_{n+1})|$$
Assign data point to the nearest centre
End for
Return number of clusters C_l
End

Figure 2 Pseudo-code of the proposed Chebyshev-K-means algorithm.

Step 3: To set the new centroid of every single cluster, the variance is gauged.

Until forming the required number of clusters C_l, the aforementioned steps are repetitive. Figure 2 displays the proposed CD-KMA algorithm's pseudo-code.

Subsequent to clustering, the features are extracted from the clustered data. Then, the features being extracted are wielded for the removal of null features along with pre-processing in the training phase. After that, for testing, the pre-processed data is utilized.

3.3 Data Prevention

For preventing the data from several security attacks, this module, which is a structure of system security, is adopted. The TCSLR is implemented to safeguard the data from intruders. By employing this methodology, the data's ($IP_{\psi_{non-att(n)}}$) source along with the destination IP address is altered, which is then amassed for further evaluation. The process of performing a 1's complement operation followed by the inclusion of 1 to this outcome is mentioned as the 2's complement. At first, in the data source along with the destination IP address, 2's complement operation is performed. Next, the outcome obtained from the 2's complement is utilized for the left shift operation. In the left shift operation, by utilizing the operator $\ll$, the number

of bits is shifted to the left. Afterward, by utilizing the function Rev, the shifted bits are reversed. The process is illustrated as,

$$IP_{\psi_{non-att(n)}} \xrightarrow{1'scomp(IP_{\psi_{non-att(n)}})+1} 2'c(IP_{\psi_{non-att(n)}}) \xrightarrow{\ll} LS(IP_{\psi_{non-att(n)}}) \xrightarrow{Rev} Rev(IP_{\psi_{non-att(n)}}) \tag{17}$$

Where, the final modified IP address is specified as $Rev(inp)$.

4 Results and Discussion

Here, by performing various experiments on Java, the proposed IDPS' performance is analyzed. For performance assessment, the proposed technique utilizes the CICDDoS2019 data set, which consists of benign along with most modern general attacks that seem like real-world data. From the database, for testing together with training, 80% and 20% of data are utilized respectively. The system is employed on the working platform of JAVA with the system configuration of Intel i5/core i7 processor, CPU Speed: 3.20 GHz CPU Speed, Windows 10 System OS 64 bit type, and 4GB RAM. The dataset link is given as follows

https://www.unb.ca/cic/datasets/ddos-2019.html

4.1 Performance Analysis of the Clustering Algorithm

For the proposed Chebyshev-K-means technique and the prevailing Kernal-K-means, K-means, Constrained-K-means, along with Fuzzy-K-means techniques, the performance evaluation is executed on the basis of clustering time.

Discussion: In Figure 3, the proposed and prevailing techniques clustering time is given. For clustering, the proposed technique fetches 12744 ms, but the prevailing approaches take more times of 15468 ms, 18346 ms, 20446 ms, and 28576 ms for K-means, Fuzzy K-means, kernel K-means, and Constrained K-means, respectively. Thus, when analogized to other techniques, the proposed approach takes lesser time for clustering.

4.2 Performance Analysis of the Classification Method

Regarding sensitivity, specificity, accuracy, and F-measure, the proposed SS-LSACNN technique's efficacy is analogized with the prevailing

