Lightweight Intrusion Detection for IoT Systems Using Artificial Neural Networks

Radhwan A. A. Saleh[1], Louai Al-Awami[1,2], Mustafa Ghaleb[2], and Anas A. Abudaqa[2(✉)]

[1] Computer Engineering Department, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran 31261, Saudi Arabia
louai@kfupm.edu.sa

[2] Interdisciplinary Research Center for Intelligent Secure Systems, KFUPM, Dhahran 31261, Saudi Arabia
{Mustafa.ghaleb,anas.abudaqa}@kfupm.edu.sa

Abstract. Internet of Things (IoT) systems, due to their vulnerability to a plethora of security attacks, suffer significant detrimental impacts on their reliability. Additionally, the inherent constraints in IoT devices necessitate the incorporation of lightweight security schemes endowed with the capacity to identify intrusions and serve as a robust line of defense. This study presents a lightweight Intrusion Detection System (IDS) that leverages the power of Artificial Neural Network (ANN) while addressing the challenges within IoT systems. The effectiveness of the proposed IDS has been validated through empirical testing by utilizing the ToN IoT Telemetry dataset. The proposed IDS exhibits superior performance compared to other Machine Learning-based IDS solutions identified in the literature, particularly in terms of binary classification where the feature set is continuous. The proposed IDS consistently exceeds expected benchmarks in multi-class classification, with metrics including recall, precision, accuracy, and F-score ranging between 91% and 100%. More importantly, its high accuracy and low time complexity make it an ideal choice for real-time applications, offering a superior alternative to existing AI-based IDS.

Keywords: Intrusion Detection System · Lightweight · Internet of Things · Machine Learning · Artificial Neural Network

1 Introduction

Statistical forecasts predict that the proliferation of IoT devices will escalate to 75.44 billion by 2025 [10]. This exponential increase, paired with the diverse applications of IoT devices, has consequently amplified the security vulnerabilities of Internet of Services [8]. IoT systems, often interconnected with critical infrastructures, present numerous security concerns. For instance, false alarms in household appliances can expose system resources to high-risk vulnerabilities, compromising both user and system security [9]. Experiments conducted with

H. Duan et al. (Eds.): SecureComm 2023, LNICST 568, pp. 45–59, 2025.
https://doi.org/10.1007/978-3-031-64954-7_3

smart devices as detailed in [17], further highlight the ease with which security and privacy of IoT devices can be breached, primarily due to their limited power and computational capabilities in large-scale deployments. Thus, developing a suitable security solution for IoT systems presents a significant challenge, as it must balance the benefits of IoT devices while adhering to stringent security standards [3,27].

Traditional security measures, such as access control and firewalls, are typically employed to protect IoT devices. These mechanisms focus on data confidentiality and authenticity, regulating access to IoT devices, and implementing security and privacy protocols to bolster user confidence [27]. Despite these safeguards, IoT networks remain susceptible to security breaches [4], necessitating systems capable of detecting intrusions and serving as robust defense lines.

An Intrusion Detection System (IDS), usually integrated within a cloud or fog computing infrastructure, is deployed to detect malicious, abnormal, or suspicious activities, alerting the network administrator to potential attacks [6,30]. Although IDS models are essential security tools in traditional networks due to their monitoring and alerting capabilities, their development for IoT networks remains a prominent research interest [31]. This is largely attributed to the inherent constraints of IoT systems, such as low processing and storage capacities of IDS agent nodes [2], which render intricate algorithm-based security solutions impractical.

Machine Learning (ML)-based algorithms, however, present a promising alternative for IoT device security [29]. With their ability to learn, recognize, and respond to varying types of attacks, ML algorithms provide prospective security mechanisms that can increase the reliability and accessibility of IoT devices [29].

When deploying Artificial Neural Networks (ANNs) for Intrusion Detection Systems (IDSs) in IoT environments, several unique challenges arise compared to traditional applications. First, the heterogeneity of application requirements makes the spectrum of data and flows extremely diverse, which in turn complicates the design and operations of the predictive models. Second, IoT data is characterized by simplicity with little or no meta-data available, thus the number of features to be used in the predictive model can be limited. Further, the data landscape in IoT can vary greatly, both in types and formats, unlike in conventional environments where data is often more standardized; this makes ANN implementation more challenging. Third, IoT networks often include a large number of nodes which impacts the performance and computational requirements of the predictive model. Fourth, due to the fact that IoT devices often lack the computing resources that are readily available in traditional settings, it is more challenging to run resource-intensive ANN algorithms or utilize distributed algorithms such as Federated Learning. Finally, IoT environments are highly dynamic and subject to frequent changes, requiring ANNs to be more adaptive and flexible in their operation to remain effective. These factors present unique challenges to employing ANNs in IoT-based IDSs compared to their use in more traditional stable environments.

