sciendo Journal of Electrical Engineering, Vol. 76, No. 1, 2025, pp. 34-47

Cross-layer DDoS attack detection in wireless mesh networks using deep learning algorithm

Anil Kumar Gankotiya[1], Vishal Kumar[2], Kunwar Singh Vaisla[2]

Wireless mesh networks (WMNs), owing to its decentralized design and resource limitations, are susceptible to several security vulnerabilities, including distributed denial of service (DDoS) attacks. Traditional DDoS detection techniques are usually unable to effectively mitigate such attacks in WMNs due to their dynamic and complex nature. In this work, we show the capability of a Deep Convolutional Neural Network (DCNN) algorithm at the cross-layer of the network protocol stack to accurately and robustly detect Distributed Denial-of-Service (DDoS) attacks in WMNs. DDoS attack assessment and recognition use a practical dataset varying standard actions such as end-to-end delay, energy consumption, packet delivery ratio, mean packet latency, detection ratio, and packet loss rate when using the CICDDoS2019 dataset. The result shows the proposed method's strong performance compared to previous detection methods. The simulation results show DCNN-DDoS has a better detection ratio metric than D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML, which grew by 78.12%, 38.54%, 22.8%, 16.33%, and 15.67% respectively. DCNN-DDoS has exhibited superior performance compared to other essential methods, which is evident from the empirical results, which have higher levels of accuracy.

Keywords: convolutional neural networks, wireless mesh networks, DDoS attacks, deep learning, malicious nodes

1 Introduction

The ability of wireless mesh networks (WMNs) to provide wider coverage and flexible connections in various settings has contributed to their meteoric rise in popularity [1]. In WMNs, a self-organized network architecture is formed by many wireless nodes linked to one another. The network's dispersed structure provides many advantages, including enhanced resilience, higher capacity, and efficient routing [2-3]. Distributed denial of service (DDoS) assaults is major cause for worry regarding the security of WMNs, which are vulnerable to various threats due to their openness and dynamic topology. Consequently, to defend WMNs against vulnerabilities of this kind, it is necessary to have security procedures that are both resilient and efficient [4]. Traditional security methods developed for wired networks or wireless networks with a single hop are unsuitable for WMNs. Customized security solutions are required because of the distinctive qualities of WMNs, which include their dynamic topology and resource-constrained nature. Learning discriminative character-ristics and reliably identifying distributed denial of service attacks in real time was made possible by the model by using the geographical correlation of network traffic data [5].

The DDoS attack's characteristics include the overwhelming network resources, which renders the targeted services or systems unavailable to lawful users. These attacks can disrupt the normal operation of WMNs, degrade network performance, and compromise the availability of critical services [6]. Traditional DDoS detection methods designed for wired networks often need to address the unique challenges WMNs pose. Therefore, there is a need to develop advanced detection techniques specifically tailored for wireless mesh environments. Mesh nodes provide a significant challenge for researchers in managing data transmission through routing chains [7]. In addition, wireless mesh networks offer their functionalities within an un-restricted framework, and malevolent nodes have the potential to act as mesh routers to forward data. In this scenario, there exists the potential for multiple forms of denial of service (DoS) attacks. Consequently, network communication may be disrupted and jeopardized [8-9].

This research is motivated by a pressing requirement to enhance the security of WMNs against the increasing risk of DDoS attacks. WMNs are crucial as essential communication infrastructures in various applications, such as wireless network deployments and disaster recovery situations. Nevertheless, the decentralized and ever-changing nature of these systems makes them susceptible to advanced cyber threats, such as DDoS assaults, that can potentially interrupt network services and undermine the integrity of data [10-11]. To render a target system's services unavailable, attackers employ a multitude of dump terminals, machines, and botnets to initiate DDoS attacks. This simultaneous onslaught effectively depletes the primary resources of the target system. A wide range of legitimate and potent tools exist that have the potential to be misused for launching DDoS attacks on both large and small targets. In a recent DDoS

[1] VMSB Uttarakhand Technical University, Dehradun, India
[2] Bipin Tripathi Kumaon Institute of Technology, Almora, India
anil.gankotiya@galgotiasuniversity.edu.in, vishalkumar@kecua.ac.in, ksvaisla@kecua.ac.in

https://doi.org/10.2478/jee-2025-0004, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

attack, the perpetrators exploited the legitimate functionality of the Memcached tool, which is primarily designed to alleviate the burden on the underlying network resources [12-15].

This paper highlights a new DCNN-based framework (DCNN-DDoS) for efficiently detecting and defending against DDoS attacks targeting WMNs. The conventional approaches concentrate on interpreting values from just one domain to improve detection accuracy, while the proposed method analyses metrics like energy usage, packet delivery ratio, end-to-end delay, and mean packet latency representative of cross-layer data for achieving secure networks. In the experiment, validated by the CICDDoS2019 dataset, DCNN-DDoS presents better performance than state-of-the-art methods, i.e., D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML while achieving higher detection ratio, lower packet loss, and high routing efficiency. Finally, this research enables a scalable and effective solution for robust security against challenging cyber threats in next-generation WMNs. The DCNN-DDoS model proposed in this study can realize the practical implementation of real-time threat detection and adaptive threat mitigation for WMNs, strengthening the network resilience to dynamic and sophisticated cyberattacks. The growing use of IoT and 5G technologies makes extending the framework to accommodate high-density, low-latency networks important. The resulting research can develop into an optimization for detecting and preventing multi-vector DDoS attacks in such advanced network environments.

The primary objective of this research paper is to propose and assess a DCNN model for the efficient detection of DDoS attacks in wireless mesh networks. By leveraging the inherent spatial correlation of network traffic data, the proposed model aims to learn discriminative features and accurately identify DDoS attacks in real-time. The CICDDoS2019 dataset [16] is substantial in size. Sixty percent of the data used during training is chosen randomly, whereas all data is used during testing. The dataset known as CICDDoS2019 is exclusively composed of flow-based data and is considered to be at the forefront of current research and technological advancements. However, these datasets lack the critical flow-based characteristics and qualities WMNs require. Existing significant techniques are compared with the suggested methodology in the evaluation. The f1-score, recall, accuracy, and precision are defined metrics upon which this evaluation is based.

The primary contributions of this paper are as follows:

- Proposed DCNN-DDoS method aims to detect DDoS attacks through a cross-layer architecture spanning network layers in WMNs.
- The DDoS attack evaluation and detection are measured based on various metrics, such as end-to-end delay, energy consumption, packet delivery ratio, mean packet latency, detection ratio, packets lost rate, malicious nodes, and accuracy, using a real-world dataset.
- This paper presents the DDoS attack detection and evaluation methods, focusing on deep CNN approaches that use the CICDDoS2019 dataset.
- The evaluation of the efficacy of the proposed method in comparison to state-of-the-art methods such as D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML with a focus on achieving a high level of detection accuracy.
- It secures communication for traffic monitoring, surveillance, and smart grids in the intelligent network and also improves the reliability of emergency communication networks during disaster response.

The remaining parts of the paper are organized as shown in the following structure. Section 2 provides a comprehensive literature review. Section 3 presents the proposed methodology. In Section 4, performance result analysis is presented, and the conclusion and future research in Section 5.

2 Literature review

WMNs are susceptible to a range of security risks as a result of their distinctive attributes. The inherent dynamism of WMNs, characterized by the dynamic addition and removal of nodes, poses issues in ensuring secure communication and avoiding unwanted intrusion. In addition, using a shared wireless channel increases the vulnerability to eavesdropping, unauthorized node impersonation, and packet interception. WMNs' weaknesses provide them appealing targets for attackers, including DDoS assaults [17-18]. Table 1 shows the abbreviation description.

