ShieldNetMapper: Internet of Things Powered Predictive Model for Real-Time Network Threat Detection and Response

R. Krishnamoorthy[1(✉)], Kazuaki Tanaka[2], and M. Amina Begum[3]

[1] Centre for Advanced Wireless Integrated Technology, Chennai Institute of Technology, Chennai 600069, Tamilnadu, India
krishnamoorthyr@citchennai.net

[2] Computer Science and System Engineering, Kyushu Institute of Technology, Kitakyushu, Japan

[3] Department of Electronics and Communication Engineering, Sri Venkataswaraa College of Technology, Chennai, Tamilnadu, India

Abstract. Numerous of IoT devices are in fact vulnerable to cyber threats. These vulnerabilities may be exploited over the internet and by remote access by the malefactors. Motivated by this, we suggest a reliable, knowledgeable IoT networks threat detection system. In this research, it aims at evaluating the integration of dual key design concepts in the development of a deep learning based intelligent threat detection system located at the edge of the IoT network. Given these notions, we introduce the ShieldNetMapper (SNM-Rand-N-MMOA) model. Finally, real time IoT traffic data is pre-processed using Spark, Enhanced Random Neural Network (Rand-NN) and Multifaceted Mayfly Optimization Algorithm (MMOA). A deep model is extracted by MMOA from the range of significant weights and bias values for the purpose of reconstructing the optimal network traffic data. On the other hand, we are utilizing Rand-NN for classification as well as to avoid the deep learning model from overfitting. This hereby proposes the model that evaluates IoT real time dataset using the recall rate of 99.78 % and average accuracy is 99.81%. With these impressive results, this is proof that the model can actually separate between different types of IoT traffic and this can help enhance the security and efficiency of operations in the network. Therefore, by integrating MMOA and Rand-NN, not only are data processed in an optimal manner but also the robust classification in dynamic environments is also guaranteed.

Keywords: Detecting IoT Threats · IoT Network Data Traffic · Improved Rand-NN · Multifaceted Mayfly Optimization

K. TanSaka and M. Amina Begum—These authors contributed equally to this work.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2026
O. Gervasi et al. (Eds.): ICCSA 2025 Workshops, LNCS 15890, pp. 205–216, 2026.
https://doi.org/10.1007/978-3-031-97606-3_14

1 Introduction

As the numbers of IoT devices keep increasing, this also resulted in a large increase of the attack surface, making these networks an easy target for sophisticated cyberattacks. The main objective is to make the Internet of Things (IoT) networks and devices unusable by those people whom it is supposed to serve by flooding these networks or devices with too many requests. Basic network resource flooding and other more elaborate types of attacks, exploiting vulnerabilities in certain protocols or services are possible in these attacks [1]. A rather big threat to the IoT networks is represented by botnets networks of devices that were compromised by a single attacker (bot herder). It is done to execute DDoS attack, spread malware, etc. on compromised devices, which are usually small low powered IoT devices lacking adequate security mechanisms [2]. Due to the nature of botnets as being decentralized and spread, they are difficult to locate and deactivate. The use of deep learning models enables botnet traffic patterns to be effectively detected as they feature the specific communication patterns and the main command and control framework of botnets [3]. However, the design and communication method of botnets will continue to improve the problem is that botnet designs and communication methods are always improving so deep learning models need to adapt to new botnet behaviours [4]. Furthermore, there is a need for developing deep learning models to detect botnet for the same reason that most IoT devices have very limited resources, and deep learning models should be light and effective to run on the devices that have limited memory and processing power [5]. Lastly, the immense difficulty remains in detecting zero day exploits, which are not present in training datasets. Advanced Persistent Threats (APTs) are established and long lasting attacks, caused to hide from detection for a very long duration. Many of these assaults have many phases and methods, which makes them hard to perceive with traditional security tools. The deep learning models, and especially those that are able to identify very small changes in a long term behavior pattern can be used to detect APTs. Proficient APT identification requires the capacity to recognize and associate apparently irregular cases over the long haul. Even today, the creation of deep learning models that can detect APTs with high precision in real-time with very low false positives remains to be a problem. When threat intelligence is merged with contextual data, APT detection systems are much better and more accurate. It is the sequential nature of network traffic data that makes RNN's, LSTMs and GRUs perfect at evaluating it [6]. These models can capture time dependencies in the data which is important to detect the threats which develop with time [7]. There are advanced types of RNNs, LSTMs and GRUs, which have solved the problem of vanishing gradients. It enables them to learn long dependencies in data faster than RNNs. To identify slow moving attacks such as APTs, it's even more critical. Botnet assaults and other kinds of intrusions have already been successfully identified using LSTMs [8]. The combination of CNNs with RNNs, LSTMs or GRUs makes threat detection systems perform better in hybrid models. This method improves accuracy and flexibility because it combinations of the finest protocol of both CNNs and RNNs. Extracting spatial patterns are

