Babylonian Journal of Internet of Things
Vol.2025, **pp**. 185–195
DOI: https://doi.org/10.58496/BJIoT/2025/012; ISSN: 3006-1083
https://mesopotamian.press/journals/index.php/BJIoT

Research Article

Smart IoT Attack Detection Through AI-Optimized Routing Methods

Nabaa Ahmed Noori [1,*], Atheel Sabih Shaker [2], Rana Abdulrahman Lateef [3], Hiba Alzubaidi [4]

[1] *Dept. of Communications Engineering, Al-Mansour university college, Baghdad, Iraq.*

[2] *Dept. of Computer Engineering Techniques, Al-Iraqia Science University, Baghdad, Iraq.*

[3] *Dept of Cybersecurity Science, Al-Iraqia Science University, Baghdad, Iraq.*

[4] *Technical College of Management/Baghdad, Middle Technical University, Baghdad, Iraq.*

ARTICLE INFO

Article History

Received 15 Sep. 2025
Revised 20 Oct. 2025
Accepted 11 Nov. 2025
Published 7 Dec. 2025

Keywords
IoT,
DDoS attacks,
AI,
E2E,
ARP,
IDS systems,
FFNN,
CNN,
Network security.

ABSTRACT

The Internet of Things (IoT) model possesses much flexibility and mobility, hence the IoT system becomes more prone to security threats such as Distributed Denial-of-Service (DDoS) attacks due to its decentralized control, dynamic node moving, energy scarces and bandwidth limited. This development has demonstrated the capabilities of AI in providing improvements to performance of IoT network in terms of achieving faster rate, provide more throughput and achieve higher packet delivery ratio. Utilization of AI-based analytics technology with the use of adaptive methods can improve metrics such as End to End delay (E2E) and Average Received Packets (ARP), by improving response time and increasing intelligence in Intrusion Detection Systems in IoT based setups. In this paper, we use Feedforward Neural Networks (FFNN) and Convolutional Neural Networks (CNN) to detect malicious activities which can strengthen the capabilities of IDS in IoT routing. The suggested AI-optimized routing model outperforms existing models in detection accuracies, which are improved up to 82% and 85% with respective processing time of 18s and 17s. These findings demonstrate the potential to significantly improve security of IoT systems along with increasing overall network robustness and efficiency using this framework.

1. INTRODUCTION

Decentralized networks, particularly MANETs, are growing in the Internet of Things (IoT). These networks enable the free movement of hosts with no wired connectivity, due to the absence of centralised hardware devices such as routers, gateways This flexibility is crucial for flexible and diverse device interactions in the Internet of Things [1]. Similar to the networks, MANETs leverage P2P routing over numerous intermediaries nodes present in IoT. This allows them to bypass the centralised nodes of traditional networks. Multi-hop communication is required when, due to the use of an RF link, nodes are not able to communicate directly. All the IoTs in this design are routing nodes that update network topology when a node enters or leaves [2]. The decentralization of the Internet of Things exposes them to security attacks. This is particularly so in the face of mobility-induced dynamic route creation and fragile network. Denial-of-service (DoS) attacks are dangerous in this dynamic scenario, because of the route discovery complexity [1,3]. It is essential in rapid military or certain other emergencies to have assured communication lines. Given the vulnerability, a robust security enforcement is required to prevent unwanted activities on the network. As IoT nodes are on the move, indeed it is quite difficult to ensure security. It is to be agile and efficient [4]. The intercommunication between Internet of Things (IoT) nodes is conducted through a wireless channel, thus leading to high vulnerability and security risks. These are further compounded by multi-hop connectivity and data packet transfer between the Internet of Things nodes. It can be observed from Fig 1, that in IoT virtually every node is interconnected and these devices are portable, which highlights the dynamic and complex nature of IoT.

*Corresponding author. Email: Nabaa.ahmed@muc.edu.iq

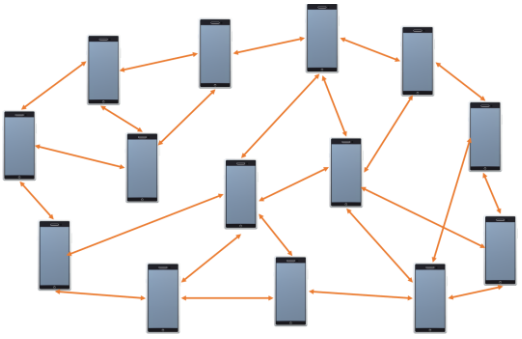

Fig. 1. Building a Sensor Network.

Internet of Things (IoT) contribute to a sophisticated architecture where traffic maintenance for every node is a core problem [2-4], whether functioning autonomously or in a larger IoT network. Figure 2 shows IoT-based Mobile Ad-hoc network traffic routing methods. The adaptability of IoT-based MANETs makes them valuable in military, rescue, and healthcare applications [5].

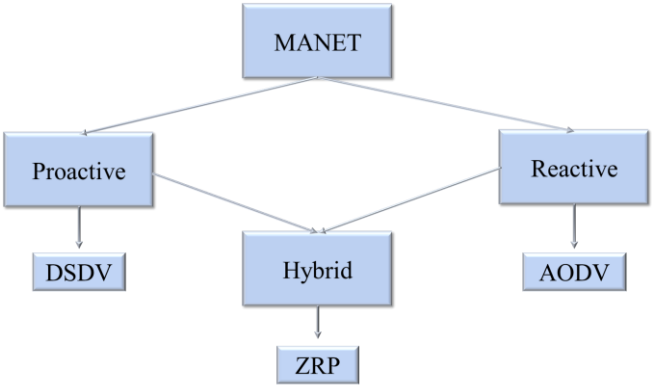

Fig. 2. Different kinds of MANET network traffic.

