Computers, Materials & Continua
DOI: 10.32604/cmc.2023.032591
Article

Tech Science Press

Optimal Fuzzy Logic Enabled Intrusion Detection for Secure IoT-Cloud Environment

Fatma S. Alrayes[1], **Nuha Alshuqayran**[2], **Mohamed K Nour**[3], **Mesfer Al Duhayyim**[4,*], **Abdullah Mohamed**[5], **Amgad Atta Abdelmageed Mohammed**[6], **Gouse Pasha Mohammed**[6] **and Ishfaq Yaseen**[6]

[1]Department of Information Systems, College of Computer and Information Sciences,
Princess Nourah Bint Abdulrahman University, P. O. Box 84428, Riyadh, 11671, Saudi Arabia
[2]Department of Information Systems, College of Computer and Information Sciences,
Imam Mohammad Ibn Saud Islamic University, Saudi Arabia
[3]Department of Computer Sciences, College of Computing and Information System, Umm Al-Qura University,
Saudi Arabia
[4]Department of Computer Science, College of Sciences and Humanities-Aflaj, Prince Sattam bin Abdulaziz University,
Saudi Arabia
[5]Research Centre, Future University in Egypt, New Cairo, 11845, Egypt
[6]Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University,
AlKharj, Saudi Arabia
*Corresponding Author: Mesfer Al Duhayyim. Email: m.alduhayyim@psau.edu.sa

Received: 23 May 2022; Accepted: 24 June 2022

Abstract: Recently, Internet of Things (IoT) devices have developed at a faster rate and utilization of devices gets considerably increased in day to day lives. Despite the benefits of IoT devices, security issues remain challenging owing to the fact that most devices do not include memory and computing resources essential for satisfactory security operation. Consequently, IoT devices are vulnerable to different kinds of attacks. A single attack on networking system/device could result in considerable data to data security and privacy. But the emergence of artificial intelligence (AI) techniques can be exploited for attack detection and classification in the IoT environment. In this view, this paper presents novel metaheuristics feature selection with fuzzy logic enabled intrusion detection system (MFSFL-IDS) in the IoT environment. The presented MFSFL-IDS approach purposes for recognizing the existence of intrusions and accomplish security in the IoT environment. To achieve this, the MFSFL-IDS model employs data pre-processing to transform the data into useful format. Besides, henry gas solubility optimization (HGSO) algorithm is applied as a feature selection approach to derive useful feature vectors. Moreover, adaptive neuro fuzzy inference system (ANFIS) technique was utilized for the recognition and classification of intrusions in the network. Finally, binary bat algorithm (BBA) is exploited for adjusting parameters involved in the ANFIS model. A comprehensive experimental validation of the MFSFL-IDS model is carried out using benchmark dataset and the outcomes are assessed under distinct aspects. The experimentation outcomes highlighted the superior performance of the MFSFL-IDS model over recent approaches with maximum accuracy of 99.80%.

Keywords: Cloud computing; security; fuzzy logic; intrusion detection; internet of things; metaheuristics

1 Introduction

Recently, the Internet of Things (IoT) network and its smart gadgets are widening in all respects of our community, namely smart grids, smart homes, intelligent distributed networks, smart manufacturing industries, smart virtual learning environments, smart hospitals, and intelligent vehicles, [1]. Currently, various activities associated with IoT have obtained interest within the industry as well as in the academic region because of its potential usage in various human actions. IoT depicts a potential solution for enhancing standards of people's life (for example, the smartwatch, that observes health via sensors [2]), and various technologies made familiar with the decline in sensor prices, the outreach of remote memory services, and huge data. It has been made clear that accessibilities to these resources reinforce IoT while gadgets with various resources are linked to a network, therefore endowing to the advent of new applications [3]. Such a new whole ground has accompanied by a price they are, the need for security. Therefore, large-scale diffusion of such technology is joined by various security difficulties regarding the huge amount of streams of data flowing from or to the smart gadgets themselves [4]. As a result, many IoT applications need protection and security, including precise validation and categorization methods in turn, along with those sufficient solutions for guaranteeing integrity and confidentiality. Furthermore, offering the extensive usage of IoT gadgets, immoral actions would have deep effects on the strength and the security of the whole Internet [5,6]. Fig. 1 illustrates the process invovled in deep learning (DL) techniques.

Currently, there seems to be developing attention to the exploration of the efficiency of machine learning (ML) methods for improving IoT security [7–9]. ML methods vary in their computational and efficiency necessities. For instance, few supervised models have high memory and central processing unit (CPU) necessities while their training stage since it demands low resources once applied in comparison to unsupervised learning model [10]. When ML methods are utilized for enhancing the network's security they are performed on conventional networked devices and they are specially placed in order to safeguard the IoT architecture.

In contrast, DL has obtained lots of momentum in the past few years as it was utilized for solving, with a growing extent of precision, distinct issues, that were generally solved by conventional ML methods, namely prediction, recession, classification, and son on [11]. Actually DL includes a group of well-known ML methods on the basis of artificial neural networks permitting one to experiment with the information processing of biological nervous systems comprises of multiple perceptron' layers [12]. Governing security difficulties in another network includes 3 broad approaches such as detection, mitigation, and prevention. All the three mentioned measures must be adopted for successful security solutions for IoT networks. For the scope of this work, we aim at Intrusion Detection Systems (IDS) and assume DL related IDS for classifying and identifying network traffic inside an IoT environment [13].

This paper presents novel metaheuristics feature selection with fuzzy logic enabled intrusion detection system (MFSFL-IDS) in the IoT environment. The presented MFSFL-IDS technique employs data pre-processing to transform the data into useful format. Besides, henry gas solubility optimization (HGSO) technique is applied as a feature selection approach to derive useful feature vectors. Moreover, adaptive neuro fuzzy inference system (ANFIS) technique was utilized for the recognition and classification of intrusions in the network. Finally, binary bat algorithm (BBA) is exploited for adjusting parameters involved in the ANFIS model. A comprehensive experimental validation of the MFSFL-IDS model is carried out using benchmark dataset and the results are assessed under distinct aspects.

