Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.034095
Article

Tech Science Press

Check for updates

An Efficient Intrusion Detection Framework for Industrial Internet of Things Security

Samah Alshathri[1], Ayman El-Sayed[2], Walid El-Shafai[3,4,*] and Ezz El-Din Hemdan[2]

[1]Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O.Box 84428, Riyadh, 11671, Saudi Arabia
[2]Department of Computer Science and Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf, 32952, Egypt
[3]Security Engineering Lab, Computer Science Department, Prince Sultan University, Riyadh, 11586, Saudi Arabia
[4]Department of Electronics and Electrical Communications Engineering, Faculty of Electronic Engineering, Menoufia University, Menouf, 32952, Egypt
*Corresponding Author: Walid El-Shafai. Email: walid.elshafai@el-eng.menofia.edu.eg
Received: 07 July 2022; Accepted: 07 November 2022

Abstract: Recently, the Internet of Things (IoT) has been used in various applications such as manufacturing, transportation, agriculture, and healthcare that can enhance efficiency and productivity via an intelligent management console remotely. With the increased use of Industrial IoT (IIoT) applications, the risk of brutal cyber-attacks also increased. This leads researchers worldwide to work on developing effective Intrusion Detection Systems (IDS) for IoT infrastructure against any malicious activities. Therefore, this paper provides effective IDS to detect and classify unpredicted and unpredictable severe attacks in contradiction to the IoT infrastructure. A comprehensive evaluation examined on a new available benchmark TON_IoT dataset is introduced. The data-driven IoT/IIoT dataset incorporates a label feature indicating classes of normal and attack-targeting IoT/IIoT applications. Correspondingly, this data involves IoT/IIoT services-based telemetry data that involves operating systems logs and IoT-based traffic networks collected from a realistic medium-scale IoT network. This is to classify and recognize the intrusion activity and provide the intrusion detection objectives in IoT environments in an efficient fashion. Therefore, several machine learning algorithms such as Logistic Regression (LR), Linear Discriminant Analysis (LDA), K-Nearest Neighbors (KNN), Gaussian Naive Bayes (NB), Classification and Regression Tree (CART), Random Forest (RF), and AdaBoost (AB) are used for the detection intent on thirteen different intrusion datasets. Several performance metrics like accuracy, precision, recall, and F1-score are used to estimate the proposed framework. The experimental results show that the CART surpasses the other algorithms with the highest accuracy values like 0.97, 1.00, 0.99, 0.99, 1.00, 1.00, and 1.00 for effectively detecting the intrusion activities on the IoT/IIoT infrastructure on most of the employed datasets. In addition, the proposed work accomplishes high performance compared to other recent related works in terms of different security and detection evaluation parameters.

Keywords: Attacks; intrusion detection; machine learning; deep learning; industrial IoT; TON_IoT dataset

1 Introduction

In recent years, superlative modern systems like the Internet of Things (IoT) [1] and cloud computing generally depend on network and Internet connectivity for data sharing. Therefore, cybersecurity turns out to be one of the key arenas for plentiful researchers around the world in diverse subjects such as cloud and IoT forensics [2], big healthcare data security [3], data hiding [4], and critical IoT infrastructure Security [5].

The IoT has recently rewarded interest in various real-world applications such as healthcare, manufacturing, and agriculture. The IoT is a connected network of several types of systems and technology involving cloud computing, intelligent sensors, the Internet, and many other modern systems. As a result, the IoT becomes more exposed to severe incidents and malicious activities that can cause breaches of IoT security and privacy. These attacks can be from both inside and outside of enterprise IoT-based infrastructure. The typical IoT system consists of several levels, as shown in Fig. 1 as the following:

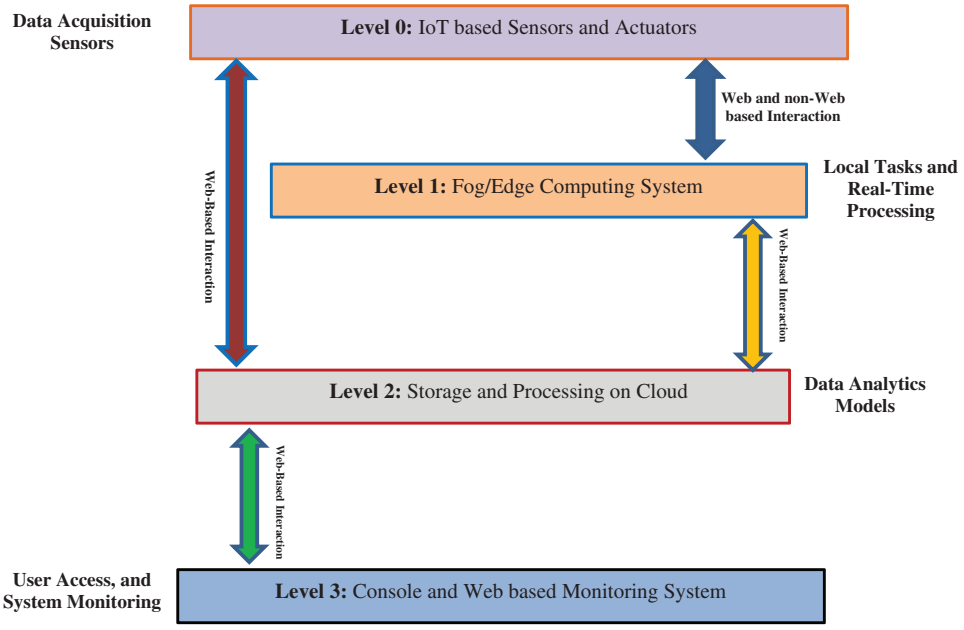

Figure 1: Typical IoT system

- **Level 0:** IoT sensors are used for collecting and observing IoT infrastructure.
- **Level 1:** The edge devices in this level are operated among the sensor network and cloud system for effective data processing near the IoT sensors.
- **Level 2:** The cloud system is employed for storing and processing the collected data from level 0.
- **Level 3:** Console and web-based monitoring applications that communicate with cloud systems for visualizing and presenting obtained results from applying data analytics models to help make suitable decisions by staff in smart IoT systems.