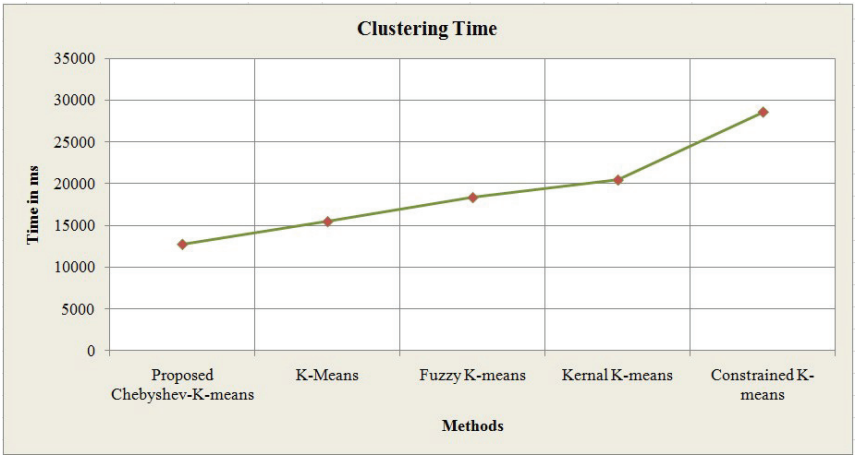


Figure 3 Illustrates the clustering time of the proposed and existing Clustering techniques.

Table 1 Clustering time of proposed model and existing models

Techniques	Clustering Time (ms)
Proposed Chebyshev-K-means	12744
K-Means	15468
Fuzzy K-means	18346
Kernel K-means	20446
Constrained K-means	28576

Table 2 Performance analysis of the proposed and existing methods

Methods	Accuracy	Sensitivity	Specificity	F-Measure
Proposed SS-LSACNN	98.15	97.59	95.72	95.48
CNN	94.36	96.02	94.96	94.67
ANN	93.02	95.55	94.19	93.44
RNN	92.42	92.75	92.78	91.34
DNN	91.26	91.74	90.8	90.84

Convolution Neural Network (CNN), Artificial Neural Network (ANN), Recurrent Neural Network (RNN), along with Deep Neural Network (DNN) techniques.

Discussion: The performances of the proposed together with the prevailing techniques are evaluated concerning f-measure, sensitivity, accuracy, and specificity, which should be higher for accurate detection. When analogized

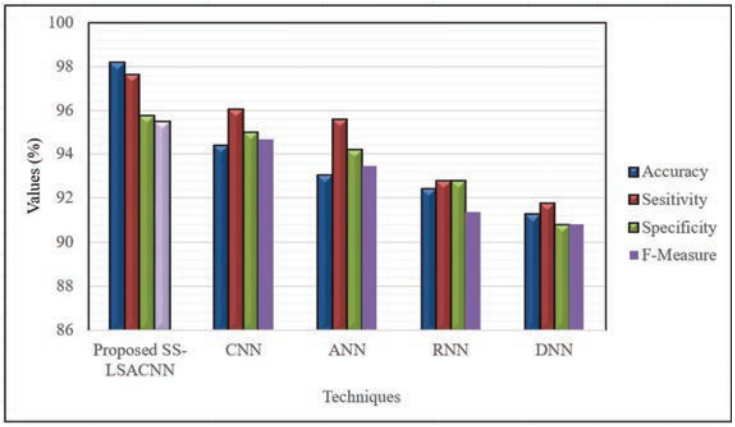
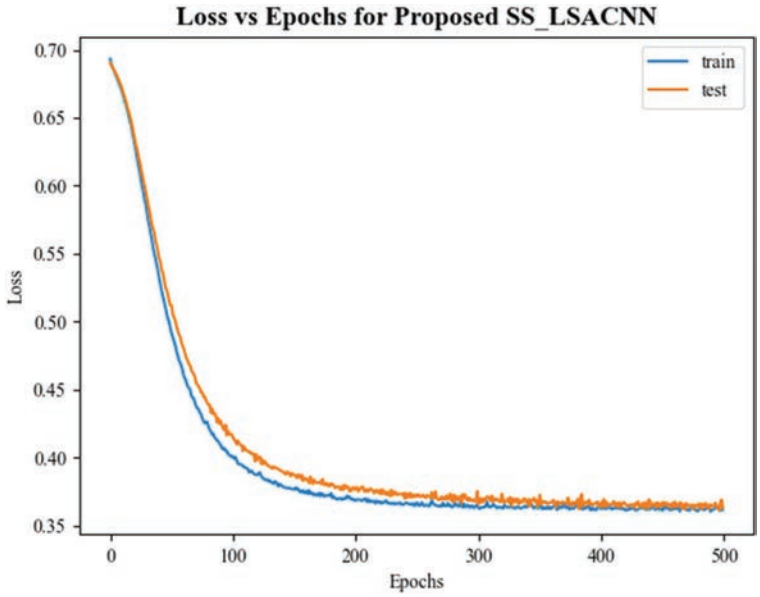


Figure 4 Performance analysis of the proposed model and existing models.