Significant threats to the IoT networks and attack categories There are a number of that potentially can significantly affect operations [6,7]. Denial of Service (DoS) attacks where attackers send packets to a network in large volume and it uses, bandwidth up to the point that resources are overloaded. Flooding attacks also seek to deplete network resources, and typically use on-demand routing protocols in order to construct a Distributed Denial of Service (DDoS) situation. Routing Table Runoff is another important threat, where routing databases are manipulated by fictitious route request packets to the extent that the router begins to run low on RAM and deletes all its own entries. Impersonating attacks refers to the situation where a node pretends be other nodes axes another in order to send fake routing information to them, accessing the network illegitimately and perhaps also issue wrong commands for certain axis one. Finally, power consumption attacks are represented when malicious nodes produce a great amount of packet traffic causing other nodes to lose energy completely, this is especially harmful in mobile node characterized by limited power supply. On the other hand, IoT networks are subject to a number of attacks such as distributed control, dynamic node operations, wireless channel limitations and physical security problems [8]. There is no central security authority in legacy networks, so it is vital to have strong security. Nodes are dynamic in nature and difficult to isolate potentially compromised devices. The downlink multicasting is also challenged by the limitation of energy due to wireless channel condition which makes it difficult to deploy secure solutions. Physical security is valuable in IoT networks, because adversaries can compromise or tamper with nodes, which would impede defenses against threats from the organization. Hence, IoT networks are easily exposed to a number of security attacks including data breach, denial of service (DoS), and device penetration. Such attacks are difficult to detect by the prevailing security mechanisms in a timely manner because of their high diversity and disparateness. To tackle this, AI models can be embedded in anomaly detection mechanisms of the IoE networks. These models are then trained on data collected from actual IoT networks, namely by means of supervised learning approaches for detecting familiar patterns and machine-learning methods to detect anomalies. This may provide detection and trigger services.

2. RELATED WORKS

Growth of the Internet of things (IoT) has surged across various sectors due to advancements such as data analytics, communication technology, embedded devices and internet protocols [9]. The intersecting of technologies is giving rise

to many applications in areas such as industry, logistics, healthcare and intelligent environments. By taking advantage of IoT devices for data gathering, assessment and communication, these applications improve productivity, efficiency and decision quality. Estimations predict by 2025 the economic impact of IoT to be huge, ranging from $2.7T to $6.2T [10]. Much of this market will be claimed by the healthcare industry, which is a great example of how IoT-based services can be disruptive in terms of potential to improve patient care, remote monitoring and healthcare management. AI is expected to result in massive transformation in a variety of industries by automating the tedious process of analysis, decision making and problem solving as well as knowledge intensive tasks [11]. AI is set to contribute between $5.2 and $6.7 trillion to the global economy by 2025, which if anything underestimates their pervasiveness across so many industries. Deep Learning (DL) has recently emerged as a fundamental technology for Internet of Things (IoT) type applications, feature-exceeding traditional machine learning techniques, due to its capability of dealing with continuously evolving analytic requirements [12]. Deep learning techniques, such as neural networks, have demonstrated outstanding performance across a range of data analysis problems e.g. regression, classification, clustering and pattern recognition. Therefore, such methods are good candidates for Internet of Things applications where different types of data analysis are needed. Despite the growing interest and use of DL in the Internet of Things, research on its actual implementation and effects is sparse. Some research has applied machine learning to context-aware computing systems and wireless sensor networks (WSNs) [13-16], but DL algorithms have received little attention. DL approaches have been studied in network traffic control systems [17], but further research is needed to identify how they may be applied to Internet of Things jobs. Table I summarizes major IoT and machine learning studies. Each study is classified by its main algorithm/method, focus, and notable findings. Tsai et al.'s study excluded DL exploration, Perera et al. focused on context-aware computing without investigating DL, Alsheikh et al. focused on traditional ML in wireless sensor networks, Fadlullah et al. focused on DL in network traffic control, and Qiu et al. analyzed various ML techniques, including DL, for processing general big data.

TABLE I: OVERVIEW OF INTERNET OF THINGS AND AI APPLICATIONS.

Study	Technique / Approach	Area of Interest	Key Findings
Tsai et al. [13]	Pattern Mining, Clustering, and Classification	IoT Services and Architecture	Focuses mainly on offline data analysis; deep learning is not addressed.
Perera et al. [14]	Multiple Machine Learning Methods	Context-Aware IoT Computing	Explores contextual intelligence but does not incorporate deep learning for context reasoning.
Alsheikh et al. [15]	Traditional ML Algorithms	Wireless Sensor Networks (WSNs)	Concentrates on classic ML techniques and WSN frameworks, without extending to deep learning.
Fadlullah et al. [16]	Deep Learning Strategies	Network Traffic Management	Investigates deep learning but limits the analysis to traffic control systems and network-level structures.
Qiu et al. [17]	ML and Deep Learning Techniques	Big Data Analytics	Discusses deep learning from a broad big-data perspective rather than IoT-oriented applications.

3. METHODOLOGY

IoT networks need a lot of sensing nodes, and MANNASIM is one of the tools that can make wireless networking applications on a computer. Network monitoring lets you check and modify network protocols as well as operational standards as needed. The AODV routing protocol is used in this study for simulation. The Blackhole attack is one of the biggest threats in IoT environments. In this attack, a bad node lies about route discovery information to make it look like it is the best way to get to the destination. A single node or a group of nodes can carry out these kinds of attacks, and they are often used in research to test routing performance and security. It is even harder to find bad behavior when both the assails and the target are on the move. Figure 3 shows a situation in which an attacker informally targets the relay node after the initial comm. link has been broken.

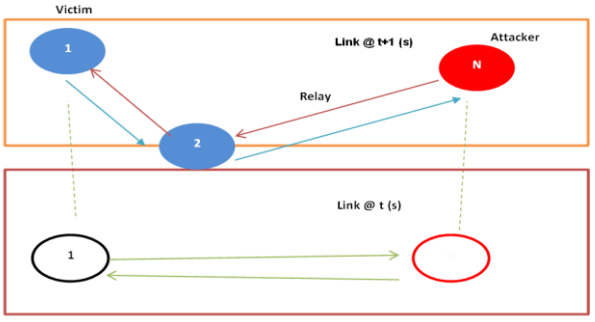

Fig. 3.3. The Victim-Attack Mobility Demonstration at Two Distinct Link Durations.