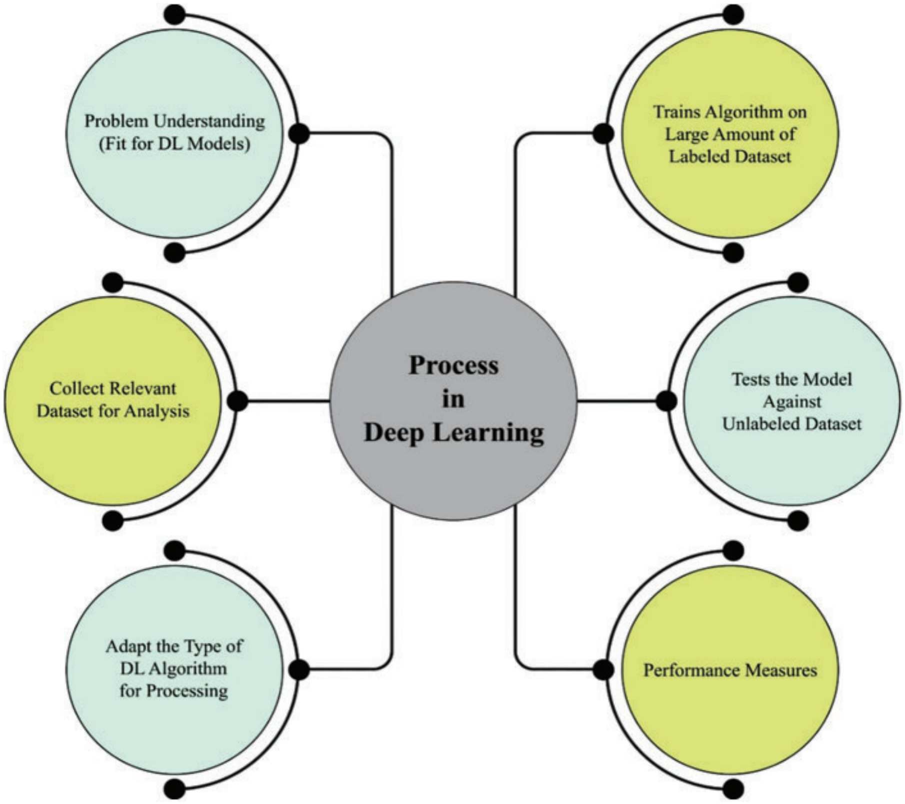

Problem Understanding (Fit for DL Models)
Collect Relevant Dataset for Analysis
Adapt the Type of DL Algorithm for Processing
Process in Deep Learning
Trains Algorithm on Large Amount of Labeled Dataset
Tests the Model Against Unlabeled Dataset
Performance Measures

Figure 1: Process in DL technique

2 Literature Review

In this section, a detailed survey of recently developed approaches for intrusion detection classification in IoT environments is provided. Nguyen et al. [14] presented real guard and deep neural network (DNN)-based network IDS (NIDS) operated directly on local gateways for protecting IoT gadgets. The advantage of the presented method is that it could precisely identify cyberattacks in real time with a smaller computation footprint. It can be accomplished by a light weighted feature extraction method and an effectual attack detection system powered by deep neural network. In [15], introduce a hybrid ML method using extreme gradient boosing (XGB) with random forest (RF) called XGB-RF for identifying intrusion attacks. The presented model was employed for the N-BaIoT data comprising hazardous botnet attacks. The RF has been utilized for selecting the features and XGB technique was utilized for detecting various kinds of attacks on IoT environment.

In [16], introduced a hybrid mechanism of shallow learning and DL to identify the intrusion in the IoT device. Firstly, the presented method with spider monkey optimization feature selection model search for selecting key features. In [17], proposed an ensemble-based IDS. In the presented method, DT, logistic regression (LR), and Naïve Bayes (NB) were deployed by voting classification afterward analyzing model efficiency with modern technologies. In [18], proposed a new a distributed combined DL-IDS for the Internet of Vehicles (IoV) based on Apache Spark architecture regarding the problem. The cluster integrates DL based convolutional neural network (DL-CNN) and long short term memory (LSTM) models for extracting features.

Alohali et al. [19] presented a novel IDS based ensemble-based voting classification model which integrates traditional classifier as a basic learner and provides the vote to the prediction of the classic classifier to obtain the concluding prediction. Friha et al. [20] a federated learning-based IDS (FELIDS) is proposed to secure agriculture based IoT infrastructure. Especially the FELIDS method protects the data through local learning, whereas device get benefitted in the knowledge of its peers by sharing upgrades in the model with aggregation server which produces an effective method.

3 The Proposed Model

A novel MFSFL-IDS model was established to recognize the existence of intrusions and accomplish security in the IoT environment. The MFSFL-IDS model primarily employed data pre-processing to transform the data into useful format. Followed by, the HGSO algorithm has been utilized to derive useful features. In addition, the BBA-ANFIS classifier is employed for the recognition and classification of intrusions in the network. Fig. 2 illustrates the block diagram of MFSFL-IDS technique.

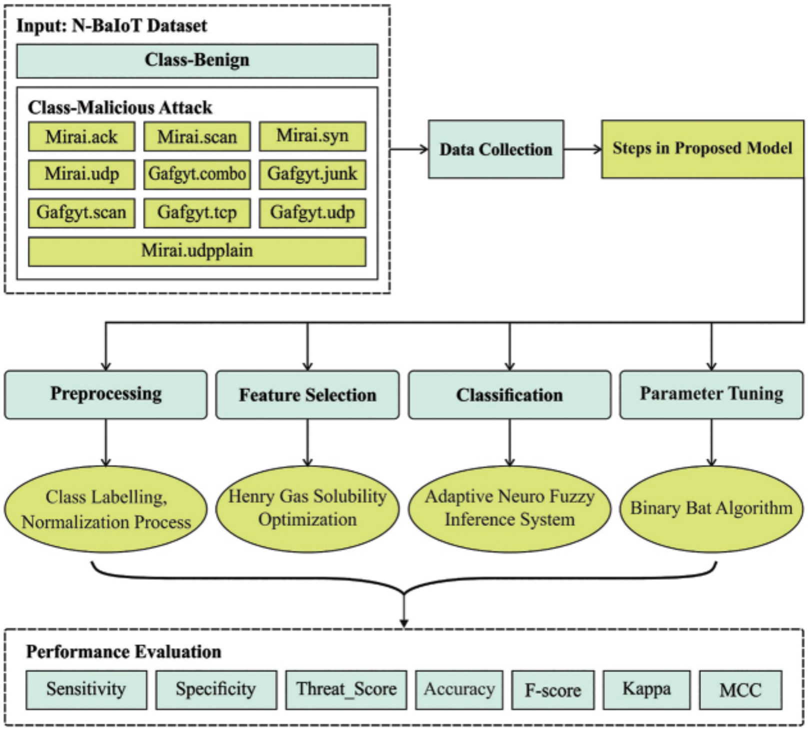


Figure 2: Block diagram of MFSFL-IDS technique

3.1 Data Pre-Processing

In several ML techniques, data normalized was generally utilized to obtain effectual performances. The value of features can vary in lower to higher values. Thus, the normalized procedure was utilized for data scaling, as given in Eq. (1):

$$s = \sqrt{\frac{1}{N-1}\sum\nolimits_{i=1}^{N}\left(x_i - \overline{x}\right)^2} \tag{1}$$

3.2 Feature Selection Using HGSO Algorithm

In order to optimally choose the features, the HGSO algorithm has been exploited for choosing an optimal subset of features [21]. The presented FS solution is a three step process, which can be defined as follows: initially, the HGSO creates population initialization of N candidate solutions, whereas all the individuals characterize a set of features that chosen for calculation. The population X^0 is arbitrarily generated. Here, lb_i and ub_i lower and upper boundaries for every candidate solution i lies within [0, 1]. Hence, every solution x_i^0 need to be transformed into binary x_i^{bin} as follows:

$$x_i^{bin} = \begin{cases} 1 & if\ x_i^0 > 0.5 \\ 0 & otherwise. \end{cases} \tag{2}$$

In order to explain the conversion procedure, we assumed the solution x_i that has six components as $x_i^0 = [0.6, 0.1, 0.7, 0.43, 0.2, 0.81]$. The conversion operation can be used for making a binary vector: $x_i^{bin} = [1, 0, 1, 0, 0, 1]$, while 1 denotes that the feature that selected; or else considered as unselected when 0. Afterward defining the set of features, the fitness function can be estimated for all the solution x_i^{bin} for determining the quality of feature. The objective value for i^{th} solution can be described by the following equation.

$$Fit_1 = w_1 \times Err_i + w_2 \times \frac{d_i}{D} \tag{3}$$

Here $w_1 = 0.99$ and $w_2 = 1 - w_1$, D indicates the overall size of attribute in original data.