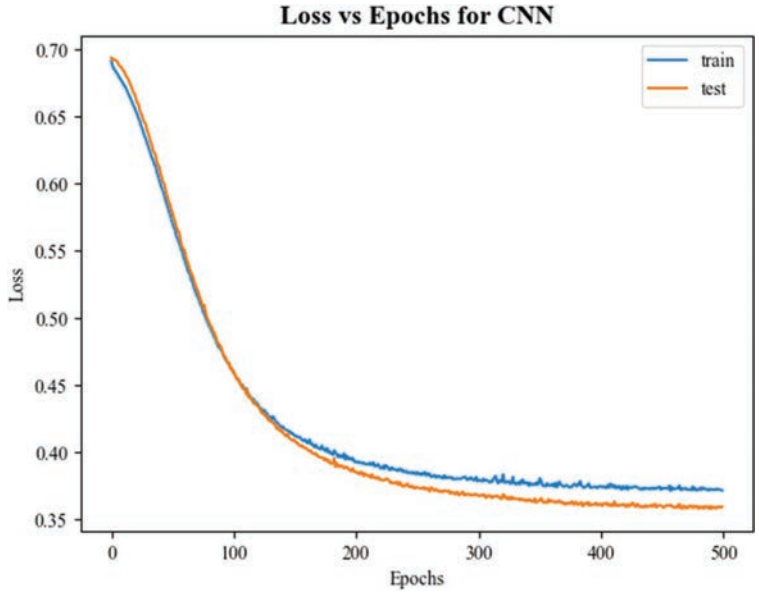
to the prevailing techniques like CNN, ANN, RNN, along with DNN, the proposed SS-LSACNN gives high specificity, sensitivity, accuracy, along with F-Score. The DDoS attack is detected by the proposed approach with an accuracy of 98.15, but the prevailing techniques like CNN, ANN, RNN, and DNN attain less accuracy of 94.36, 93.02, 92.42, and 91.26 respectively. For the proposed approach, the sensitivity together with specificity values are 97.59 and 95.72; and regarding F-measure, it has the highest value of 95.48. Hence, regarding every performance metric, the proposed SS-LSACNN attains enhanced outcomes.

Figure 5 illustrates the training loss and validation loss for the proposed SS-LSACNN, and existing CNN, ANN, RNN, and DNN. The existing RNN and DNN model causes underfitting as both the training and validation loss are high. It indicates that the CNN model performs well on testing data but poorly on the training data. This indicates that the model is overfitting, and cannot be generalized on new data. But, the proposed model indicates an optimal fit, as the training loss and validation loss both decreases and stabilizes at a specific point i.e., a model does not overfit or underfit.