done with CNNs, and temporal dependencies are modelled by RNNs. However, even though RNN models based on RNNs might come with high computational costs as well, if the data sequences are long, they will not be as effective in IoT settings with limited resources.

2 Related Study

This research [9] examines the difficulties associated with managing and safeguarding the expansive Internet of Things (IoT) network, noted for its substantial scale and data flow. It is shown to provide intrinsic challenges in terms of the reliability and security in this context and innovative approaches that need to be taken to the network infrastructure management. These functions cannot be supported by the present dependence on individual IoT devices. The model's input layer examines characteristics of network traffic, and forwards their results to many hidden layers that can spot complicated patterns in the data. To avoid overfitting, a dropout strategy is used. A softmax activation function is used by them on output layer for classification of network traffic as normal or abnormal and hence as a multi class discrimination. The results indicate that it is important to process data carefully and to run a structured evaluation process that uses several performance criteria. The proposed model is compared with existing anomaly detection algorithms to validate its effectiveness in terms of accuracy and efficiency of handling IoT network data flow. In this study [10], the author identify and improve from Distributed Denial of Service (DDoS) attacks involving botnet drive attacks against Internet of Things (IoT) networks. As IoT devices become bigger and more sophisticated; it creates a growing landscape weak to cyber-attack and data breaches. Directly, research addresses the steady growing threat from botnets within the augmented IoT environment. The research makes use of a multidimensional method that leverages multiple ML models to accomplish its task. More details of the examination involves many techniques such as K-Nearest Neighbor (KNN), KMeans clustering, Fuzzy clustering, and deep learning models like Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and Long Short Term Memory (LSTM) networks. These models are assessed on the Bot-IoT dataset and we show their effectiveness. To aid in algorithm selection, the method employs preprocessing of data as well as a performance guided selection that is aided by accuracy percentages. Additionally, the study makes the machine learning algorithms work better and uses feature selection and the Synthetic Minority Over-sampling Technique (SMOTE). This suggests the suggested architecture is effective in determining botnet related attacks of the IoT network. In [11], the author examine the issue of growing security being faced in the stoopid growing Internet of Things (IoT) ecosystem more specifically, and how botnet assault is a big risk to IoT network operations. New approach, GA HDLAD (Genetic approach with Hybrid Deep Learning based Anomaly Detection) is proposed in the study for increasing security with Anomaly Detection for Botnet activities. In this approach, genetic algorithm is used to select feature for the high dimensionality existed in

the IoT network data. Before effective and precise defects identification is possible, that requires this dimensionality reduction. HDL element is the essence (core) of the GA-HDLAD system. The advantage of multiple strategies is used by this component to achieve the result of effective botnet detection. At the same time, feature extraction techniques (FETs) are used to efficiently and effectively extract relevant features from the spatial data that is available in IoT network traffic. Attention mechanisms are integrated into the model so that the model can pay greater attention to the most significant part of the data and this leads to higher accuracy. Simulated annealing (SA) of the HDL method guarantees the optimal performance of the deep learning model, and hyperparameter optimization is performed for the HDL method with simulated annealing. Empirical assessment of the effectiveness of the proposed GA-HDLAD system based on a standardized benchmark botnet dataset is performed. Results from the experimental study show that the GA-HDLAD algorithm surpasses existing botnet detection techniques and thus has the promise of being a valuable addition to IoT security. In [12], the researcher proposed to advance the field of study by investigating how to utilize AI to address crucial cybersecurity problems, particularly from the point of view of enhancing security in the Internet of Things (IoT) environment. This objective is accomplished by an innovative architecture based on Cognitive Digital Twin Systems (CDTS) introduced by the research. Using AI methods to a high degree of sophistication, CDTS enables the digitalization of actual IoT devices, real time monitoring, real time prediction, real time pre-emptive security. It is a result of integrating cognitive learning ability, anomaly detection and machine learning capabilities to predict and counter security issues. In the adopted methodology, the patterns of security incidents are analyzed to reduce response time to 87.5% based on cognitive computing and predictive security techniques, and on detecting zero day attacks with accuracy rate of 92.4%. As expected, these results strongly indicate that the proposed CDTS architecture can provide a comprehensive solution to the issue of adaptive and intelligent threat control in complex IoT networks.