1.1 Problem Statement and Motivation

The necessity for effective security systems that can detect and deter intrusions in IoT networks is paramount. However, due to the constrained nature of the majority of IoT devices, employing complex and traditional IDS solutions is not practically efficient. As an alternative, Machine Learning (ML)-based algorithms emerge as promising candidates for IDS for IoT. However, the implementation of ML-based security techniques in IoT devices necessitates careful consideration of the constraints inherent to both the machine learning algorithms and the IoT devices [34]. As a result, there is a pressing need for a lightweight ML approach that respects the specifics of IoT while fulfilling the role of an IDS within the IoT system. The suggested ToN IoT Telemetry dataset [1] consists of three types of data. Sensor data, e.g., humidity changes, defines the data sent by the devices, log data which defines the unprocessed network analyzer pcap data files, and raw data concerning these IoT devices' internal and external behavior [5].

For validating models, heterogeneity is a key aspect of modern IoT intrusion detection datasets. The heterogeneity of the dataset should be reflected as variability in traffic data, attack types, used devices, and network technologies. Obviously, the ToN-IoT datasets combine data from heterogeneous sources, including telemetry data from IoT services, operating system logs, and network traffic from IoT networks [1]. In this research, we focus on proposing lightweight IDS based on ML techniques which can easily be implemented on the devices. Thus, the ToN-IoT Telemetry data is the most appropriate data among the heterogeneous ToN-IoT datasets.

1.2 Related Work

Before we proceed and summarize the ML-based IDS approaches that have been proposed in the literature, we need to dive more into the description of the ToN IoT Telemetry dataset. IoT and Industrial IoT (IIoT) sensors across the IIoT network were subjected to nine different forms of cyberattacks in the ToN IoT Telemetry dataset (Weather, Thermostat, Modbus, Garage Door, Motion Light, Tracker, GPS, and Fridge); more details can be found in [18].

The ToN IoT Telemetry dataset provides two kinds of classification options. The Binary Classification option wherein recorded row is classified into normal or attack record. The other option is the Multi-class Classification option wherein each record has to be classified into its corresponding type of attack or into a normal record. Table 1 shows the statistics of attack and normal data records in each of the Train-Test sets.

Since the ToN IoT Telemetry dataset has been published recently, few researchers have utilized it to train their proposed ML approaches. In [1], eight ML approaches have been suggested. The authors have also combined all the per-device datasets into a single dataset. Then they tested their proposed eight ML approaches and figured out that on both per-device and combined datasets, Random Forest (RF) and Classification and Regression Trees (CART) had the highest score in all criteria metrics. In [13], based on an ensemble voting classifier

Table 1. Statistics of Training and Testing IoT Records (No of Rows)

Attack type	Fridge	GPS Tracker	Motion Light	Garage Door	Modbus	Thermostat	Weather	Total
Password	5000	5000	5000	5000	5000	5000	5000	35000
Scanning	–	550	1775	529	529	61	529	3973
XSS	2042	577	449	1156	577	449	866	6116
DDOS	5000	5000	5000	5000	–	–	5000	25000
Ransomware	2902	2833	2264	2902	–	2264	2865	16030
Injection	5000	5000	5000	5000	5000	5000	5000	35000
Backdoor	5000	5000	5000	5000	5000	5000	5000	35000
Total	24944	23960	24488	24587	16106	17774	24260	156119
Normal	35000	35000	35000	35000	35000	35000	35000	245000

Table 2. Summary of ML-based IDS-based on Telemetry Dataset

Ref.	Year	IDS Models	Dataset	Performance Metrics	Summary
[1]	2020	kNN, RF, NB, LR, CART, LSTM, SVM, LDA	All ToN IoT Telemetry dataset	Accuracy, Recall, Precision, F score, Train Time, Test Time	CART performed best for binary (F-score 0.88) and multiclass classification (F-score 0.75)
[13]	2021	DT-RFkNN-NB, DT-RFNB, DT-RFkNN	All ToN IoT Telemetry dataset	Accuracy, Recall, Precision, F-score	DT-RFNB was best for binary (F-score 0.88) and DT-RFkNN for multiclass (F-score 0.74). The proposed models generally outdid ML-based approaches from reference [1]
[7]	2021	DNN, CNN, RNN, DT, RF, NB, LR	All ToN IoT Telemetry dataset with more samples	Accuracy, Recall, Precision, F-score	CNN was best for binary (F-score 0.99) and multiclass (F-score 0.90). Comparing with ML approaches in references [1] and [13] is not fair due to use of duplicate (not augmented) dataset
[19]	2021	GWO-SVM	Only weather dataset from ToN IoT Telemetry dataset	Accuracy	The proposed model achieves 90.28% accuracy for binary classification. No model is proposed for multiclass classification
[21]	2022	RF, VC, ANN, CNN	All ToN IoT Telemetry dataset	Accuracy, Precision, Recall, F-score	The voting classifier (VC) achieves the highest accuracy of 99.7% for the Thermostat, GPS Tracker, Garage Door, and Modbus datasets, while RF achieves best accuracy, that is, 99.3% on the Weather dataset
[26]	2022	ET	Tuned ToN IoT Telemetry dataset	Accuracy, AUC, DR, FAR, F-score, PT	The binary-class classification results show accuracy of 97.86%, 99.66% and 99.64% for ToN-IoT, NF-ToN-IoT, and NF-ToN-IoT-v2 datasets, respectively

technique three novel ML-based approaches have been suggested. The authors have used the traditional classifiers as base learners that have to vote together to get the final prediction. Although the competitive performance of the suggested ensemble voting classifier performance on the binary classification problem, it is still struggling in the multiclass classification problem.