Intrusion Detection Systems (IDSs) are commonly used as a backline of defense to supervise and examine network events to detect possible nasty actions that profitably evade security perimeters, such as firewalls. To evaluate the performance of intrusion detection systems, it is necessary to use IoT-based

datasets that reflect real-world IoT scenarios. Consequently, it can provide an efficient evaluation for developing an efficient intrusion system for IoT applications. In the last years, many machine learning (ML) approaches have been commonly used for detecting and classifying malicious attacks against network infrastructure in an automatic fashion. However, a variety of challenges arise due to malicious attacks that are recurrently changing. In addition, numerous malware datasets are widely available for further cybersecurity study and research.

However, few present studies have indicated the detailed performance analysis of several machine learning algorithms on a dataset for a realistic representation of medium-scale IoT infrastructure. Therefore, this work focuses on classifying and recognizing the intrusion and malicious activity based on several machine learning algorithms using the new open-source TON_IoT dataset [6]. The contribution of this paper is summarized as the following:

- Review of the perception of intrusion detection methods on IoT applications.
- Proposes an intelligent intrusion detection framework for detecting malicious actions in the industrial IoT environment.
- The system is applied to different types of data, such as IoT/IIoT, based telemetry data involving operating system logs and IoT-based traffic networks collected from a realistic medium-scale IoT network.
- The techniques in the proposed framework were carefully chosen based on their broad use in the security realm, as they have verified a good performance in the design of intrusion detection systems.
- Conducting different experiments and analyses for metrics such as accuracy, F1 score, precision, and recall, along with the Receiver Operating Characteristic (ROC) curve, to evaluate the effectiveness of the proposed work for efficiently detecting intrusion trials.
- Performing a detailed comparative analysis to compare the security and detection performance of the proposed IDS and the other related IDSs.

The rest of this paper is organized as follows. First, Section 2 presents the existing work regarding intrusion detection systems, while Section 3 provides candidate machine learning methods. Then, Section 4 defines the proposed intrusion detection system in IoT infrastructure, while the experimental environment with results assessment is given in Section 5. Finally, the conclusion and future scope of this manuscript are introduced in Section 6.

2 Related Work

Recently, the security of the IoT has become perilous; therefore, different studies are related to this area [7–19]. Several works have been proposed for detecting intrusion using intelligent approaches like classification and detection [20–30].

In [7], they used classification methods on the water data where they accomplished different scenarios on the selected features in the water dataset that were gathered from real critical infrastructure with calculated only the accuracy of the classification methods. In [8], they presented a review on detecting Cyber-Physical Systems (CPS) intrusion actions. Their work classified the CPSs into two types based on the potential detection system. In [9,10], they presented a security analysis of a critical cyber-physical system. This assessment comprises various layers like supervisory and control networks, where they presented a methodology of grey-box penetration testing to penetrate an attack on the target system to perform as an intrusion detection system (IDS).

In [11], they developed an Intrusion detection system using Naive Bayesian networks. In this work, a class of a connection is represented by a root node, while the leaf nodes present features of a connection.

In [12], the authors proposed a network-based IDS using a genetic algorithm to recognize anomalous behavior. In [13], they proposed host-based IDS using an ensemble approach and language modeling to decrease the false alarm rates. In [14], they presented a dataset to recognize Denial-of-Service (DoS) attacks in an IoT system. Their system classified the traffic into two types: normal and several DoS attacks.

In [15], they proposed a novel type of anomaly intrusion detection algorithm for unlabeled data clustering to detect new-found intrusions. Reference [16] presented an effective method based on hybrid classifiers to classify the data with high detection and low false alarm rates. In [17], they introduced an optimal feature selection algorithm to help in detecting network intrusion. In [18], they provided an online attack detection model to collect evidence related to the attack that can be used in computer forensics. In [19], they presented an intrusion detection approach using 10% of the knowledge discovery and data mining (KDD) cup'99 dataset to compare the attack types and the protocol used by the attackers.

3 Machine Learning Approaches

In the last years, several ML techniques have been used for detecting intrusion activities [31]:

- **Gaussian Naive Bayes (NB):** The NB uses probability to categorize the features, where it uses normal probability distributions and presumes that the data are normally distributed. This method is used for efficiently performing the classification process.
- **Linear Discriminant Analysis (LDA):** To reduce the dimensions of a provided classification job, the LDA is used by focusing on maximizing the separability among identified types.
- **K-Nearest Neighbors (KNN):** The KNN is used for the regression and classification. The KNN does not have a training phase like the other machine learning algorithms. Instead, the key idea of KNN is to detect the K number of neighbors, and various predefined classes assign a class to the unspecified point.
- **Random Forest (RF):** The RF is used for combining multiple decision trees that use arbitrarily picked data points as their input, so it is known as ensemble learning. It classifies the data according to the results of a decision tree collection. The last result of the classification process can be decided by majority or weighted voting.
- **Classification and Regression Tree (CART):** A decision tree is defined as a structure in which every node indicates a test feature, every branch indicates a test result, and every leaf or node contains a class name. The CART can be used with both numerical and categorical data.

4 Proposed System

The projected intelligent intrusion detection system for malicious activities analysis and classification for IoT structure is shown in Fig. 2. The proposed framework is accomplished in the following phases:

- **Phase 1:** The data is gathered from the IoT/IIoT testbed (TON_IoT dataset) that contains IoT/IIoT services-based Telemetry data that involves logs of Operating Systems and IoT-based traffic network logs collected from a realistic medium-scale IoT network.
- **Phase 2:** The TON_IoT dataset is used for applying the experimental study for intrusion detection and classification in IoT systems.
- **Phase 3:** Pre-processing process is performed on the datasets, and the features (date, time, timestamp, and type) in IoT datasets were removed from feature vectors as they may cause some machine learning methods to over-fit the training data.
- **Phase 4:** The data has been split into training and testing data, where the training data has 80% while the testing data has 20% of the entire dataset.