Here, the ANFIS classification is employed as evaluator or expert system in the FS method. The Err_i indicates the error rate of testing set calculated by ANFIS mechanism. Now, updating solution comprises of using the clustering stage that focuses on splitting the population into distinct clusters. Next, evaluate the fitness of the new population for determining the optimal solution x_{best}^0. Afterward completing HGSO procedure, we return the optimal solution x_{best}^0. In original dataset, we keep the feature with the value corresponding to one in x_{besf}.

3.3 ANFIS Classifier

Next to feature selection, the ANFIS classifier is employed for the recognition and classification of intrusions in the network [22]. The rules are listed below

Rule 1: if x is A_1 and y is B_1 then $f_1 = p_1x + q_1y + r_1$

Rule 2: if x is A_2 and y is B_2 then $f_2 = p_2x + q_2y + r_2$

where x and y inputs, the fuzzy sets are B_i and A_i. p_i, q_i and r_i indicates the parameter of model that is represented in training and f_i indicates the output defined by the fuzzy rule.

Layer 1: Each node i in layer 1 are adopted with node function

$$O_i^1 = \mu_{A_i}(x) \tag{4}$$

whereas x indicates the input to node i, μ_{A_i} denotes the membership function of A_i. The membership function can be chosen as follows:

$$\mu_{A_i}(x) = \frac{1}{1 + \left[\left(\frac{x-c_i}{a_i}\right)^2\right]^{b_i''}}, \tag{5}$$

Or

$$\mu_{A_i}(x) = exp\left(-\left(\frac{x-c_i}{a_i}\right)^2\right) \tag{6}$$

Here a_i, b_i, c_i denotes the principle parameter set and x indicates the input

Layer 2: The fixed node is presented in layers 2 and 3. The resultant is given for Layer 2:

$$O_i^2 = \omega_i = \mu_{A_i}(x) .\mu_{B_i}(y), \ i = 1, 2 \tag{7}$$

These are firing strengths of rules.

Layer 3: The firing strength of layer 2 can be normalizing with the nodes and marked as N and it is given in the following:

$$O_i^3 = \overline{\omega_i} = \frac{\omega_i}{\omega_1 + \omega_2}, \ i = 1, 2 \tag{8}$$

Layer 4: The product of 1st-order polynomial as well as normalize firing strength. The output is demonstrated in the following:

$$O_i^4 = \overline{\omega} f_i = \overline{\omega}_{\mathrm{i}}(p_i x + q_i y + r_i), \ i = 1, 2 \tag{9}$$

Now $\overline{\omega}_{\mathrm{i}}$ indicates the output of layer 3 and $\{p_i, \ q_i, \ r_i\}$ denotes the subsequent parameter set

Layer 5: There is a single fixed node marked as S in layer 5. Every incoming signal is summed up through the node. The output can be shown in the following:

$$O_i^5 = overalloutput = \sum_i (\overline{\omega}_i f_i) = \frac{\sum_i \omega_i f_i}{\sum_i \omega_i} \tag{10}$$

3.4 Parameter Tuning Using BBA

Finally, the optimization is exploited for adjusting parameters involved in the ANFIS model [23–25]. The BBA distinguishes amongst barrier and prey [26]. It is also noticeable the bat is variation the wavelength of its emitted pulse and the rate of emissions dependent upon its comparative position in terms of the targets. During the context of FS, this provides the technique flexibility for adapting modifies from the feature space and exploring superior solutions. Afterward initialized the place, frequency, and velocity vector, and optimum solution was noted and upgraded throughout the technique. It can be completed mostly utilizing the subsequent formulas:

$$Q_i = Q_{min} + (Q_{min} - Q_{max}) \, . \, rand \tag{11}$$

$$v(i,j) = v(i,j) + (x(i,j) - best(j) \cdot Q_i \tag{12}$$

$$\widetilde{x}(i,j) = x(i,j) + v(i,j) \tag{13}$$

whereas *rand* signifies the arbitrarily created number from the interval of zero and one, and $\widetilde{x}$ implies the novel solution. These novel solutions could not always be implemented and are upgraded according to specific other parameters from the technique. The threshold was chosen according to the value of velocity of the bat that controls the count of exploration, the bat is able of attaining as is provided in Eq. (14). When the specific arbitrary number was lesser than this threshold value, a novel solution is upgraded and bat moves on to novel solution space.

$$V{-}value = \left|\frac{2}{\pi} \arctan\left(\frac{\pi}{2} \mathrm{v}\left(i,j\right)\right)\right|. \tag{14}$$

The rate of pulse emission resolves if the bat is stuck to preceding optimum solution attained or implements the recently upgrade solution. It can be same as an optimum global solution adoption stage is one of the metaheuristics and uses to steer and clear off too much redundant exploration. The loudness parameter establishes more filtering to implementation of solutions as the novel accepted solutions. The solution is only established when the arbitrary number selected is lower than the loudness value and fitness of novel solution was superior to the old solution.

Algorithm 1: Binary bat algorithm

```
Parameter initialization: swarm size $SS(N)$, a, $r$, $Q_{min}$, $Q_{max}$ and $\max_{iter}$.
Random population initialization
$x_i = (x_{i1}, x_{i2}, \ldots, x_{iD}) \in S$ for every solution, the frequency vector $v$ as $D$ dimensional zero vector and velocity vector $v$ as $D$ dimensional zero vector
Find fitness of all solutions
Initialization of $x_{temp}$ as $D$ dimension zero vector.
Save optimal solution and minimal fitness in $F_{min}$
Set $t := 0$.
for $(i = 1; j < SS; i++) do$
    for $(j = 1; j < D; j++)$ do
        $Q_i = Q_{min} + (Q_{min} - Q_{max}) \cdot rand$
        $v(i,j) = \mathrm{v}(i,j) + (x(i,j) - best(j) \cdot Q_i$
        $\widetilde{x}(i,j) = x(i,j) + v(i,j)$
        Binarize $\widetilde{x}(i,j)$
        $V\_value = |\frac{2}{\pi} \arctan\left(\frac{\pi}{2} v\ (i,j)\right)|$
        if $(rand < V\_value)$ then
            $x_{remp}(i,j) = \widetilde{x}(i,j)$
        else
            $x_{temp}(i,j) = x(i,j)$
    end if
    if(rand> $r$) then
        $x_{femp}(i,j) = besr(j)$
    end if
    end for
    fit = fitness of $x_{femp}$.
    If $(fit < F_{min}$ & $rcmd < \zeta l)$ then
        $x(i) = x_{remp}(i)$
        upgrade fitness.
```

(Continued)

Algorithm 1: Continued

```
        end if
        upgrade best and $F_{min}$.
        end for
t = t + 1
until ($r < \max_{iter}$) //Stopping condition gets fulfilled.
Display optimal solution
```

