Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH Open Access

Intelligent intrusion detection framework for multi-clouds – IoT environment using swarm-based deep learning classifier

Syed Mohamed Thameem Nizamudeen[1*]

Abstract

In the current era, a tremendous volume of data has been generated by using web technologies. The association between different devices and services have also been explored to wisely and widely use recent technologies. Due to the restriction in the available resources, the chance of security violation is increasing highly on the constrained devices. IoT backend with the multi-cloud infrastructure to extend the public services in terms of better scalability and reliability. Several users might access the multi-cloud resources that lead to data threats while handling user requests for IoT services. It poses a new challenge in proposing new functional elements and security schemes. This paper introduces an intelligent Intrusion Detection Framework (IDF) to detect network and application-based attacks. The proposed framework has three phases: data pre-processing, feature selection and classification. Initially, the collected datasets are pre-processed using Integer- Grading Normalization (I-GN) technique that ensures a fair-scaled data transformation process. Secondly, Opposition-based Learning- Rat Inspired Optimizer (OBL-RIO) is designed for the feature selection phase. The progressive nature of rats chooses the significant features. The fittest value ensures the stability of the features from OBL-RIO. Finally, a 2D-Array-based Convolutional Neural Network (2D-ACNN) is proposed as the binary class classifier. The input features are preserved in a 2D-array model to perform on the complex layers. It detects normal (or) abnormal traffic. The proposed framework is trained and tested on the Netflow-based datasets. The proposed framework yields 95.20% accuracy, 2.5% false positive rate and 97.24% detection rate.

Keywords Multi-clouds- IoT services, Intrusion detection framework, Netflow-based datasets, Rat inspired optimizer and convolutional neural network

Introduction

The recent development of web technologies has greatly influenced today's world. The deployment of the Internet of Things (IoT) plays a crucial part in the environment. As per the report [1] from Ericson, a plethora of devices and gadgets would be interlinked with each other. Similar to data published by Statista Research in 2019 [2], the volume of interconnecting devices would go beyond 75 trillion by 2025, which could lead to a serious financial crisis on the IoT effect. These reports depict the scope of the IoT in a real-life scenario. A PC comprises several PCs with diverse channels for proportionate data and resources. The gadgets inside the PCs are termed hubs [3]. An anomaly is a major issue that prevails among interlinked devices and other secured networks. Intruders find a new path to steal sensitive information (or) perform any malicious activities that could affect IoT users' lifestyles. Security plays a vital role in protecting IoT users.

The preservation of connected devices and the recognition of intruders are defined as the Intrusion Detection Framework (IDF). Several IDFs have been introduced to secure communication via

*Correspondence:
Syed Mohamed Thameem Nizamudeen
smthameem@gmail.com
[1] University of Houston-Clear Lake, Houston, Texas 77058, USA

Internet technologies. The performance of the associated devices is keenly monitored to eradicate malicious behaviour. In case of detecting any malicious activities, the users are awakened by a notification from the connected devices [4]. The advancements in IoT technologies have undergone different versions to make them compact and portable. Recently, it's been widely adopted in remote locations. Regardless of the limited computing power in terms of size and battery capacity, the usage of the IoT environment is still in the developmental stage. On the other hand, several lightweight protocols have been established to preserve the communication medium. The design of detecting and predicting intruders in real-time networks is becoming a more challenging task due to the dynamic network traffic changes. Many studies have depicted the detecting model for intrusion systems by learning the intended data type. Though the results are enhanced on the specified dataset, the deployment of generated IDF is a tedious task in real-time due to the deviation between actual and observed network traffic [5]. The data type used for training the model makes it more difficult to observe similar features in real-time. Prior IDFs can detect only the limited attacks on which it has been trained to detect the intended attacks. In [6], the author has stated that the restricted computational resources will take enormous computational time and communication loads. Henceforth, deploying IDFs in resource-constrained IoT devices is not feasible. The design of security tools that balance security and performance is quite complicated.

Cloud computing is one of the superior technologies associated with users based on on-demand services. It has greatly revolutionised the IT organisation in delivering elegant services, including servers, spaces, databases, networking, applications, resources, etc. Relied on the usage of computational assets, the concept of multiple clouds is adapted to meet the client's requirements [7]. Inspired by the advantages of service quality and the optimisation of heterogeneous resources, the multi-cloud concept is widely adopted in the IoT environment to deal with security and performance issues. Multi-cloud acts as an intermediary among connected IoT devices, and the necessity of concentrating security is the major focus of this study. Motivated by the efficacy of the deep learning techniques, the design of IDFs is introduced to intelligently detect intruders on a security-aware network. The deep learning concept is gaining interest in resolving diverse issues. It is an extended version of the machine learning technique [8, 9] that resolves the diverse issues of data-intensive computational analysis. Testing dataset needs to be efficient in designing a better IDS. The available datasets have different characteristics from the non-uniform set of data attacks. This challenging issue inspires us to design intelligent and scalable IDS to forecast real-time normal and abnormal traffic.

The main contributions of the study are summarised as follows:

- The significance of the IDF in a multi-cloud-IoT environment is discussed in this study. State-of-the-art works related to machine learning and deep learning techniques are presented.
- Propose Integer-Grading Normalization (I-GN), a simple pre-processing technique to leverage the collected data in a unique grading form. This ensures the fairness of the preserved data for different purposes.
- Opposition-Based Learning (OBL)- Rat Inspired Optimizer (RIO) is a novel feature selection technique that extracts significant features by exploring and exploiting the local searching process. The design of rat inspired optimiser analyses the entire data to select the fittest features.
- 2D-Array-based Convolutional Neural Network (2D-ACNN) is employed to classify the attack classes. The proposed model resolves the overfitting issue by incorporating filtering layers for regularisation.
- The designed framework is tested and implemented on a combined dataset NF-UQ-NIDS, yielding the best detection accuracy than the prior method.

The paper is presented as follows:

The "Related Surveys" discusses the IDS model using machine learning and deep learning classifiers in Part 2.
The "Proposed framework" that discusses the proposed design of IDF in Part 3.
The "Experimental results and discussion" portrays the implementation setup and result in part 4.
The "Experimental results and discussion" portrays the implementation setup and result in part 4.
"Conclusion" concludes the proposed work's findings in part 5.

Related surveys

This section presents the reviews of existing Machine learning and Deep learning techniques explored in the Intrusion Detection Framework (IDF) to identify the research gaps. In [7], the author presented the issues in IDS exploring the threats like intended attacks, data exfiltration and identity attacks. A comparative study is done to analyse the characteristics of the attacks. However, the current attacks were not explored. A systematic study was done on the security approaches [8] and the solutions in IoT networks. The demerits of the ML and DL techniques were discussed to meet the security requirements. Along with that, the research scope of IoT security was discussed. Similarly, the author in [9] has discussed the overview of security and privacy approaches in defining the heterogeneous devices using the Edge Computing (EC) model. The adoption of ML and DL approaches in EC-oriented applications was presented. The advancement of intrusions and their scope under fog computing with IoT were studied by [10]. Different solutions to the intrusions for IoT architecture were studied [11]. A diverse set of architectures, detection models and prevention models was portrayed wherein the detection efficiency was improved. The review study focused on the kinds of architectural prototypes related to IoT security.