In [7], three Deep Learning-based (DL) approaches have been used as IDS models for DDoS attack detection. Each approach has been tested in both binary and multiclass problems. Despite the fact that the proposed approaches outperformed ML-based approaches, they are still not good alternatives. This is due to the fact that IoT devices are constrained devices and need lightweight models whereas the suggested DL approaches are not [22]. In [19], the Support Vector Machines (SVM) classifier has been used after using the Grey Wolf Optimizer (GWO) [16] for feature selections. The proposed approach has been only tested on the weather problem and the obtained accuracy was 90.28%. In [21], the authors applied random forest, voting classier, ANN, and 1-D Convolutional Neural Network (CNN) to find the normal and abnormal traffic in IoT. In [26] extra trees (ET) classifier is utilized to evaluate the ToN-IoT, and other three datasets after converting them to NetFlow-based feature set [25]. The considered performance metrics are accuracy, area under the curve (AUC), F1-score, detection rate (DR), false alarm rate (FAR), and time needed to predict a single test sample in microseconds. Table 2 shows a summary of these papers.

1.3 Contribution and Paper Organization

Motivated by the aforementioned statements, we propose a lightweight IDS predicated on ML techniques, designed with considerations to both power-consumption and processing time, to align with the constraints of IoT devices. The selection of artificial neural networks (ANNs) as the basis for the lightweight intrusion detection system (IDS) was motivated by their ability to effectively address the complex issues associated with security in the Internet of Things (IoT) domain, while also ensuring the necessary level of efficiency and adaptability required for practical implementation. ANNs have exhibited outstanding abilities in distinguishing complicated patterns within datasets, making them a highly suitable option for identifying anomalies and intrusions in IoT network traffic [23]. The complex and ever-changing characteristics of IoT data need the use of a dynamic and robust model capable of adjusting to new types of attacks. ANNs possess the capacity to acquire knowledge and adapt to various data distributions, hence providing a flexible solution for handling the wide array of inputs encountered inside IoT networks [28]. The main contributions of this paper are as follows:

- Recognizing the challenge of sensor-by-sensor intrusion detection [12], we propose a high-performance sensor-by-sensor IDS.
- We demonstrate that the proposed IDS is a lightweight system, particularly in terms of power consumption and processing time.

- We compare the proposed IDS with other machine learning methods, highlighting its superior performance in terms of accuracy and time complexity, thereby establishing the merits of our approach.
- The proposed IDS is trained not only to detect the existence of intrusion attacks (binary classification problem), but also to identify the type of intrusion attack (multi-class classification problem).

The subsequent parts of this paper delve deeper into various aspects of this research. In Sect. 2, 'Proposed Model (ANN-based IDS)', the novel Artificial Neural Network-based IDS is presented in detail. Section 3, 'Experiments and Results', discusses the testing and validation of this model, demonstrating its performance against the ToN IoT Telemetry dataset [1]. Finally, Sect. 4, 'Conclusion and Future Direction', summarizes the research findings and outlines potential future research directions.

2 Proposed Model

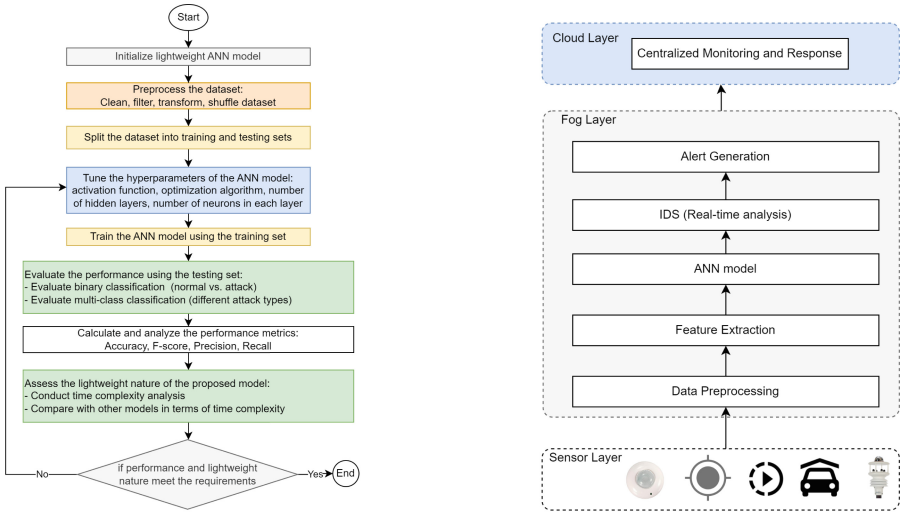
Various Machine Learning techniques have been evaluated using the ToN IoT Telemetry dataset. Yet, the application of a lightweight ANN approach remains unexplored. Thus, this study proposes a lightweight ANN model, tailored to the constraints of IoT devices, particularly concerning power consumption and processing time. To meet these lightweight requisites, minimizing the ANN parameters without compromising the model's performance is imperative, thus necessitating the optimization of ANN parameters for an improved IDS.