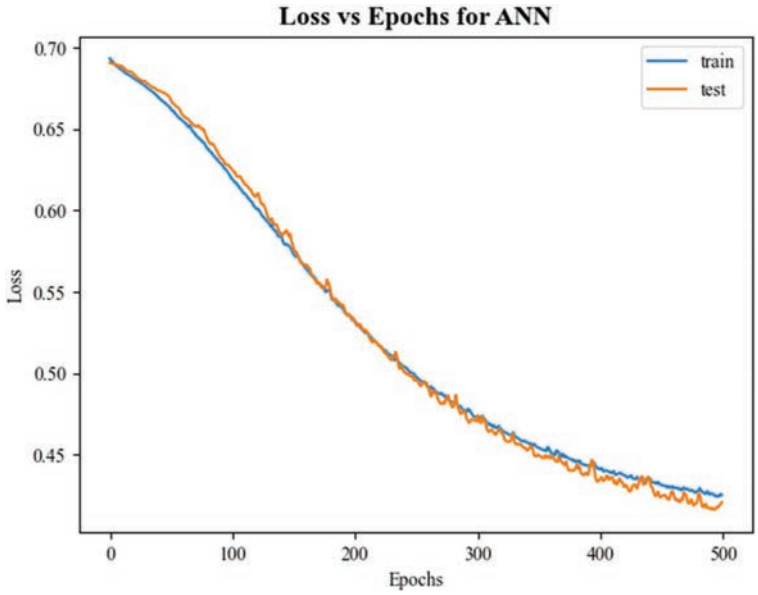
Figure 6 illustrates the training accuracy and validation accuracy for the proposed SS-LSACNN, and existing CNN, ANN, RNN, and DNN. The training and validation accuracy for the existing RNN and DNN models are low. But, the proposed model indicates good training accuracy and validation accuracy. Thus, the proposed attains better results.


Loss vs Epochs for Proposed SS_LSACNN
train
test
Loss
Epochs
0.70
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0
100
200
300
400
500

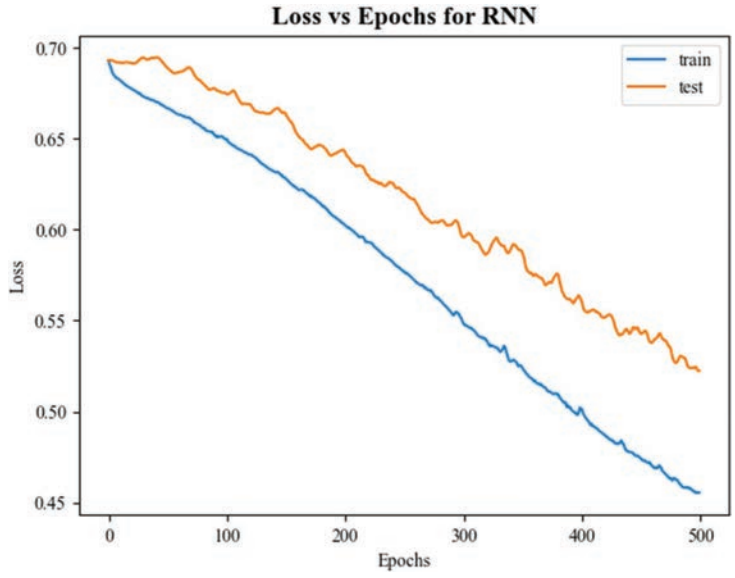
(a)


Loss vs Epochs for CNN
train
test
Loss
Epochs
0.70
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0
100
200
300
400
500

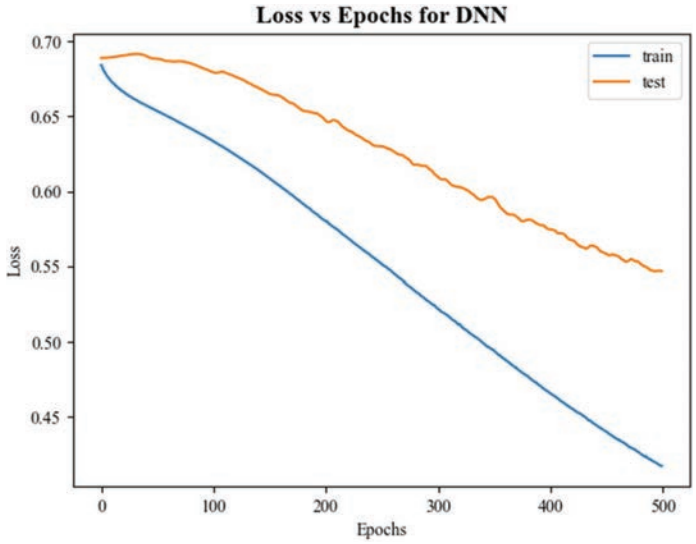
(b)