Machine learning-oriented IDFs

In [12], the author has discussed distributed IDFs using fog computing that specifically detected the Distributed Denial of Services (DDoS) attacks. The combinations of AI and fog nodes were employed to detect the attacks. The design of IPFS based data storage model has balanced the IoT data. Relied on the estimation of mutual information gain, a significant performance was achieved. Key-based and clustering-based algorithms were introduced to recognise the routing attacks under IoT networks [13]. The deployment of a cluster-based approach [14] has significantly improved the detection performance of similar attacks. In [15], a two-stage IDS model that depicted the naïve Bayes classification and the k-mean algorithm was introduced. These algorithms have improved the detection accuracy of low-scale computational loads during training. A lightweight IDS model was introduced to detect the Sybil attack [16]. Under the RPL network system, the employment of Artificial Bee Colony (ABC) was modelled to identify the same identifiers using the Cooja simulator. The system has improved detection accuracy with a high false positive rate. An intelligent three-layer IoT security architecture [17] that includes the types of associated devices and their activities and findings of threats based on the specified activity monitoring modules and the classification of attacks based on monitored results. A software-defined AI-assisted two-stage IDS was studied [18] to improve the throughput rate. Initially, the common features were aggregated using bat algorithms. Using random forest classifiers, the weights of the data samples were calculated to detect the classes. Likewise, the characteristics of attackers were studied widely in [19]. In [20], honeypot modules were discussed to prevent real-time IDS. The author has developed a simulation tool to detect malicious activities early. However, it is not suitable to detect heterogeneous devices. The class imbalance problem was resolved using smote and random sampling techniques [21] on CIC IDS 2018 datasets. Similarly, IIoT unbalanced datasets were resolved using the XGBoost model [22]. To address the threats in smart homes [23], the author has introduced a testbed to capture the data packets. The power consumption of the data packets was minimised. An intrusion detection model with an ensemble classifier [24] was introduced to compare the performance of naïve Bayes, logistic regression, decision tree, and voting technique. Though the detection accuracy was improved, the new instances were not detected properly.

Deep learning-based IDF

In [25], a deep learning-based IDF was studied using a Recurrent Neural Network (RNN). The designed RNN method was tested and validated on the NSL-KDD dataset. This system has resolved the class imbalance problem on multi-class classifiers. With several hidden nodes at a learning rate of 0.5, a higher accuracy is achieved for multi-class classification. Auto-encoder [26] is also studied to reduce computational loads. The encoder phase was improved using the time-series data. The system has balanced the data overloads without compromising efficiency. A collaborative IDS [27] was introduced to secure and preserve data privacy in IoT-cloud networks. The data transmission process was managed using blockchain technology. The network attack was classified then using a bi-directional LSTM model, which was tested on UNSW NB15 and BoT IoT datasets. The efficiency of the classifier was proved on the small-scale data. The detection of online attacks using the URL was studied [28]. In connection to the Edge of Things environment, the security preservation on web apps was analysed using deep models. Each data was trained independently and then deployed on the servers to ease the detection process, yet, the computational time was decreased. Similarly, the cyber risks were reduced using the auto-encoder model comprising the convolutional and recurrent operators [29]. In coordination with it, CNN-related anomaly detection [30] was investigated to extend the capabilities of traffic prediction. Data imbalance [31] issues were resolved to detect the anomalies using conditional GANs. The feature set was improved to build the minority data classes. It was detected using Feed Forward Neural Network (FFN) by the concept of ocGAN and bcGAN systems. However, this system has reduced the false positive rate.

Then, a light Deep Neural Network (DNN) [32] was examined to reduce the high-dimensional features with an improved Principal Component Analysis (PCA). This has significantly reduced the network traffic. In [33], deep learning-based automatic IDS is introduced to improve the accuracy and scalability issues. A Secured Automatic Two-level Intrusion Detection System (SATIDS) is designed by Long Short-Term Memory (LSTM) network. This scheme has detected the subtypes of the network attacks. This has significantly increased the accuracy rate. A semi-supervised learning technique [34] is designed for IDS using outlier-aware deep autoencoder and self-tuning threshold selection. It is intended for cyber-physical systems. The imperfections of the training data are modulated under different normal operations. It has improvised and reduced computational efforts. Fog-cloud-based IDS [35] is introduced to mitigate the effects of unnecessary features. Time-series data has been incorporated using a Recurrent Neural Network and a Bi-directional Long Short-term memory framework. The system has significantly reduced the data sizes without compromising the attack detection ability. In [36], a hybrid CNN and LSTM framework is formulated to detect intruders in Industrial IoT networks. It is mainly intended to preserve the sensitive data access model. These hybrid combination has determined the highest accuracy rate.

From the above-conducted reviews, it is clearly stated that an intelligent design of IDF is still in the developmental stage. There are many reasons to yield the lowered performance of the designed IDFs. The challenges identified by the review studies are compiled as follows:

- *Data inconvenience:* Intrusion detection is a sensitive topic wherein the security and privacy of the organisation and the users are highly involved. Hence, synthetic datasets display the enhanced efficiency of the designed techniques for the intervening period.
- *Running time:* It involves both training and testing the data samples. In some cases, high-complexity models must be designed to achieve optimal solutions.
- *Number of parameters:* Parameter setting defines the consecutive steps to accomplish the goals. It has two types, viz, learnable parameters, which are defined during training the features and the other, hyperparameters which are manually defined before the initialisation of the training process. Therefore, it must be addressed for efficient detection and predictions.
- *Feature representation:* Machine learning models take the feature vector from raw data that reduces the issues of data overlapping, overfitting and underfitting.
- *Interpretability:* Data mining techniques like decision trees, Bayesian networks etc., have strong interpretability, whereas solution converging speed estimation, related to data skewness issues, is always a challenging task. Thus, the use of machine learning will give scalable convergence speed.
- *Class imbalance:* It is a common problem. To some extent, machine learning models will resolve by designing single-label and multi-label classifications.

Proposed framework

In this study, a novel and intelligent Intrusion Detection Framework (IDF) is designed by improvising the feature selection and classification phases.

System model

The deployment of multiple clouds is in two forms, namely, federated and multi-cloud. The federated cloud forms an agreement among the providers and fails to ensure the usage of different clouds for the users. Regardless of it, multi-cloud develops dynamic coordination among the providers according to the user's requirements. And also, the users are aware of resource utilisation. Henceforth, this work discusses multi-cloud usage on IoT security services wherein the users and organisations dynamically access the services. The communication scenario between the multi-cloud and the IoT services to the users is shown in Fig. 1.

Under the roof of a multi-cloud environment, the users use IoT services that could affect the conventional cloud architecture. The theme of a multi-cloud environment is to ensure a fair collaboration among the industry and the users. The administration of the required services is still an open issue. Besides service administration, the adoption of multi-clouds also fetches security issues. The conventional security solutions for the available networks impact the need for dynamic coordination between IoT users and the services. It brings problems like communication overhead, data unavailability and unauthorization. Consider an instance service is instantiated from a foreign cloud. The cloud_IoT users are not aware of the received network service, so they will rely on the provider to verify the service modelling. The communication on malicious or faulty services will severely affect the security properties like confidentiality, integrity and authentication, data modification and so on without the consent of users.