Mahadik et al. [18] introduced an intelligent intrusion detection system (IDS) called HetIoT-CNN IDS. This IDS system employs a convolutional neural network (CNN) built around deep learning methodologies. The HetIoT-CNN IDS is tailored to function inside the HetIoT ecosystem. This technique primarily emphasizes the binary categorization of assaults, with a remarkable accuracy rate of 99.75%. A security solution based on Convolutional Neural Networks (CNN) has been created to combat DDoS threats in real time inside the IoT environment. The technique proposed achieves a binary classification accuracy rate of 99.9% in recognizing DDoS assaults.

Table 1. Abbreviations

Abbreviation	Description
DCNN	Deep Convolutional Neural Network
DDoS	Distributed Denial-of-Service
D-ConCReCT	Distributed congestion control by duty-cycle restriction
SVM-DoS	Support vector machine-based denial-of-Service
FSO-LSTM	Firefly-swarm-optimized long short-term memory
AIDS-HML	Advanced intrusion detection systems based on hybrid machine learning
T_k	The total number of convolutional layers is denoted by k
$G_{i\ \ L,M}^{j}$	Fixed feature map from the G^{th} layer, cantered at coordinates L and M
G_i^j	i^{th} non-linear layer with feature maps
H_n^j	Expected output
$E_{i\,,w}^{h}$	The relationship between the j^{th} component in layer i and a unit in the w^{th} characteristic vector
W_k	Weight of the kernel size k
T_w	Defined as the input weights
b_k	Bias value of kernel k
y_t	The initial instant of the gradient

Sharma et al. [19] developed a security technique for protecting the RPL model against black hole attacks. One encouraging step towards guaranteeing the security of these networks is the RPL mechanism's use of sophisticated capabilities like 6LoWPAN network discovery. Modern, cutting-edge solutions exist to counter these dangers, and they do not have to be cumbersome or disable certain nodes' ability to function. Because of the promiscuity of certain nodes, this issue has solutions that might compromise network security. Even though ID systems could need substantial processing and network overheads, there are ways to address these challenges. The proposed study employs a promising strategy for malicious node detection in the distributed timer-based technique. The simulation approach has been used to evaluate the work, and the results demonstrate that it can accurately locate black holes, leading to less packet loss [20].

Gandhimati et al. [21] created a model that examines threat detection using a cross-layered method and flow-based intrusion detection system. To enhance the communication security of the crucial application, it should consider replacing its current cryptographic technology with flow-based anomaly detection. Moreover, transitioning from a single-layer detection strategy to a multi-layer detection approach might be advantageous in enhancing attack detection. The technology evaluated has a two-stage approach to detection. The authors used a flow-based approach to look for strange behavior in the network's traffic during the first stage. The second step involves analyzing cross-layer properties to lessen the likelihood of assaults. Regarding detection precision, the suggested detection technique shows promising results in simulation. Compared to layer-based and packet-based methods, this will also result in lower energy usage and a lower false positive rate [22].

Ramesh et al. [23] presented a DoS attack on Wireless Multimedia Sensor Networks (WMSN) utilizing an improved DNN system. The selection of parameters is performed utilizing the adaptive PSO method. The method's efficacy is evaluated by assessing various factors, including energy consumption, throughput, packet delivery ratio, latency, and network longevity. The obstacles sensor nodes rise to many attacks on Wireless Multimedia Sensor Networks (WMSN), notably DoS attacks. The crux of this assault is to hinder the adequate operation of the system. The primary concern is launching attacks and restricting the access of authorized nodes to network resources. The distributed attack refers to a scenario where multiple assailants attack a network. The present form of assault engenders more functional complications within the network than attacks directed toward a solitary node. Introducing an advanced machine learning algorithm is imperative to mitigate the risks of DoS attacks on the network.

Liu et al. [24] introduced an intelligent IDS method for Wireless Sensor Networks (WSNs). Their approach uses machine learning's k-NN algorithm and evolutionary computation's AOA [26]. This integration creates a WSN DoS detection edge intelligence architecture. To increase model accuracy, use a parallel technique for inter-population communication and the Lévy flight strategy for optimization modifications. In the benchmark function test, the PL-AOA algorithm improves the kNN classifier. Borgiani et al. [25] introduced the DConCReCT, a distributed variant of the ConCReCT mechanism. The goal was to exhibit the practicality of implementing the DConCReCT in time-constrained critical situations and extensive IoT-driven WSNs. Empirical investigations demonstrate that utilizing a decentralized approach leads to a re-duction in detection and mitigation time when compared to the centralized approach.

Additionally, the proposed mechanism demonstrates the capability to function effectively in networks of up to 500 nodes, even when resources are limited. In addition, deploying D-ConCReCT allows for identifying and reducing multi-target assaults by allocating these tasks to various nodes. Using the ConCreCT in large-scale WSNs with limited resources like memory, computing power, and battery life is not entirely shown by the aforementioned research, which is especially problematic in time-sensitive and essential scenarios [27]. Table 2 shows the reference study of various methods advantages and research gaps.

Table 2. Reference study of various methods advantages and research gaps

References	Methods	Attacks type	Advantages	Research gap
Gowdhaman et al. [36]	Deep recurrent neural network and SVM	Probing attack and DoS attack	Data transmission with low latency.	Low accuracy and used one dataset.
Almomani et al. [37]	SHO with LSTM model	Various attacks such as malicious node	Integration of metaheuristic algorithm.	Lack of specific measurable results.
Naser et al. [38]	Classification methods using ML	Cyber attacks	Scalability of the network improved.	Impact of network topology is not analysed.
Maheswari et al. [39]	Hybrid deep learning model	Jamming and DoS attacks	Reduce the jamming flooding	Overhead the information and low accuracy.
Gankotiya et al. [40]	Hybrid DAD	DoS attack and malicious node	Detection of duplicate address.	Overhead the wait time.
Premkumar et al. [41]	Deep radial basis network	DoS attack and probing	Work better with changing topologies.	Increased lag time for first connections

3 Materials and methods

This section develops a deep learning-based DDoS attack detection system for Wireless Mesh Networks. To train and evaluate the proposed model, the dataset contains typical traffic patterns and numerous simulated DDoS assaults. It details the DCNN-DDoS architecture and algorithm through data collection and preprocessing, feature selecting, data cleaning, memory optimization, and feature grading.

3.1 Data collection and preprocessing

The CICDDoS2019 dataset is used in research on deep CNN based identification and alleviation of DDoS assaults. The dataset consists of regular and malicious traffic examples, covering ten different categories of assaults. Each dataset contains the precise name of the assault, the number of samples that indicate the attack's magnitude, and the corresponding characteristics. The efficacy of a learning mechanism relies on the pre-processing of data conducted [29]. The study used the following dataset preparation procedures to develop a DCNN-DDoS model.

Feature selection is discovering a given dataset's most relevant and informative characteristics. The method aims to enhance performance, and selecting features is a critical data preprocessing technique that plays a vital role in lowering the number of features and improving performance [30]. The investigation included all aspects except Flow Id, Timestamp, Similar HTTP, Destination IP, and Source IP. The decision to introduce exclusions was driven by either inherent shortcomings in the informative content of the features or the deliberate creation of these attributes within a simulated environment [31]. During program execution, memory optimization is used to deallocate or free up memory. When working with massive datasets, it is not uncommon to get an out-of-memory issue when developing a model. To make sure the model can manage the dataset effectively with the resources it has, memory optimization is an essential technique. Data cleaning is a crucial operation in data processing, aiming to eliminate occurrences of infinity, NAN, and null values from the dataset. The values are substituted with constant numerical values. Data preparation involves a critical step called feature scaling. The Standard Scaler is generally acknowledged as the predominant method for feature scaling. The DCNN-DDoS model incorporates the use of the Standard Scaler.