4 Experimental Validation

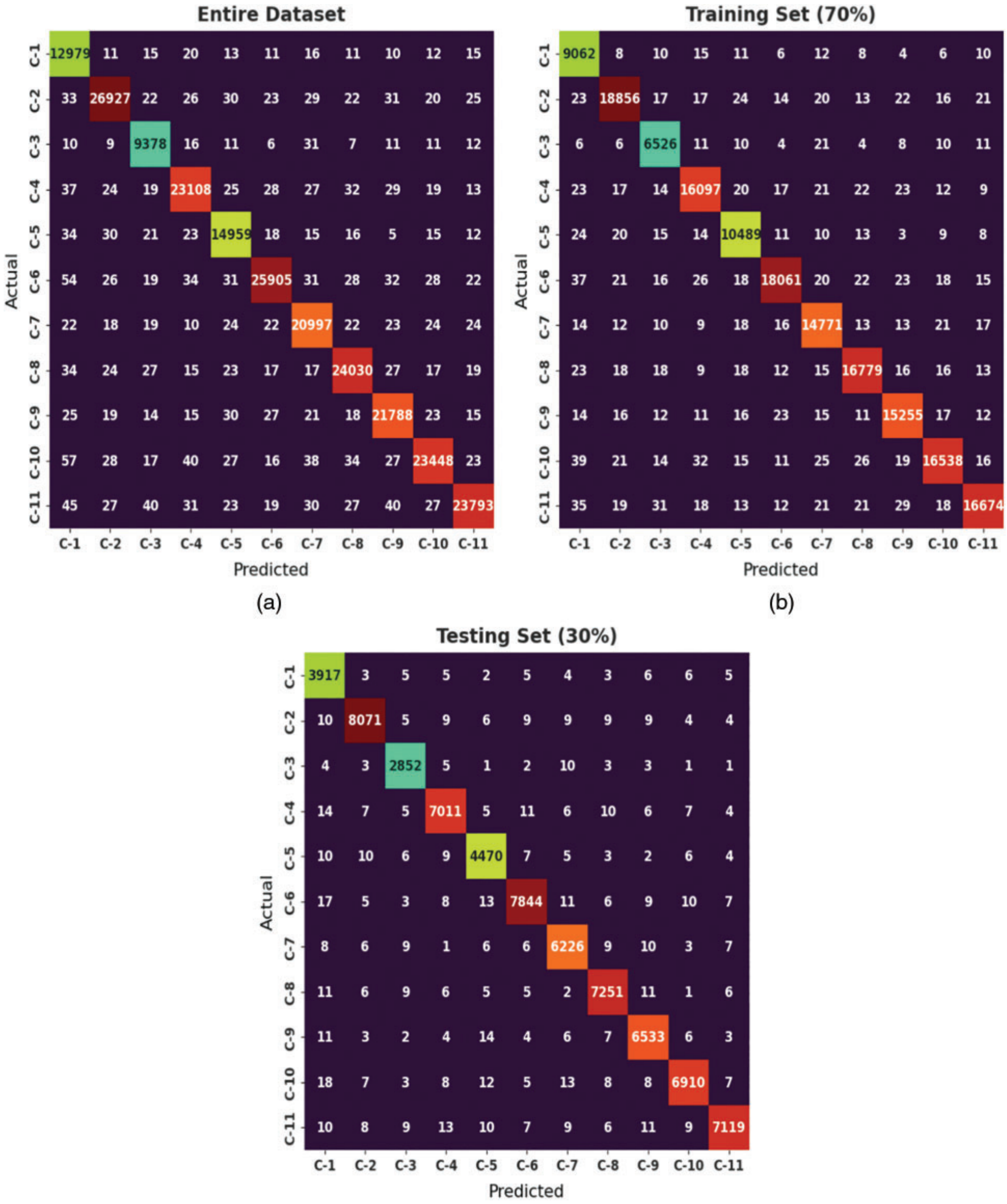
This section investigates the performance of the MFSFL-IDS model using Kaggle dataset (available at https://www.kaggle.com/datasets/mkashifn/nbaiot-dataset). It comprises of 115 attributes with 11 class labels. In this work, a set of 64 features are chosen from the dataset. Tab. 1 provides the details related to the dataset.

Table 1: Dataset details

Categories	Class labels	Class name	No. of instances
Normal	C-1	benign	13113
Malicious	C-2	Mirai.ack	27188
	C-3	Mirai.scan	9502
	C-4	Mirai.syn	23361
	C-5	Mirai.udp	15148
	C-6	Mirai.udpplain	26210
	C-7	Gafgyt.combo	21205
	C-8	Gafgyt.junk	24250
	C-9	Gafgyt.scan	21995
	C-10	Gafgyt.tcp	23755
	C-11	Gafgyt.udp	24102
Total number of instances			229829

Fig. 3 exemplifies the confusion matrix generated by the MFSFL-IDS model. Tab. 2 offers a detailed classification outcome of the MFSFL-IDS model on the entire dataset. The experimental outcomes implied that the MFSFL-IDS model has able to improve performance under all class labels. For instance, with class C-1, the MFSFL-IDS model has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and Mathew Correlation Coefficient (MCC) of 99.79%, 98.98%, 99.84%, 98.17%, and 98.06% respectively. In addition, with class C-3, the MFSFL-IDS method has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.85%, 98.70%, 99.90%, 98.23%, and 98.16% correspondingly. Also, with class C-10, the MFSFL-IDS technique has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.78%, 98.71%, 99.90%, 98.94%, and 98.82% respectively.

Fig. 4 provides an average classification result of the MFSFL-IDS model on entire dataset. The figure portrayed that the MFSFL-IDS model has gained effective outcome with an average $accu_y$, $reca_l$, $spec_y$, and F_{score}, and MCC of 99.80%, 98.89%, 99.89%, and 98.82% respectively.



Entire Dataset

	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8	C-9	C-10	C-11
C-1	12979	11	15	20	13	11	16	11	10	12	15
C-2	33	26927	22	26	30	23	29	22	31	20	25
C-3	10	9	9378	16	11	6	31	7	11	11	12
C-4	37	24	19	23108	25	28	27	32	29	19	13
C-5	34	30	21	23	14959	18	15	16	5	15	12
C-6	54	26	19	34	31	25905	31	28	32	28	22
C-7	22	18	19	10	24	22	20997	22	23	24	24
C-8	34	24	27	15	23	17	17	24030	27	17	19
C-9	25	19	14	15	30	27	21	18	21788	23	15
C-10	57	28	17	40	27	16	38	34	27	23448	23
C-11	45	27	40	31	23	19	30	27	40	27	23793

(a)

Training Set (70%)

	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8	C-9	C-10	C-11
C-1	9062	8	10	15	11	6	12	8	4	6	10
C-2	23	18856	17	17	24	14	20	13	22	16	21
C-3	6	6	6526	11	10	4	21	4	8	10	11
C-4	23	17	14	16097	20	17	21	22	23	12	9
C-5	24	20	15	14	10489	11	10	13	3	9	8
C-6	37	21	16	26	18	18061	20	22	23	18	15
C-7	14	12	10	9	18	16	14771	13	13	21	17
C-8	23	18	18	9	18	12	15	16779	16	16	13
C-9	14	16	12	11	16	23	15	11	15255	17	12
C-10	39	21	14	32	15	11	25	26	19	16538	16
C-11	35	19	31	18	13	12	21	21	29	18	16674

(b)

Testing Set (30%)