- **Phase 5:** Categorizing the accumulated data and splitting it into normal or attack.
- **Phase 6:** Customary performance metrics such as accuracy, precision, recall, and F1-score are used to evaluate the methods employed in the proposed framework.
- **Phase 7:** Demonstrating the accomplished results to decide if there is any malicious action from criminals appearing in the collected data in the IoT environment.
- **Phase 8:** Present a comparative study evaluation and analysis amongst different scenarios using various selected ML algorithms for binary classification for normal or intrusion activity.

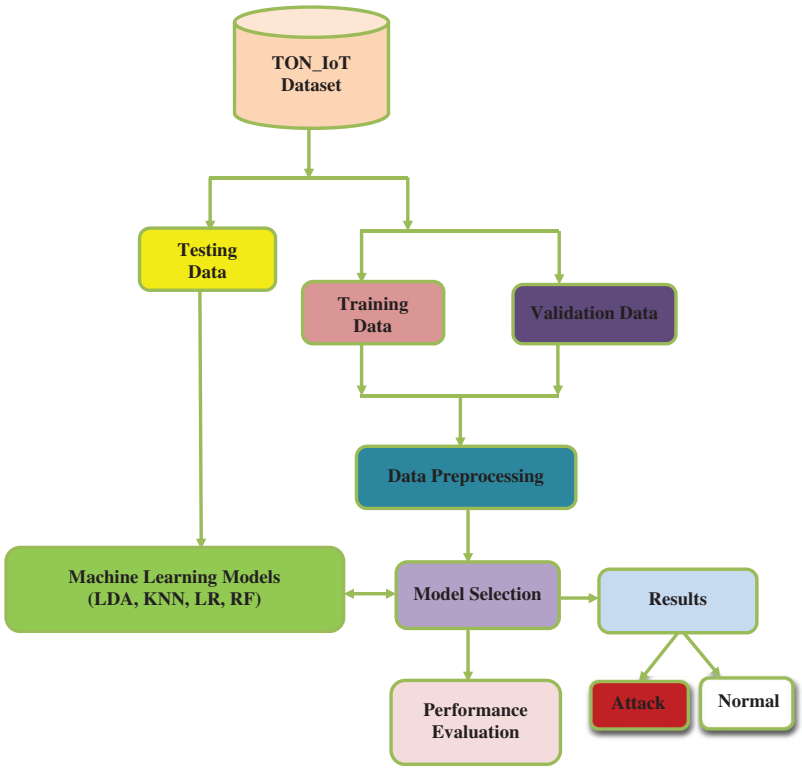

Figure 2: Proposed IoT-based intrusion detection system

5 Experimental Study and Results Analysis

This division delivers the evaluation and analysis of experimental results of the planned intrusion system on the Internet of Things.

5.1 Datasets and Experimental Environment

The planned model estimates the experimental IoT/IIoT data. Then, the TON_IoT datasets are applied within the proposed system for intrusion detection objectives. Finally, the comprehensive information of these datasets, like the name of attributes and their numbers for training and testing, is highlighted in Table 1. The test scenarios are executed using machine learning algorithms written in Python running on Windows 8.1 with Intel® Core™ i5-4288U CPU @ 2.60 GHz processor and 12.00 GB RAM.

Table 1: Datasets description

Datasets	Attributes names	Number of attributes
Dataset 1 (Train_Test_IoT_Fridge)	ts, date, time, fridge_temperature, temp_condition, label	6
Dataset 2 (Train_Test_IoT_Garage_Door)	ts, date, time, door_state, sphone_signal, label	6
Dataset 3 (Train_Test_IoT_GPS_Tracker)	ts, date, time, latitude, longitude, label	6
Dataset 4 (Train_Test_IoT_Modbus)	ts, date, time, FC1_Read_Input_Register, FC2_Read_Discrete_Value, FC3_Read_Holding_Register, FC4_Read_Coil, label	8
Dataset 5 (Train_Test_IoT_Motion_Light)	ts, date, time, motion_status, light_status, label	6
Dataset 6 (Train_Test_IoT_Thermostat)	ts, date, time, current_temperature, thermostat_status, label	6
Dataset 7 (Train_Test_IoT_Weather)	ts, date, time, temperature, pressure, humidity, label	7
Dataset 8 (Train_Test_Linux_disk)	ts, PID, RDDSK, WRDSK, WCANCL, DSK, CMD, label	8
Dataset 9 (Train_Test_Linux_memory)	ts, PID, MINFLT, MAJFLT, VSTEXT, VSIZE, RSIZE, VGROW, RGROW, MEM, CMD, label	12
Dataset 10 (train_Test_Linux_process)	ts, PID, TRUN, TSLPI, TSLPU, POLI, NICE, PRI, RTPR, CPUNR, Status, EXC, State, CPU, CMD, label	16
Dataset 11 (Train_Test_Network)	ts, src_ip, src_port, dst_ip, dst_port, proto, service, duration, src_bytes, dst_bytes, conn_state, missed_bytes, src_pkts, src_ip_bytes, dst_pkts, dst_ip_bytes, dns_query, dns_qclass, dns_qtype, dns_rcode, dns_AA, dns_RD, dns_RA, dns_rejected, ssl_version, ssl_cipher, ssl_resumed, ssl_established, ssl_subject, ssl_issuer, http_trans_depth, http_method, http_uri, http_version, http_request_body_len, http_response_body_len, http_status_code, http_user_agent, http_orig_mime_types, http_resp_mime_types, weird_name, weird_addl, weird_notice, label	44
Dataset 12 (Train_Test_Windows_7)	Processor(_Total) DPC Rate, Processor(_Total) pct_ Idle Time, Processor(_Total) pct_ C3 Time, Memory Pool Paged Resident Bytes, label	134
Dataset 13 (Train_Test_Windows_10)	Processor_DPC_Rate, Processor_pct_ Idle_Time, Processor_pct_ C3_Time, Processor_pct_ Interrupt_Time, Processor_pct_ C2_Time, label	126