The CICDDoS2019 dataset is significant for training purposes; 60% of the data from the training dataset is randomly selected. Conversely, the entire testing dataset is utilized without any reduction. The datasets above are utilized for classification. This research collects a representative dataset of network traffic in wireless mesh networks. The dataset should include normal network traffic and instances of DDoS attacks. The data collection can be performed in a controlled testbed or by capturing real-world network traffic data, i.e., represented in Table 3. Ensuring the dataset covers various network conditions and attack scenarios is essential.

Table 3. The CICDDoS2019 dataset details

Sl. No.	Sample size	Name of the various attacks file	Features
1	987,128	DrDoS_NTP	64
2	2,524,422	DrDoS_MSSQL	64
3	4,062,418	DrDoS_DNS	64
4	16,107,739	TFTP	64
5	2,636,051	DrDoS_UDP	64
6	270,602	UDP-Lag	64
7	1,011,573	DrDoS_LDAP	64
8	1,377,682	Syn	64
9	3,124,986	DrDoS_NetBIOS	64
10	4,761,372	DrDoS_SNMP	64

Once the dataset is collected, preprocessing steps are applied to prepare the data for training the deep CNN model. It involves cleaning the data, removing irrelevant features, and normalizing the input to a suitable range.

3.2 The proposed DCNN method

The Deep Convolutional Neural Network (DCNN) method used convolutional layers skill to detect potential features in data which makes it best suitable for difficult tasks as DDoS detection. In the literature, DCNN is designed to process network traffic data from various OSI model layers in the context of Wireless Mesh Networks (WMNs) learning intricate correlations among diverse behaviors to identify possible DDoS attack patterns. The CNN architecture, exploiting its deep structure, learns a hierarchal set of features, from the core statistical properties related to network traffic toward more abstract and high-level ones that can represent attack aspects. This staged learning approach allows the model to highly effectively separate legit traffic from attack traffic.

The input layer of the DCNN architecture: This is the initial step in the entire structure which receives raw network traffic data as input. The data from the various layers of WMN like, MAC layer, are preprocessed in to an appropriate format usually as a multi-dimensional matrix. Each input vector may correspond to various traffic characteristics such as the packet duration, arrival time, and some protocol-specific features. The cross-layer one enhances the input data, which offers a variety of information to reveal patterns in CNN through the optimized process to detect DDoS attacks. Such multi-layered input is crucial in order to allow CNN be able to learn relationships of higher layers throughout the network.

DCNNs are built upon a network where the heart of it lies within its convolutional layers, that is, each layer learns to detect local patterns in the input data. A stack of convolutional layers each process the image with a set of filters that slide across traffic volumes, disrupted packet rates or high latency suggesting an intrusion. At the beginning layers, these filters capture low-level patterns like packet anomalies, while in deeper layers learn abstract features such as complex attack behaviors that spread between different network layers. Upon performing convolution to an input, the output is passed through an exponential linear activation function which introduces non-linearity and aids the model in sifting negative from attack traffic. An input to the convolutional layer is signified as Q and can be represented by

$$T = \{T_1, T_2, \ldots T_b, \ldots, T_k\} \tag{1}$$

The total number of convolutional layers is denoted by k in this case. Equ. (2) shows that the convolutional layers take an input into account before producing an output, and that unit (L, M) also generates an output.

$$G_i^j{}_{L,M} = U_i^j{}_{L,M} + \sum_{w=1}^{K_1^{w-1}} \sum_{q=-K_1^h}^{L_1^h} \sum_{t=-K_2^h}^{K_2^h} E_{i\,,w}^{h\,*}\, G_w^{h-1}{}_{L+q,M+t} \tag{2}$$

Here, * represents the convolutional operator. $G_i^j{}_{L,M}$ refers to the fixed feature map from the G^{th} layer, centered at coordinates L and M. Consider that the CNN optimizes the weights of convolutional layers $E_{i\,,w}^h$ and bias $U_{i\ \ L,M}^{j}$. DCNNs are built upon a network where the heart of it lies within its convolutional layers, that is, each layer learns to detect local patterns in the input data. A stack of convolutional layers each process the image with a set of filters that slide across traffic volumes, disrupted packet rates or high latency suggesting an intrusion. At the beginning layers, these filters capture low-level patterns like packet anomalies, while in deeper layers learn abstract features such as complex attack behaviors that spread between different network layers. Upon performing convolution to an input, the output is passed through a exponential linear activation function which introduces non-linearity and aids the model in sifting negative from attack traffic. The expression for the i^{th} non-linear layer with feature maps is

$$G_i^j = fun\left(G_w^{j-1}\right) \tag{3}$$

Then, to decrease the number of parameters for computation and make the model invariant to small changes in input data, The pooling layers are added following convolutional layers. Pooling: Pooling is a down-sampling operation that reduces the size of the feature map while retaining its important features; by taking in our input image, applying filters to get feature maps and then applying pooling on those feature maps we extract useful information from these images called Max-pooling. commonly used techniques such as max-pooling which simply extracts only dominant features i.e. it takes the maximum value that covers some patch of your picture therefore reduces spatial resolution. The pooling layers do help summarize the important characteristics of our network which allows us to store it memory wise and thus improve generalization with unseen traffic data, as opposed to benchmark data. The output of a wholly linked layer is expressed as

$$H_n^j = J\left(G_i^j\right) + G_{i\ \ L,M}^{j} \tag{4}$$

In this case, H_n^j is the expected result, and $E_{i,w}^h$ is the relationship between the j^{th} component in layer i and a unit in the w^{th} characteristic vector of layer $(i-1)$. Figure1 illustrates the structure of Deep Convolutional Neural Networks.

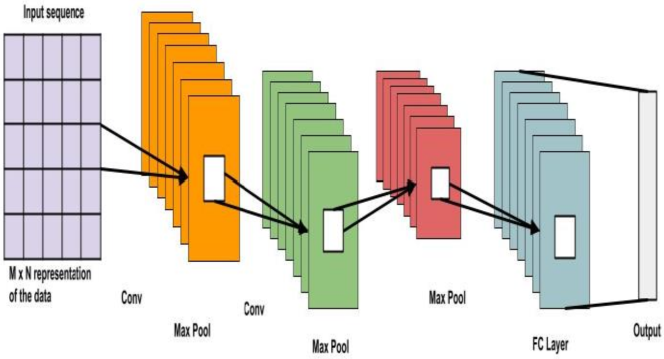

Fig. 1. Architecture of deep CNN

3.3 Training and evaluation

This training of DCNN involves tweaking the parameters of that model such as to minimise the error while predicting DDoS attacks. The model is trained on labeled network traffic data, including normal and attack samples. This loss is regularized using a specific loss function to establish the dissimilarity between the predicted output or target output and then may specify an optimization methodology like, Adam. For this to happen during training, we use techniques like dropout and batch normalization to prevent overfitting as showing below so the model generalizes well on unseen data. The stability of the model processing in real-time traffic in a wireless mesh network can be maintained using these techniques.

