Journal of Intelligent Systems and Internet of Things
Vol. 16, No. 01, PP. 176-188, 2025
DOI: https://doi.org/10.54216/JISIoT.160115

A New Descriptor Based on Machine Learning for Intrusion Detection in Wireless Sensor Networks WNSs

Esraa Saleh Alomari[1], Oday Ali Hassen[1, 2, *], Wisam Makki Salim[3], Selvakumar Manickam[4], Nur Azman Abu[5]

[1]Computer Department, College of Education for Pure Sciences, Wasit University, Iraq

[2]Ministry of Education, Wasit Education Directorate. Iraq

[3]College of Dentistry, Al-Iraqia University, Baghdad, Iraq

[4]National Advanced IPv6 Centre (NAv6), Universiti Sains Malaysia, Gelugor 11800, Penang, Malaysia

[5]Department of Information Technology, University Technical Malaysia Melaka, Hang Taya, Melaka 76100, Malaysia

Emails: ealomari@uowasit.edu.iq; oday123456789.oa@gmail.com; wisam.m.salim@aliraqia.edu.iq; selva@nav6.usm.my; nura@utem.edu.my

Abstract

Wireless sensor networks have become a vital component of the infrastructure for many modern applications. With the increasing use of wireless sensor networks, the challenges facing these networks in the field of security are escalating and growing, and with the rapid advancement of wireless communication technology, these networks are exposed to increasing, complex and continuous threats. Our research is characterized by innovation in the field of security technology to enhance protection, repel attacks and detect intrusions, among these innovations are intrusion detection systems based on machine learning as a creative and new solution. In this research, we highlight the effectiveness of different machine learning algorithms, such as supervised and unsupervised learning, in detecting anomalies and intrusions within wireless sensor networks, as our goal focuses on enhancing the security of wireless sensor networks (WSNs) by adopting intrusion detection systems (IDS) based on machine learning techniques. In this context, with a focus on using the WSN-DS dataset. The results of this research showed that machine-learning models could improve the security efficiency of wireless sensor networks by achieving accuracy ranging from 91% to 99.7% and testing time ranging from 0.006 to 0.1249, which enhances the ability to effectively retrieve and detect threats in real time.

Received: November 28 2024 Revised: January 21, 2025 Accepted: February 17, 2025

Keywords: WSN; IDS; WSN-DS; Random Forest; Decision Tree; Logistic Regression; MLP

1. Introduction

WSNs are one of the vital technologies for modern applications, as they evolve from the smart internet to environmental control and healthcare, these networks collect data from diverse environments like never before [1] [2]. This wireless sensor network consists of several independent and separate sensor nodes placed in different areas of interest. Crucial data can be obtained from these nodes, which are then wirelessly transmitted to a more powerful node known as a base station or sink node [3] [4]. With the widespread deployment of these networks, security issues and information protection from attacks have become critical issues that require continuous attention. These challenges embody an urgent necessity with the development of robust and effective security systems. Intrusion detection systems are the cornerstone of building security for wireless sensor networks [5].
Denial of service (DoS) in wireless sensor networks (WSNs) is a type that aims to cover resources not allocated to legitimate users or wireless sensor systems. These systems monitor data traffic and activities within the network, allowing for the detection of any abnormal patterns or suspicious activities. The core functions of an intrusion detection system include scanning networks and hosts, assessing network activities, generating alarms, and

responding to suspicious behaviors. Using machine learning techniques, the ability of these systems to adapt to and effectively deal with evolving threats can be improved [6]. In wireless sensor networks (WSNs), where all nodes act as hosts and network devices (routers and switches), each node is responsible for performing its own intrusion detection process. These systems are typically deployed in close proximity to secured network devices while monitoring connected hosts and connections, such as switches [7].