Loss vs Epochs for ANN
train
test
Loss
Epochs
0.70
0.65
0.60
0.55
0.50
0.45
0
100
200
300
400
500

(c)


Loss vs Epochs for RNN
train
test
Loss
Epochs
0.70
0.65
0.60
0.55
0.50
0.45
0
100
200
300
400
500

(d)



(e)

Figure 5 Training and validation loss for proposed and existing classifiers.

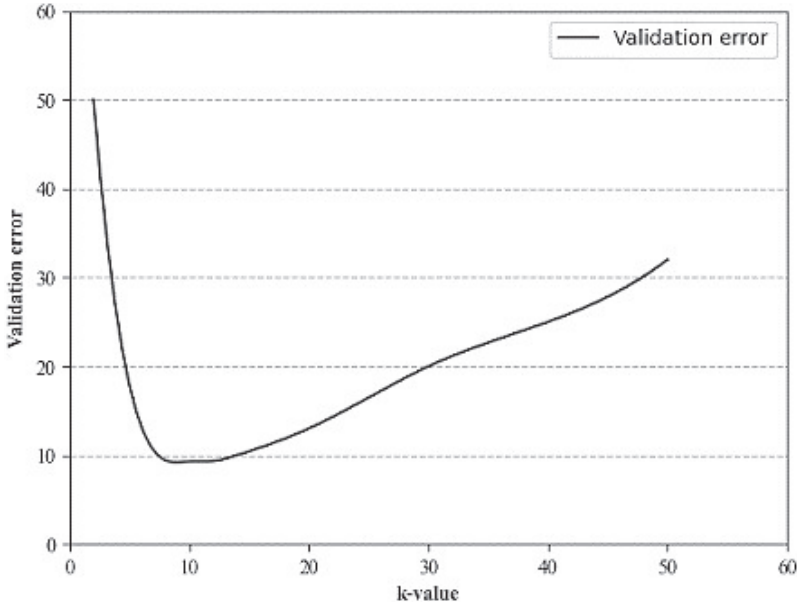


Figure 6 Validation error of KNN algorithm.

4.3 Comparative Analysis with the Literature Papers

In this section, the proposed SS-LSACNN is compared with the literature papers such as Multi-kernel extreme learning machine (MK-ELM) [12], and binary grey wolf optimizer with support vector machine- intrusion detection system (GWOSVM-IDS (7 wolves)) [18].

Table 3 depicts the comparative assessment of the proposed SS-LSACNN with the literature papers regarding accuracy. The proposed technique achieves 98.15% and existing techniques attain accuracy of 97.85% and 96.00%. Hence, it is clear that the proposed system attains superior results in terms of accuracy.

Table 3 Comparative analysis of the proposed SS-LSACNN with the literature papers

Techniques	Accuracy (%)
Proposed SS-LSACNN	98.15
MK-ELM [12]	97.85
GWOSVM-IDS [18]	96.00

5 Conclusion

To damage the user's data along with the application, the vulnerability in the network and the protocol are utilized by numerous attacks. In the WSN, an SS-LSACNN in conjunction with TCSLR methodologies-centric IDPS is proposed for detecting along with preventing the WSN from several intrusions. (i) The Intrusion detection training phase, (ii) the intrusion detection testing phase, and (ii) the intrusion detection prevention phase are the 3 phases included in the proposed model. The DDoS attacks that existed in the data are identified and data confidentiality is ensured by the proposed framework. In performance analysis, the proposed SS-LSACNN is analogized with various other prevailing methodologies. The experiential outcomes display that whilst validating along with evaluating the performance metrics with graphical outcomes, the proposed model obtained better outcomes. An accuracy rate of 98.15 % is achieved by the proposed SS-LSACNN. In accordance with the outcomes, it is established that the proposed methodology is extremely effective, secure, along with highly accurate than the other methodologies. To elevate the security of data transmission in the WSN, the proposed framework might be expanded by pondering the cryptography algorithm in the future.