Design of intelligent intrusion detection framework (IDF)

The designed framework concentrates on devising all the steps of the intrusion detection phases, such as pre-processing, feature selection and classification. The detection model is designed between the web and the IoT gateway to recognise the attacks on the IoT network and protect the multi-cloud infrastructure. Figure 1 presents the integrated process of the

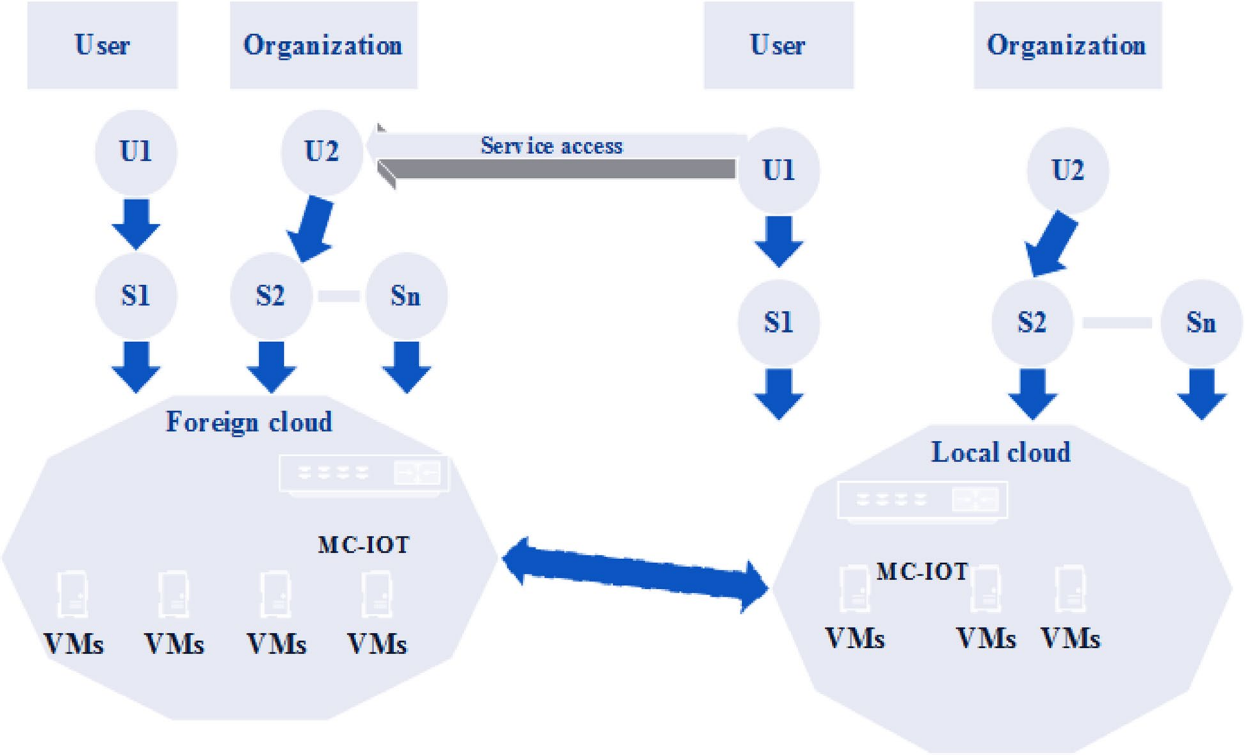


Fig. 1 IoT network communication model in multi-cloud

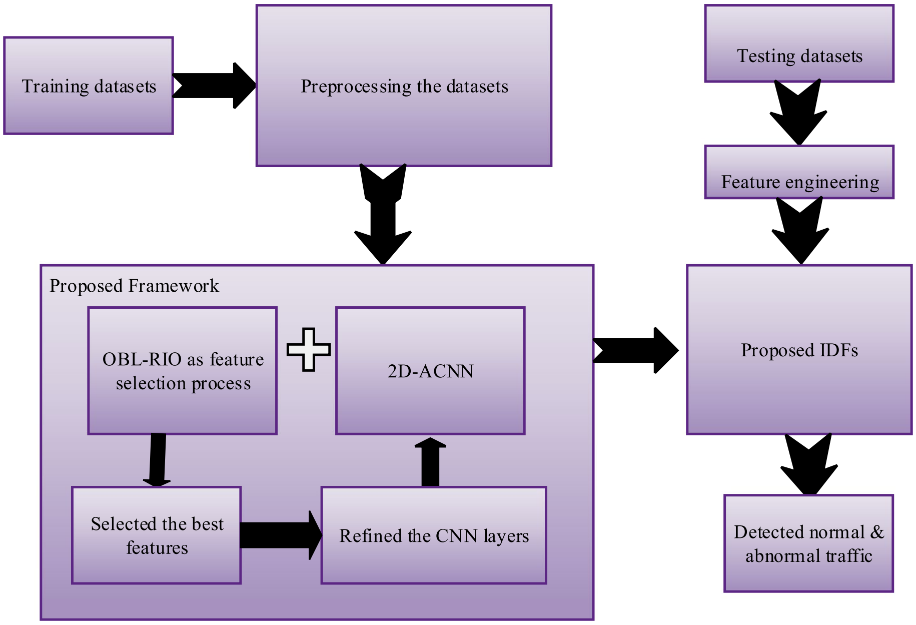

Training datasets
Preprocessing the datasets
Testing datasets
Feature engineering
Proposed Framework
OBL-RIO as feature selection process
2D-ACNN
Proposed IDFs
Selected the best features
Refined the CNN layers
Detected normal & abnormal traffic

Fig. 2 Design of proposed IDF

proposed intention. The different forms of IoT devices are connected with the various forms of cloud layers. Therefore, the theme of the proposed framework is to detect intruders at the IoT-Cloud network gateway. Figure 2 presents the overall workflow of the proposed IDFs. It includes phases such as pre-processing, normalisation, feature extraction and classification. They are explained as follows:

Data pre-processing phase

Pre-processing is the earliest phase that deals with the collected records to make them useful for future data analysis. Some information about the records is invalid, which is eliminated at the initial stage, e.g. IP addresses and flow duration. Each feature is normalised using a novel Integer-oriented Grading Normalization (I-GN) technique. Like the other normalisation techniques like min–max and z-score, the proposed I-GN yields well-structured data from 0 to 1. It is expressed as follows:

$$GN_I = \frac{(|A|) - \left(10^{d-1}\right) * (|P|)}{10^{d-1}} \tag{1}$$

where,

A ➔ Specified element of the data;
d ➔ Count of digits in element A;
P ➔ First digit of the element A;
GN_I ➔ Normalized value ranging between 0 and 1.

Since the collection of datasets is tremendous in volume, the proposed pre-processing technique will leverage the data elements irrespective of size and volume.