Detection methods can be either signature-based or anomaly-based, with a preference for anomaly-based detection, particularly focusing on the learning capabilities of WSN nodes. But the machine learning training procedure is still a recurring problem, as the WSN Challenges part describes. For this reason, a great deal of research in this field has attempted to improve the wireless sensor network training procedure by cutting down on training duration, relying on smaller datasets, and improving overall accuracy. In this context, this research seeks to improve the security of wireless sensor networks by adopting intrusion detection systems based on machine learning techniques, with a focus on using the WSN-DS dataset as a basis for achieving the required security, as the usage of this data set provides a reality-based framework that enables the evaluation of systems performance in diverse conditions and under the influence of real security challenges. Therefore, main purpose is to provide an effective framework for improving the security of wireless sensor networks using intrusion detection systems based on machine learning, which enhances the ability of these networks to effectively confront increasing security challenges.

To detect intrusions in wireless sensor networks (WSNs), machine learning requires powerful tools. These tools are compatible with the features of machine learning. We will mention some of them: (1) It can detect complex intrusion patterns with very high accuracy, (2) Depending on the types of attacks, machine-learning can adapt through retraining, which makes it flexible against threats, especially advanced ones. (3) It reduces human intervention due to automation and detection of intrusions before they occur. In addition, (4) Processing data streams and different network traffic with algorithms in a short and immediate time, which makes it suitable for large-scale wireless sensor networks.

There are also disadvantages to this use, including (1) resource exhaustion and limitations of wireless sensor networks, which makes machine learning algorithms require a lot of computational power, (2) it is difficult to obtain high-quality wireless sensor network data, although it is necessary, and the data trained from it cannot work on new environments due to the complex design and maintenance of its systems, in addition to the fact that its computational requirements lead to the exhaustion of sensor batteries. There are also challenges for machine learning in wireless sensor networks, including (1) the balance between latency and energy efficiency is a major and ongoing challenge, (2) despite the smooth performance of the network, machine learning systems are targets for hostile attacks, which compromises the effectiveness of detection, which requires adapting machine learning models to heterogeneous sensor networks with diverse capabilities and configurations, (3) and ensuring minimal data collection overhead is difficult because frequent changes in the topology of the wireless sensor network disrupt the predictions of the machine learning model.

The rest of the paper is organized as follows: A brief summary of relevant works is provided in Section II, the suggested approach is described in Section III, the results and their discussion are presented in Section IV, and the article is concluded in Section V.

2. Related Work

Several relevant previous studies were reviewed that analyzed and applied machine learning-based intrusion detection systems in the context of wireless sensor networks (WSNs). Previous studies have highlighted significant progress in understanding how to achieve security in a complex WSNs environment. Studies vary in their use of a wide range of approaches and techniques, with some exploring the effectiveness of machine learning techniques in dealing with challenges caused by sensor resource limitations. Some research has also analyzed the impact of these technologies on the accuracy of intrusion detection and their ability to deal with evolving security threats, Below we review the most prominent of these studies.

In Ahmad et al.'s (2022) [8] research, they explain how machine learning algorithms can help wireless sensor networks save money on security in a number of areas. Besides, it covers the difficulties and suggested fixes for enhancing sensors' capacity to recognize dangers, attacks, threats, and malicious nodes through self-improvement and machine learning. It also addresses unresolved challenges with the adaptation of machine learning algorithms to sensor capabilities in this kind of network. In addition, Liu et al.'s (2022) [9] provided a comparative analysis of the impact of machine-based intrusion detection systems technology in enhancing protection against security threats, demonstrated the system's efficacy using performance indicators including accuracy, F1 score, recall, FPR, FAR, and so on. A study by Gowdhaman, V and R. Dhanapal. (2021) [10] suggested a deep neural network (DNN)