A simulation environment with a 50-node communication model and an IEEE 802.11-based MANET architecture is used to simulate four different attack scenarios that start with four malicious nodes. The Blackhole attack is used to measure network latency, and the attacker nodes are put in random places on the network. Analyzing trace files can find bad behavior. A node is considered malicious if a user's trace pattern matches a known attack signature. Figure 4 shows the first step in gathering data for both normal and attack situations. This data is used to train and test how well the AI models work.

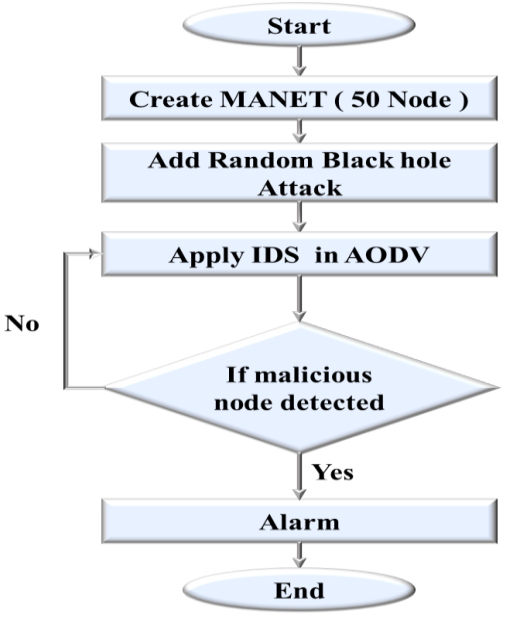

Fig. 4. AODV-based flow diagram for the IDS's routing protocol.

3.1 Data Training Using Artificial Intelligence Algorithms

The proposed framework uses AI to set up a distributed Intrusion Detection System (IDS) in IoT environments. This makes detection more accurate and transmission faster even when the network is limited. Although AI enables efficient data analysis and attack identification, it may occasionally produce false alarms due to misinterpretation of node behavior. To fix this, both Feedforward Neural Networks (FFNN) and Convolutional Neural Networks (CNN) are used to better handle changing network topologies and node mobility, which lowers the number of misclassifications. IoT Network Operation and Data Transmission. Malicious or prying nodes can be configured to inject false data or drop packets, leading to incorrect packet generation within the network.

- Data transmission begins at the source node, which also initiates the route discovery process.
- All nodes forward data packets using the AODV routing protocol operating under IDS supervision.
- The IDS continuously observes node behavior, monitors route formation and replies, and performs physical-layer eavesdropping when required.
- AI-based detection leverages AODV routing data and IDS observations to identify attack patterns during the detection phase.

$$R = net\ (r) \tag{1}$$

$$R = W \times r\ \ + b \tag{2}$$

Equations 1-3 show the model's bias and output vector. Nets can optimize R-T correlation by adjusting W coefficients. Find the minimum Eq (4) learning rate.

$$E\ =\ R - T \tag{3}$$

$$MSE = \frac{\sum_{n=1}^{i} e(n)^2}{i} \tag{4}$$

MSE, a training/learning performance parameter, estimates the velocity that best detects attacker node activity to assess network attack recognition (Eq..5).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + PN} \times 100\% \tag{5}$$

The transmission ratios of control and data messages, the dependability of original and fake packets, and the anticipated amounts of the E2E, ARP, RMSE rate, MAE rate, and MSE rate can distinguish traffic congestion from malicious activity. In order to assess and forecast nodes based on trust criteria and other inputs, the suggested model employs a flow diagram. Altering the connection and discarding assaulted nodes follow its decision to deliver a packet or partition in order to avoid the connection. A flow diagram illustrating the proposed concept is displayed in Figure 5.

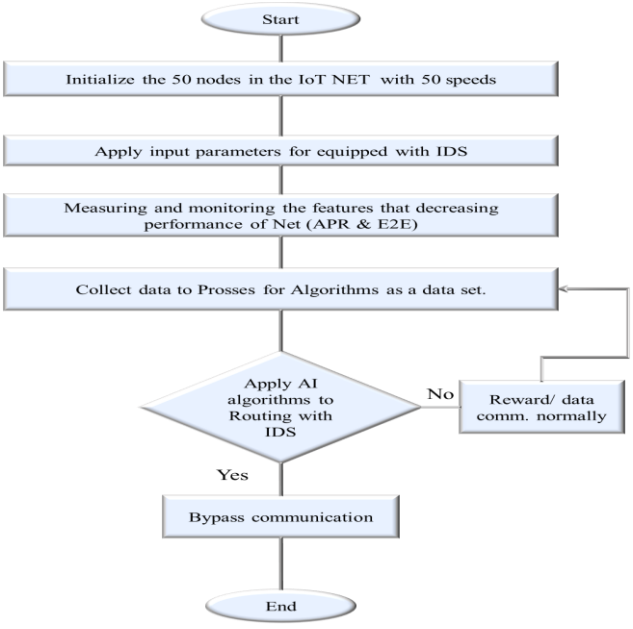

Fig. 5. Illustration of the Suggested Approach.

3.2 First Algorithm: Feedforward Neural Network (FFNN)

The Feedforward Neural Network (FFNN) represents one of the simplest forms of artificial neural networks, characterized by the absence of any cyclical or recursive connections. Data flows strictly in a single forward direction—from the input layer, through one or more hidden layers, and finally to the output layer—without looping back at any point. Regardless of how many hidden nodes or layers the signals traverse, the information never returns to previous layers. In FFNN, input values are multiplied by their corresponding weights as they propagate through the network layers. As illustrated in Figure 6, the FFNN algorithm computes outputs by applying weighted summations across all inputs. The final aggregated value is compared against predefined thresholds; outputs less than one typically occur when the weighted sum falls between the lower and upper activation thresholds. The learning process in FFNN relies on gradient descent to adjust the model parameters. While weight updates in multilayer perceptrons follow a similar concept, the backpropagation (BP) algorithm provides a more precise and efficient method of updating weights. In BP, the error computed at the output layer is propagated backward through the network, allowing each preceding layer to adjust its weights based on the error contribution from the layers ahead.