3 Proposed Methodology

The datasets are referred to as $'ToN_IoT'$since they consist of such data sources drawn from Telemetry datasets of IoT and IIoT sensors. Furthermore, the dataset contains created new Industry 4.0 test-bed network consisting of both IoT and IIoT networks. In order to make easier talking between three levels of IoT, Cloud and the Edge/cloud system, the test-bed was created with bunch of virtual machines and hosts running numerous of Windows platform, Linux and Kali. Different types of threats, like DoS, DDoS, as well as ransomware, were to be used against web apps, Traffic ofIoT networks. The dataset creation was used parallel processing to collect network traffic, Windows audit logs, Linux audit logs and IoT service telemetry data for both normal and threat events. In this proposed method, termed ShieldNetMapper (SNP-Rand-NN-MMOP), should employ Random Neural Networks and Mayfly in conjunction, and it has

many parts. An Artificial Neural Network (ANN) known as the Random Neural Network (Rand-NN) places its neurons in such a way that connection weights are randomly allocated in the middle of its neurons. Unlike traditional neural networks, the epitome of the Rand-NN differs in that its architecture is realized randomly. It is unique in the sense that it makes it easier to experiment with a whole accumulation of network architectures. The Rand-NN consists of an input layer, number of hidden layers, and an output layer. The neurons of the Rand-NN are interconnected in a random manner and thus, allow for efficient data flow through the network. The goal of Rand-NN is to discover the essence of neuronal interconnections and activation functions in a network. Rand-NN has the advantage that is easier to use to make random patterns, study properties of neural networks, etc. because their weights are randomized at the beginning. The particle swarm optimization algorithm enhances the multifaceted Mayfly optimization algorithm (MMOA). This bridges between the advantages of evolutionary as well as swarm intelligence algorithms and produces a high power hybrid algorithmic framework, which is applicable to solve both continuous and discrete problems. This method was derived from mayflies' flying and reproductive behaviours. We applied MMOA to a set of 20 test functions with 20 different test functions including fixed dimensional, unimodal, multimodal and multi object optimization functions for which we compare it to seven of the best performing meta heuristic algorithms. The proposed method synergizes the benefits of swarm intelligence and on one hand evolutionary computation.

Figure 1 shows that there are many phases in Rand-NN-MMOA. Scrambling, initialization, scheming the fitness function, selection of the best location, global best evaluation, and updating are the procedures. To obtain better solutions, the most important step in swarm intelligence processes is the exhibition of potential solutions. This work provides a solution to this problem by encoding the Rand-NN weights and biases as swarm mayflies. The three types of weights that can be modified in a Rand-NN network with respect to input activation state weights are: input weights (In W), neuron weights (NW), and bias. A particle has an initial velocity of $V_{\text{ini}} = 0$. The mayfly needs to be set up with a number of parameters. They consist of the total number of population for the swarm (S_{pop}), the maximum number of iterations or generations (Max_{iter}), the individual coefficient (k_{in}), the social ratio (k_{so}), and the inertia coefficient ($\mho$). The proposed Multifaceted MAO will be differentiated from the traditional MOA by using the inertia component x, the direction that linearly changes from $\mho_{\max}$ to $\mho_{\min}$ over time.

$$\mho = \mho_{\max} - \frac{(\mho'_{\max} - \mho'_{\min})t}{T} \tag{1}$$

The variable t in Eq. (1) represents the current iteration number and the Variable T represents the total number of iteration counts. Calculation of the cost function in the middle of the actual and perceived values is the measure by which any population solution is effective. Position Best of each mayfly is used to find out its fitness, while Global Best value is the optimal location which is occupied by any mayfly in the swarm. The positions and velocities of the mayflies