Feature selection phase

Assortments of the features are the second phase of selecting the best features to design an efficient classifier. The theme is to observe all features keenly to yield the best features using the extractor module. The Rat-Inspired Optimizer (RIO) has recently been widely adopted to resolve complex design solutions. An adaptive feature selection technique is designed for IDFs by improving the RIO algorithm. The concept of Opposition based learning (OBL) is combined with the RIO algorithm to increase the performance and efficiency of the feature selector module. The course of the proposed feature selection phase is given in Fig. 3. In general, RSO is a population-oriented optimisation that begins with a random set of initial data points and tries to modulate the solutions to the best. The initial position of the rats is expressed in Eq. (2).

$$\boldsymbol{R_i} = r_{i-min} + rand \times (r_{i-max} - r_{i-min})i = 1, 2, \ldots N \tag{2}$$

where,

r_{i-max}➔ Upper bounds for the i$^{\text{th}}$ variable;
r_{i-min}➔ Lower bounds for the i$^{\text{th}}$ variable;
N ➔ Aggregate count of used agents.

The upper and lower bound implies the minimum and maximum set of iterations. Generally, the rat follows the bait behaviours wherein the highest search agents know the bait's placement. The local search agent is adopted, which suffers from being trapped in local optima and brings complex problems. Some agents rely on the local minimum during the local search process and become static for several iterations. Henceforth, the search agent updates the position from its highest search agents. The revised position of the rat is estimated from Eq. (3).

$$\overrightarrow{\underset{i}{pos}}\,(r+1) = |\overrightarrow{\underset{a}{pos}}\,(r) - \overrightarrow{pos}| \tag{3}$$

where,

$\overrightarrow{\underset{i}{pos}}\,(r+1)$ ➔ The updated positions of i$^{\text{th}}$ rats;
$\overrightarrow{\underset{a}{pos}}\,(r)$ ➔ The latest estimated optimal solution.

The instant position $\overrightarrow{pos}$ of the rats is estimated from Eq. (4).

$$\overrightarrow{pos} = M \times \overrightarrow{\underset{i}{pos}}\,(r) + N(\overrightarrow{\underset{a}{pos}}\,(r) - \overrightarrow{\underset{i}{pos}}\,(r)) \tag{4}$$

Pertaining to the above, the parameters M and N are expressed as follows:

$$M = rand - j \times (\frac{rand}{max_{iter}})\text{j} = 1, 2, 3 \ldots.\text{max}_{\text{iter}} \tag{5}$$

$$N = 2 \times rand \tag{6}$$

where,

rand ➔ Assuming a random number from [1–5];
N ➔ Random number between [0, 2];
j ➔ Present use of iteration;
max$_{\text{iter}}$ ➔ Maximum use of iterations to execute the task.

The random function set is evaluated by an objective function. It is improved based on the behaviors of rats. A global optimizer is designed for the best local search process to increase the convergence speed performance and estimate the global features. The contrary position of the rat is also calculated using the OBL process, which is expressed as:

$$\overrightarrow{\underset{i}{Pos}} = \left(r^{i_{\max}} - r^{i_{min}}\right) - r^{i}\text{i} = 1, 2 \ldots \text{n} - \text{dimensional} \tag{7}$$

Relying on the objective function f(.), the positions of the rat at pos_i and $\overrightarrow{\underset{i}{pos}}$ are estimated. The task of the fitness function is to evaluate the derived solution that

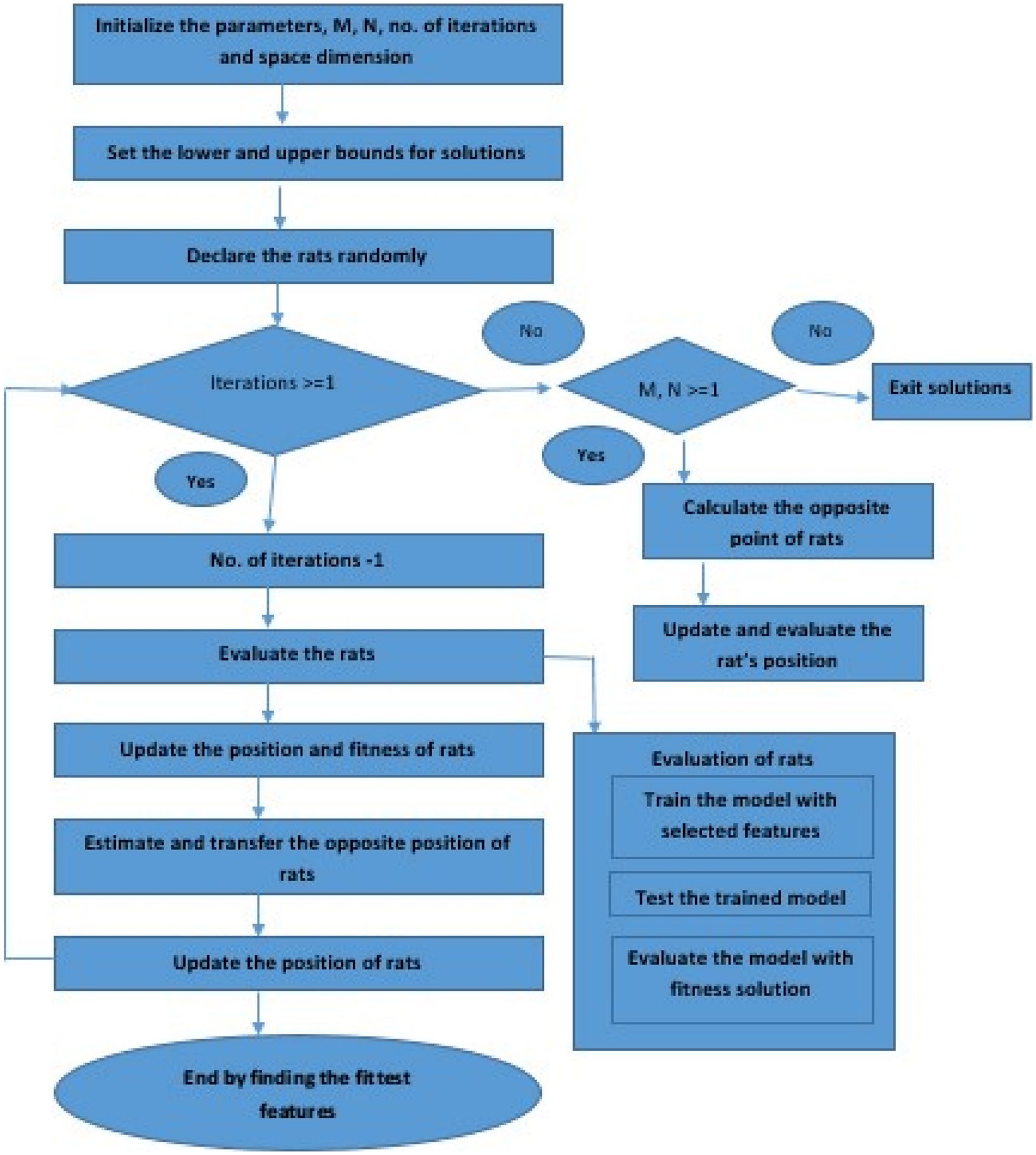

Initialize the parameters, M, N, no. of iterations and space dimension
Set the lower and upper bounds for solutions
Declare the rats randomly
Iterations >=1
No
M, N >=1
No
Exit solutions
Yes
Yes
Calculate the opposite point of rats
No. of iterations -1
Update and evaluate the rat's position
Evaluate the rats
Update the position and fitness of rats
Evaluation of rats
Train the model with selected features
Estimate and transfer the opposite position of rats
Test the trained model
Update the position of rats
Evaluate the model with fitness solution
End by finding the fittest features