5.2 Performance Evaluation Metrics

The performance assessment of the proposed system in a precise classification process is calculated by various metrics like accuracy, precision, recall, and F-score can be used. First, the remarks are accomplished by considering true negatives (TN), true positives (TP), false positives (FP), and false negatives (FN). After assessing the parameters in the confusion matrix as tabulated in Table 2. Then, the evaluation metrics are calculated as in Eqs. (1) to (4):

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

$$\text{F1} - \text{score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

Table 2: Representation of confusion matrix (CM)

	Predicted (−)	Predicted (+)
Actual (−)	TP	FN
Actual (+)	FP	TN

The intrusion detection model in the IoT system, which is consists of four essential phases, as shown in Fig. 3 like the following:

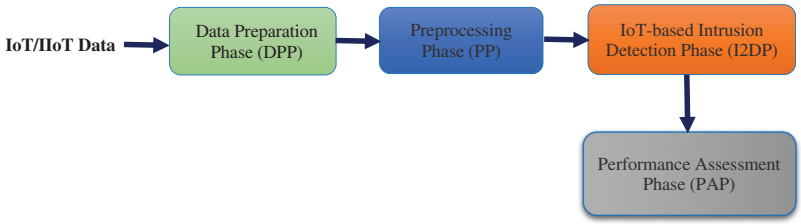

Figure 3: Phases of the proposed intrusion detection model

1) **Data Preparation Phase (DPP):** In this phase, the real dataset from the IoT system was pre-processed to be organized during the training and testing stages. The key procedure achieved in this stage is data handling. Most instances for every situation have been certain from the collected data. The collected data is in the form of a TON_IoT dataset containing IoT/IIoT services-based Telemetry data that includes operating systems logs and IoT traffic network gathered from a realistic medium-scale IoT network. The data are split into two cliques; the first is 80%, and the second is 20% for training and testing.
2) **Preprocessing Phase (PP):** This phase includes the pre-processing process that is accomplished on the data with selected features like date, time, and timestamp, type in the IoT datasets which are

uninvolved from feature vectors as they may cause some machine learning algorithms to over-fit the training data.

3) **IoT-based Intrusion Detection Phase (I2DP):** This phase concerns classifying the collected data and splitting it into normal or attacks for the used datasets. Then, utilizing the obtained results to decide if any intrusion activity from attackers can happen in the IoT infrastructure.
4) **Performance Assessment Phase (PAP):** This phase uses common evaluation metrics like accuracy, precision, recall, and F1-score to estimate the candidate approaches within the proposed system. Finally, effectively perform a comparison analysis between particular circumstances using nominated machine learning methods for normal or intrusion actions classification.

The projected intrusion detection approach can be described in a comprehensive Framework for smart IoT/IIoT infrastructure, as shown in Fig. 4, which includes three main stages as follows:

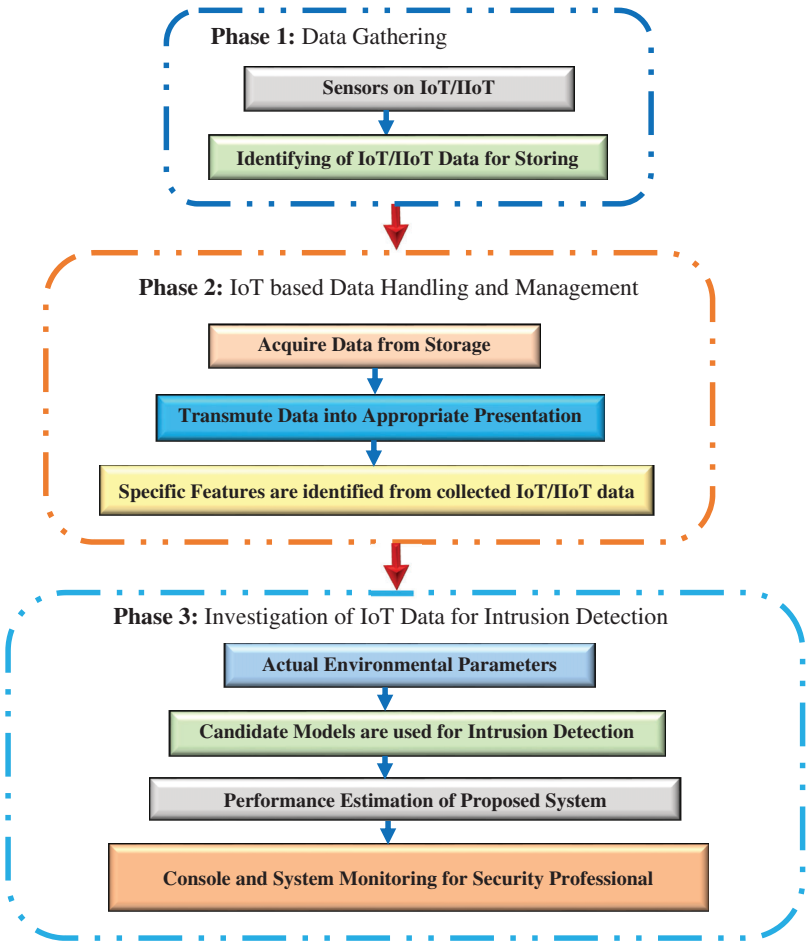

Figure 4: Comprehensive intrusion detection framework in smart IoT/IIoT infrastructure

- **Stage One:** Data Acquisition
  - Data gathering from IoT/IIoT sensors in a smart environment like the smart city.
  - After collecting all data, it is used to create an IoT/IIoT dataset.
- **Stage Two:** Data Handling and Management
  - Pre-processing involves data handling, like selecting specific features to be suitable for the analytical engines.
  - The data has been split into two sets; one set has 80% for training, and another set has 20% for testing.
- **Stage Three:** IoT Data Classification for Intrusion Detection
  - Classifying the gathered data and grouping them into normal or intrusion classes.
  - Evaluate classification models using assessment metrics with the comparative analysis of several ML approaches for detecting intrusion actions in an IoT environment.
  - Visualizing the results on the management console to make an alarm if any intrusion appeared in the gathered data. Then, finally, the security professional can decide and take the appropriate action.