The proposed research introduces a lightweight IDS for IoT networks, leveraging ANN techniques. The primary objective of this IDS is to address the identified challenges of network vulnerability and device constraints within the IoT ecosystem. By incorporating machine learning techniques, the proposed model aims to provide a robust line of defense against intrusions while being compatible with IoT devices' limited processing power and storage capacities. For dataset preparation, the ToN IoT Telemetry dataset is utilized following the recommended data cleaning, filtering, and transformation steps. Additionally, shuffling of the dataset is performed to prevent overfitting during training. The proposed ANN-based IDS's performance is evaluated through binary and multi-class classification experiments. The ANN model consists of multiple layers where each layer processes and transfers the inputs to the subsequent layer through weighted connections. The input layer receives the feature data, which remains constant throughout the network. The hidden layers process the received data, applying activation functions to produce outputs. Finally, the output layer provides the results of the intrusion detection process. To ensure the lightweight nature of the proposed IDS, careful optimization of ANN parameters is performed without compromising the model's performance. The following hyperparameters are tuned to achieve an improved intrusion detection system: activation function, weights, number of layers, and number of neurons in each layer.

The first step in the experiments involves tuning the activation function and optimization algorithm. Through empirical testing and analysis, the activation

function 'tanh' and the optimization algorithm 'adam' demonstrate superior performance in terms of accuracy, F-score, precision, and recall. These settings are chosen as the optimal configurations for the proposed model. Next, the number of neurons in the hidden layers is tuned. Starting with a fixed number of three hidden layers, the experiments aim to determine each layer's optimal number of neurons. By evaluating various configurations, the best-performing model is identified based on accuracy, F-score, precision, and recall metrics. The chosen configuration is utilized for further testing and validation.

The proposed model's flow, as depicted in the accompanying flowchart (Fig. 1a), includes dataset preprocessing, hyperparameters tuning, training and evaluation, and iterative refinement for achieving optimal performance and lightweight characteristics.



(a) Flowchart (b) Architecture

Fig. 1. Proposed Model

The architecture of our proposed IDS is designed to operate in a sensor-by-sensor manner, enabling comprehensive monitoring and detection of security threats in IoT networks. It consists of several key components as shown in Fig. 1b, including:

- Sensor Layer: This layer comprises IoT devices equipped with various sensors, such as temperature, and motion sensors. Each sensor collects data from its corresponding environment or device.
- Data Preprocessing: The collected sensor data undergoes preprocessing to clean and normalize it, ensuring accuracy and reliability for subsequent analysis.

- Feature Extraction: Relevant features are extracted from the preprocessed sensor data, capturing distinctive characteristics of attacks and normal behavior patterns.
- ANN-based IDS model: The extracted features are input to the ANN, which serves as the core of our IDS.
- IDS (Real-time analysis): The trained ANN performs real-time analysis on incoming sensor data, flagging deviations from expected behavior as anomalies or potential security threats.
- Alert Generation: Detected anomalies trigger the generation of alerts, providing information about the nature of the anomaly, the affected sensor, and relevant details for further investigation.
- Centralized Monitoring and Response: Alerts are centrally monitored and logged in a centralized system in the cloud, offering comprehensive visibility into the network's security status. Security personnel can analyze alerts, correlate them with other system events, and initiate appropriate response actions to mitigate threats.

This sensor-by-sensor architecture ensures granular monitoring and detection capabilities, enabling effective threat identification and real-time response.

3 Experiments and Results

This section consists of three subsections. The first subsection introduces the experimental setup, highlighting how the minimum optimal number of parameters was achieved. In the second subsection, we present the performance of the proposed model in terms of accuracy, F-score, recall, and precision. Finally, the third subsection compares the time complexity of the proposed model with existing literature to demonstrate its lightweight nature in terms of power consumption and processing time.

3.1 Experiments Setup

In our experiments, we employed the Grid-Search algorithm to fine-tune the hyperparameters of the Fully Connected ANN, including the number of hidden layers and the number of neurons within each layer. Through systematic exploration of different combinations, we discovered that the best structure consists of three hidden layers, each comprising 50 neurons. Afterwards, the remaining hyperparameters, such as the activation function and optimization algorithm, were thoroughly compared to further optimize the performance of the proposed ANN. The results in Table 3 indicate that the optimal combination for our problem is achieved with the 'tanh' activation function and the 'adam' optimization algorithm.

Next, we focused on tuning the number of neurons in the hidden layers while keeping the activation function and optimization algorithm fixed at 'tanh' and 'adam', respectively. By iteratively adjusting the ANN structure, we obtained the best performance with three hidden layers, as demonstrated in Tables 4 and 5. Throughout all experiments, the number of iterations was set to 500.