Fig. 3 Design of OBL_RIO

contains the subset of features. The estimation of True Positive Rate (TPR), False Positive Rate (FPR) and the number of features are considered. The number of features is considered to display the quality of the solutions from TPR and FPR. The fitness function is expressed as:

$$Fitness_r = w_1 \times \frac{Chosenfeat.}{No.offeat} + w_2 \times FPR + w_3 \times \frac{1}{TPR} \tag{8}$$

wherein the values of weight W is 1, $w_1 = 0.1$ and w_2 and $w_3 = 0.45$. Henceforth, if the $f(\overrightarrow{pos}_i)$ is better than the f(pos_i), then the search agent at pos_i will be altered with the $\underset{i}{\overrightarrow{Pos}}$. In some cases, the largest optimal value can arrived at from the worst solution that can be altered with the new solution using the Eq. (8)

$$f(r_{worst}) = \begin{cases} rand_1 \times \underset{i}{\overrightarrow{pos}}(r) if rand_3 \leq 0.5 \\ (r^{i_{\max}} - r^{i_{min}}) - r_i, if rand_3 > 0.5 \end{cases} \quad (9)$$

where,

r_{worst} ➔ Holds maximum value of the objective function;

rand$_1$, rand$_2$ and rand$_3$ are the random numbers from 0 to 1.

Classification phase

The classification phase is the third phase of designing a classifier for attack classes. 2D array-based Convolutional Neural Network (2D-ACNN) is introduced to classify the attacks. In general, the CNN consists of three layers, viz, input layers include the set of chosen features; hidden layers include the stacked set of filtering layers, Convolutional layers, ReLU layers, pooling layers, and the fully connected layers and output layers predict the outcome from the set of combined pooling layers. Each layer has its characterisation according to the assigned neurons. The inputs to the CNN layer have been devised by the concept of 2D arrays wherein the information is preserved in arrays. This data analysis format will optimise the classification time and heavy computational efforts. The entities of the 2D-ACNN are the filtering layer, convolutional layers, ReLU layers, pooling layers and fully connected layers.

In the proposed study, 20 filtering layers, 5 convolutional layers and 5 subsampling layers are considered fully connected layers. The variation of the features subset and the layers defines the performance of each layer. The roles of the proposed layers are explained as follows:

1. Filtering layers: A rich set of feature subsets is taken as the input. The features are stored in column-wise ordering. Here, the 1st column is held first, followed by the 2nd row of an array. This process continues until all features are preserved.
2. Convolution layers: It intends to learn the data of the chosen features. An input layer holds the feature detector and feature map modules. The convolutional filters work by column-wise feature mapping. The filter is multiplied by the original matrix values of the data. Finally, an aggregator function operates on the multiplications to derive a single set of layers. Likewise, the convolutional filter performs on all data of the chosen features. It is displayed in 2D array form representing a 'feature map'. The set of feature maps is used to define the first convolutional layer. It is mathematically expressed as:

$$S_x^y = f(\sum_{i=1}^{N} S_i^{c-1} * Q_{ij}^c + L_j^c) \quad (10)$$

Where,

S_x^y ➔ Output of convolution operation between input and vth convolutional kernel;

S_i^{c-1} ➔ Output of the predecessor layer i.e. the ith vector receiving the data of present convolution layer c;

Q_{ij}^c ➔ jth convolution kernel of c;

L_j^c ➔ The offset term of the present convolution kernel;

N ➔ local feature information extracted traversing to the convolution kernel.

3. Pooling layers: It uses a random sampling operator to generate the matrix with the help of convolution operators. At this layer, the most significant features that deliberately reduce the dimensionality issue are obtained.
4. Fully connected layers: The final classified outcomes are yielded from the consecutive functioning of the convolution kernel and pooling operators.

Experimental results and discussion

The proposed IDF is employed on the combined datasets, namely, NF-UQ-NIDS datasets with four databases [37]. They are NF-UNSWNB15, NF-BoTIoT, NF-ToNIoT and NF-CSE CI-CIDS2018. The prior researchers have converted the captured raw packet information into the Netflow format. This Netflow format aggregates a similar feature set which has resolved the issues of various features on diverse datasets. Since it is practically suitable in different scenarios, the datasets have better scalable properties. The different versions of datasets have been encompassed properly and the categories of all attacks are shown in Tables 1, 2 and 3. In this study, the biased nature of the record is removed by eliminating the features like IP addresses and the flow duration at the initial

Table 1 NF-UQ-NIDS datasets and its characterization

Datasets	Volume of data	Volume of Training data	Volume of testing data
NF-BoTIoT	600100	480080	120020
NF-ToNIoT	1379274	1103419	275855
NF-UNSWNB15	1623118	1298494	324624
NF-CSE CI-CIDS2018	8392401	6713920	1678481

Table 2 List of attacks and their classes

Datasets	Attacks classes	
NF-BoTIoT	Benign	13859
	DDoS	56844
	DoS	56833
	Reconnaissance	470655
	Theft	1909
NF-ToNIoT	Benign	270279
	Backdoor	17247
	DDoS	326345
	DoS	17717
	Injection	468539
	Mitm	1295
	password	156299
	Ransomware	142
	Xss	99944
	Scanning	21467
NF-UNSWNB15	Analysis	1995
	Backdoor	1782
	benign	1550712
	DoS	5051
	Exploits	24736
	Fuzzers	19463
	Reconnaissance	12291
	shellcode	1365
	Worms	153
	Generic	5570
NF-CSE CI-CIDS2018	Benign	7373198
	Bot	15683
	Brute force web	2613
	Brute force XSS	1745
	DDoS attack HOIC	230
	DDoS attack LOIC UDP	1667
	DDoS attacks LOIC HTTP	378199
	DDoS attacks Golden Eye	32850
	DDoS attacks hulk	108136
	DDoS attacks slow HTTP test	105550
	DoS attacks slowloris	22825
	FTP bruteforce	193360
	Infiltration	62072
	SQL injection	36
	SSH bruteforce	94237

stage. The proposed work is implemented and tested using Python programming version 3.6.9. It consists of Pandas libraries. Keras is also used for classification purposes.

The above Tables 4, 5, 6 and 7 represents the performance of the proposed I-GN technique. A simplified preprocessing approach is made to leverage the collected data. The proposed preprocessing technique can work on a large amount of data with the motive of an individual element grading process. All data values are graded between 0 and 1. The preprocessing steps are applied to all data to achieve fairness. It has been seen that the percentage of unstructured data increases, and the time taken for the inferences also increases. Henceforth, the proposed I-GN employs traditional round-off values for easy interpretation.

To our knowledge, the Netflow datasets are simulated under an optimisation problem. Thus, the conventional RSO and the proposed OBL-RSO techniques are compared, and the parameters are shown in Table 8. The Table 9 and Figs. 4 (a-d) and 5 represent the selected features from their evaluated fitness solutions. The results show that the proposed OBL-RSO has yielded the best solutions than the conventional RSO. Since the technique aims to lower the objectives of the solution, the quality of the solution at 30–40 iterations is achieved. Estimating the opposite point helps enhance the solutions and removes the non-steadiness in local search. The choice of selected features and the deep learning model has been analysed and learnt from numerous correlations between the normalised data. The selected features significantly contribute to learning more about intrusion detection ability. Even though OBL-RSO performs better in yielding the most notable features, more efforts are required in multi-class systems, which is the limitation of this study.