5.3 Results Analysis

This part discusses the performance of the candidate machine learning methods for intrusion detection objectives using IoT datasets. To evaluate the performance of several machine learning approaches on the TON_IoT dataset for thirteen different datasets, the next several test cases were studied:

- Classifying the seven IoT sensors linking records as either normal or attack.
- Classifying the network connection record as either normal or malicious activity (i.e., attack).
- Classifying the Linux Operating system data from three different scenarios Linux OS disk, Memory, and process records as either normal or malicious activity (i.e., attack).
- Classifying the Windows operating system data from three different scenarios, Windows7 OS, and Windows10 OS records as either normal or malicious activity (i.e., attack).

Publicly accessible TON_IoT datasets are used to assess the performance of candidate machine-learning algorithms to identify a baseline system. These datasets include the IoT/IIoT services-based Telemetry data that involves logs of Operating Systems and IoT-based traffic networks collected from a realistic medium-scale IoT network. The total number of datasets is thirteen that are separated into train and test datasets, with applying the necessary pre-processing mechanism as normalization to be ready to use by the machine learning models such as LR, LDA, KNN, NB, CART, RF, and AB for classification and detection of intrusion activities in an efficient way. First, train and test datasets are used to train and test the models. Then, metrics such as accuracy, precision, recall, F1-score, and ROC curve are used to assess the proposed framework.

To understand more interpretation for the obtained results, a comparative analysis among the models on the used datasets using each metric individually is presented in Tables 3–6. These tables show the machine learning models' results on the thirteen examined datasets. The results represent the comparison among the used models based on accuracy, precision, recall, and F1-score, respectively. For dataset 1, the KNN outperforms the other models, while the NB gives fewer results based on all metrics. For dataset 2, all models give the same results, which are 100% for all metrics. For dataset 3, the RF outperforms the other models, while the NB gives fewer results based on all metrics. For dataset 4, the CART outperforms the other models, while the LR, LDA, and NB give fewer results based on all metrics. For dataset 5, all models give the same results, except the KNN gives fewer results based on all metrics. For dataset 6, all models give the same results except KNN and CART, where the KNN gives fewer results based on

all metrics. For dataset 7, the CART outperforms the other models, while the LR and LDA give fewer results based on all metrics. For dataset 8, the CART outperforms the other models, while the LR gives fewer results based on all metrics. For dataset 9, KNN, CART, AB, and RF models give the same results, which are the highest, while the LR gives fewer results based on all metrics. Like dataset 9, KNN, CART, AB, and RF models give the same results, which are the highest, while the LR gives fewer results based on all metrics for dataset 10. The NB outperforms the other models, while the RF gives fewer results based on all metrics for dataset 11. For dataset 12, The RF outperforms the other models while the NB gives fewer results based on all metrics. Finally, for dataset 13, the AB outperforms the other models, while the KNN gives low results based on all tested metrics.

Table 3: Accuracy results of dataset 1(D1) to dataset 13 (D13)

Algorithm	Accuracy												
	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13
LR	0.81	1.00	0.86	0.68	0.58	0.66	0.61	1.00	0.94	0.61	0.63	0.73	0.62
LDA	0.80	1.00	0.87	0.68	0.58	0.66	0.61	1.00	0.99	0.78	0.64	0.73	0.83
NB	0.50	1.00	0.85	0.68	0.58	0.66	0.69	0.69	0.96	0.96	0.65	0.67	0.73
KNN	0.99	1.00	0.91	0.78	0.42	0.56	0.80	1.00	1.00	1.00	0.38	0.97	0.43
CART	0.97	1.00	0.99	0.99	0.58	0.59	0.88	1.00	1.00	1.00	0.37	0.88	0.88
AB	0.94	1.00	0.90	0.68	0.58	0.66	0.72	1.00	1.00	1.00	0.57	0.88	0.90
RF	0.97	1.00	0.99	0.98	0.58	0.66	0.86	1.00	1.00	1.00	0.36	0.97	0.66

Table 4: Precision results of dataset 1(D1) to dataset 13 (D13)

Algorithm	Precision												
	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13
LR	0.87	1.00	0.86	0.34	0.29	0.33	0.59	0.50	0.96	0.64	0.78	0.66	0.31
LDA	0.84	1.00	0.86	0.34	0.29	0.33	0.59	1.00	0.99	0.85	0.81	0.67	0.86
NB	0.51	1.00	0.84	0.34	0.29	0.33	0.70	0.50	0.95	0.96	0.80	0.60	0.72
KNN	0.99	1.00	0.91	0.75	0.29	0.50	0.80	0.68	1.00	1.00	0.42	0.95	0.51
CART	0.98	1.00	1.00	0.99	0.29	0.51	0.87	1.00	1.00	1.00	0.41	0.85	0.87
AB	0.95	1.00	0.89	0.75	0.29	0.54	0.71	1.00	1.00	1.00	0.51	0.86	0.89
RF	0.98	1.00	0.99	0.99	0.29	0.51	0.86	1.00	1.00	1.00	0.40	0.96	0.75

The ROC curve is a standard metric used to evaluate the system performance where the machine learning model is better if the AUC is higher. For example, Fig. 5 shows the ROC results for all models on the used dataset1. In addition, the confusion matrix shows the correct and incorrect classification percentages for all examined ML models. For the simple presentation of results, the obtained confusion matrices of all employed ML models for the tested dataset 12, as shown in Fig. 6. It is clear for the attained results that the utilized ML models introduce high classification ratios and low misclassification ratios.