based intrusion detection system. Experimental results confirm that the proposed deep neural network effectively detects attacks and outperforms traditional machine learning models including decision trees, random forests, and support vector machines. The best features are selected from datasets using cross-correlation procedure, and then the selected parameters are used as the basis for the structure of a deep neural network that detects these intrusions. With more efficiency and accuracy, the scheduling process obtains coverage breaches and detects new intruders. The scheduling method-based intrusion detection system has been reported in a research study (Wang et al. 2020) [11], using an artificial neural network-based intrusion detection system, as described in Caterusio et al. (2019) [12], a variety of features are used to identify normal and abnormal network events. Artificial neural networks are used to classify the extracted features, and performance measurements show that these networks perform better than conventional techniques. According to Sivagaminathan1 et al. (2023) [13], decision trees and artificial neural networks were combined with PSO for a specific case study. The results of this case study revealed that the PSO+ANN (particle swarm optimization + artificial neural network) classification method outperformed PSO+KNN (particle swarm optimization + nearest neighbors) and PSO+DT (particle swarm optimization + decision tree). Evzarn et al. (2021) [14] proposed a novel intelligent machine learning-based approach that addresses intrusion detection and intrusion detection. Using a clustered wireless sensor network (WSN) design, the model has the ability to detect intrusions in real time and identify the type of presence. The proposed ID-GOPA model efficiently and quickly identifies intrusions while minimizing resource consumption. A notable feature of this model is its use of gain ratio for feature selection, which is a crucial component that enhances the effectiveness of the algorithm. The model has passive-aggressive methods as an incremental learning machine, which contributes to reducing the overall features and processing burden. In the simulation results, the model shows an impressive accuracy rate of 96 percent when compared to offline models, demonstrating its exceptional accuracy. This model outperforms previous systems and is versatile, making it applicable for various purposes.

Overall, previous studies highlight the importance of using machine learning-based intrusion detection techniques as an effective means to improve the security of wireless sensor networks and make valuable contributions to the field of cybersecurity. Also, reflect a growing interest in analyzing the robustness of security and the resilience of systems in the face of modern challenges, with a focus on improving the performance of intrusion detection systems.

Machine learning (ML) applications for intrusion detection in wireless sensor networks (WSNs) are categorized according to their use cases: (1) Unsupervised learning: clustering algorithms such as K-Means to cluster normal and abnormal activities and detect different patterns in network behavior to identify potential intrusions (2) Supervised learning: decision trees, random forests (SVM) and neural networks to classify incoming packets or behaviors where network events are classified as normal or malicious. Using deep learning models such as CNNs and RNNs to analyze sequential data to detect complex attack patterns. (3) Developing lightweight machine learning models to reduce energy consumption while maintaining detection accuracy, energy and privacy and monitoring energy usage patterns to detect malicious nodes that are trying to drain network resources.

3. Strengths and Limitations of using Machine Learning in WSNs: -

3.1 Strengths WSNs: -

A. **High Accuracy to Detection:** Machine-learning algorithms can recognize any type of hacking patterns, especially complex ones that are very difficult to detect using traditional rule systems. Advanced algorithms such as (deep learning) provide very high accuracy in identifying all attacks, specifically Sybil or DoS attacks [13].

B. **Adaptability and Scalability:** Adaptability is a good feature of machine learning models and learning on training is one of their most important features, so machine-learning models using these two features can repel attacks and threats, making them highly resistant to all attacks, especially advanced attacks in the future. Also, with these features, they are able to deploy large-scale WSNs and even in the presence of different nodes and streaming data [14][17].

C. **Automation and Real-Time Detection:** The processes of detecting any intrusion or attacks are done through automation, which is a software technology that is done with prior orders to ensure the implementation of instructions and most importantly without human intervention and reliance on the computer. Therefore, machine learning had the distinct ability to complete the automation of detection processes for anomalies and threats with accuracy and strength, in addition to reducing time with immediate response [9][18].

3.2 Limitations WSNs: -

A. **Resource Constraints & Training Dependence:** WSN nodes have limited computing power and resources, which makes it difficult to develop and deploy machine learning models, and requires high-quality datasets that are often unavailable, which limits their operation due to the cost of creating them.