To build an IDF, all features are not necessary, and thus, the significant features are chosen using the OBL-RSO technique. This process is important to eliminate the unnecessary features that cause heavy computational time and effort. Once the feature selection process is completed, then the set of chosen features are fed into the 2D-CNN classifier. This classifier aims to forecast the classes of normal traffic and abnormal traffic. There are several metrics to validate the proposed classifiers. Here, a simple confusion matrix is employed

Table 3 Intrusion types and its target

Target part of the intrusion	Origin of the intrusion
Application layer	Bruteforce; XSS; SQL injection; Fuzzers; DoS flood; DoS slowloris
Network layer	Bruteforce; DoS based amplification; DoS synflood; Unsolicited traffic; Backdoor

Table 4 Preprocessing the NF-BoTIoT- sample data

Original data (L4 SRC port as features)	Proposed I-GN technique
80	0.800
49160	0.491
3456	0.345
80	0.800
80	0.800
0	0
365	0.365
80	0.800
80	0.800
50850	0.508

Table 5 Preprocessing the NF-ToNIoT-Sample data

Original data (L4 SRC port as features)	Proposed I-GN technique
63318	0.633
57442	0.574
57452	0.574
138	0.138
51989	0.519
53927	0.539
60453	0.604
49866	0.498
36125	0.361
0	0

Table 6 Preprocessing the NF-UNSWNB15- Sample data

Original data (L4 SRC port as features)	Proposed I-GN technique
62073	0.620
32284	0.322
21	0.210
23800	0.238
63062	0.630
57349	0.573
41660	0.416
29259	0.292
1813	0.181
20139	0.201

Table 7 Preprocessing the NF-CSE CI-CIDS2018-Sample dataset

Original data (L4 SRC port as features)	Proposed I-GN technique
51128	0.511
443	0.443
12262	0.122
61023	0.610
443	0.443
55252	0.552
443	0.443
63445	0.634
49248	0.492
51109	0.511

Table 8 OBL-RSO parameters for the feature selection phase

Parameters	Representation
No. of rats (N_r)	Count of the available solutions
Position of each rat	Solutions include chosen features
Best rat	The solution has the optimal fitness value
Opposite position of the rat	Change towards the best rat
Fitness function	Evaluates by TPR, FPR and no. of features
max_{iter}	Usage of iterations

to do the experimental process. The confusion matrix consists of four labels, namely,

a) True Positive (TP): The count of abnormal traffic classes classified accurately.
b) True Negative (TN): The count of normal traffic classes classified accurately.
c) False Positive (FP): The count of normal traffic established as abnormal traffic.
d) False Negative (FN): The count of abnormal traffic established as normal traffic.

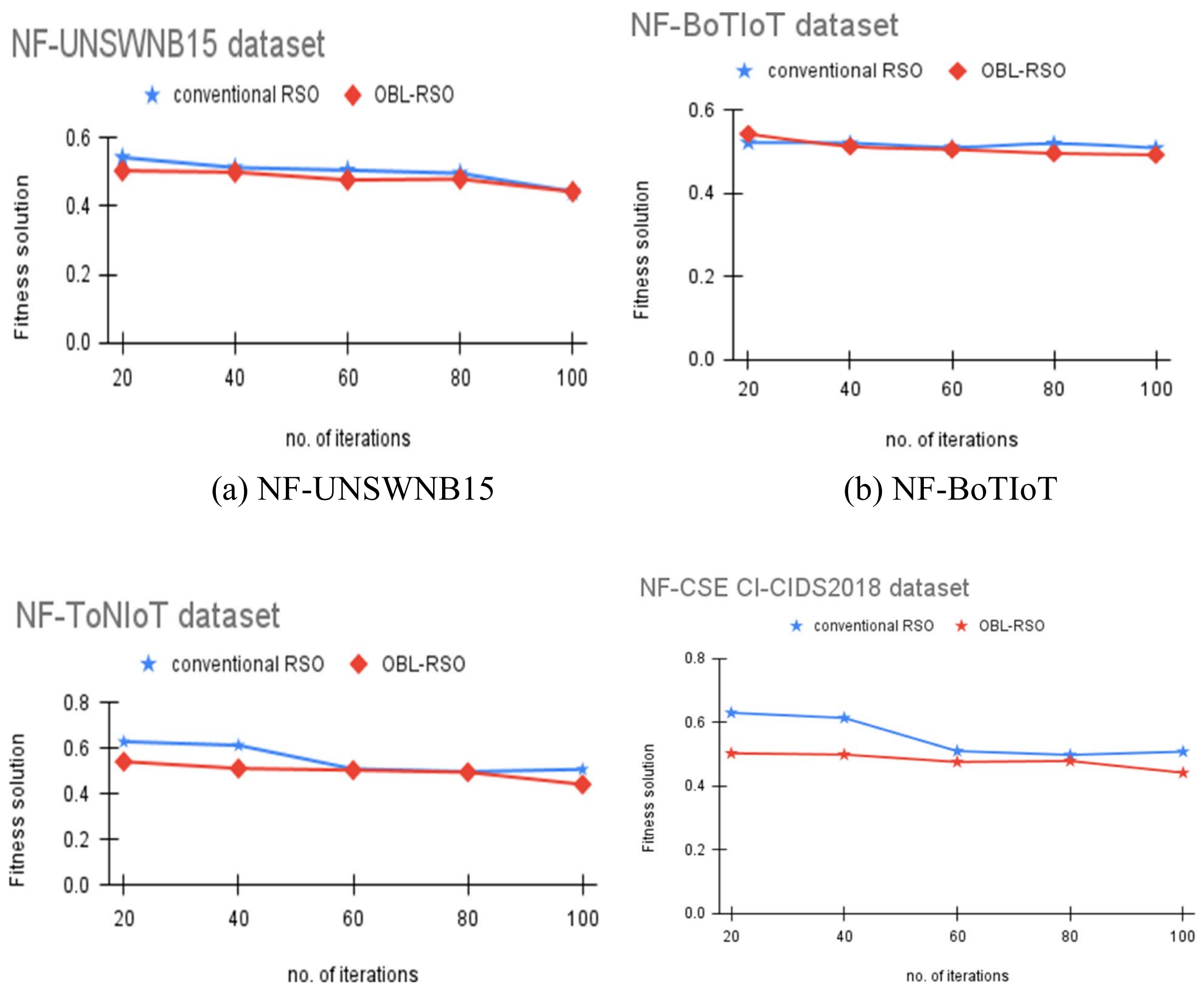
Pertaining to it, the performance parameters are studied as:

a) Recall (or) Detection rate: It dictates the correct measures of the attacks to the attack classes. It is expressed as:

$$TPR\,(Detection\,rate) = \frac{TP}{TP + FN}$$

Table 9 Selected features of the datasets

Features	Chosen features
IPV4 SRC ADDR; IPV4 DST ADDR; L4SRC PORT, L4 DST PORT; L7 PROTO, PROTOCOL; IN BYTES; OUT BYTES; TCP FLAGS; IN PKTS; OUT PKTS; FLOW DURATION MILLISECONDS	L4 SRC PORT; L4 DST PORT; PROTOCOL; TCP FLAGS; L7 PROTO; IN BYTES; OUT BYTES; IN PKTS & OUT PKTS