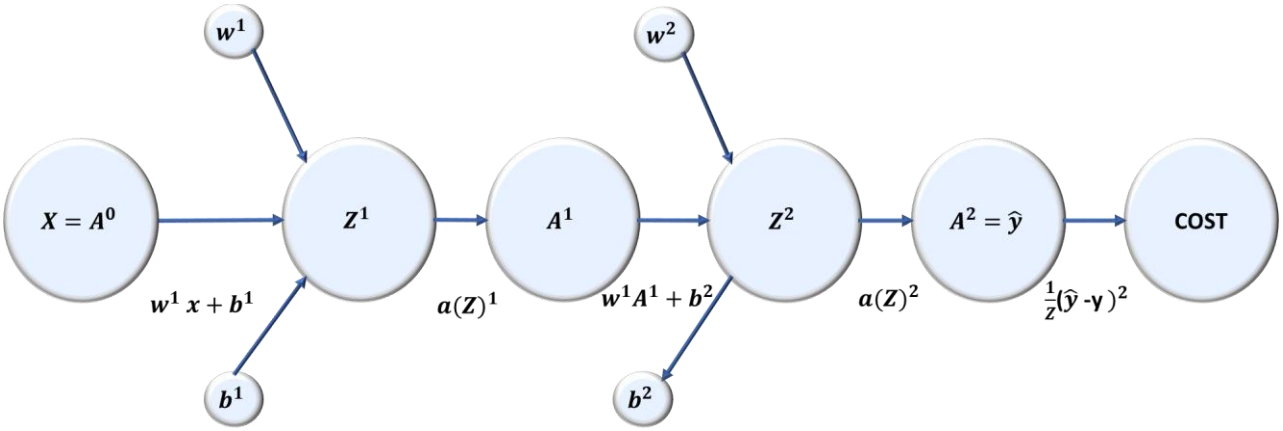

Fig. 6. Computation of weight for Feedforward Neural Network Algorithm.

3.3 Second Algorithm: Convolutional Neural Network (CNN)

Convolutional neural network (CNN) or ConvNet, as one of the most effective and widely applicable deep learning architectures have been employed in IOT-based analysis even in the current chapter. Differing from the densely connected neural network, CNNs are based on convolution operation whose idea is that two functions can be combined to form a third function relation between them. This allows the model to effectively capture complicated patterns and fold hierarchies from input data such as network packets. CNNs operate by learning features at different levels of abstraction. Shallow layers are in charge of high-level or coarse features, while deep layers gradually capture minor and finer details with more complexity. The architecture is generally composed of one or more convolutional layers and fully connected layers, plus optional subsampling or pooling layers that are added in between to decrease dimensionality and increase generality. CNNs are especially well-suited to IoT systems that need a good balance between fast training and adaptive detection, and is capable of supporting large-scale, multi-element dataset processing with low computational cost. As shown in Figure 7, each convolutional layer produces activation maps that act as the input of its following convolutional layers and we can continuously update our weights and feature representations everywhere within the network.

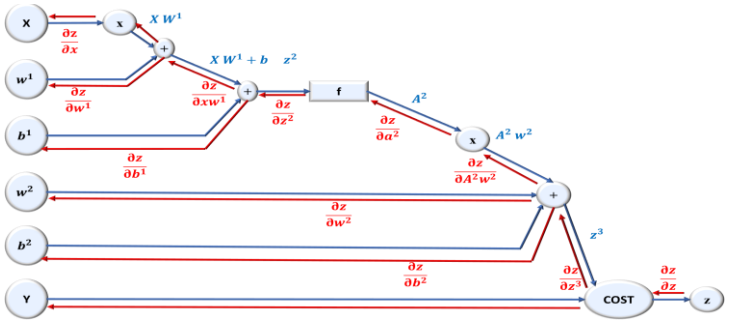

Fig. 7. Wight computation for convolutional neural network algorithms.

4. RESULTS

With the NS2.4 network simulator, the IoT network experiment was developed. It has four malicious nodes, four Mbps bandwidth, a node mobility random rate of sixty seconds, a transmission range of 500 m, AODV routing traffic, and a packet size of 1024 bytes. Twenty, thirty, fifty nodes and 10, twenty, thirty, forty, fifty node speeds were considered, each with its own goals. Table II lists model parameters.

TABLE II. MODEL PARAMETER VALUE.

Parameters	Values
Type of Channel	WSN
Number of nodes	20, 30, 50 nodes
Number of speeds	10 speed
Arena of Workspace	3000m x 3000m
Protocol of Routing	AODV
Range of Transmission	500m
Time of Simulation	300s
Types of Connection	Antenna, omni-directional
Type of Attack	DoS (blackhole)
IDS Type	Cooperative
Size of Packet	1024 bytes
Bandwidth-(BW)	4Mbps
Number of nodes	50 nodes
Number of speeds	10,20,30,40,50 speed
Channel frequency	2.4 GHz
Traffic Type	TCP
Mobility type	Random
Traffic sending rate	32 kbps

The performance of IoT networks is affected by the number and speed of nodes, with increased E2E delay and ARP rates. Despite an intrusion detection system, these networks still face difficulty detecting attacks. Deep learning systems are needed to monitor these networks for unusual behavior and enhance security. The proposed deep learning system can train routing protocols with IDS systems to detect attacks faster and increase protection. DDoS attacks are a vulnerability in IoT networks, leading to less network protection. Anomaly analysis systems can detect previously unknown attacks but sometimes fail to distinguish more recognized safety attacks. The results include data from 50 nodes and different speeds, which were used to train AI algorithms. MATLAB code was used to ARP as well as E2E delay estimations. The dataset was modified to train AI algorithms using nodes and characteristics of IoT, allowing for the evaluation of packet integrity in large networks. This approach could help address the issue of generating random data and improving the performance of IoT networks.