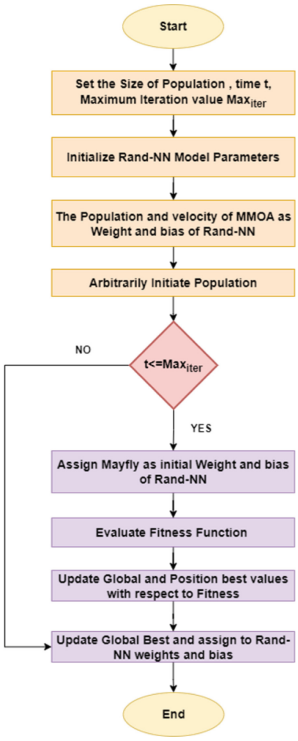

Fig. 1. Proposed Rand-NN-MMOP Flow Diagram.

are modified so that Pos_{best} and Glo_{best} are optimised in each iteration. This is an iterative process which continues till a stop criterion Max_{iter} is met. In the end, it produces the optimum solution that serves as the swarm's collective intelligence in its quest for solve problem in the shortest time possible.

The velocity and position of the ith mayfly at time t are denoted by $(\text{Vel})_t^i$ and $(\text{Pos})_t^i$. Subsequently, the equations cause modifications in the velocity $(\text{Vel})_{t+1}^i$ and the position $(\text{Pos})_{t+1}^i$ of the ith mayfly at the time $(t+1)$. Equations (2) and (3) represent their designated locations.

$$(Vel)_{t+1}^i = \upsilon \times (Vel)_t^i + k_{\text{in}} r_1 \times (\text{Pbest})^i - (\text{Pos})_t^i + k_{so} r_2 \times (\text{Gbest})^i \times (\text{pos})_t^i \quad (2)$$

$$(\text{Pos})_{t+1}^i = (\text{Pos})_t^i + (Vel)_{t+1}^i \quad (3)$$

To continue the exploitative and exploratory characteristics of the multi-faceted mayfly, we use the velocity of the complex mayfly, which is known to be the local and global optimum. The Pbest, Gbest, location, and speed of a mayfly are updated during the subsequent process until the iteration number exceeds the maximum (max_{iter}) or the tolerance reaches the specified threshold. The

Rand-NN, with all weights and biases calibrated and the feedforward layer fully regulated, yields the final Gbest, which is the optimal solution. The weights and biases of the MMOA-Rand-NN are taken as the starting point of the Rand-NN as well as Feed-forward RNN. They are then improved with Adam optimiz-er.

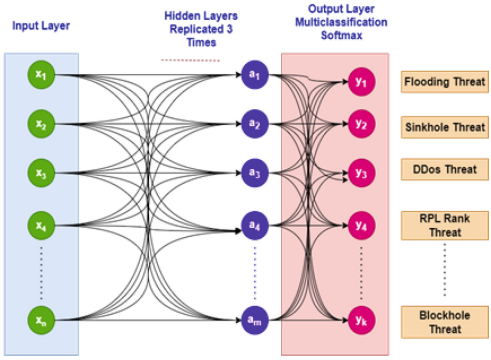

Fig. 2. IoT Network Threat Detection using Proposed SNM: Rand-NN-MMOP Model.

The suggestion is to have the input and output neurons and all of the positive and negative weights relating to each of the neurons. It also has clusters that denote highly interconnected cells. The neurons of each cluster are assembled differ-ently and may be inhibited by external cellular units. The Rand-NN-MMOA architec-ture is depicted in Fig. 2 MMOA decodes how likely a single cell in a cluster is to become active and therefore model suggests the likeliness by exploiting strengths and weaknesses of inputs that stimulate and inhibit with MMOA. With back-propagation over the specified interval, different characteristics and initialization weights were establishedbeginning of input neurons, link matrices, information knowledge rate and total count of epochs, and the algorithms were run. It stores and retrieves weights, calculates firing rates of neurons and computes loss, correctness. The proposed Architecture which has a dual layered framework and three layers of concealing layers considered for binary and multiclass classification. The model employs learning rates of 0.10, and 0.010 over 120 epochs. For binary classification the input X is a dimension of 72 and for multiclass classification 78. It is in the connectivity and weight matrices that the iterative updates of the MMOA optimal value weights of neurons, hence the prediction output, are dictated. A novel approach to simulating network risks is developed by Rand-NN-MMOA through the use of multiple attacks. The proposed architecture is based on a different way of sending data where path where the suspicious attack happened is blocked to flow data and is aware of quality of service (QoS). To protect the network from intrusion, this data is housed in two databases: one for defensive purposes and another for mitigation purposes. It is an active approach, increases the real time responsiveness of network defenses and also allows real time changes based on evolving threat set. The system can