B. **Generalization, threat and energy:** Generalization is the similarity between a group of entities that may combat specific scenarios or are trained on new environments or types of attacks. Therefore, machine learning systems are vulnerable to threats and attacks, as these threats are carefully designed to deceive the system, which leads to draining and consuming energy in sensor nodes[10][19].

4. Motivation behind the proposed model

In this research, we use the approach of addressing the inherent limitations of traditional systems for intrusion detection in wireless sensor networks (WSNs) using machine learning for sensor networks while taking advantage of its strengths in WSNs, which is the main motivation in this research. Due to the weakness of current and traditional methods, including the lack of recognition of the accuracy of detection of the threat due to limitations or failure with the dynamic and different nature of WSNs, our research considers this the ultimate goal of detecting threats to WSNs using deep learning. Among the motivations that can be summarized in the following points:

1. Develop and implement machine learning resource models to balance high detection accuracy with low computational cost on the one hand and implement models that are adaptable to evolving intrusion patterns in real time on the other hand

2. Generalization to provide machine-learning technology via transfer diverse network scenarios, this in turn reduces the need for intensive training that consumes time. This method is called bias mitigation. On the one hand, on the other hand, the trained models are combined to make the model robust to repel attacks under another name, which is flexibility.

3. Obtaining compatibility between the engineering network for wireless sensing and protocols without cost and in a short time.

5. Gap the Proposed Model

Our proposed model aims or focuses on addressing the gaps of WSN intrusion detection systems based on machine learning technology. We decided to make a detailed table (Table 1) that shows the identification of gaps with the methods of addressing each gap for the proposed model and re-mentioning the gap motivation for each model.

Identified Gaps	Mechanism for addressing gaps	Motivation behind the gaps
Resource-intensive ML models unsuitable for WSN nodes [20[.	Utilizes lightweight ML techniques, model compression strategies to reduce computational, and energy costs.	To make ML feasible in resource-constrained WSN environments.
Poor adaptability to dynamic and heterogeneous WSN topologies [21].	Employs reinforcement learning and transfer learning for adaptive detection in diverse scenarios.	To ensure robustness in changing network configurations and attack patterns.
Vulnerability to adversarial attacks compromising model reliability [22].	Integrates adversarial training and defensive techniques to enhance model security.	To make the system resilient to intelligent attackers targeting ML vulnerabilities.
Dependency on extensive labeled training data, which is often unavailable [23].	Advantages unsupervised and semi-supervised learning methods to detect anomalies with minimal labeled data.	To reduce reliance on costly and time-consuming data annotation processes.
Limited transparency and explainability in ML-based systems [24].	Implements interpretable models using explainable AI (XAI) techniques to clarify decision-making processes.	To foster trust and usability among network administrators and stakeholders.

3. Algorithms Suitable for Intrusion Detection in WSNs

Deep learning algorithms are good and suitable for intrusion detection or attack in wireless sensor networks (WSNs) because they process large amounts of data and have the ability to handle diverse and complex data and adapt to dynamic environments [19].

Below is a comparison of deep learning algorithms with a detailed explanation for intrusion detection in wireless sensor networks. The table.1, 2. Illustrates this

A. Convolutional Neural Networks (CNNs):- The nature of the design of this algorithm makes it important to extract features from data using convolutional filters, and therefore it is effective in detecting patterns that indicate intrusion during network traffic. Its applications include detecting network anomalies and identifying unusual network traffic flows.

B. Recurrent Neural Networks (RNNs):- The nature of the work of this algorithm is that it specializes in sequential data to retain previous information, as the changes in it overcome the problem of the gradient that vanishes, such as long-short-term memory (LSTM) networks and gated repetition units (GRUs), which enables it to capture long-term dependencies. Its applications include gradually detecting attacks by analyzing the data traffic pattern over time, identifying network transfers and stopping slow ones.