The resulting feature map, denoted as f_{conv1}, comprises thirty-two filters. This f_{conv1} feature map is then used as the new input shape, specifically (f_{conv1}, 32), for the initial max-pooling layer. The max-pooling layer performs downsampling using a kernel =2 and stride = 2, resulting in 32 feature maps f_{max1} of size. This calculation is carried out as follows:

$$f_{max1} = {}^{\left(f_{conv1} - K_{size} + 2 * padding\right)}\!/_{stride} + b_0 \tag{5}$$

The second convolution layer applies convolution to the input feature map f_{max1}, using sixty-four filters of kernel size 5. The resulting output is a feature map of size f_{conv2}, consisting of sixty-four features. The subsequent max-pooling layer iteratively applies the method with a tweaked input shape (f_{conv2}, 64). This results in producing sixty-four feature maps, each with a size of f_{max2}. The input, with a shape of (f_{max2}, 64), is subjected to two convolutional layers, followed by two max-pooling layers. The resultant output is further compressed and used as the input for the ultimate layer. The ultimate layer in the network architecture is a densely linked layer that efficiently categorizes various forms of DDoS assaults using the SoftMax process. The SoftMax process is expressed mathematically as follows:

$$\delta(x) = e^{z_i} / \sum_{n=1}^{K} e^{z_i} \tag{6}$$

A sigmoid activation function, defined as in [32], is applied to the output of each convolution layer as follows

$$\delta(x) = 1 / 1 + e^{-x} \tag{7}$$

where x is the output that is generated as a consequence of each layer of convolution, i.e.

$$x = \sum T_w * W_k + b_k \tag{8}$$

where T_w is defined as the input weights, b_k is the bias value of kernel k, and W_k is the weight of the kernel size k.

Finally, the Adam optimizer is employed for adaptive moment estimation. Because of its improved performance and widespread use, we included the Adam optimizer [33-34] in the model to optimize. The Adam optimizer maintains a weighted average of past gradients, gd_t [35], whose value decays exponentially over time.

$$y_t = \alpha_1 y_{t-1} + (1 - \alpha_1) * gd_t \tag{9}$$

$$m_t = \alpha_2 y_{t-1} + (1 - \alpha_2) * gd_t^2 \tag{10}$$

The initial instant of the gradient is denoted by y_t and the second instant by m_t, and the decay rates are denoted by α_1 and α_2.

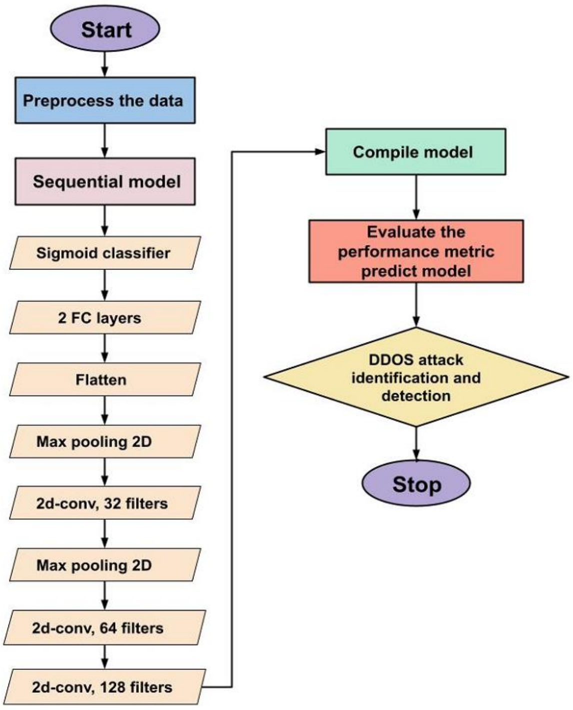

Fig. 2. Proposed method architecture of DCNN-DDoS

After each convolution layer, the inputs to the next layer are weighted using a non-linear activation function called the sigmoid. To further categorize the different DDoS assaults, SoftMax is employed as an alternative activation function. Figure 2 depicts the suggested architecture for the DCNN-DDoS.

The proposed model is a DCNN that utilizes the sigmoid activation function for each unit. The section additionally presented the pseudocode employed for the suggested DCNN-DDoS method. DDoS assaults are detected and classified using Algorithm 1.

Algorithm - 1. DCNN-DDoS Algorithm.

```
Input: Encoded data $T = \{T_1, T_2, \ldots T_b, \ldots, T_k\}$
Output: DDoS detection and classification
Begin
  While round <= Imax do
  Creation of trained data set.
   A deep convolutional NN is created.
   Weight initialization Wi.
    For i belongs to m_i do
       Calculate the $H_n^j$ by equ. (4)
     Update the hidden layer values.
     DoS attack is identified.
  End While
  Initialize sequential model
  For i belong to $I_{max}$ do
       Calculate convolution with filters f and kernel size k using equation 2.
       Maximal pooling layer activation function extraction using equation 6.
       Update the resultant output
       Categorize DDoS attacks using a fully connected Dense layer and activation = 'SoftMax'
       Create the model with optimizer = 'adam'
       Optimize the model on the training data.
  End for
       Predictive model evaluation and testing
       Classify the output
End
```

By following the DCNN-DDoS method, the objective is to develop a deep CNN model that effectively detects DDoS attacks in wireless mesh networks. The subsequent sections of the research paper will provide details on the experimental setup, results, and analysis, further validating the proposed approach.

3.4 Computational complexity

The computational complexity of the suggested DCNN-DDoS model is $O(n * m * k)$ where n represents network layers, m refers to feature maps size and k indicates filters number which proves that its scalability for big datasets such as CICDDoS2019. In comparison, the older techniques (e.g. SVM-DoS) are affected with a quadratic complexity $O(N^2)$ for larger datasets resulting longer training time. The performances of advanced models e.g., FSO-LSTM and AIDS-HML are similar to DCNN-DDoS, although these models are more complex with a lack of cross-layer feature integration. Thus, the performance drops in mesh networks with time-varying scenarios (in dynamic network environments).

4 Experiment setup and result analysis

The simulations for the proposed work were conducted using MATLAB, and the results were obtained for multiple performance metrics. The simulation analysis incorporates a comprehensive set of parameters, as outlined in Table 4. The simulation we performed had 1000 mesh nodes spread throughout 200 m × 200 m. Table 2 summarizes the consistency of mesh nodes and DCNN-DDoS parameters and offers accurate normative values for the sample size. The efficacy of the DCNN-DDoS technique has been validated through its application in a security model designed for detecting intrusions and providing solutions for secure routing in the context of DoS attacks. The present study examines the outcomes of the suggested approach compared to the pre-existing solutions, considering different quantities of malevolent nodes and mesh clients. The configuration for the velocity of mesh clients falls somewhere in the range of 2 to 5 meters per second. The parameters are as follows: a learning rate of 0.001, a batch size of 64, the activation function, an Adam optimizer and a dropout value of 0.3 to avoid overfitting in the proposed DCNN-DDoS model. The architecture includes a series of convolutional and pooling layers designed to extract features from the network traffic data, with fully connected layers for classification at the top level. It is usual practice to classify mesh routers as gateway devices because of their ability to ease the routing of data. Ten nodes exhibit malicious behavior. The assessment of the proposed scheme is conducted about end-to-end delay, energy consumption, packet delivery ratio, mean packet latency, detection ratio, packet loss rate, malicious nodes, and accuracy.

Table 4. Simulation setting for DCNN-DDoS

Parameters	Values
Area covered	200 m ×200 m
Mesh nodes	1000
Malicious nodes	10
Initial energy E_0	0.5
Essential transceiver energy E_{el}	50 nJ/bit
Threshold-distance d_0	86 m
Packets size	4000 bits
Filter	256 Cov1/2
Kernel Size	3
Learning rate	0.001
Optimizer	Adam
Max pooling	2
Activation function	Sigmoid
Simulation runs	30

The first stage of parameter selection in which is to determine those parameters that are very important by affecting algorithm performance. In DCNN-DDoS model, the parameters like the learning rate, which determines the step is updates weights by measuring loss function and a batch size, the number of samples processed before updating. The number of layers and filters (depth & complexity), type of activation functions in hidden layers and dropout rate are also one other vital parameter. The Adam optimizers optimize the regularization parameter.