learn from these suggested algorithms, so that it con-tinues to learn how it finds threats and how it responds to it in order to have the strongest defenses against future attacks.

4 Result and Discussions

At the beginning of experiment analysis, the IoT routing network dataset is gener-ated from normal and Threat classes. Table 1 and Fig. 3 of 3D heat map show that there possible more than 6,375 instances, 12 meaningful characteristics, and 48 sub-class characteristics of the final routing dataset.

The three groups of the processed data are the training, validation, and testing, on the feature processing stage. For binary classification data, the data has 64 dimen-sions and for multi class classification data it has 69 dimensions. These sets have the labels within them, which refer to regular and attack threats. This representation is depicted in Fig. 4. Evaluation measures are used from mainstream literature, name-ly accuracy, precision, recall and F1 score in order to evaluate the proposed model.

Accuracy: This term is used to indicate the level of correctness of a model's pre-dictions. In other words, the accuracy rate is calculated as the number of

Table 1. Proposed Simulation Dataset Collection

S. No	Type of Data	No. of Samples
1	Normal Data	6375 (25*25)
2	Threat Data – DDoS	375
3	Threat Data – Flood	165
4	Threat Data – Sinkhole	186
5	Threat Data – RPL Rank	148
6	Threat Data – Blockhole	119

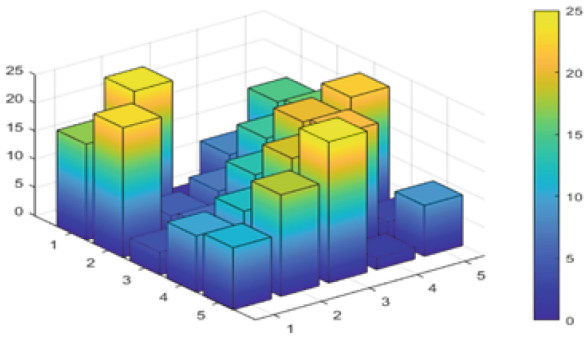

Fig. 3. 3D Heatmap Visualization of Proposed Model"s Input Dataset

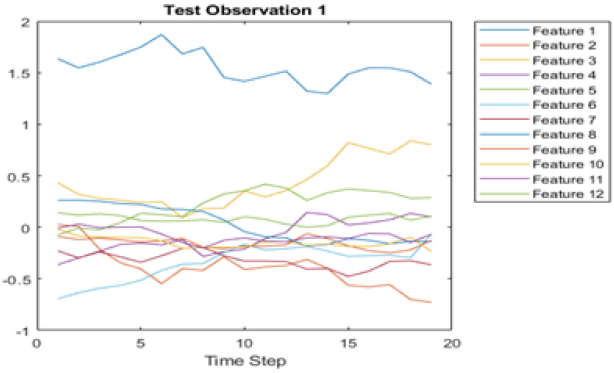

Fig. 4. Proposed Model Test observation Feature counts

correctly classified instances (true positives, true negatives) divided by total instances. We compute it using Eq. (4).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

- **TP** It is a count of *'correct'* identified cases as positive.
- **TN** The number of correctly identified instances as negative.
- **FP** Number of positive incorrectly classified instances.
- **FN** The number of incidents misclassified as negative.