4.1. Training IOT Network Data Using Artificial Intelligence Algorithms

The suggested method uses two Artificial Neural Network (ANN) models, the Feedforward Neural Network (FFNN) and the Convolutional Neural Network (CNN), to look at and sort IoT network behavior. Both models were trained with MATLAB. Table III shows the training configuration parameters, and Figure 8 shows the dataset after it has been preprocessed.

TABLE III. AI MODEL CONFIGURATION PARAMETERS

Unit	Information
Hidden Layers	2 hidden layers
Training Technique	Supervised Learning (SL) and Deep Learning (DL)
Epochs	100
Maximum Gradient	(1 \times 10^{-30})
Mean Square Error (MSE)	(1 \times 10^{-20})
Types of AI Models	FFNN and CNN
Validation Method	K-fold cross-validation
Number of Testing Sets	10

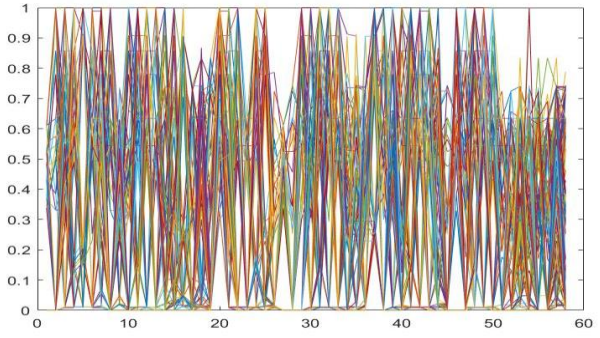

Fig. 8. Processed dataset used in the proposed system.

4.2 FFNN Results: Neural Network-Based Detection

Figure 9 shows how the FFNN model is built. You can use MATLAB's graphical interface to train and sort data using the AI algorithms you choose. In the simulation environment, there are nodes that are spread out randomly. Four of these nodes act as malicious blackhole nodes. One input layer, two hidden layers with ten neurons each, and one output layer make up the FFNN configuration. As well as Table 4 shows the performance metrics, which are accuracy, MSE, MAE, and RMSE. Figure 10 shows the visual output that goes along with it, which shows how well the FFNN model can detect attacks.

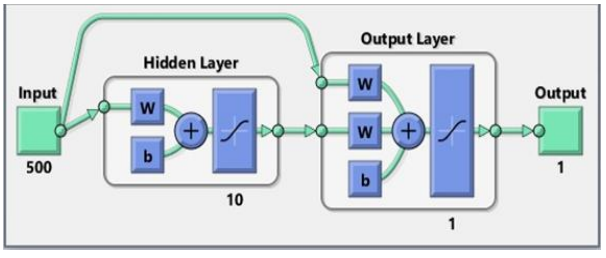

Fig. 9. The FFNN algorithm's design.

TABLE IV. FFNN PERFORMANCE EVALUATION

Metrics	Results
Validation Method	K-fold cross-validation
Number of Observations	50
Number of Test Sets	10
Accuracy	82%
MSE	0.706
MAE	0.327
RMSE	0.84
Runtime	18 seconds (out of 60 seconds)

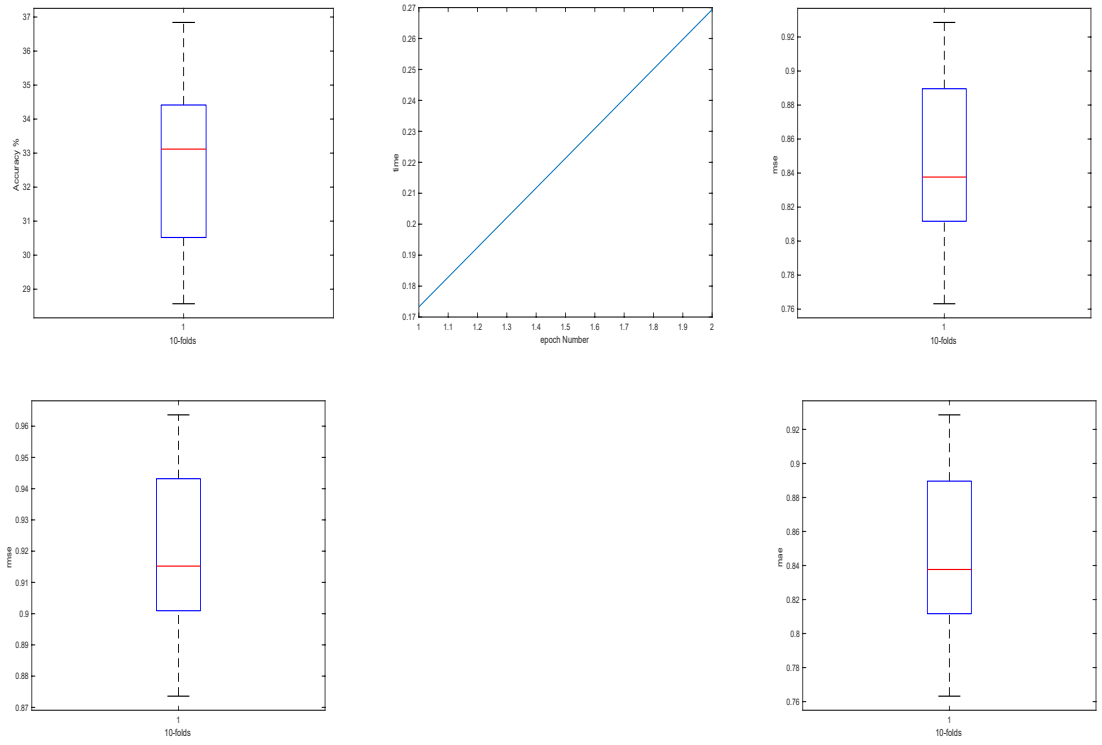

Fig. 10. FFNN attack detection performance metrics.