Table 5: Recall results of dataset 1(D1) to dataset 13 (D13)

Algorithm	Recall												
	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13
LR	0.78	1.00	0.87	0.50	0.50	0.50	0.54	0.50	0.92	0.60	0.50	0.55	0.50
LDA	0.77	1.00	0.88	0.50	0.50	0.50	0.54	0.88	0.99	0.78	0.51	0.55	0.79
NB	0.51	1.00	0.85	0.50	0.50	0.50	0.65	0.84	0.95	0.96	0.52	0.59	0.70
KNN	0.99	1.00	0.91	0.73	0.50	0.50	0.79	0.55	1.00	1.00	0.43	0.97	0.51
CART	0.97	1.00	0.99	0.98	0.50	0.51	0.87	1.00	1.00	1.00	0.43	0.87	0.87
AB	0.93	1.00	0.90	0.50	0.50	0.50	0.69	0.99	1.00	1.00	0.51	0.84	0.89
RF	0.97	1.00	0.99	0.97	0.50	0.50	0.84	0.99	1.00	1.00	0.42	0.97	0.72

Table 6: F1-Score results of dataset 1(D1) to dataset 13 (D13)

Algorithm	F1-score												
	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13
LR	0.79	1.00	0.86	0.41	0.37	0.40	0.49	0.50	0.93	0.58	0.40	0.53	0.38
LDA	0.77	1.00	0.86	0.41	0.37	0.40	0.50	0.93	0.99	0.77	0.42	0.52	0.80
NB	0.50	1.00	0.85	0.41	0.37	0.40	0.65	0.42	0.95	0.96	0.43	0.59	0.71
KNN	0.99	1.00	0.91	0.74	0.37	0.50	0.79	0.58	1.00	1.00	0.38	0.96	0.41
CART	0.97	1.00	0.99	0.99	0.37	0.50	0.87	1.00	1.00	1.00	0.37	0.86	0.88
AB	0.94	1.00	0.89	0.42	0.37	0.40	0.70	1.00	1.00	1.00	0.50	0.85	0.89
RF	0.97	1.00	0.99	0.98	0.37	0.40	0.85	1.00	1.00	1.00	0.35	0.96	0.66

To further clarify the security and detection efficacy of the proposed IDS compared to the other related IDSs, a comparative analysis is performed on the same TON_IoT dataset as given in Table 7. The average values of accuracy, precision, recall, and F1 score are estimated. As a result, it is declared that the suggested IDS framework accomplished high detection and accuracy performance compared to the recent related IDS frameworks in terms of all examined assessment parameters.