Declarations

Funding: Authors did not receive any funding.

Conflicts of interests: Authors do not have any conflicts.

Data Availability Statement: No datasets were generated or analyzed during the current study.

Code availability: Not applicable.

Authors' Contributions: All authors contributed to the design and methodology of this study, the assessment of the outcomes, and the writing of the manuscript.

References

[1] Gavel, S., A. S. Raghuvanshi, and S. Tiwari. 2020. A novel density estimation based intrusion detection technique withPearson's divergence for Wireless Sensor Networks. *ISA Transactions* (Pre-proof). https://doi.org/10.1016/j.isatra.2020.11.016.

[2] Premkumar, M., and T. V. P. Sundararajan. 2020. DLDM Deep learning-based defense mechanism for denial of service attacks in wireless sensor networks. *Microprocessors and Microsystems*. 79(8): 1–10.

[3] Prabakaran K, N. Kumaratharan, P. S. D. Epsiba. 2020. An evaluation of effective intrusion DoS detection and prevention system based on SVM classifier for WSN. *IOP Conference Series Materials Science and Engineering*. 925(1): 1–11.

[4] Bisen, D., B. Barmaiya, R. Prasad and P. Saurabh. 2021. Detection and prevention of black hole attack using trusted and secure routing in wireless sensor network. Springer, Cham, 1st Edition, ISBN: 978-3-030-49335-6.

[5] Borkar, G. M., L. H. Patil, D. Dalgade and A. Hutke. 2019. A novel clustering approach and adaptive SVM classifier for intrusion detection in WSN a data mining concept. *Sustainable Computing Informatics and Systems*. 23(5): 120–135.

[6] Ajeetha, G., and G. M. Priya. 2019. Machine learning based DDoS attack detection. *Innovations in Power and Advanced Computing* Technologies (i-PACT), 22–23 March, Vellore, India.

[7] Belej, O. 2020. Development of a technique for detecting distributed denial-of-service attacks in security systems of wireless sensor network.

15th International Conference on Computer Sciences and Information Technologies (CSIT), 23–26 September, Zbarazh-Lviv, Ukraine, 2020.

[8] Liu, G., H. Zhao, F. Fan, G. Liu, Q. Xu and S. Nazir. 2022. An enhanced intrusion detection model based on improved kNN in WSNs. *Sensors*. 22(4): 1–18.

[9] Narayanan, K. L., S. R. Krishnan, E. G. Julie, Y. H. Robinson and V. Shanmuganathan. 2021. Machine learning based detection and a Novel EC-BRTT algorithm-based prevention of DoS attacks in wireless sensor networks. *Wireless Personal Communications*. https://doi.org/10.1007/s11277-021-08277-7.

[10] Mehetre, D. C., S. E. Roslin and S. J. Wagh. 2018. Detection and prevention of black hole and selective forwarding attack in clustered WSN with active trust. *Cluster Computing*. https://doi.org/10.1007/s10586-017-1622-9.

[11] Pajila, P. J. B., E. G. Julie and Y. H. Robinson. 2021. FBDR-Fuzzy based DDoS attack detection and recovery mechanism for wireless sensor networks. *Wireless Personal Communications*. https://doi.org/10.1007/s11277-021-09040-8.

[12] Zhang, W., Han, D., Kuan-Ching Li and Massetto, F. I. 2020. Wireless sensor network intrusion detection system based on MK-ELM. *Soft Computing*. 24(1): 12361–12374.

[13] Jiang, S., Zhao, J., and Xu, X. 2020. SLGBM an intrusion detection mechanism for wireless sensor networks in smart environments. *IEEE Access*. 8: 169548–169558.