3.2 Accuracy Evaluation

Table 4 presents the optimal results obtained through binary classification using our proposed ANN model, while Table 5 showcases the optimal results achieved for multi-class classification. These results are evaluated based on four key performance metrics: accuracy, recall, precision, and F-score.

When assessing ML models, the literature commonly utilizes two evaluation methods: hold-out validation and k-fold cross-validation [14,33]. The hold-out method's reliability is influenced by how the data is partitioned into training and testing sets. Additionally, this approach may introduce bias towards either the training or testing set, leading to inaccurate performance estimation. Therefore, for this research, we employed k-fold cross-validation with a value of five ($k = 5$), which aligns with the suggested limit in [33] and is appropriate for our dataset size.

Table 3. Activation Function and Optimization Algorithm Tuning Results

Activation Fun.	Optimization Alg.	Accuracy	F-score	Precision	Recall
tanh	adam	89%	91%	92%	89%
tanh	lbfgs	83%	87%	90%	83%
tanh	sgd	80%	84%	89%	80%
logistic	adam	84%	87%	91%	84%
logistic	lbfgs	59%	74%	100%	59%
logistic	sgd	59%	74%	100%	59%
relu	adam	84%	86%	87%	85%
relu	lbfgs	78%	83%	88%	78%
relu	sgd	80%	82%	79%	86%

Table 4. Binary Classification Results of the Proposed ANN Model

Datasets	Best Model	Accuracy	F-score	Precision	Recall	Time Complexity	# Multiplications
Fridge Sensor	ANN (25, 25, 25)	58%	74%	**100%**	58%	$O(ij + jk + kr + rs)$ [20]	1325
	LSTM [1]	**100%**	**100%**	**100%**	**100%**	$O(4ij + 3(j + k + r) + 4(j^2 + k^2 + r^2) + rs)$ [20]	123884
Garage Door	ANN (25, 25, 25)	59%	74%	**100%**	59%	$O(ij + jk + kr + rs)$ [20]	1325
	LSTM [1]	**100%**	**100%**	**100%**	**100%**	$O(4ij + 3(j + k + r) + 4(j^2 + k^2 + r^2) + rs)$ [20]	123884
GPS Sensor	ANN (50, 25, 100)	**91**	**92%**	**89%**	**95%**	$O(ij + jk + kr + rs)$ [20]	3950
	kNN [1]	88%	88%	**89%**	88%	$O(mn)$ [15]	94336
Modbus	ANN (100, 100, 25)	70%	81.5%	98%	70%	$O(ij + jk + kr + rs)$ [20]	12625
	CART [1]	**98%**	**99%**	**99%**	**98**	$O(mn \log_2 n)$ [24]	98460
Light Motion	ANN (25, 25, 25)	**59%**	**74%**	**100%**	**59%**	$O(ij + jk + kr + rs)$ [20]	1325
	LSTM [1]	**59%**	44%	35%	**59%**	$O(4ij + 3(j + k + r) + 4(j^2 + k^2 + r^2) + rs)$ [20]	123884
Thermostat	ANN (25, 25, 25)	**66%**	**80%**	**100%**	**66%**	$O(ij + jk + kr + rs)$ [20]	1325
	RF [1]	**66%**	62%	59%	**66%**	O(depth of tree* number of trees) [15]	10*depth of tree
Weather	ANN (100, 100, 100)	**91%**	**92%**	**93%**	**91%**	$O(ij + jk + kr + rs)$ [20]	20175
	CART [1]	87%	87%	88%	87%	$O(mn \log_2 n)$ [24]	67858

Table 4 highlights an interesting observation: when at least one feature is discrete, the precision achieved is 100%, indicating that the number of normal records incorrectly classified as attacks is minimized. Consequently, all normal records are correctly identified. However, this improvement in precision comes at the expense of a decreased recall, meaning that the number of attack records incorrectly classified as normal increases. In other words, the system avoids false alarms, but it may fail to detect certain attack records. Hence, the proposed ANN model is not the preferred choice when dealing with datasets containing discrete features. Conversely, when all features are continuous, our proposed ANN model outperforms other ML models, exhibiting superior performance.

Table 5 demonstrates the competitive performance of the proposed IDS in the multi-class classification problem, with measurement metrics ranging from 91% to 100%. Notably, the recall metric consistently hovers around 100%, indicating rare instances of incorrectly detecting attack records as normal records. However, three cases exhibit zero recall (TP = 0): the GPS Sensor dataset with scanning attack, the Weather dataset with scanning attack, and the Thermostat dataset with normal records. These occurrences can be attributed to the limited number of scanning attack records in the first two cases and the imbalanced class distribution in the third case. Therefore, to enhance the accuracy of the multi-class IDS based on the proposed ANN model, it is recommended to employ a balanced dataset.