	C-1	C-2	C-3	C-4	C-5	C-6	C-7	C-8	C-9	C-10	C-11
C-1	3917	3	5	5	2	5	4	3	6	6	5
C-2	10	8071	5	9	6	9	9	9	9	4	4
C-3	4	3	2852	5	1	2	10	3	3	1	1
C-4	14	7	5	7011	5	11	6	10	6	7	4
C-5	10	10	6	9	4470	7	5	3	2	6	4
C-6	17	5	3	8	13	7844	11	6	9	10	7
C-7	8	6	9	1	6	6	6226	9	10	3	7
C-8	11	6	9	6	5	5	2	7251	11	1	6
C-9	11	3	2	4	14	4	6	7	6533	6	3
C-10	18	7	3	8	12	5	13	8	8	6910	7
C-11	10	8	9	13	10	7	9	6	11	9	7119

(c)

Figure 3: Confusion matrix of MFSFL-IDS technique on distinct datasets

Table 2: Outcomes of MFSFL-IDS method on entire dataset

Entire dataset							
Classes	Accu.	Reca.	Spec.	F-Score	MCC	Kappa score	Threat score
C-1	99.79	98.98	99.84	98.17	98.06	-	-
C-2	99.79	99.04	99.89	99.12	99.00	-	-
C-3	99.85	98.70	99.90	98.23	98.16	-	-
C-4	99.79	98.92	99.89	98.97	98.85	-	-
C-5	99.81	98.75	99.89	98.60	98.50	-	-
C-6	99.79	98.84	99.91	99.06	98.94	-	-
C-7	99.80	99.02	99.88	98.91	98.80	-	-
C-8	99.81	99.09	99.89	99.10	98.99	-	-
C-9	99.81	99.06	99.89	99.00	98.89	-	-
C-10	99.78	98.71	99.90	98.94	98.82	-	-
C-11	99.79	98.72	99.91	98.98	98.86	-	-
Average	99.80	98.89	99.89	98.82	98.72	98.79	97.83

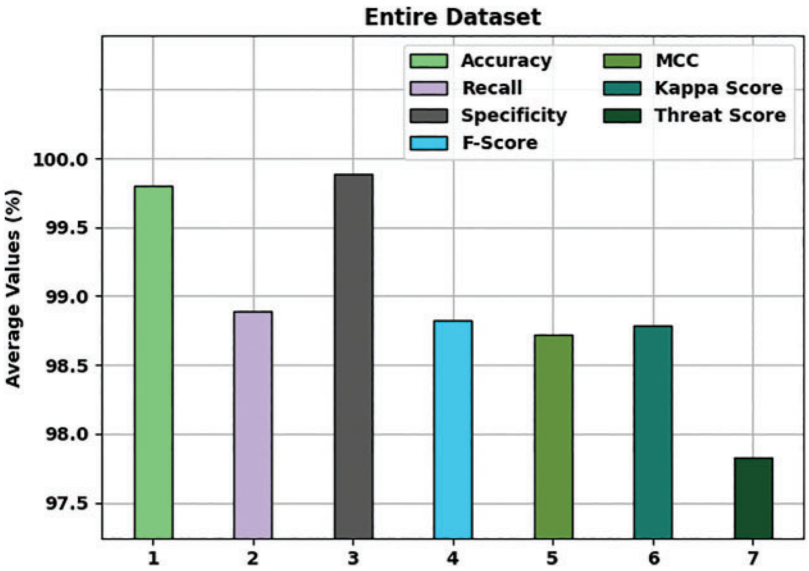


Figure 4: Average analysis of MFSFL-IDS technique on entire dataset

Tab. 3 provided a brief classification outcome of the MFSFL-IDS approach on 70% of training set (TRS) dataset. The experimental outcomes implied that the MFSFL-IDS methodology has accomplished higher performance under all class labels. For instance, with class C-1, the MFSFL-IDS approach has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.80%, 99.02%, 99.84%, 98.22%, and 98.12% correspondingly. Moreover, with class C-5, the MFSFL-IDS model has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.82%, 98.80%, 99.89%, 98.64%, and 98.54% correspondingly. Besides, with class C-10, the MFSFL-IDS approach has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.78%, 98.70%, 99.90%, 98.92%, and 98.80% correspondingly.

Table 3: Result analysis of MFSFL-IDS method with various measures on 70% TRS dataset

Training Set (70%)							
Class labels	Accuracy	Recall	Specificity	F-Score	MCC	Kappa score	Threat score
C-1	99.80	99.02	99.84	98.22	98.12	-	-
C-2	99.79	99.02	99.89	99.09	98.97	-	-
C-3	99.85	98.62	99.90	98.14	98.06	-	-
C-4	99.79	98.91	99.89	98.95	98.84	-	-
C-5	99.82	98.80	99.89	98.64	98.54	-	-
C-6	99.79	98.82	99.91	99.06	98.94	-	-
C-7	99.80	99.04	99.88	98.92	98.81	-	-
C-8	99.81	99.07	99.89	99.08	98.97	-	-
C-9	99.81	99.05	99.89	99.00	98.90	-	-
C-10	99.78	98.70	99.90	98.92	98.80	-	-
C-11	99.78	98.72	99.91	98.96	98.84	-	-
Average	99.80	98.89	99.89	98.82	98.71	98.78	97.82

Fig. 5 gives an average classification result of the MFSFL-IDS model on 70% of TRS dataset. The result portrayed that the MFSFL-IDS system has reached effective outcome with an average $accu_y$, $reca_l$, $spec_y$, F_{score}, MCC, kappa, and threat score of 99.80%, 98.89%, 99.89%, 98.82%, 98.71%, 98.78%, and 97.82% correspondingly.

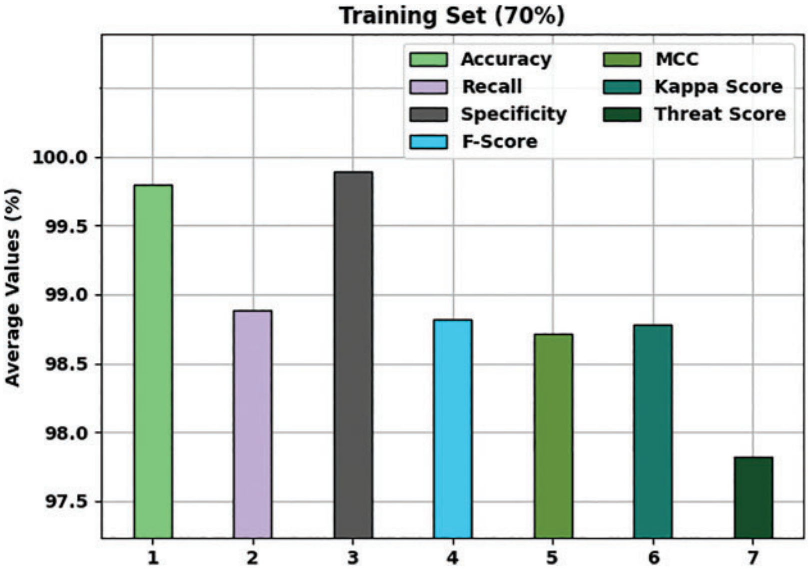


Figure 5: Average analysis of MFSFL-IDS technique on 70% TRS dataset

Tab. 4 provides a detailed classification outcome of the MFSFL-IDS method on the 30% of testing set (TSS) dataset. The experimental outcomes implied that the MFSFL-IDS system has accomplished enhanced performance under all class labels. For sample, with class C-1, the MFSFL-IDS model has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.77%, 98.89%, 99.83%, 98.04%, and 97.92% correspondingly. Furthermore, with class C-3, the MFSFL-IDS model has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.87%, 98.86%, 99.92%, 98.46%, and 98.40% correspondingly. Also, with class C-10, the MFSFL-IDS technique has reached $accu_y$, $reca_l$, $spec_y$, F_{score}, and MCC of 99.79%, 98.73%, 99.91%, 98.98%, and 98.87% correspondingly.