It is therefore essential to know that the parameters chosen are optimal. K-fold cross-validation and other similar methods allow evaluating the performance of a model on different partitions of data. Further, the different evaluation metrics of accuracy, precision, recall and F1-score, detection ratio gives a clear view of performance. The regularization terms, in turn equilibrate the complexity of the models with their generalization. Last but not the least, validating the model on a separate test set ensures that selected parameters generalize better than training data. To on the best model, a fine-tuning is performed to reach near-optimal parameters. It consists of reducing the search space for certain parameters and tuning them to improve performance over multiple iterations.

4.1 Performance evaluation

The proposed DCNN-DDoS is validated using industry-standard performance measures. Five criteria are used to determine secure routing: end-to-end delay, energy consumption, packet delivery ratio, mean packet latency, detection ratio, packets lost rate, malicious nodes, and accuracy. To measure the DCNN-DDoS against D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML methods.

4.1.1 Energy consumption

It has been shown that the DCNN-DDoS method expects a decrease in network energy use due to data transmission. As predicted, networks' energy consumption performance improved with the increase of iterations. As shown in Fig. 3, DCNN-DDoS outperforms protocols like D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML because it uses more iterations and improves data transmission. Furthermore, DCNN-DDoS uses less energy each round than competing protocols in dual-hop communication.

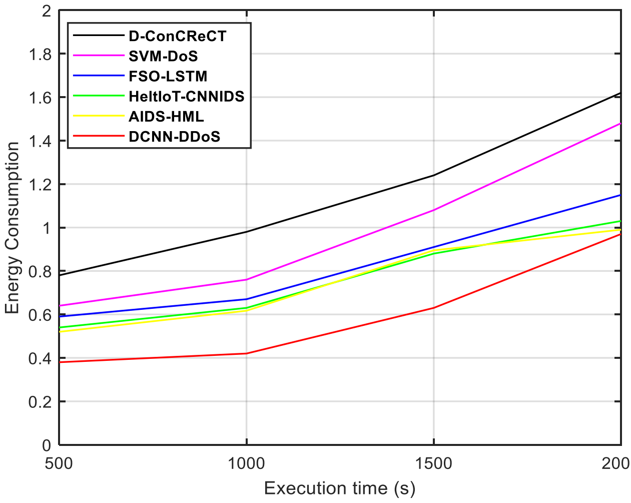

Fig. 3. Energy consumption comparison of DCNN-DDoS with existing methods

4.1.2 Detection ratio

The detection ratio accurately predicts the proportion of DoS attacks detected and assesses the resulting permanent environmental damage. The DCNN-DDoS method relies heavily on utilizing the DoS attack detection ratio to safeguard the environment's security effectively. The evaluation of the Denial of Service (DoS) attack is illustrated in Fig. 4. The comparison demonstrates the DCNN-DDoS method exhibits a higher detection ratio compared to the existing approaches, namely, D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML. The comparison analysis reveals that the Deep Neural Network (DNN) model proposed in this study achieves a detection ratio that is 15.67% higher than that of AIDS-HML, 22.8% higher than that of FSO-LSTM, 38.54% higher than that of SVM-DoS, and 78.12% higher than that of D-ConCReCT.

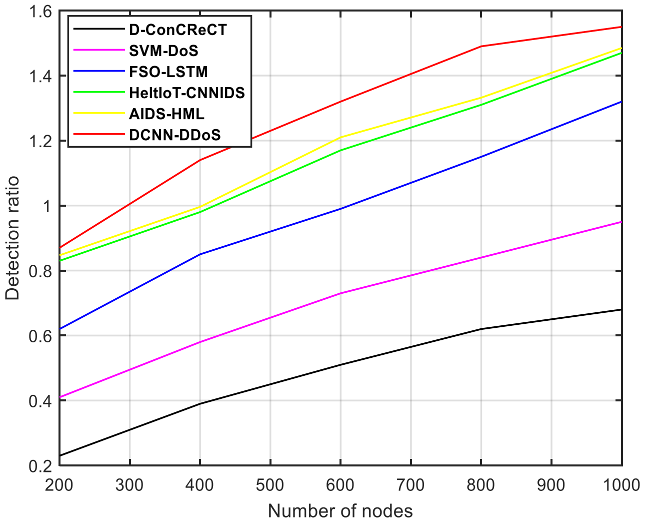

Fig. 4. Detection ratio comparison of DCNN-DDoS with existing method

4.1.3 Mean packet latency

The lower forms of the mean latency packet are used to calculate the average number of hops for the shortest path. The efficacy of the suggested technique is enhanced by incorporating countermeasures designed to mitigate the impact of malicious assaults. Fig. 5 presents a visual depiction of the mean packet delay. The current system uses the D-ConCReCT method rather than the DCNN-DDoS technique, demonstrating a reduced packet delay. The proposed deep convolutional neural network (DCNN) system would use a hop-to-hop methodology for data transmission, with a special focus on identifying high-count distance pathways that are hypothesized to be possible attack routes. The comparison demonstrates that the proposed DCNN-DDoS exhibits a lower mean packet latency value than D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML. Specifically, the DCNN-DDoS reduces 52.72% compared to D-ConCReCT, 42.61% compared to SVM-DoS, and 28.34% compared to FSO-LSTM.

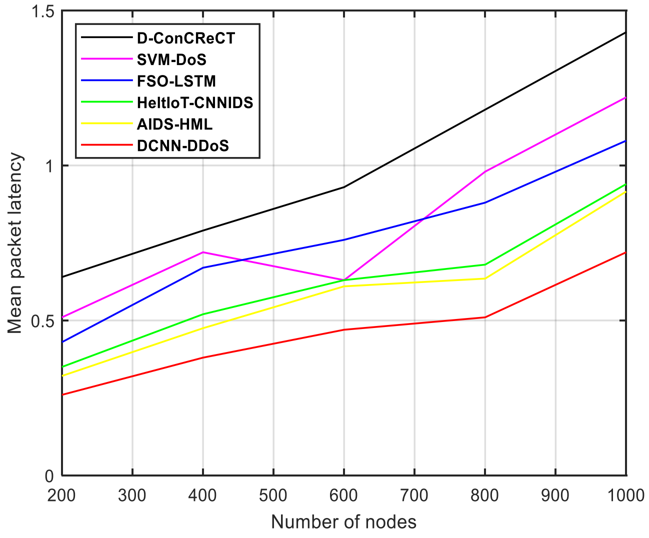

Fig. 5. Mean packet latency of DCNN-DDoS *versus* existing methods

4.1.4 End-to-end delay

A delay in processing, queuing, and propagation delay has been encountered by the data packet transmission time from the sink's source. This delay has been passed on to end-to-end locations. The deployment of mesh nodes in military settings has been broken down via graphical representations of end-to-end latency to provide clarity. There will be a delay towards the end of the route where communication occurs if the distance between hops is significant. If the DCNN-DDoS system source were to be targeted for it, the short end-to-end latency is shown in Fig. 6 by counting the number of hops between each node in the data transmission chain. Long hop-to-hop route distances are disqualified as data transmission in the context of the DCNN-DDoS system.