Additionally, there are cases where precision and F-score are reported as "NAN", denoting both TP and FP values are zero. These cases involve normal records in the Light Motion and Thermostat datasets. However, it should be noted that these two datasets rely on dependent features, resulting in the proposed model being trained with only one feature. This outcome highlights the impracticality of the proposed IDS when trained using a single sensor reading. Indeed, it is logical to deduce and generalize that any ML-based IDS cannot effectively extract specific patterns from a dataset if it is solely based on one sensor reading. Furthermore, as stated by the authors in [32], any performance improvement achieved by a model on one dataset is offset by a decline in performance on another dataset.

3.3 Time Complexity

The time complexity of the proposed ANN-based IDS is calculated using the Big O-Notation, aiming to demonstrate its lightweight nature [11]. Power consumption and processing time are inversely related, meaning that reducing processing time leads to lower power consumption and vice versa. Consequently, minimizing the number of required multiplication operations results in faster data processing and reduced power consumption. In this subsection, we employ the Big O-Notation to calculate the number of multiplication operations in the worst-case scenarios of the proposed ANN-based IDS compared to the ML-based IDS mentioned in Table 4.

The time complexity of the proposed ANN-based IDS is determined in two stages. Firstly, multiplication is performed between the weights and the inputs

Table 5. Multi-class Classification Results of the Proposed ANN Model for Fridge Sensor, Garage Door, GPS Sensor, Modbus, Light Motion, Thermostat, and Weather Activities

Datasets	Attack Type	Accuracy	F-score	Precision	Recall
Fridge	ddos	92%	96%	92%	100%
	backdoor	92%	96%	92%	100%
	injection	92%	96%	92%	100%
	normal	93%	92%	85%	100%
	password	92%	96%	92%	100%
	ransomware	96%	98%	95%	100%
	xss	81%	98%	97%	100%
	Average	93%	96%	92%	100%
Garage	ddos	92%	96%	92%	100%
	backdoor	92%	96%	92%	100%
	injection	92%	96%	92%	100%
	normal	100%	100%	100%	100%
	password	92%	96%	92%	100%
	ransomware	95%	98%	95%	100%
	xss	98%	99%	98%	100%
	scanning	99%	100%	99%	100%
	Average	95%	97%	95%	100%
GPS	ddos	96%	98%	96%	100%
	backdoor	95%	97%	96%	99%
	injection	95%	98%	97%	98%
	normal	96%	94%	95%	94%
	password	93%	96%	94%	99%
	ransomware	97%	98%	97%	99%
	xss	100%	100%	100%	100%
	scanning	0%	0%	0%	0%
	Average	84%	85%	84%	86%
Modbus	injection	93%	96%	94%	98%
	backdoor	92%	95%	93%	98%
	normal	83%	70%	77%	64%
	password	91%	95%	93%	98%
	scanning	99%	99%	99%	100%
	xss	99%	99%	99%	100%
	Average	93%	93%	92%	93%
Motion	ddos	92%	96%	92%	100%
	backdoor	92%	96%	92%	100%
	injection	92%	96%	92%	100%
	normal	59%	nan%	nan%	0%
	password	92%	96%	92%	100%
	ransomware	98%	96%	100%	100%
	scanning	97%	98%	97%	100%
	xss	0%	0%	0%	99%
	Average	77%	–	–	87%
Thermostat	injection	91%	95%	91%	100%
	backdoor	91%	95%	91%	100%
	normal	66%	nan%	nan%	0%
	password	91%	95%	91%	100%
	ransomware	96%	98%	96%	100%
	scanning	100%	100%	100%	100%
	xss	99%	100%	99%	100%
	Average	90%	–	–	86%
Weather	ddos	97%	98%	98%	98%
	backdoor	96%	98%	98%	98%
	injection	98%	99%	99%	99%
	normal	90%	88%	88%	87%
	password	97%	98%	98%	98%
	ransomware	98%	99%	98%	100%
	xss	100%	100%	100%	100%
	scanning	0%	0%	0%	0%
	Average	84%	85%	85%	85%

of each layer. For instance, if we have i neurons in the first layer and j neurons in the second layer, the time complexity is $O(ij)$. Secondly, applying the activation function requires a time complexity of $O(j)$. Thus, the total time complexity for every two layers can be expressed as $O(ij+j) = O(j(i+1)) = O(ij)$. Considering the input layer with i neurons, three hidden layers with j, k, and r neurons respectively, and the output layer with s neurons, the overall time complexity of the proposed ANN-based IDS is given as $O(ij+jk+kr+rs)$ [20]. Consequently, Table 4 presents the time complexity for each of the seven proposed models and the models from the literature, where n represents the number of features and m represents the number of training examples. In conclusion, the proposed ANN-based IDS demonstrates higher efficiency in terms of time complexity compared to the other models listed in Table 4, highlighting its lightweight nature.