(a) NF-UNSWNB15

(b) NF-BoTIoT

(c) NF-ToNIoT

(d) NF-CSE CI-CIDS 2018

Fig. 4 **a**-**d** Convergence curve of all datasets – fitness value vs no. of iterations. **a** NF-UNSWNB15. **b** NF-BoTIoT. **c** NF-ToNIoT. **d** NF-CSE CI-CIDS 2018

b) False positive rate (FPR): It dictates the correct measures of the non-attacks to the attack classes. It is expressed as:

$$FPR = \frac{FP}{TN + FP}$$

c) Classification accuracy: It dictates the measures of the correctly classified attack classes to the aggregate count of classes. It is expressed as:

$$Classification\,accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

DoS	2241	0	1	0	1	0
DDoS	758	69347	830	542	14236	0
Fuzzers	2	0	845	1	40	0
Backdoor	0	0	0	1235	0	0
Benign	1	13	15	1	119474	0
Brute force Web	0	0	0	0	0	2
	DoS	DDoS	Benign	Fuzzers	Backdoor	Brute force Web

Fig. 5 Confusion matrix for multi-class classification

Table 10 NF-UQ-NIDS dataset with feature selection technique followed by 2D-ACNN

Datasets	TPR ± STDV	FPR ± STDV	Accuracy ± STDV	F-measure ± STDV
NF-BoTIoT	0.8012 ± (0.011)	0.056 ± (0.007)	0.859 ± (0.004)	0.845 ± (0.006)
NF-ToNIoT	0.856 ± (0.014)	0.085 ± (0.000)	0.893 ± (0.008)	0.884 ± 0.014
NF-UNSWNB15	0.707 ± (0.012)	0.057 ± (0.003)	0.652 ± (0.010)	0.795 ± (0.005)
NF-CSE CI-CIDS2018	0.613 ± (0.002)	0.037 ± (0.003)	0.805 ± (0.010)	0.761 ± (0.002)

Table 11 Performance analysis between existing and proposed methods

Performance (%)	Proposed(NF_ UQ-NIDS dataset)	LeNet network model (NSL-KDD dataset) [38]
Accuracy	95.20	97.29
False positive rate	2.5	6.5
Detection rate	97.24	98.55

d) F-measure: It dictates the measure of the classification accuracy of the model with special reference to the recall and precision parameters. It is expressed as:

$$F - measure = \frac{2 * TP}{2 * TP + FP + FN}$$

The Tables 10 and 11 and Fig. 6 portrays the performance analysis of the proposed classifier. The design of the proposed rat swarm optimiser is to develop robust IDF without deteriorating the performance of accuracy,

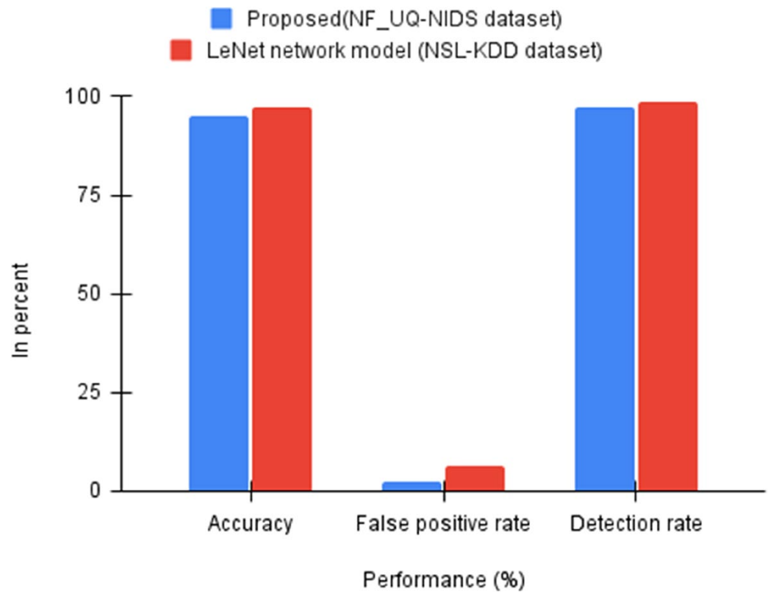


Fig. 6 Comparative graph – existing and proposed method

false positive rate and detection rate. The first challenge is to design a consecutive set of attack signatures for the best feature selection process. Since the 2D-CNN scheme lacks attack data, adopting I-GN and OBL-RSO is useful for creating relevant training data with its attack types. Though the generated attacks are not actual attacks, the formation of complex attacks is impossible due to heterogeneity. The heterogeneous nature of multi-cloud-IOT is more vulnerable because of the complex topology formation and high network connectivity between traffic variables. The proposed result has maintained better TPR and accuracy with minimal time.

Conclusion

This paper is a novel and intelligent attempt to upgrade the Intrusion Detection Framework (IDF) using a swarm-based deep learning classifier. The proposed IDF consists of three phases to detect the network and application layer-based attacks. The collected dataset is pre-processed using Integer- Grading Normalization (I-GN) technique that easily scales the data to ensure fairness. Then, the Opposition-based Learning- Rat Inspired Optimizer (OBL-RIO) is designed as a feature selection technique to extract the significant features by exploring the local search of all features. Finally, a 2D-Array-based Convolutional Neural Network (2D-ACNN) is proposed as the binary class classifier. The input features are preserved in the 2D-array model to perform on the complex layers. It detects normal traffic and intruder traffic. The proposed framework is trained and tested on the Netflow-based datasets. The proposed framework yields 95.20% accuracy, 2.5% false positive rate and 97.24% detection rate.

Acknowledgements
Not applicable.

Authors' contributions
The Manuscript in its entirety has been written the Sole Author—Syed Mohamed Thameem Nizamudeen.

Author's information
Syed Mohamed Thameem Nizamudeen is an Independent Researcher with a Masters degree in Management Information Systems from University of Houston – Clearlake, Texas, Houston, USA, 77058. E-mail: smthameem@gmail.com. His areas of interests include Application Modernization leveraging Cloud Technologies. Syed is currently a Technology Executive with one of the Top cloud Service Providers in the world. Syed has also worked with C-Suite executives of Fortune 100 firms in the past advising them on their Application Modernization efforts in aspects of IaaS, PaaS, SaaS, Cloud Security, Multi cloud, Internet Of Things during his tenure with reputed technology Advisory firms – "PricewaterHouseCoopers" and "Ernst & Young". His main interests include computer applications, Cloud Computing, Artificial Intelligence in Cloud Computing.

Funding
Not applicable.