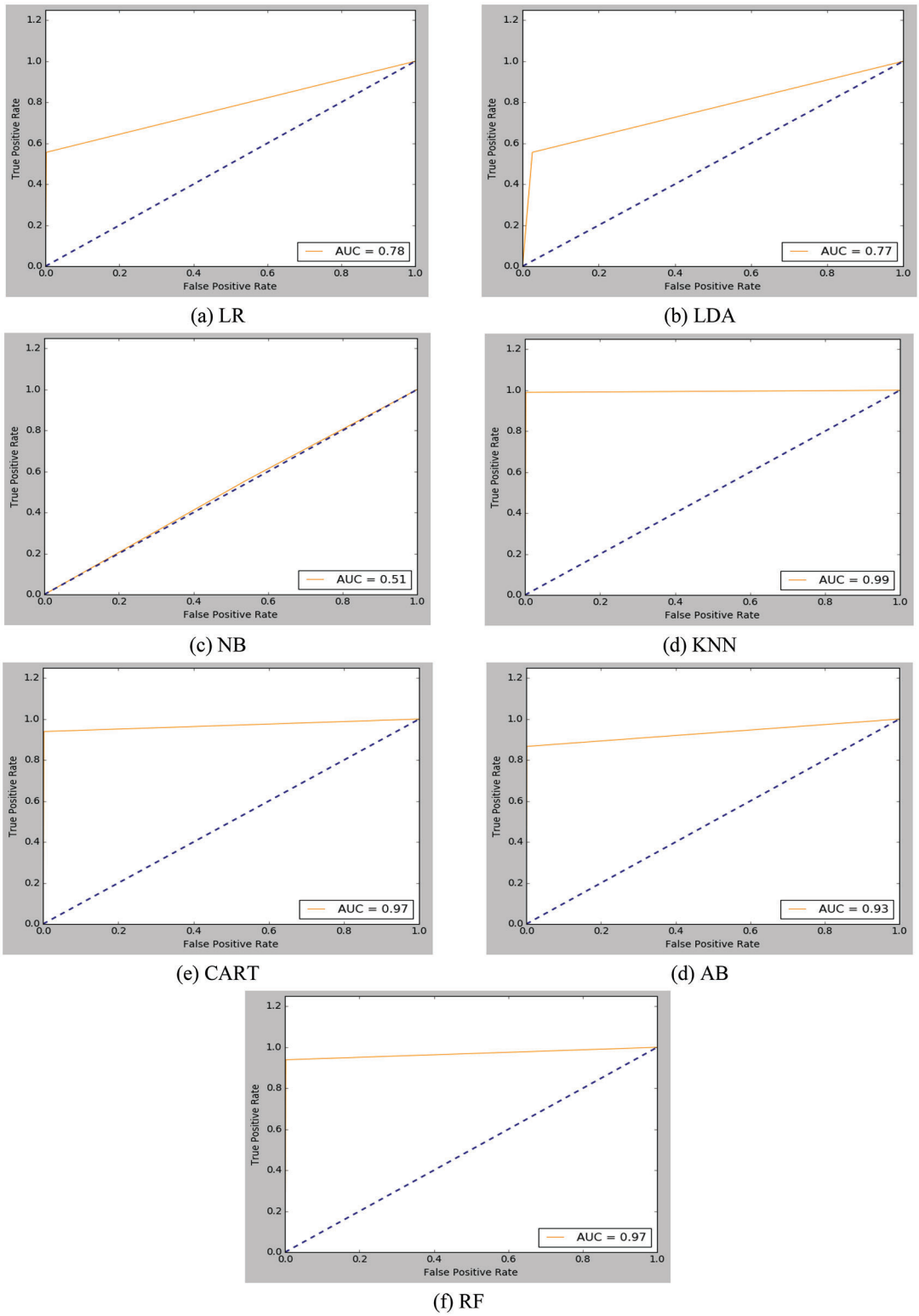

Figure 5: ROC curves of all examined ML models for dataset 1

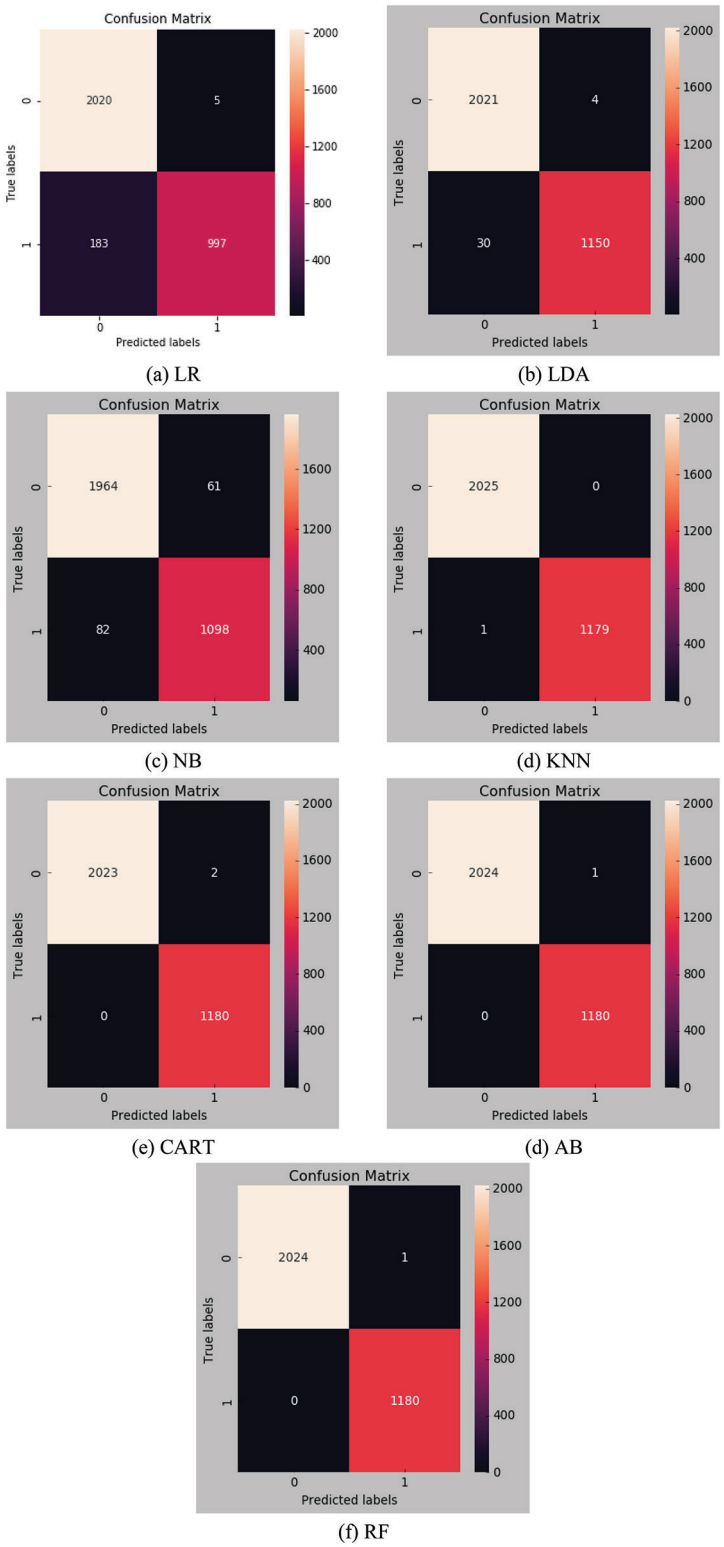

Figure 6: Confusion matrices of all examined ML models for dataset 12

Table 7: Comparative analysis between the proposed IDS and related IDS frameworks

IDS framework	Accuacy (%)	Precision (%)	Recall (%)	F1 score (%)
Proposed	98.4	97.8	95.6	98.2
[16]	79.8	86.4	91.7	93.5
[23]	96.5	96.7	93.5	96.4
[27]	97.7	97.68	95.4	98.14

6 Conclusion and Future Scope

This work aims to present an efficient framework to analyze and classify intrusion and malicious activities in IoT Infrastructure using machine learning methods. This system was comprehensively evaluated based on extensive experiments on thirteen datasets collected from IoT/IIoT testbed. Several machine learning-based approaches, such as LR, NB, CART, LDA, KNN, RF, and AB, are used. The obtained results in this paper urge that the CART algorithm gives the highest scores for classification and detection based on the evaluation metrics such as accuracy, precision, recall, and F1-score. Also, the ROC curve results confirmed the same conclusion for the proposed framework. The performed comparisons clarified that the suggested IDS framework accomplished high detection and accuracy performance compared to the recent related IDS frameworks in terms of all examined assessment parameters. In future work, we plan to apply different deep learning algorithms within the proposed scheme to the same datasets. In addition, we plan to investigate the multiclass classification problem for industrial IoT.

Acknowledgement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R197), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R197), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

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

References

[1] I. Akyildiz and J. Jornet, "The internet of nano-things," *IEEE Wireless Communications*, vol. 17, no. 6, pp. 58–63, 2010.

[2] A. Alarifi, S. Sankar, T. Altameem, K. Jithin, M. Amoon *et al.,* "Novel hybrid cryptosystem for secure streaming of high efficiency H. 265 compressed videos in IoT multimedia applications," *IEEE Access*, vol. 8, pp. 128548–128573, 2020.

[3] Y. Essa, A. El-Mahalawy, G. Attiya and A. El-Sayed, "IFHDS: Intelligent framework for securing healthcare big data," *Journal of Medical Systems*, vol. 43, no. 5, pp. 124–135, 2019.

[4] F. Akyildiz, M. Pierobon, S. Balasubramaniam and Y. Koucheryavy, "The internet of bio-nano things," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 32–40, 2015.

[5] F. Dressler and S. Fischer, "Connecting in-body nano communication with body area networks: Challenges and opportunities of the internet of nano things," *Nano Communication Networks*, vol. 6, no. 2, pp. 29–38, 2015.

[6] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood and A. Anwar, "TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems," *IEEE Access*, vol. 8, pp. 165130–165150, 2020.