Table 4: Result analysis of MFSFL-IDS method with various measures on 30% of TSS dataset

Testing set (30%)							
Class labels	Accuracy	Recall	Specificity	F-Score	MCC	Kappa score	Threat score
C-1	99.77	98.89	99.83	98.04	97.92	-	-
C-2	99.81	99.09	99.90	99.19	99.08	-	-
C-3	99.87	98.86	99.92	98.46	98.40	-	-
C-4	99.79	98.94	99.89	98.99	98.87	-	-
C-5	99.80	98.63	99.89	98.50	98.40	-	-
C-6	99.78	98.88	99.90	99.05	98.93	-	-
C-7	99.80	98.97	99.88	98.89	98.78	-	-
C-8	99.82	99.15	99.90	99.14	99.04	-	-
C-9	99.80	99.09	99.88	98.98	98.87	-	-
C-10	99.79	98.73	99.91	98.98	98.87	-	-
C-11	99.80	98.72	99.92	99.03	98.91	-	-
Average	99.80	98.90	99.89	98.84	98.73	98.80	97.86

Fig. 6 provides an average classification result of the MFSFL-IDS system on 30% of TSS dataset. The figure outperformed that the MFSFL-IDS approach has gained effectual outcome with an $accu_y$, $reca_l$, $spec_y$, F_{score}, MCC, kappa, and threat score of 99.80%, 98.90%, 99.89%, 98.84%, 98.73%, 98.80%, and 97.86% correspondingly.

The training accuracy (TA) and validation accuracy (VA) attained by the MFSFL-IDS model on test dataset is demonstrated in Fig. 7. The experimental outcomes implied that the MFSFL-IDS model has gained maximum values of TA and VA. In specific, the VA is seemed to be higher than TA.

The training loss (TL) and validation loss (VL) achieved by the MFSFL-IDS model on test dataset are established in Fig. 8. The experimental outcomes inferred that the MFSFL-IDS model has accomplished least values of TL and VL. In specific, the VL is seemed to be lower than TL.