Precision is the number of the extent of accuracy in optimistic forecasts. The pro-portion of expected positive occurrences (both real and false) to actual positive oc-currences is known as the false positive rate. A measure of accuracy, precision may be expressed as the proportion of correct results to the total number of results, includ-ing both true and false positives. The ratio of the number of positively identified events to the total number of positive events (including both true positives and false negatives) is called recall, which is sometimes called sensitivity or the true positive rate. It is also a statistic that can be used to define precision and recall as the same. Hence, there is a need to find a F1 score in the middle of two metrics. Using accuracy and memory average is beneficial in case of imbalanced precision and recall score. As a categorization model, this method is commonly used by machine learning, deep learning, and statistical analysis; to determine an effective categorization model. Our proposed optimization involves the mathematical expression that the neces-sary losses is for to minimize with respect to the iteration and an iteration is a single step in an algorithm assessing the objective function towards the optimal solution step by step. In the Fig. 5, exemplify the objective function which shows that pro-posed M-MOA optimization has such characteristics and the iterations stands for the repeated steps followed to find the best possible value of the objective function.

Setting up experiments on the suggested models and using three benchmark datasets, we perform experiments on both multiclass and binary classification.

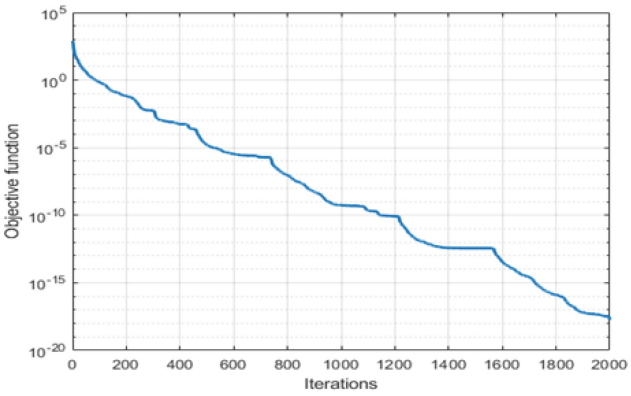

Fig. 5. Proposed Model Objective Function with respect to Iterations

To exhibit efficiency, we used the MATLAB 2022b Tool on a PC with a 2.4 GHz Xeon processor, 16 GB RAM, and NVIDIA Basic GPU for training and deployment of our suggested models. Its evaluation was performed on Windows 11. The stages of an assessment of a neural network include training, validation, and testing. The classification takes preference for the majority class if the dataset is unevenly distributed. In order to fix the problem of imbalanced class distribution in the datasets, we used class weights and AdaSyn for that purpose. The instance quantity of a class has an effect on the class weights, and a class with a small fraction of instance count will be under substantial weight. For the Multifaceted Mayfly Optimizer (MMOA), we trained each Rand-NN classifier for iterations with batch size one.

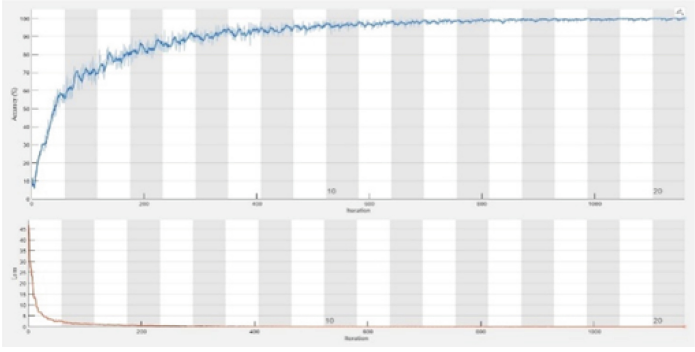

Fig. 6. Proposed Model Training Accuracy and Loss Function of Real Time Dataset

In this work, we employ a dual-type method for training and evaluating our proposed Rand-NN-MMOA model. In this experiment, we allocated 70% of the

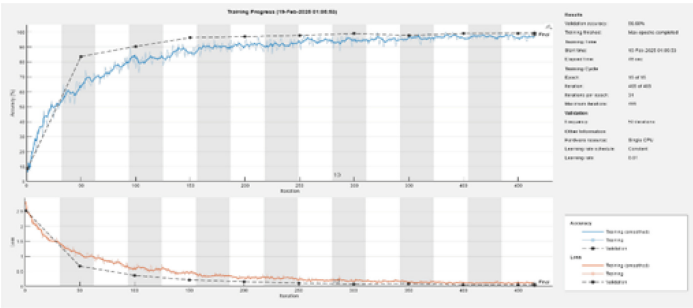

Fig. 7. Proposed Model Testing and Validation Accuracy and Loss Function of Real Time Dataset

data for training and the remaining 30% for testing. In the first case, results are obtained with the best of the standard MOA-based Rand-NN weight and bias assignment. During the evaluation, the Multifaceted Mayfly Technique (MMOA) uses sample points as an activation function to fine-tune hyperparameter values. During the tuning procedure, we also identified the epoch with the ideal learning rate and performance batch size as the best optimizers. Specifically, Figs. 6 and 7 illustrate accuracy and error rate of detection in training and validation as a function of testing data, in which the accuracy and loss components come from a separate set of testing data. Tests indicate that the model performance has considerably improved, especially under complex data distribution conditions. The comparative analysis also shows the robustness of the proposed method in improving the overall predictive capacity of the model on most of the datasets.