[7] H. Hindy, D. Brosset, E. Bayne, A. Seeam and X. Bellekens, "Improving SIEM for critical SCADA water infrastructures using machine learning," *Journal of Bioinformatics*, vol. 11, no. 3, pp. 3–19, 2019.

[8] R. Mitchell and I. Chen, "A survey of intrusion detection techniques for cyber-physical systems," *ACM Computational Survey*, vol. 46, no. 4, pp. 55–71, 2014.

[9] S. Amin, X. Litrico, S. Sastry and A. Bayen, "Cyber security of water SCADA systems part II: Attack detection using enhanced hydrodynamic models," *IEEE Transactions Control Systems Technolgy*, vol. 21, no. 5, pp. 1679–1693, 2012.

[10] S. Amin, X. Litrico, S. Sastry and A. Bayen, "Cyber security of water SCADA systems part I: Analysis and experimentation of stealthy deception attacks," *IEEE Transactions Control Systems Technolgy*, vol. 21, no. 5, pp. 1963–1970, 2012.

[11] W. El-Shafai, S. El-Rabaie, M. El-Halawany and F. Abd El-Samie, "Security of 3D-HEVC transmission based on fusion and watermarking techniques," *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 27211–27244, 2019.

[12] W. El-Shafai, S. El-Rabaie, M. El-Halawany and F. Abd El-Samie, "Efficient hybrid watermarking schemes for robust and secure 3D-MVC communication," *International Journal of Communication Systems*, vol. 31, no. 4, pp. 1–23, 2018.

[13] W. El-Shafai, F. Mohamed, H. Elkamchouchi, M. Abd-Elnaby and A. ElShafee, "Efficient and secure cancelable biometric authentication framework based on genetic encryption algorithm," *IEEE Access*, vol. 9, pp. 1–25, 2021.

[14] W. El-Shafai, I. Almomani and A. Alkhayer, "Optical bit-plane-based 3D-JST cryptography algorithm with cascaded 2D-FrFT encryption for efficient and secure HEVC communication," *IEEE Access*, vol. 9, pp. 35004–35026, 2021.

[15] N. El-Hag, A. Sedik, F. El-Samie, H. El-Hoseny, A. Khalaf *et al.,* "Classification of retinal images based on convolutional neural network," *Microscopy Research and Technique*, vol. 84, no. 3, pp. 394–414, 2021.

[16] M. Panda, A. Abraham and M. Patra, "A hybrid intelligent approach for network intrusion detection," *Procedia Engineering*, vol. 30, no. 5, pp. 1–9, 2012.

[17] S. Kang and K. Kim, "A feature selection approach to find optimal feature subsets for the network intrusion detection system," *Cluster Computing*, vol. 19, no. 1, pp. 325–333, 2016.

[18] N. Soliman, S. Abd-Alhalem, S. Abdulrahman and F. Abd El-Samie, "An improved convolutional neural network model for DNA classification," *Computers, Materials and Continua*, vol. 70, no. 3, pp. 5907–5927, 2022.

[19] M. Siddiqui and S. Naahid, "Analysis of KDD CUP 99 dataset using clustering-based data mining," *International Journal of Database Theory and Application*, vol. 6, no. 5, pp. 23–34, 2013.

[20] O. Faragallah, M. AlZain, H. El-Sayed, J. Al-Amri, F. El-Samie *et al.,* "Secure color image cryptosystem based on chaotic logistic in the FrFT domain," *Multimedia Tools and Applications*, vol. 79, no. 3, pp. 2495–2519, 2020.

[21] M. Nasir, S. Khan, S. Mehmood, M. Khan, M. Zubair *et al.,* "Network meddling detection using machine learning empowered with blockchain technology," *Sensors*, vol. 22, no. 18, pp. 1–22, 2022.

[22] H. El-Hoseny, W. Abd El-Rahman, F. El-Samie, G. El-Banby, E. El-Rabaie *et al.,* "Efficient multi-scale non-sub-sampled shearlet fusion system based on modified central force optimization and contrast enhancement," *Infrared Physics & Technology*, vol. 10, no. 2, pp. 102–123, 2019.

[23] T. Le, Y. Oktian and H. Kim, "XGBoost for imbalanced multiclass classification-based industrial internet of things intrusion detection systems," *Sustainability*, vol. 14, no. 14, pp. 87–105, 2022.

[24] G. Alshammri, A. Samha, E. Hemdan, M. Amoon and W. El-Shafai, "An efficient intrusion detection framework in software-defined networking for cybersecurity applications," *CMC-Computers Materials & Continua*, vol. 72, no. 2, pp. 3529–3548, 2022.

[25] R. AbuKhurma, I. Almomani and I. Aljarah, "IoT botnet detection using salp swarm and ant lion hybrid optimization model," *Symmetry*, vol. 13, no. 8, pp. 13–77, 2021.

[26] A. Dahou, M. Abd Elaziz, S. Chelloug, M. Awadallah, M. Al-Betar *et al.,* "Intrusion detection system for IoT based on deep learning and modified reptile search algorithm," *Computational Intelligence and Neuroscience*, vol. 2, no. 3, pp. 1–17, 2022.

[27] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustainable Cities and Society*, vol. 7, no. 2, pp. 1–13, 2021.

[28] R. Qaddoura, A. M. Al-Zoubi, H. Faris and I. Almomani, "A multi-layer classification approach for intrusion detection in IoT networks based on deep learning," *Sensors*, vol. 21, no. 9, pp. 29–87, 2021.

[29] A. Dina and D. Manivannan, "Intrusion detection based on machine learning techniques in computer networks," *Internet of Things*, vol. 1, no. 6, pp. 100–117, 2021.

[30] H. El-Hoseny, W. El-Rahman, F. Abd El-Samie, S. M. El-Rabaie, K. Mahmoud *et al.,* "Optimal multi-scale geometric fusion based on non-subsampled contourlet transform and modified central force optimization," *International Journal of Imaging Systems and Technology*, vol. 29, no. 1, pp. 4–18, 2019.

[31] K. Randhawa, C. Loo and A. Nandi, "Credit card fraud detection using AdaBoost and majority voting," *IEEE Access*, vol. 6, pp. 14277–14284, 2018.