[14] Premkumar, M., and Sundararajan, T.V.P. 2020. DLDM: Deep learning-based defense mechanism for denial of serviceattacks in wireless sensor networks. *Microprocessors and Microsystems*.79: 1–10.

[15] Narayanan, K.L., Krishnan, R.S., Julie, E.G., Robinson, Y.H., and Shanmuganathan, V. 2022. Machine learning based detection and a novel EC-BRTT algorithm based prevention of dos attacks in wireless sensor networks. *Wireless Personal Communications*. 127: 479–503.

[16] Ademola, P. A., and Kabaso, B. 2021. Lightweight models for detection of denial-of-service attack inwireless sensor networks. *IET Networks*. 10: 185–199.

[17] Suryaprabha, E., and Kumar, N. M.S. 2020. Enhancement of security using optimized DoS (denial-of-service)detection algorithm for wireless sensor network. *Soft Computing*. 24: 10681–10691.

[18] Safaldin, M., Otair, M., andAbualigah, L. 2021. Improved binary gray wolf optimizer and SVM for intrusion detectionsystem in wireless sensor networks. Journal of Ambient Intelligence and Humanized Computing. 12: 1559–1576.

[19] Balaji, S., and Sasilatha, T. 2019. Detection of denial of service attacks by domination graph applicationin wireless sensor networks. *Cluster Computing*. 22: 15121–15126.

[20] Sinha, S., and Paul, A. 2020. Neuro-fuzzy based intrusion detection system for wireless sensor network. *Wireless Personal Communications*. 114: 835–851.

[21] Siva, S.S.S., Geetha, S., and Kannan, A. 2012. Decision tree based light weight intrusion detection using a wrapper approach. *Expert Systems with Applications*. 39: 129–141.

[22] Elreedy, D., and Amir, F. Atiya. 2019. A novel distribution analysis for SMOTE oversampling method in handling class imbalance. 1st ed, ISBN: 978-3-030-22743-2. Springer, Cham.

[23] Thara, D. K., Sudha, B. G. P., and Fan Xiong. 2019. Auto-detection of epileptic seizure events using deep neural network with different feature scaling techniques. *Pattern Recognition Letters*. 128: 544–550.

[24] Abdulkareem, S.S., Akgul, A., Jalal, V.J., Faraj, B.M., and Abdullah, O.G.H. 2020. Numerical solution for time period of simple pendulum with large angle. *Thermal Science*. 24: 25–30.

[25] Faraj, B., and Modanli, M. 2017. Using difference scheme method for the numerical solution of telegraph partial differential equation. *Proceedings in Journal of Garmian University*. pp. 157–163, Iraq.

[26] Faraj B.M., and Ahamed, F.W. 2019. On the matlab technique by using laplace transform for solving second order ode with initial conditions exactly. *Matrix Science Mathematic*. 3(2): 8–10.

[27] Modanli, M., Faraj, B.M., and Walyahamed, F. 2019. Using matrix stability for variable telegraph partial differential equation. *An International Journal of Optimization and Control: Theories and Applications*. 10(2): 237–243.

Biographies

Vikash Kumar Singh received his PGDBA specialization Operation from the Symbiosis Pune. Pursued BTech from (NIT PATNA) IT specialization. Currently having of 7.5 years, experience. Research interest includes AI, Machine Learning, Data science.

Durga Sivashankar received her PGDBA specialization Operation from the Symbiosis Pune. Pursued B.E from (GEC Gandhinagar) Instrumentation and Technology specialization. Currently having of 6 years, experience. Current research interest includes AI, IOT, Machine learning.

Kishlay Kundan received his B. Tech CSE (NIT PATNA-2011) 11 years, experience in software industry. Current research interest includes System programming, Cryptography, AI.

Sushmita Kumari have completed B.Tech in CSE and have well experience in IT Industry with over an experience of 3 years.