5 Conclusion

This is especially true when evaluating the effectiveness of the proposed SNM-Rand-NN-MMOA strategies, as shown by the fact that they can achieve near-optimal results under various data distribution complexities. A comparison investigation showing the efficacy of several strategies usually attaining performance metrics between 0.97 and 0.98 is performed. Notably, we also assessed the IoTReal time dataset for which the recall rate obtained is 99.78% and the average accuracy is 99.81%. Therefore, these results give confirmation that these methods are also practical in real-life applications and can be further improved in data analysis and machine learning to obtain more advanced decision-making processes in threat detection systems. These findings indicate that operational efficiency would be greatly increased by the use of modern data processing methods. With the continued development of these data-driven strategies becoming more and more necessary as the cycle of challenges and opportunities continues to rise in these firms, they will be essential in the ongoing enhancement.

Conflicts of Interest. The authors declare that they have no conflict of interest.

Ethics Approval. Not applicable.

References

1. Nuhu, A.A., Raffei, A.F.M., Razak, M.F.A., Ahmad, A.: Distributed denial of service attack detection in IoT networks using deep learning and feature fusion: a review. None (2024). https://doi.org/10.58496/mjcs/2024/004
2. Haque, S., El-Moussa, F., Komninos, N., Rajarajan, M.: A systematic review of data-driven attack detection trends in IoT. In: Italian National Conference on Sensors (2023). https://doi.org/10.3390/s23167191
3. Alsoufi, M.A., et al.: Anomaly-based intrusion detection systems in IoT using deep learning: a systematic literature review. In: Multidisciplinary Digital Publishing Institute (2021). https://doi.org/10.3390/app11188383
4. Abdullahi, M., et al.: Detecting cybersecurity attacks in internet of things using artificial intelligence methods: a systematic literature review. Multidisciplinary Digital Publishing Institute (2022). https://doi.org/10.3390/electronics11020198
5. Yaras, S., Dener, M.: IoT-based intrusion detection system using new hybrid deep learning algorithm. Electronics (2024). https://doi.org/10.3390/electronics13061053
6. Manimaran, A., Kartheesan, L., Kumutha, D., Surendran, R.: An optimized hybrid deep learning framework for monitoring botnet attacks in IoT networks (2024). https://doi.org/10.1109/ICICNIS64247.2024.10823344
7. Tekleselassie, H.: DDoS detection on internet of things using unsupervised algorithms. EDP Sciences (2021). https://doi.org/10.1051/e3sconf/202129701005
8. Avci, S., Koca, M.: Predicting DDoS attacks using machine learning algorithms in building management systems. Multidisciplinary Digital Publishing Institute (2023). https://doi.org/10.3390/electronics12194142
9. Singh, N.J., Hoque, N., Singh, K.R., Bhattacharyya, D.K.: Botnet-based IoT network traffic analysis using deep learning. Secur. Priv. (2023). https://doi.org/10.1002/spy2.355
10. Clinton, U.B., Hoque, N., Singh, K.R.: Botnet-based IoT network attacks identification using LSTM. In: Proceedings of International Conference on Computing Communication and Networking Technologies (ICCCNT) (2023). https://doi.org/10.1109/ICCCNT56998.2023.10307716
11. Kumar, A.K., et al.: Enhanced hybrid deep learning approach for botnet attacks detection in IoT environment (2024). https://doi.org/10.1109/ICSPIS63676.2024.10812621
12. Alkhudaydi, O.A., Krichen, M., Alghamdi, A.D.: A deep learning methodology for predicting cybersecurity attacks on the internet of things. Multidisciplinary Digital Publishing Institute (2023). https://doi.org/10.3390/info14100550