4.3. CNN Results: Neural Network-Based Detection

Figure 11 shows how the CNN model is built. Again, MATLAB's interface is used to process, train, and test the data. Four nodes act as malicious blackhole attackers, and the nodes are randomly placed, just like in the FFNN experiment. The CNN architecture consists of convolutional layers succeeded by dense layers, enabling the model to derive hierarchical features from the dataset. Table V shows the CNN model's performance results, such as accuracy, MSE, MAE, and RMSE. Figure 12 shows the model's performance metrics, which show how well CNN works at finding attacks.

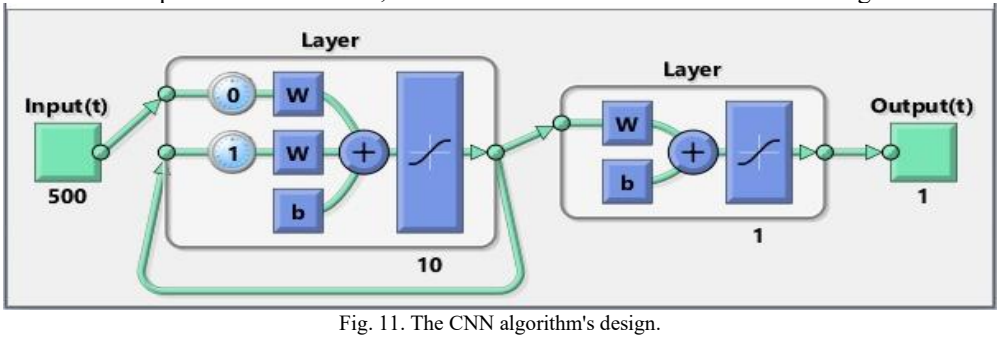

Fig. 11. The CNN algorithm's design.

TABLE V. CNN PERFORMANCE EVALUATION

Metrics	Results
Validation Method	K-fold cross-validation
Number of Observations	50
Number of Test Sets	10
Accuracy	85%
MSE	0.982
MAE	0.327
RMSE	0.991
Runtime	17 seconds (out of 60 seconds)

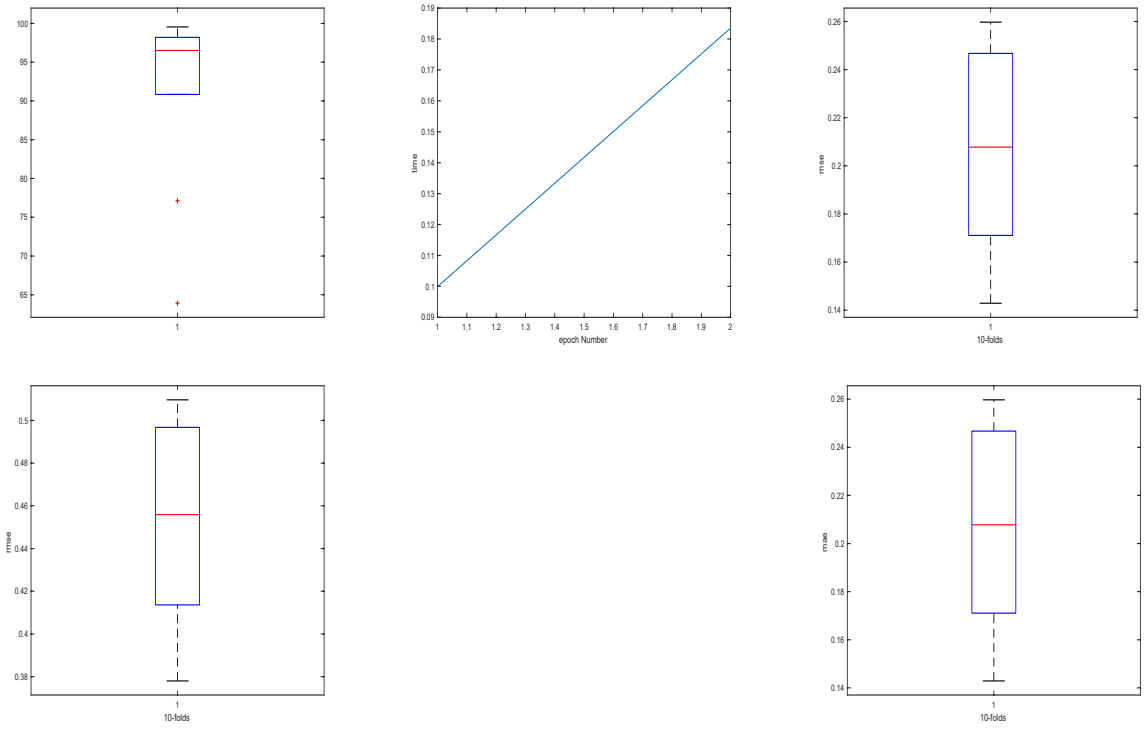

Fig. 12. CNN attack detection performance metrics.