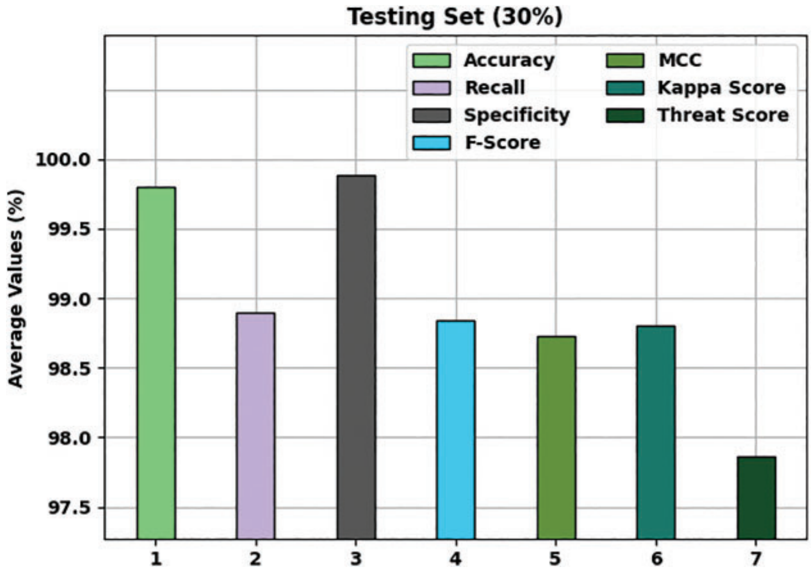


Figure 6: Average analysis of MFSFL-IDS technique on 30% TSS dataset

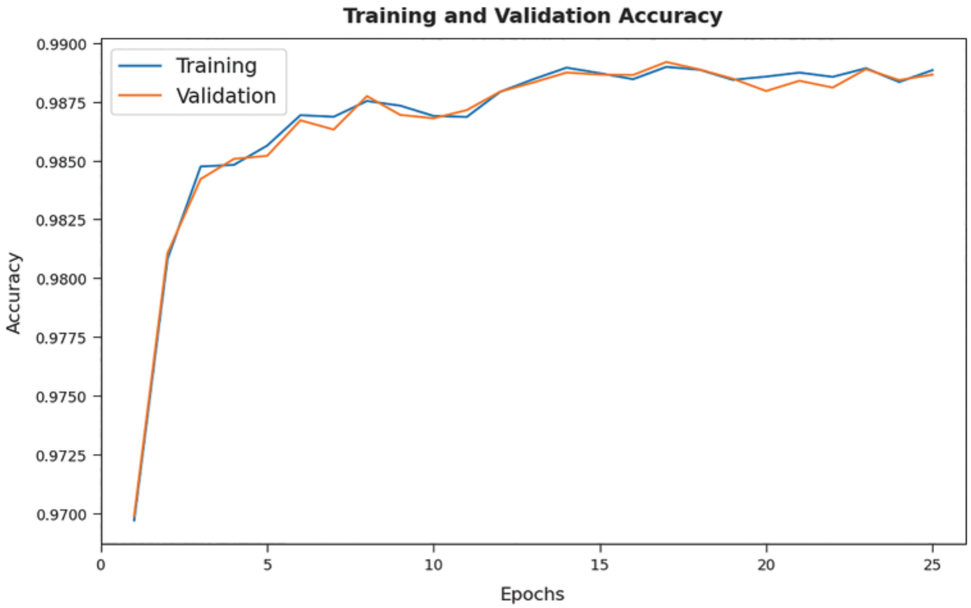


Figure 7: TA and VA analysis of MFSFL-IDS technique

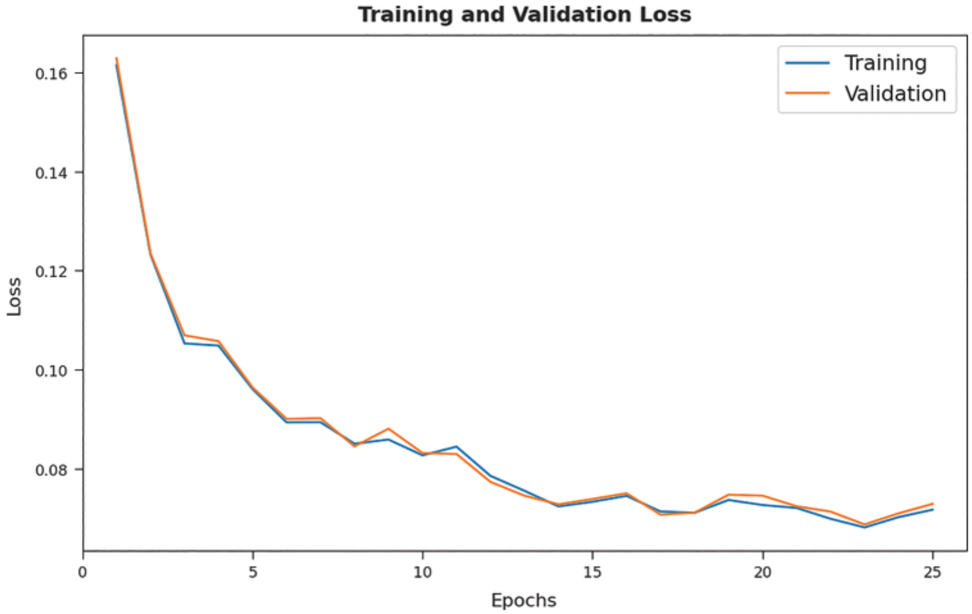


Figure 8: TL and VL analysis of MFSFL-IDS technique

Tab. 5 provides a detailed comparative study of the MFSFL-IDS model with recent models such as RF-RF, XGB-RF, RF-recursive feature elimination (RFE), RF-Feature Selection with Recursive Feature Elimination (RFECV), RF-SelectK, and RF-weighted feature selection (WFS) interms of different measures. Fig. 9 illustrates an extensive comparative investigation of the MFSFL-IDS model with existing models. The experimental results implied that the RF-RF, RF-RFE, RF-RFECV, and RF-SelectK models have accomplished lower values of $accu_y$, $sens_y$, and$spec_y$. Moreover, the RF-WFS model has resulted in slightly improved values of $accu_y$, $sens_y$, and $spec_y$. Though XGB-RF model has exhibited reasonable performance with $accu_y$, $sens_y$, and $spec_y$ of 99.42%, 98.82%, and 99.51%, the presented MFSFL-IDS model has showcased maximum outcome with $accu_y$, $sens_y$, and $spec_y$ of 99.80%, 98.90%, and 99.89% respectively.

Table 5: Comparison study of MFSFL-IDS with recent algorithms

Method	Accu.	Sens.	Spec.	F-Score	Kappa	MCC	Threat score
RF-RF	89.56	89.93	98.97	86.00	88.71	89.77	85.93
XGB-RF	99.42	98.82	99.51	98.44	98.32	98.49	97.67
RF-RFE	89.74	90.05	98.33	86.83	88.49	90.11	86.46
RF-RFECV	89.23	88.86	98.31	86.17	88.17	89.34	85.82
RF-SelectK	89.84	90.20	98.51	86.67	88.67	88.92	86.39
RF-WFS	90.17	89.34	98.53	86.92	89.21	89.74	85.93
MFSFL-IDS	99.80	98.90	99.89	98.84	98.80	98.73	97.86

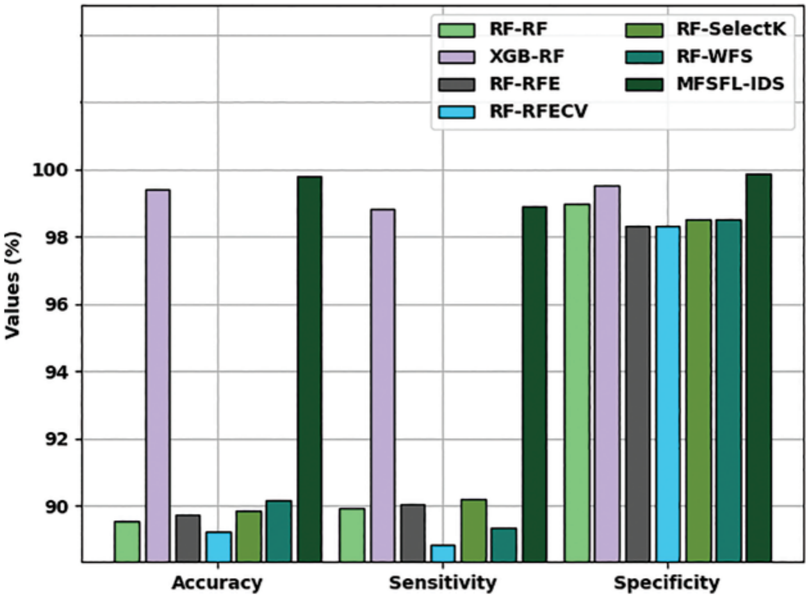


Figure 9: Acc_y, $sens_y$, and $spec_y$ analysis of MFSFL-IDS technique with existing algorithms

Fig. 10 depicts an extensive comparative investigation of the MFSFL-IDS model with existing models. The experimental outcomes referred that the RF-RF, RF-RFE, RF-RFECV, and RF-SelectK models have accomplished lower values of F_{score}, $kappa$, MCC, and TS. Moreover, the RF-WFS approach has resulted in somewhat improved values of F_{score}, $kappa$, MCC, and TS. Though XGB-RF technique has outperformed reasonable performance with F_{score}, $kappa$, MCC, and TS of 98.44%, 98.32%, 98.49%, and 97.67%, the presented MFSFL-IDS algorithm has showcased maximal outcome with F_{score}, $kappa$, MCC, and TS of 98.84%, 98.80%, 98.73%, and 97.86% correspondingly.

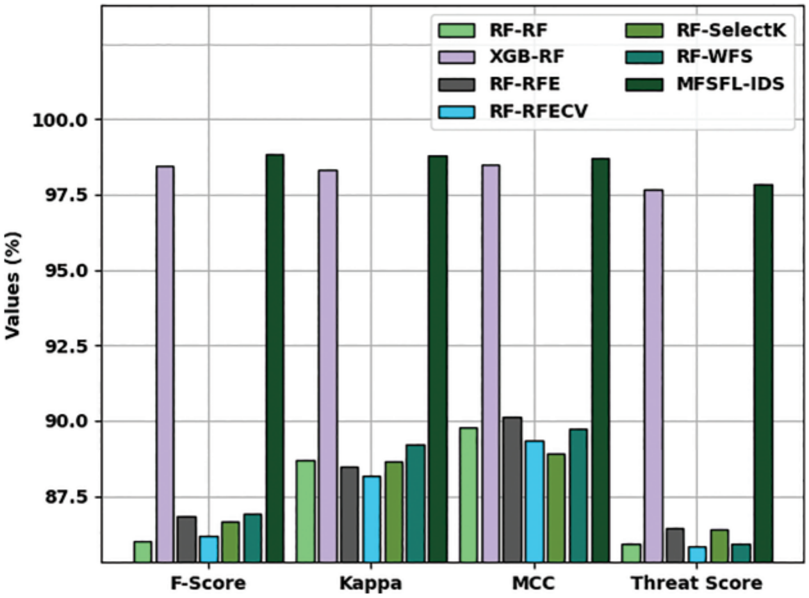


Figure 10: Comparison study of MFSFL-IDS technique with recent models

5 Conclusion

In this study, a novel MFSFL-IDS model was established to recognize the existence of intrusions and accomplish security in the IoT environment. The MFSFL-IDS model primarily employed data pre-processing to transform the data into useful format. Followed by, the HGSO algorithm has been utilized to derive useful features. In addition, the ANFIS classifier is employed for the recognition and classification of intrusions in the network. At last, the BBA is exploited for adjusting parameters involved in the ANFIS model. A comprehensive experimental validation of the MFSFL-IDS model is carried out using benchmark dataset and the outcomes are assessed under distinct aspects. The experimentation outcomes highlighted the better performance of the MFSFL-IDS model over recent approaches. Therefore, the MFSFL-IDS model can be utilized for accomplishing security in the IoT environment. In future, feature reduction and outlier removal approaches can be developed for enhancing security in the IoT environment.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R319), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: (22UQU4310373DSR27).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Asharf, N. Moustafa, H. Khurshid, E. Debie, W. Haider *et al.,* "A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions," *Electronics*, vol. 9, no. 7, pp. 1177, 2020.