C. Generative Belief Networks (GANs): - The nature of this algorithm consists of two neural networks: (1) Generator, (2) Discriminator. The generator network generates synthetic data while the Discriminator distinguishes between the synthetic data generated by the generator and the real data. One of its applications is to create trained data to improve models to detect breaches between generated and observed traffic.

D. Hybrid models: - They are integrated algorithms to find a good and previously known result. These algorithms complement each other for a higher goal. An example of this is the CNN-RNN hybrid network algorithm, as its advantage is to combine deep learning structures to take advantage of their strengths and process spatial and temporal features. Its applications include detecting attack patterns now of occurrence and then gradually until eliminating them by analyzing the different features. It is characterized by the accuracy of its detection and the flexibility of its work in WAN environments.

Table 1: Comparison of Deep Learning Algorithms for WSN Intrusion Detection

Algorithm	Strengths	Limitations	Best Use Case
CNN[25]	Feature extraction from spatial data.	High resource consumption.	Packet inspection and spatial anomaly detection.
RNN (LSTM, GRU)[26]	Temporal data analysis.	Computational complexity.	Time-series traffic analysis.
Autoencoders[27]	Unsupervised anomaly detection.	Threshold tuning required.	Resource anomaly detection.
DBN[28]	Abstract representation learning.	Training complexity.	Pretraining and feature extraction.
GAN[29]	Data augmentation for imbalanced datasets.	Sensitive training balance.	Synthetic attack simulation.
Hybrid Models [30]	Comprehensive detection capabilities.	High complexity.	Multi-modal intrusion detection.

Table 2: Explain Algorithms with advantages for WSNs

Algorithms	Advantages
Decision Trees DT	Lightweight and interpretable, ideal for WSNs with limited resources. Can be pruned to reduce size and complexity.
Naïve Bayes NB	Probabilistic approach with low computational overhead. Works well for anomaly detection with categorical or discrete features.

Random Forests: RF	Ensemble learning with higher accuracy than single decision trees. Suitable for cluster heads with moderate computational power.
Support Vector Machines (SVM)	Effective for binary classification problems. Kernel trick enables non-linear decision boundaries, useful for diverse attack patterns.
Deep Learning **DL**	Convolutional Neural Networks (CNNs): For spatial data features in routing attacks. Recurrent Neural Networks (RNNs): For time-series analysis of network traffic.

6. Proposed Methodology

This section outlines the proposed methodology of implementing the framework of intrusion detection in wireless sensor networks based on machine learning models. In this context, the proposed methodology of the research is as shown in figure 1

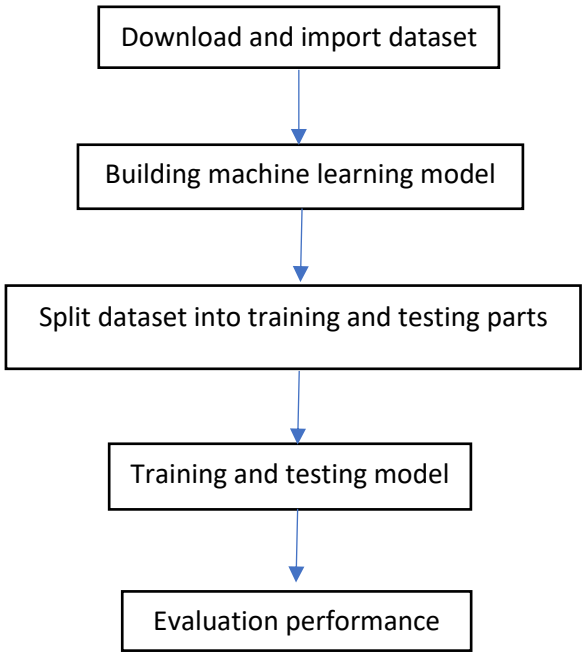


Figure1. Proposed Methodology

i. Download and import dataset, the dataset contains different features about the traffic performance and column named as attack type.
ii. Building machine learning model: building model contains different type of machine learning algorithm as decision tree, random forest, logistic regression and MLP classifier.
iii. Split dataset into two parts, the first part with 80% from dataset used for training model, and the second part with 20% from dataset used for testing.
iv. Training and testing model on these parts can predicate label.
v. Evaluate performance system for all type of machine learning algorithm from measure some metrics.
vi. After evaluate performance select the best algorithm that sustainable with the real time environment.