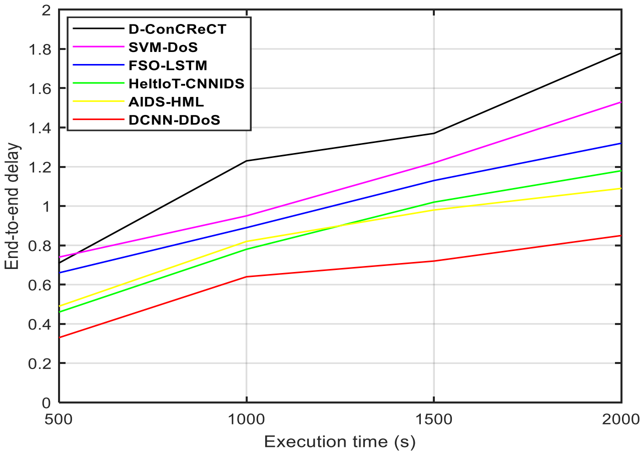

Fig. 6. End-to-end delay of DCNN-DDoS *versus* existing methods

4.1.5 Packet delivery ratio

By comparing the total number of data packets sent with the total number of data packets received, the destination of the data is identified. D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML have all been tested and compared to the proposed DCNN-DDoS regarding round count and Packet Delivery Ratio. This evaluation is illustrated in Fig. 7. The proposed DCNN-DDos has successfully achieved a higher packet delivery ratio (PDR) by efficiently receiving packets at the destination without encountering any failures. After conducting a thorough analysis of both methods, it has been determined that the proposed DCNN-DDoS exhibits a higher delivery packet rate than D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML. Specifically, the DCNN method demonstrates a 32.56% increase in delivery packets compared to ConCReCT, a 19.54% increase compared to SVM-DoS, 11.34% increase compared to FSO-LSTM, and 10.75% increase compared to AIDS-HML.

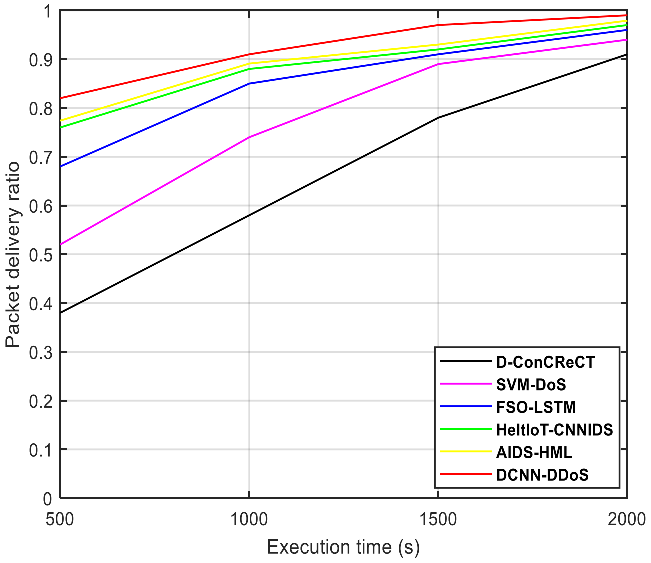

Fig. 7. Packet delivery ratio of DCNN-DDoS *versus* existing methods

4.1.6 Malicious nodes versus packet received

Figure 8 illustrates the received packet in the presence of a variable number of malevolent nodes. The results of the DCNN-DDoS investigation demonstrate that the network throughput performance has been enhanced by 15% in comparison to other established techniques such as D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML. The observed enhancement can be attributed to using an authentic Deep Maxout Network classifier and Deep Auto Encoder. The proposed scheme employs the Deep

Maxout Network classifier and Deep Auto Encoder to attain resilient authentication and safeguard data confidentiality. In this scenario, the possibility of a malevolent node successfully intercepting, discarding, or modifying the information contained within data packets is mitigated. While link estimation between a mesh's clients and its router is critical for assuring the security and dependability of data transmission, a large fraction of the currently available approaches ignore it. Both network throughput and routing efficiency will suffer as a result of this shortcoming.

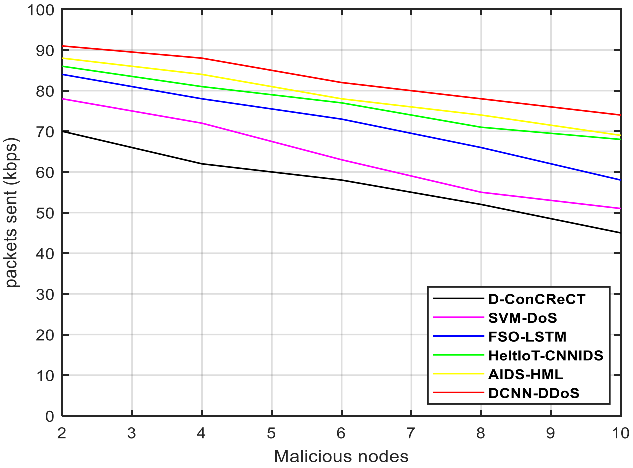

Fig. 8. Comparison of the packet received versus malicious nodes.

4.1.7 Packet loss rate

Figure 9 depicts the comparative evaluation performance of the DCNN-DDoS and existing methods such as D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML concerning varying numbers of nodes. The experiment's findings indicate that the suggested approach has reduced the rate of lost packets by an average of 40% when compared to alternative solutions. Selecting trustworthy and secure mesh clients for data routing improves packet delivery ratios. In contrast, unreliable and insecure data networks may lower delivery ratios in other systems. Due to the optimization model, the suggested technique reduces congestion and distributes mesh client load better in situations with bigger nodes. The suggested technique determines the lost packet rate factor to assess source-destination connectivity. This method improves message delivery without control messages to rebuild routes. Deep CNNs secure network data in the proposed approach. This approach may detect abnormal authentication activity even with malicious nodes. A hostile activity detection system reduces packet loss and improves network performance.

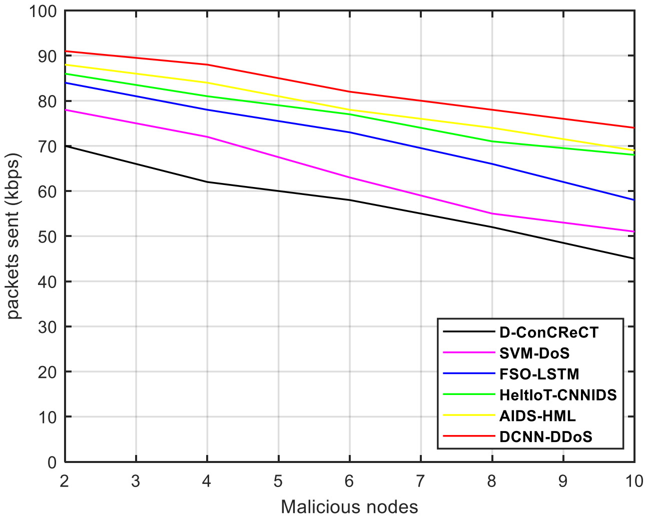

Fig. 9. Comparison of the packet received *versus* malicious nodes

We analyse the DCNN-DDoS model using evaluation metrics such as F1-score, recall, accuracy, and precision. According to the results of the experiments, a two-level binary classification system is adequate for classifying intrusions. Once this is done, the training process is ended and the model is tested against the experimental dataset using specified metrics for classification.

True Positives (T_{Pos}): The number of DDoS assaults that have been correctly detected. **True Negative (T_{Neg}):** this statistic pertains to quantifying precisely diagnosed negative DDoS attacks that are displayed as DDoS attacks.

False Positives (F_{Pos}): the term "false positive DDoS attack" refers to the number of negative DDoS attacks mistakenly classified as positive. **False Negative (F_{Neg})** refers to positive DDoS attacks that are incorrectly classified as negative DDoS attacks.