4.4. Comparison of Algorithmic Outcomes

Fig.13 exhibits the performance of proposed FFNN and CNN models since training and testing phases were completed on the pro-posed IoT dataset. A gradient threshold of 1 was applied during the assessment. For the most part, the CNN worked better overall – at least in terms of finding attacks and how long it took to run. The CNN was more accurate than the FFNN, achieving 85% true positive in 17s. The best CNN setup in Figure 14 for example had a gradient of 1.1567, Mu parameter of 0.001, and validated at every 8th check after an epoch was reached (i.e., the validation frequency is x10 epochs) However, the CNN model did poorly for location 0 and 2. The best validation result of the CNN model is shown in Figure 15, in which a minimum Mean Squared Error (MSE) was found to be 2.8118 on epoch 2. This was the quickest verification ever. The desired trajectory, the setpoint and the convergence of the model are also depicted in Figure 16 shows the training visualization for the optimized CNN architecture in MATLAB. The FFNN and CNN models were tested in the same simulation conditions. The FFNN got 82% accuracy after 18 seconds, and the CNN got 85% accuracy after 17 seconds. We used a tenfold cross-validation method, which meant splitting the dataset into ten parts and training on nine of them while testing on one of them. Table VI shows the performance metrics (MSE, MAE, RMSE) for both algorithms. The results show that CNN is more accurate and gives more consistent results than FFNN. Both methods give useful information, but CNN was clearly better at classifying and detecting things.

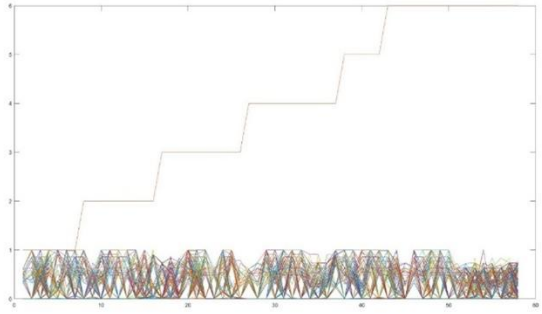

Fig. 13. Results obtained using the MATLAB software for the two algorithms: FFNN & CNN.

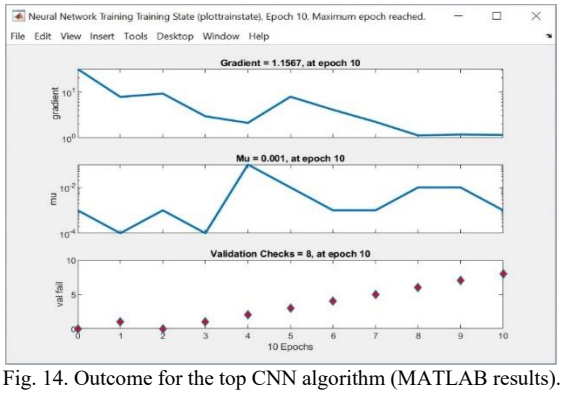

Fig. 14. Outcome for the top CNN algorithm (MATLAB results).

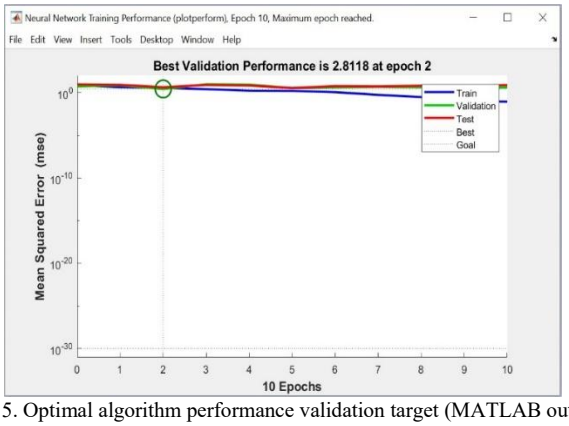

Fig. 15. Optimal algorithm performance validation target (MATLAB output).

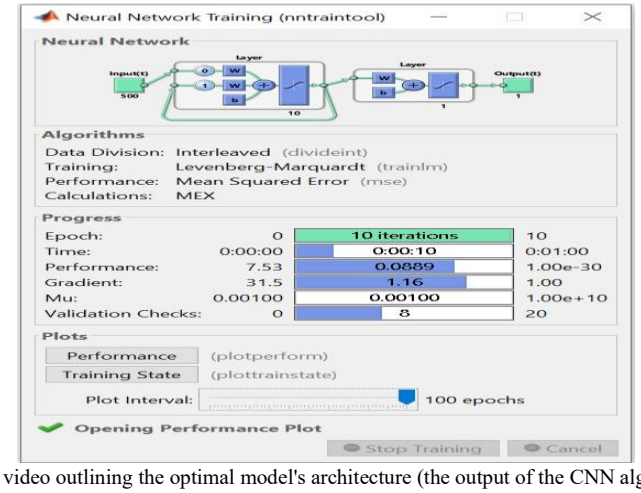

Fig. 16. Instructional video outlining the optimal model's architecture (the output of the CNN algorithm in MATLAB).

TABLE VI. PERFORMANCE COMPARISON OF FFNN AND CNN

Nodes	AI Model	Accuracy (%)	MSE (%)	MAE (%)	RMSE (%)	Runtime
50	FFNN	82.7586	0.7069	0.3276	0.8408	18 s
50	CNN	85.7586	0.9828	0.3276	0.9913	17

4.5. Comparison With Prior Studies

To assess the efficacy of the proposed CNN-based intrusion detection model, its performance was evaluated against prior studies ([18–20]) utilizing RNN, BLSTM, DBM, and CNN algorithms. These works used ten hidden layers, each with 100 nodes, and a training-to-target ratio of 0.8. Table VII shows a comparison of the different models' attack detection accuracy, false positive rate (FPR), and detection rate (DR). After 10 iterations, the proposed CNN–IDS model had an accuracy of 82% and a detection rate of 99%. This was better than the models that had been reported before.

a) The RNN model in [18] achieved 68% accuracy and 83% detection rate.
b) The BLSTM and DBM models in [19] obtained accuracies of 75% and 66%, with detection rates of 67% and 54%, respectively.
c) The CNN model in [20] achieved 77% accuracy and 80% detection rate, which is lower than the performance recorded in this study. These findings highlight significant advancements in IoT intrusion detection through enhanced CNN architectures and AI-driven routing behavior analysis.