The proposed approach outperforms the other models in terms of different measurement metrics such as recall, precision, f-score, accuracy, and time complexity on different case studies such as GPS Sensor, Light Motion, Thermostat, and Weather. The suggested approach demonstrates outstanding efficiency in successfully detecting intrusions in every case study, while simultaneously achieving a notable level of precision and recall. This results in a reduction in both false positives and false negatives, hence enhancing the overall resilience of the intrusion detection system. Additionally, the utilization of the f-score metric, which effectively combines precision and recall, serves to enhance the balanced performance of the suggested approach in different scenarios. The advantages of the suggested methodology extend beyond performance indicators alone. Significantly, the reduced time complexity of the system guarantees rapid and efficient real-time identification, highlighting its practical feasibility in dynamic Internet of Things (IoT) situations. The suggested approach demonstrates its efficacy as an enhanced solution for IoT intrusion detection by surpassing previous models in several dimensions and across a range of scenarios.

4 Conclusion and Future Directions

The IoT ecosystem faces significant security challenges, threatening the reliability of IoT systems. Lightweight security solutions are crucial for resource-constrained IoT devices to detect intrusions effectively. This research conducts a comprehensive analysis of ML-based IDS approaches for edge computing using the ToN IoT Telemetry dataset. We select the ToN IoT Telemetry dataset due to its relevance and unique characteristics. ML approaches commonly used in IoT security systems are discussed, providing a foundation for our analysis. Our findings show that ML-based IDSs hold promise as lightweight security solutions. We propose and extensively test an ANN model on the ToN IoT Telemetry dataset. Results reveal the superiority of our model in binary classification with continuous features. Additionally, the proposed ANN model consistently achieves competitive performance in multi-class classification, with metrics ranging from 91% to 100%. To establish its lightweight nature, we utilize Big O-Notation to calculate the time complexity of our proposed ANN-based IDS.

The analysis confirms its efficiency compared to models in Table 4, highlighting reduced computational requirements and power consumption. In conclusion, this research advances lightweight IDS-ML models for IoT security. The proposed ANN model effectively detects intrusions in resource-constrained IoT environments. Future research directions include exploring diverse datasets, hybridizing optimization strategies, and utilizing alternative algorithms. By addressing these areas, we strengthen IoT security and enhance system resilience. This research has focused on evaluating lightweight IDS-ML models using the ToN IoT Telemetry dataset. However, there are potential avenues for further exploration and improvement. Firstly, it is recommended to test the proposed model on diverse datasets from different domains to assess its generalizability and robustness. This would provide valuable insights into the model's performance across various application scenarios. Secondly, future work can involve hybridizing different optimization strategies to enhance the performance of the proposed ANN model. Combining multiple optimization techniques, such as genetic algorithms, particle swarm optimization, or simulated annealing, may lead to improved convergence and more accurate results. Additionally, exploring alternative optimization algorithms for tuning the hyperparameters of the proposed model can further enhance its effectiveness. Comparative studies on the performance of different optimization algorithms would offer valuable insights into the model's behavior and help identify the most suitable optimization approach for specific datasets and scenarios. Overall, these future directions aim to broaden the scope of the research and advance the development of more efficient and effective IDS-ML models.

Acknowledgment. The authors would like to acknowledge the support provided by King Fahd University of Petroleum and Minerals (KFUPM), the Interdisciplinary Research Center for Intelligent Secure Systems, and the Department of Computer Engineering. Furthermore, the authors would like to thank Dr. Marwan Abu-Amara for his invaluable guidance which significantly elevated the quality of this paper.

References

1. Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A., Anwar, A.: TON_IoT telemetry dataset: a new generation dataset of IoT and IIoT for data-driven intrusion detection systems. IEEE Access **8**, 165130–165150 (2020)
2. Andrea, I., Chrysostomou, C., Hadjichristofi, G.: Internet of things: security vulnerabilities and challenges. In: 2015 IEEE Symposium on Computers and Communication (ISCC), pp. 180–187. IEEE (2015)
3. Azzedin, F., Ghaleb, M.: Internet-of-things and information fusion: trust perspective survey. Sensors **19**(8), 1929 (2019)
4. Benkhelifa, E., Welsh, T., Hamouda, W.: 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 (2018)
5. Booij, T.M., Chiscop, I., Meeuwissen, E., Moustafa, N., den Hartog, F.T.: TON_IoT: the role of heterogeneity and the need for standardization of features and attack types in IoT network intrusion datasets. IEEE Internet Things J. (2021)