[2] M. A. Alsoufi, S. Razak, M. M. Siraj, I. Nafea, F. A. Ghaleb *et al.,* "Anomaly-based intrusion detection systems in IoT using deep learning: A systematic literature review," *Applied Sciences*, vol. 11, no. 18, pp. 8383, 2021.

[3] A. Verma and V. Ranga, "Machine learning based intrusion detection systems for IoT applications," *Wireless Personal Communications*, vol. 111, no. 4, pp. 2287–2310, 2020.

[4] M. Ge, X. Fu, N. Syed, Z. Baig, G. Teo *et al.,* "Deep learning-based intrusion detection for IoT networks," in *2019 IEEE 24th Pacific Rim Int. Symp. on Dependable Computing (PRDC)*, Kyoto, Japan, pp. 256–25609, 2019.

[5] K. A. P. da Costa, J. P. Papa, C. O. Lisboa, R. Munoz and V. H. C. de Albuquerque, "Internet of things: A survey on machine learning-based intrusion detection approaches," *Computer Networks*, vol. 151, pp. 147–157, 2019.

[6] A. M. Hilal, J. S. Alzahrani, I. Abunadi, N. Nemri, F. N. Al-Wesabi *et al.,* "Intelligent deep learning model for privacy preserving IIoT on 6G environment," *Computers, Materials & Continua*, vol. 72, no. 1, pp. 333–348, 2022.

[7] G. Thamilarasu and S. Chawla, "Towards deep-learning-driven intrusion detection for the internet of things," *Sensors*, vol. 19, no. 9, pp. 1977, 2019.

[8] A. Al-Qarafi, F. Alrowais, S. Alotaibi, N. Nemri, F. N. Al-Wesabi *et al.,* "Optimal machine learning based privacy preserving blockchain assisted internet of things with smart cities environment," *Applied Sciences*, vol. 12, no. 12,. pp. 1–17, 2022.

[9] M. A. Rahman, A. T. Asyhari, L. S. Leong, G. B. Satrya, M. Hai Tao *et al.,* "Scalable machine learning-based intrusion detection system for IoT-enabled smart cities," *Sustainable Cities and Society*, vol. 61, pp. 102324, 2020.

[10] I. Abunadi, M. M. Althobaiti, F. N. Al-Wesabi, A. M. Hilal, M. Medani *et al.,* “Federated learning with blockchain assisted image classification for clustered UAV networks,” *Computers, Materials & Continua*, vol. 72, no. 1, pp. 1195–1212, 2022.

[11] A. Raghuvanshi, U. K. Singh, G. S. Sajja, H. Pallathadka, E. Asenso *et al.,* “Intrusion detection using machine learning for risk mitigation in IoT-enabled smart irrigation in smart farming,” *Journal of Food Quality*, vol. 2022, pp. 1–8, 2022.

[12] S. Tsimenidis, T. Lagkas and K. Rantos, “Deep learning in IoT intrusion detection,” *Journal of Network and Systems Management*, vol. 30, no. 1, pp. 1–8, 2022.

[13] I. V. Pustokhina, D. A. Pustokhin, E. L. Lydia, P. Garg, A. Kadian *et al.,* “Hyperparameter search based convolution neural network with Bi-LSTM model for intrusion detection system in multimedia big data environment,” *Multimedia Tools and Applications*, 2021. https://doi.org/10.1007/s11042-021-11271-7.

[14] X. H. Nguyen, X. D. Nguyen, H. H. Huynh and K. H. Le, “Realguard: A lightweight network intrusion detection system for IoT gateways,” *Sensors*, vol. 22, no. 2, pp. 432, 2022.

[15] J. A. Faysal, S. T. Mostafa, J. S. Tamanna, K. M. Mumenin, M. M. Arifin *et al.,* “XGB-RF: A hybrid machine learning approach for IoT intrusion detection,” *Telecom*, vol. 3, no. 1, pp. 52–69, 2022.

[16] T. Li, H. Zhao, Y. Tao, D. Huang, C. Yang *et al.,* “Power intelligent terminal intrusion detection based on deep learning and cloud computing,” *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–14, 2022.

[17] A. Abbas, M. A. Khan, S. Latif, M. Ajaz, A. A. Shah *et al.,* “A new ensemble-based intrusion detection system for internet of things,” *Arabian Journal for Science and Engineering*, vol. 47, no. 2, pp. 1805–1819, 2022.

[18] O. A. Alzubi, “A deep learning-based frechet and dirichlet model for intrusion detection in IWSN,” *Journal of Intelligent & Fuzzy Systems*, vol. 42, no. 2, pp. 873–883, 2022.

[19] M. A. Alohali, F. N. Al-Wesabi, A. M. Hilal, S. Goel, D. Gupta *et al.,* “Artificial intelligence enabled intrusion detection systems for cognitive cyber-physical systems in industry 4.0 environment,” *Cognitive Neurodynamics*, 2022.

[20] O. Friha, M. A. Ferrag, L. Shu, L. Maglaras, K. -K. R. Choo *et al.,* “FELIDS: Federated learning-based intrusion detection system for agricultural internet of things,” *Journal of Parallel and Distributed Computing*, vol. 165, pp. 17–31, 2022.

[21] N. Neggaz, E. H. Houssein and K. Hussain, “An efficient henry gas solubility optimization for feature selection,” *Expert Systems with Applications*, vol. 152, pp. 113364, 2020.

[22] A. Shoeibi, N. Ghassemi, M. Khodatars, P. Moridian, R. Alizadehsani *et al.,* “Detection of epileptic seizures on EEG signals using ANFIS classifier, autoencoders and fuzzy entropies,” *Biomedical Signal Processing and Control*, vol. 73, pp. 103417, 2022.

[23] D. K. Jain, Y. Li, M. J. Er, Q. Xin, D. Gupta *et al.,* “Enabling unmanned aerial vehicle borne secure communication with classification framework for industry 5.0,” *IEEE Transactions on Industrial Informatics*, vol. 18, no. 8, pp. 5477–5484, 2022.

[24] R. Gopi, P. Muthusamy, P. Suresh, C. G. G. S. Kumar, I. V. Pustokhina *et al.,* “Optimal confidential mechanisms in smart city healthcare,” *Computers, Materials & Continua*, vol. 70, no. 3, pp. 4883–4896, 2022.

[25] G. N. Nguyen, N. H. L. Viet, M. Elhoseny, K. Shankar, B. B. Gupta *et al.,* “Secure blockchain enabled cyber-physical systems in healthcare using deep belief network with ResNet model,” *Journal of Parallel and Distributed Computing*, vol. 153, pp. 150–160, 2021.

[26] M. A. Tawhid and K. B. Dsouza, “Hybrid binary bat enhanced particle swarm optimization algorithm for solving feature selection problems,” *Applied Computing and Informatics*, vol. 16, no. 1/2, pp. 117–136, 2018.