TABLE VII. COMPARISON OF PROPOSED METHOD WITH PREVIOUS STUDIES

Ref.	AI Algorithm	Attack Type	Accuracy (%)	FPR (%)	Detection Rate (%)
[18]	RNN	DDoS	68	2.0	83
[19]	BLSTM / DBM	DDoS	75 / 66	19 / 22	67 / 54
[20]	CNN	DDoS	77	16	80
Our Method	**CNN**	**DDoS**	**82**	**2.0**	**99**

5. CONCLUSIONS

The IoT is a flexible, decentralized network that allows devices of varying types to communicate among themselves. But such flexibility also makes it vulnerable to hackers who attack by randomly breaking up the stable data transfer. The study addressed the impact of Blackhole attacks on IoT networks and introduced an AI-based detection model to detect blackholed nodes and maintain reliable packet transmission. Deep learning methods were deployed to identify the best peers for exchanging data, allowing AI- based IDSs to enhance their classification and detection accuracy of malicious activities. We used two models: the Feedforward Neural Network (FFNN) and the Convolutional Neural Network (CNN). The FFNN achieved 82% accuracy in 18 seconds and the CNN reached 85% accuracy in 17 seconds. That is proof that deep learning is viable for IoT security. Future works might include further securing IoT routing, optimizing QoS and memory efficiency, resorting to large data approaches for training sets with higher credibility, improving detection accuracy with less packet loss by 5G integration for larger scale data management independently of the number of targets spread over more intruders to support broader attack vectors. Overall, the proposed AI-enabled approach paves a way for users to design IoT IDS systems that are not only secure but smart as well.

Conflicts of Interest

The authors declare no conflict of interest.

Funding

This research received no external funding.

Acknowledgment

Non.

References

[1] M. Mohammadi, A. Al-Fuqaha, S. Sorour, and M. Guizani, “Deep Learning for IoT Big Data and Streaming Analytics: A Survey,” *IEEE Communications Surveys & Tutorials*, 2018.

[2] *Proceedings of ICETIT 2019*, Springer Science and Business Media LLC, 2020.

[3] *Illumination of Artificial Intelligence in Cybersecurity and Forensics*, Springer Science and Business Media LLC, 2022.

[4] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi, and M. Ghogho, "Deep learning approach for Network Intrusion Detection in Software Defined Networking," in *Proc. 2016 Int. Conf. Wireless Networks and Mobile Communications (WINCOM)*, 2016.

[5] A. AlBusaidi and F. H. Mohideen, "Analysis of Wireless Sensor Network Security Models: A Salient Approach for Deeper Inspection Using Deep Neural Networks," in *Proc. 2023 Int. Conf. Emerging Techniques in Computational Intelligence (ICETCI)*, 2023.

[6] S. Sarkar, G. Choudhary, S. K. Shandilya, A. Hussain, and H. Kim, "Security of zero trust networks in cloud computing: A comparative review," *Sustainability*, vol. 14, no. 18, p. 11213, 2022.

[7] N. Kunhare, R. Tiwari, and J. Dhar, "Particle swarm optimization and feature selection for intrusion detection system," *Sādhanā*, vol. 45, pp. 1–14, 2020.

[8] M. Zaid and P. Agarwal, "Intelligent Intrusion Detection System Optimized using Nature-Inspired Algorithms," in *Proc. 2022 1st Int. Conf. Informatics (ICI)*, pp. 80–85, IEEE, Apr. 2022.

[9] S. G. Zwane, *An Intrusion Detection System for SDN-Based Tactical Networks: A Machine Learning Approach*, Ph.D. dissertation, Univ. of Zululand, 2020.

[10] P. Kumar, G. P. Gupta, and R. Tripathi, "A distributed ensemble design based intrusion detection system using fog computing to protect the internet of things networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 9555–9572, 2021.

[11] P. R. Kannari, N. C. Shariff, and R. L. Biradar, "Network intrusion detection using sparse autoencoder with swish-PReLU activation Model," *Journal of Ambient Intelligence and Humanized Computing*, 2021.

[12] P. Amudha, S. Karthik, and S. Sivakumari, "An Experimental Analysis of Hybrid Classification Approach for Intrusion Detection," *Indian Journal of Science and Technology*, 2016.

[13] S. V. S. V. P. Sanaboina, M. C. Naik, and K. Rajiv, "Examining the impact of Artificial Intelligence methods on Intrusion Detection with the NSL-KDD dataset," in *Proc. 2023 First Int. Conf. Cyber Physical Systems, Power Electronics and Electric Vehicles (ICPEEV)*, 2023.

[14] *Advanced Intelligent Systems for Sustainable Development (AI2SD'2019)*, Springer Science and Business Media LLC, 2020.

[15] B. K. Mohanta, D. Jena, U. Satapathy, and S. Patnaik, "Survey on IoT Security: Challenges and Solution using Machine Learning, Artificial Intelligence and Blockchain Technology," *Internet of Things*, 2020.

[16] A. A. Ojugo and A. O. Eboka, "Mitigating Technical Challenges via Redesigning Campus Network for Greater Efficiency, Scalability and Robustness: A Logical View," *International Journal of Modern Education & Computer Science*, vol. 12, no. 6, 2020.

[17] H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta, and C.-H. Hsu, "Efficient and secure routing protocol based on artificial intelligence algorithms with UAV-assisted for vehicular ad hoc networks in intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4757–4769, 2021.

[18] N. K. Gupta, R. S. Yadav, and R. K. Nagaria, "3D geographical routing protocols in wireless ad hoc and sensor networks: An overview," *Wireless Networks*, vol. 26, pp. 2549–2566, 2020.

[19] M. Elhoseny and K. Shankar, "Reliable data transmission model for mobile ad hoc network using encryption technique," *IEEE Transactions on Reliability*, vol. 69, no. 3, pp. 1077–1086, 2019.

[20] S. Laqtib, K. El Yassini, and M. L. Hasnaoui, "A technical review and comparative analysis of machine learning techniques for intrusion detection systems in MANET," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 3, p. 2701, 2020.