Precision (P): Precision measures classifier accuracy. The accuracy rate is the number of positive findings divided by the total of positive and negative results.

$$Precision\ (P) = \frac{T_{Pos}}{T_{Pos} + F_{Pos}} \tag{11}$$

Recall (R): DDoS attack classifier comprehensiveness is assessed using the recall measure. The recall calculation involves dividing true positives by the sum of true positives and false negatives.

$$Recall\ (R) = \frac{T_{Pos}}{T_{Pos} + F_{Neg}} \tag{12}$$

F1-Score: To find statistical significance, the F1-score takes a harmonic mean of recall and accuracy. Accuracy and memory are averaged out to get the F1 score. The range of the variable is 0-1.

$$F1 - Score = 2 * \left(\frac{P \times R}{P + R}\right) \tag{13}$$

Accuracy: Using the difference between the actual labels and the projected DDoS assaults is the accuracy measure for assessment. The score may go as high as 1 and as low as 0.

$$Accuracy = \frac{T_{Pos} + F_{Neg}}{T_{Pos} + F_{Pos} + T_{Neg} + F_{Neg}} \tag{14}$$

Table 5 shows that the proposed model outperformed other models in detecting DDoS assaults in terms of accuracy, recall, f1-score, and precision compared to D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML.

Table 5. DDoS attacks detection accuracy of proposed method and existing methods

Techniques	Precision	Recall	F1-Score	Accuracy
ConCReCT	0.9432	0.9615	0.9522	92.71%
SVM-DoS	0.9721	0.9789	0.9754	96.11%
FSO-LSTM	0.9795	0.9813	0.9803	96.93%
HeltIoT-CNNIDS	0.9811	0.9875	0.9842	97.2%
AIDS-HML	0.9823	0.9879	0.9856	97.34%
DCNN-DDoS	0.9881	0.9914	0.9892	98.98%

5 Conclusion

This study proposed a deep Convolutional Neural Network (CNN) model for efficient detection of DDoS attacks in wireless mesh networks. The model leveraged the spatial correlation of network traffic data to learn discriminative features and accurately identify DDoS attacks in real-time. The DCNN-DDoS optimization method is utilized to train the deep model. The developed approach exhibited superior performance compared to several pre-existing methods. The proposed scheme involves the utilization of mesh nodes to enhance network coverage. As a result, the proposed DCNN-DDoS model achieves enhanced throughput. Upon assessing the security parameters, the optimal pathways are selected based on their ability to yield superior energy efficiency, extended network longevity, reduced packet loss, extended throughput, and reduced DoS attacks. The assessment criteria utilized to analyse the system's effectiveness encompass the energy consumption, end-to-end delay, packet delivery ratio, mean packet latency, detection ratio, packets lost rate, malicious nodes, and accuracy. According to the simulation results, the DCNN-DDoS method demonstrates a superior detection ratio metric compared to D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML, with improvements of 78.12%, 38.54%, 22.8%, 16.33%, and 15.67%, respectively. It has been determined that the proposed DCNN-DDoS exhibits a higher delivery packet rate than D-ConCReCT, SVM-DoS, FSO-LSTM, HeltIoT-CNNIDS, and AIDS-HML. Specifically, the DCNN method demonstrates a 32.56% increase in delivery packets compared to ConCReCT, a 19.54% increase compared to SVM-DoS, an 11.34% increase compared to FSO-LSTM, and a 10.75% increase compared to AIDS-HML. The empirical findings have validated that the DCNN-DDoS method exhibited superior performance compared to existing deep learning models, as evidenced by higher precision, f1-score, and recall levels. Moreover, it achieved the highest accuracy rate of 98.98%. In the future, machine learning-driven optimization methods will be incorporated to imbue the mesh nodes with intelligence while keeping the node level and processing overhead nominal.

Funding

The authors declare that they have no known competing financial interests in this paper.

Competing interests

The authors declare that they have no known competing financial interests.

Availability of data and material

Data used in this work from DDoS Evaluation DatasetCICDDoS2019.https://www.unb.ca/cic/datasets/

References

[1] S. Tanzeel Shah, B. Shams and S. Khan, "A Survey on Secure Routing in Wireless Sensor Networks." *International Journal of Sensors Wireless Communications and Control,* 3, no. 1, pp. 37-44, 2013.

[2] T. Delwar, U. Aras, S. Mukhopadhyay, A. Kumar, U. Kshirsagar, Y. Lee, M. Singh and J. Ryu, "The Intersection of Machine Learning and Wireless Sensor Network Security for Cyber-Attack Detection: A Detailed Analysis." *Sensors,* 24, no. 19, pp. 6377, 2024.

[3] M. Hassan, M. Mohamad and F. Muchtar, "Advanced Intrusion Detection in MANETs: A Survey of Machine Learning and Optimization Techniques for Mitigating Black/Gray Hole Attacks." *IEEE Access*, 2024.

[4] D. Boubiche and A. Bilami, "Cross layer intrusion detection system for wireless sensor network." *International Journal of Network Security & Its Applications,* 4, no. 2, pp. 35, 2012.

[5] F. Catak and A. Mustacoglu, "Distributed denial of service attack detection using autoencoder and deep neural networks." *Journal of Intelligent & Fuzzy Systems,* 37, no. 3, pp. 3969-3979, 2019.

[6] M. Asad, M. Asim, T. Javed, M. Beg, H. Mujtaba and S. Abbas, "Deepdetect: detection of distributed denial of service attacks using deep learning." *The Computer Journal,* 63, no. 7, pp. 983-994, 2020.

[7] B. Sahoo and A. Sabyasachi, "A Metaheuristic Algorithm Based Clustering Protocol for Energy Harvesting in IoT-Enabled WSN." *Wireless Personal Communications,* pp. 1-26, 2024.

[8] A. Bozorgchenani, M. Jahanshahi and D. Tarchi, "Gateway selection and clustering in multi-interface wireless mesh networks considering network reliability and traffic." *Transactions on Emerging Telecommunications Technologies,* 29, no. 3, pp. e3215, 2018.

[9] B. Sahoo, H. Pandey and T. Amgoth, "A genetic algorithm inspired optimized cluster head selection method in wireless sensor networks." *Swarm and Evolutionary Computation,* 75, pp. 101151, 2022.

[10] A. Nanda, P. Nanda, X. He, A. Jamdagni and D. Puthal, "A hybrid encryption technique for Secure-GLOR: The adaptive secure routing protocol for dynamic wireless mesh networks." *Future Generation Computer Systems,* 109, pp. 521-530, 2020.

[11] K. Haseeb, A. Almogren, N. Islam, I. Din and Z. Jan, "An energy-efficient and secure routing protocol for intrusion avoidance in IoT-based WSN." *Energies*, 12, no. 21, pp. 4174, 2019.

[12] M. Nasir, S. Khan, M. Khan and M. Fatima, "Swarm intelligence inspired intrusion detection systems—a systematic literature review." *Computer Networks,* 205, pp. 108708, 2022.

[13] S. Pingale and S. Sutar, "Remora whale optimization-based hybrid deep learning for network intrusion detection using CNN features." *Expert Systems with Applications,* 210, pp. 118476, 2022.

[14] A. Alqahtani, "RETRACTED ARTICLE: FSO-LSTM IDS: hybrid optimized and ensembled deep-learning network-based intrusion detection system for smart networks." *The Journal of Supercomputing* 78, no. 7, pp. 9438-9455, 2022.

[15] E. Bernardino, A. Bernardino, J. Sánchez-Pérez, J. Pulido and M. Rodríguez, "Swarm optimisation algorithms applied to large balanced communication networks." *Journal of network and computer applications,* 36, no. 1, pp. 504-522, 2013.