Availability of data and materials
Dataset downloaded from the open source websites. The download link is given in References section.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 23 December 2022 Accepted: 14 August 2023
Published online: 22 September 2023

References

1. Ejaz W, Anpalagan A (2019) Internet of things for smart cities: technologies, big data and security, Springer, SpringerBriefs in Electrical and Computer Engineering
2. Fizza K, Banerjee A, Mitra K, Jayaraman PP, Ranjan R, Patel P, Georgakopoulos D (2021) Qoe in iot: a vision, survey and future directions. Discover Internet Things 1(1):1–14
3. Huang Z, Wu W, Shan F, Bian Y, Lu K, Li Z, Wang J, Wang J (2020) Couas: enable cooperation for unmanned aerial systems, ACM Trans. Sens Netw 16(3):1–19
4. Kreibich C (2001) Network Intrusion Detection: Evasion, Traffic Normalization, and EndTo-End Protocol Semantics
5. Kabir MF, Hartmann S (2018) Cyber security challenges: an efficient intrusion detection system design, in: 2018 International Young Engineers Forum (YEF-ECE), IEEE pp. 19–24
6. Xiao L, Wan X, Lu X, Zhang Y, Wu D (2018) Iot security techniques based on machine learning: how do iot devices use ai to enhance security? IEEE Signal Process Mag 35(5):41–49
7. Nisioti A, Mylonas A, Yoo PD, Katos V (2018) From intrusion detection to attacker attribution: a comprehensive survey of unsupervised methods. IEEE Commun Surv Tutor 20(4):3369–3388
8. Hussain F, Hussain R, Hassan SA, Hossain E (2020) Machine learning in iot security: current solutions and future challenges. IEEE Commun Surv Tutor 22(3):1686–1721
9. Singh S, Sulthana R, Shewale T, Chamola V, Benslimane A, Sikdar B (2021) Machine-learning-assisted security and privacy provisioning for edge computing: a survey. IEEE Internet Things J 9(1):236–260
10. de Souza CA, Westphall CB, Machado RB, Lof L, Westphall CM, Geronimo GA (2022) Intrusion detection and prevention in fog based IoT environments: a systematic literature review. Comput Netw. 214:109154
11. Benkhelifa E, Welsh T, Hamouda W (2018) A critical review of practices and challenges in intrusion detection systems for iot: toward universal and resilient systems. IEEE Commun Surv Tutor 20(4):3496–3509
12. Kumar P, Kumar R, Gupta GP, Tripathi R (2021) A distributed framework for detecting ddos attacks in smart contract-based blockchainiot systems by leveraging fog computing. Trans Emerg Telecommun Technol 32(6):e4112
13. Choudhary S, Kesswani N. Detection and prevention of routing attacks in internet of things, in: 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), IEEE, 2018, pp. 1537–1540
14. Choudhary S, Kesswani N (2019) Cluster-based intrusion detection method for internet of things. 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA). pp 1–8
15. Vishwakarma M, Kesswani N. A two-stage intrusion detection system (tids) for internet of things, in: Advances in Deep Learning, Artifcial Intelligence and Robotics. Springer. 2022. pp. 89–97
16. Murali S, Jamalipour A (2019) A lightweight intrusion detection for Sybil attack under mobile rpl in the internet of things. IEEE Internet Things J 7(1):379–388

17. Anthi E, Williams L, Słowińska M, Theodorakopoulos G, Burnap P (2019) A supervised intrusion detection system for smart home iot devices. IEEE Internet Things J 6(5):9042–9053
18. Li J, Zhao Z, Li R, Zhang H (2018) Ai-based two-stage intrusion detection for software defned iot networks. IEEE Internet Things J 6(2):2093–2102
19. Moustafa N, Turnbull B, Choo K-KR (2018) An ensemble intrusion detection technique based on proposed statistical flow features for protecting network trafc of internet of things. IEEE Internet Things J 6(3):4815–4830
20. Baykara M, Das R (2018) A novel honeypot based security approach for real-time intrusion detection and prevention systems. J Inform Secur Appl 41:103–116
21. Seth S, Chahal KK, Singh G (2021) A novel ensemble framework for an intelligent intrusion detection system. IEEE Access 9:138451–138467
22. Le T-T-H, Oktian YE, Kim H (2022) Xgboost for imbalanced multiclass classification-based industrial internet of things intrusion detection systems. Sustainability 14(14):8707
23. Tushir B, Dalal Y, Dezfouli B, Liu Y (2020) A quantitative study of ddos and e-ddos attacks on wif smart home devices. IEEE Internet Things J 8(8):6282–6292
24. Abbas A, Khan MA, Latif S, Ajaz M, Shah AA, Ahmad J (2022) A new ensemble-based intrusion detection system for internet of things. Arab J Sci Eng 47(2):1805–1819
25. Yin C, Zhu Y, Fei J, He X (2017) A deep learning approach for intrusion detection using recurrent neural networks, Ieee. Access 5:21954–21961
26. Shone N, Ngoc TN, Phai VD, Shi Q (2018) A deep learning approach to network intrusion detection. IEEE Trans Emerg Topics Comput Intell 2(1):41–50
27. Alkadi O, Moustafa N, Turnbull B, Choo K-KR (2020) A deep blockchain framework-enabled collaborative intrusion detection for protecting iot and cloud networks. IEEE Internet Things J 8(12):9463–9472
28. Tian Z, Luo C, Qiu J, Du X, Guizani M (2019) A distributed deep learning system for web attack detection on edge devices. IEEE Trans Industr Inf 16(3):1963–1971
29. Khan IA, Moustafa N, Pi D, Sallam KM, Zomaya AY, Li B (2022) A new explainable deep learning framework for cyber threat discovery in industrial iot networks, IEEE Internet of Things Journal
30. Saba T, Rehman A, Sadad T, Kolivand H, Bahaj SA (2022) Anomalybased intrusion detection system for iot networks through deep learning model. Comput Electr Eng 99:107810
31. Ullah I, Mahmoud QH (2021) A framework for anomaly detection in iot networks using conditional generative adversarial networks. IEEE Access 9:165907–165931
32. Zhao R, Gui G, Xue Z, Yin J, Ohtsuki T, Adebisi B, Gacanin H (2023) A novel intrusion detection method based on lightweight neural network for internet of things, IEEE Internet of Things Journal
33. Rania A. Elsayed, Reem A. Hamada, Mahmoud I. Abdalla, Shaimaa Ahmed Elsaid, Securing IoT and SDN systems using deep-learning based automatic intrusion detection. Aim Shams Engineering Journal. 2023
34. Marta Catillo, Antonio Pecchai & Umberto Villiano, CPS-GUARD: Intrusion detection for cyber-physical systems and IoT devices using outlier-aware deep autoencoders. Comput Secur. 2023;129
35. Naeem Firdous Syed, Mengmeng Ge & Zubair Baig, Fog-cloud based intrusion detection system using Recurrent Neural Networks and feature selection for IoT networks. Comput Netw 2023;225
36. Hakan Can Altunay & Zafer Albayrak, A hybrid CNN+LSTM-based intrusion detection system for industrial IoT networks, Engineering Science and Technology, an International Journal. 2023;38
37. Dataset download link: https://staff.itee.uq.edu.au/marius/NIDS_datasets/
38. Raviprasad B, Mohan CR, Devi GN, Pugalenthi R, Manikandan LC, Ponnusamy S (2022) Accuracy determination using deep learning technique in cloud-based IoT sensor environment. Meas Sens 24:100459

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.