6. Duraisamy, A., Subramaniam, M., Robin, C.R.R.: An optimized deep learning based security enhancement and attack detection on IoT using ids and KH-AES for smart cities. Stud. Inf. Control **30**(2), 121–131 (2021)
7. Ferrag, M.A., Shu, L., Djallel, H., Choo, K.K.R.: Deep learning-based intrusion detection for distributed denial of service attack in agriculture 4.0. Electronics **10**(11), 1257 (2021)
8. Ghaleb, M., Azzedin, F.: Towards scalable and efficient architecture for modeling trust in IoT environments. Sensors **21**(9), 2986 (2021)
9. Ghaleb, M., Azzedin, F.: Trust-aware fog-based IoT environments: artificial reasoning approach. Appl. Sci. **13**(6), 3665 (2023)
10. Cvitić, I., Peraković, D., Periša, M., Krstić, M., Gupta, B.: Analysis of IoT concept applications: smart home perspective. In: Perakovic, D., Knapcikova, L. (eds.) FABULOUS 2021. LNICST, vol. 382, pp. 167–180. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-78459-1_12
11. Hidary, J.D.: Complexity theory. In: Hidary, J.D. (ed.) Quantum Computing: An Applied Approach, pp. 43–50. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-83274-2_4
12. Jan, S.U., Ahmed, S., Shakhov, V., Koo, I.: Toward a lightweight intrusion detection system for the internet of things. IEEE Access **7**, 42450–42471 (2019)
13. Khan, M.A., et al.: Voting classifier-based intrusion detection for IoT networks. arXiv preprint arXiv:2104.10015 (2021)
14. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Ijcai, Montreal, Canada, vol. 14, pp. 1137–1145 (1995)
15. Kumar, P.: Computational complexity of ml models (2019). https://medium.com/analytics-vidhya/time-complexity-of-ml-models-4ec39fad2770
16. Mirjalili, S., Mirjalili, S.M., Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. **69**, 46–61 (2014)
17. Notra, S., Siddiqi, M., Gharakheili, H.H., Sivaraman, V., Boreli, R.: An experimental study of security and privacy risks with emerging household appliances. In: 2014 IEEE Conference on Communications and Network Security, pp. 79–84. IEEE (2014)
18. Nour, M.: TON-IoT datasets (2020). https://cloudstor.aarnet.edu.au/plus/s/ds5zW91vdgjEj9i
19. Rahmany, I., Mnassri, H., Moulahi, T., El Khediri, S.: Grey wolf optimizer enhanced SVM for IoT fault detection. In: 2021 International Wireless Communications and Mobile Computing (IWCMC), pp. 1483–1488. IEEE (2021)
20. Raja, S.: FNNs, RNNs, LSTM and BLSTM (2021)
21. Saba, T., Khan, A.R., Sadad, T., Hong, S.P.: Securing the IoT system of smart city against cyber threats using deep learning. Discrete Dyn. Nat. Soc. **2022** (2022)
22. SALEH, N.A., ERTUNÇ, H.M., SALEH, R.A., RASSAM, M.A.: A simple mask detection model based on a multi-layer perception neural network. In: 2021 International Conference of Technology, Science and Administration (ICTSA), pp. 1–5. IEEE (2021)
23. Saleh, R.A., Konyar, M.Z., Kaplan, K., Ertunç, H.M.: Tire defect detection model using machine learning. In: 2022 2nd International Conference on Emerging Smart Technologies and Applications (eSmarTA), pp. 1–5. IEEE (2022)
24. Sani, H.M., Lei, C., Neagu, D.: Computational complexity analysis of decision tree algorithms. In: Bramer, M., Petridis, M. (eds.) SGAI 2018. LNCS (LNAI), vol. 11311, pp. 191–197. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04191-5_17

25. Sarhan, M., Layeghy, S., Moustafa, N., Portmann, M.: NetFlow datasets for machine learning-based network intrusion detection systems. In: Deze, Z., Huang, H., Hou, R., Rho, S., Chilamkurti, N. (eds.) BDTA/WiCON -2020. LNICST, vol. 371, pp. 117–135. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-72802-1_9
26. Sarhan, M., Layeghy, S., Portmann, M.: Towards a standard feature set for network intrusion detection system datasets. Mob. Netw. Appl. 1–14 (2022)
27. Sicari, S., Rizzardi, A., Grieco, L.A., Coen-Porisini, A.: Security, privacy and trust in internet of things: the road ahead. Comput. Netw. **76**, 146–164 (2015)
28. Sumaiya Thaseen, I., Saira Banu, J., Lavanya, K., Rukunuddin Ghalib, M., Abhishek, K.: An integrated intrusion detection system using correlation-based attribute selection and artificial neural network. Trans. Emerg. Telecommun. Technol. **32**(2), e4014 (2021)
29. Tahsien, S.M., Karimipour, H., Spachos, P.: Machine learning based solutions for security of internet of things (IoT): a survey. J. Netw. Comput. Appl. **161**, 102630 (2020)
30. Thakkar, A., Lohiya, R.: Role of swarm and evolutionary algorithms for intrusion detection system: a survey. Swarm Evol. Comput. **53**, 100631 (2020)
31. Thakkar, A., Lohiya, R.: A review on machine learning and deep learning perspectives of ids for IoT: recent updates, security issues, and challenges. Arch. Comput. Methods Eng. **28**(4), 3211–3243 (2021)
32. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Trans. Evol. Comput. **1**(1), 67–82 (1997)
33. Yadav, S., Shukla, S.: Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In: 2016 IEEE 6th International Conference on Advanced Computing (IACC), pp. 78–83. IEEE (2016)
34. Zeadally, S., Tsikerdekis, M.: Securing internet of things (IoT) with machine learning. Int. J. Commun. Syst. **33**(1), e4169 (2020)