[16] DDoS Evaluation Dataset CICDDoS2019. https:// www. unb. ca/ cic/ datas ets/ ddos- 2019. html (2019). Accessed 10 Jun 2020.

[17] F. Al-Anzi, "Design and analysis of intrusion detection systems for wireless mesh networks." *Digital Communications and Networks* 8, no. 6, pp. 1068-1076, 2022.

[18] S. Mahadik, P. Pawar and R. Muthalagu, "Efficient intelligent intrusion detection system for heterogeneous internet of things (HetIoT)." *Journal of Network and Systems Management,* 31, no. 1, pp. 2, 2023.

[19] D. Sharma, S. Dhurandher, S. Kumaram, K. Gupta and P. Sharma, "Mitigation of black hole attacks in 6LoWPAN RPL-based Wireless sensor network for cyber physical systems." *Computer Communications,* 189, pp. 182-192, 2022.

[20] B. Bhati and C. Rai, "Analysis of support vector machine-based intrusion detection techniques." *Arabian Journal for Science and Engineering* 45, no. 4, pp. 2371-2383, 2020.

[21] L. Gandhimathi and G. Murugaboopathi, "A novel hybrid intrusion detection using flow-based anomaly detection and cross-layer features in wireless sensor network." *Automatic Control and Computer Sciences,* 54, no. 1, pp. 62-69, 2020.

[22] L. Gandhimathi and G Murugaboopathi, "Cross layer intrusion detection and prevention of multiple attacks in Wireless Sensor Network using Mobile agent." In *2016 International Conference on Information Communication and Embedded Systems (ICICES)*, pp. 1-5. IEEE, 2016.

[23] S. Ramesh, C. Yaashuwanth, K. Prathibanandhi, A. Basha and T. Jayasankar, "An optimized deep neural network-based DoS attack detection in wireless video sensor network." *Journal of Ambient Intelligence and Humanized Computing,* pp. 1-14, 2021.

[24] G. Liu, H. Zhao, F. Fan, G. Liu, Q. Xu and S. Nazir, "An enhanced intrusion detection model based on improved kNN in WSNs." *Sensors* 22, no. 4, pp. 1407, 2022.

[25] V. Borgiani, P. Moratori, J. Kazienko, E. Tubino and S. Quincozes, "Toward a distributed approach for detection and mitigation of denial-of-service attacks within industrial internet of things." *IEEE Internet of Things Journal* 8, no. 6, pp. 4569-4578, 2020.

[26] R. Vijayanand and D. Devaraj, "A novel feature selection method using whale optimization algorithm and genetic operators for intrusion detection system in wireless mesh network." *IEEE Access,* 8, pp. 56847-56854, 2020.

[27] R. Thillaikarasi and S. Bhanu, "Adaptive DSR to mitigate packet dropping attacks in WMNs using cross layer metrics." *Journal of Ambient Intelligence and Humanized Computing,* pp. 1-17, 2021.

[28] M. Assis, L. Carvalho, J. Rodrigues, J. Lloret and M. Proença Jr, "Near real-time security system applied to SDN environments in IoT networks using convolutional neural network." *Computers & Electrical Engineering,* 86, pp. 106738, 2020.

[29] K. Pal and K. Sudeep, "Preprocessing for image classification by convolutional neural networks." In *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, pp. 1778-1781. IEEE, 2016.

[30] K. Ghori, M. Imran, A. Nawaz, R. Abbasi, A. Ullah and L. Szathmary, "Performance analysis of machine learning classifiers for non-technical loss detection." *Journal of Ambient Intelligence and Humanized Computing,* pp. 1-16, 2023.

[31] C. Pontes, M. Souza, J. Gondim, M. Bishop and M. Marotta, "A new method for flow-based network intrusion detection using the inverse Potts model." *IEEE Transactions on Network and Service Management* 18, no. 2. pp. 1125-1136, 2021.

[32] M. Wani, F. Bhat, S. Afzal, A. Khan, M. Wani, F. Bhat, S. Afzal and A. Khan, "Training supervised deep learning networks." *Advances in Deep Learning,* pp. 31-52, 2020.

[33] S. Indolia, A. Goswami, S. Mishra and P. Asopa, "Conceptual understanding of convolutional neural network-a deep learning approach." *Procedia computer science,* 132, pp. 679-688, 2018.

[34] D. Kingma, "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980,* 2014.

[35] A. Taqi, A. Awad, F. Al-Azzo and M. Milanova, "The impact of multi-optimizers and data augmentation on TensorFlow convolutional neural network performance." In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 140-145. IEEE, 2018.

[36] V. Gowdhaman and R. Dhanapal, "An intrusion detection system for wireless sensor networks using deep neural network." *Soft Computing*, 26, no. 23, pp. 13059-13067, 2022.

[37] I. Almomani, B. Al-Kasasbeh and M. Al-Akhras, "WSN-DS: a dataset for intrusion detection systems in wireless sensor networks." *Journal of Sensors*, 2016, no. 1, pp. 4731953, 2016.

[38] S. Naser, Y. Ali and D. OBE, "Deep learning model for cyber-attacks detection method in wireless sensor networks." *Periodicals of Engineering and Natural Sciences (PEN),* 10, no. 2, pp. 251-259, 2022.

[39] M. Maheswari and R. Karthika, "A Novel hybrid deep learning framework for intrusion detection systems in WSN-IoT networks*." Intelligent Automation & Soft Computing*, 33, no. 1, pp. 365-382, 2022.

[40] A. Gankotiya, V. Kumar and K. Vaisla, "Building IPv6 addressing scheme using Hybrid Duplicate Address Detection to prevent Denial of Service Attack." *Computers and Electrical Engineering*, 117, pp. 109229, 2024.

[41] M. Premkumar and T. Sundararajan, "Defense countermeasures for DoS attacks in WSNs using deep radial basis networks." *Wireless Personal Communications,* 120, no. 4, pp. 2545-2560, 2021.

Anil Kumar Gankotiya completed his B.E. in Computer Science and Engineering from KEC Dwarahat, Almora, Uttarakhand. He obtained his M.E. degree in Computer Science and Engineering from PECUniversity of Technology, Chandigarh. Currently, he is pursuing his doctoral studies in Computer Science and Engineering Department from Veer Madho Singh Bhandari Uttarakhand Technical University, Dehradun, Uttarakhand, India in the area of network security.

Vishal Kumar works as an Assistant Professor in the Department of Computer Science & Engineering at Bipin Tripathi Kumaon Institute of Technology, Dwarahat, Almora, Uttarakhand, INDIA. He did his Bachelors (2005), M.Tech (2010), & PhD (2019) in Computer Science and Engineering respectively. He has edited 5 books and published 30 research papers in IEEE/Springer Conferences/ SCI/ Scopus indexed journals of repute. He is also a member of 5G Working Group, Telecommunication Engineering Center, Govt. of India & Member of Selection Committee, C-DAC Bangalore.

Kunwar Singh Vaisla works as a Professor in the Department of Computer Science & Engineering at Bipin Tripathi Kumaon Institute of Technology, Dwarahat, Almora, Uttarakhand, INDIA. He did his B.Sc (1994), M.C.A (1998), & Ph.D (2011) in Computer Science and Engineering respectively. He has edited 02 Books, authored 09 Books, 07 Book Chapter in Edited Books, 03 patents, and published 88 research papers in IEEE/Springer Conferences/ SCI/ Scopus indexed journals of repute. He holds the position of Head, of the Department of CSE.

Received 29 October 2024