Below, we review the details related to the dataset and machine learning models used in the proposed methodology.

A. WSN-DS Dataset

The dataset used in this research is the WSN-DS [15], which is considered an essential tool for learning machine learning-based data detection systems, and is greatly involved in developing effective solutions to improve the security of wireless sensor networks. WSN-DS dataset is a specialized dataset for intrusion detection in Wireless Sensor Networks (WSNs) to detect Denial of Service (DoS) attacks, comprising 374661 records and 19 features representing various DoS attack types as shown in the figure 2, such as Blackhole, Grayhole, Flooding, and Scheduling attacks, in addition to the normal behavior (no-attack) records. The dataset is separated to 80% training data and 20% testing data.

```
RangeIndex: 374661 entries, 0 to 374660
Data columns (total 19 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0    id               374661 non-null  int64
 1    Time             374661 non-null  int64
 2    Is_CH            374661 non-null  int64
 3    who CH           374661 non-null  int64
 4    Dist_To_CH       374661 non-null  float64
 5    ADV_S            374661 non-null  int64
 6    ADV_R            374661 non-null  int64
 7    JOIN_S           374661 non-null  int64
 8    JOIN_R           374661 non-null  int64
 9    SCH_S            374661 non-null  int64
 10   SCH_R            374661 non-null  int64
 11  Rank              374661 non-null  int64
 12   DATA_S           374661 non-null  int64
 13   DATA_R           374661 non-null  int64
 14   Data_Sent_To_BS  374661 non-null  int64
 15   dist_CH_To_BS    374661 non-null  float64
 16   send_code        374661 non-null  int64
 17  Expaned Energy    374661 non-null  float64
 18  Attack type       374661 non-null  int64
dtypes: float64(3), int64(16)
memory usage: 54.3 MB
```

Figure 2. Content of WSN-DS

The WSN-DS dataset is an essential part of this research, as it provides diverse, real-world data to test the effectiveness of machine learning-based intrusion detection systems in realistic simulation scenarios for enhancing intrusion detection accuracy, reducing false positives, and improving overall network security.

B. Selection of ML Algorithms

This research proposes to improve the security of WSNs by adopting machine learning models-based intrusion detection systems. A variety of machine learning (ML) models is selected based on a range of inspiring previous studies that have provided valuable insights into achieving progress in this field including Random Forest, Decision Trees, k nearest neighbor (KNN), Logistic Regression, Artificial Neural Networks Multi-Layer Perceptron (MLP).

i) Random Forest

Using this approach, sets of random trees are packed to create random forests. Without any pruning, trees constructed with the technique take into account a predetermined quantity of random attributes at each node. In order to merge hundreds of decision trees, the method tests a large number of models at random. Next, each decision tree is trained on a distinct subset of examples. The average of the forecasts generated for every single tree is used to create the final random forest predictions. Even while using random forests increases processing complexity, they can lessen the overfitting impact of individual decision trees. Random Forest can be used to develop a wireless sensor-based intrusion detection system. The model performs multi-tree analysis to detect abnormal patterns or attacks. It can also be used to estimate the importance of variables in identifying attacks and improving system security.

ii) Decision Trees

Decision trees can be used to examine contexts and make decisions based on different variables. A decision tree is a type of classifier that seeks to maximize a discrimination criterion (such as classification error, entropy gain, etc.) for each partition that is chosen at each stage or decision that is made in the tree. One benefit of trees is that they offer a natural method to see how a dataset is categorized. Decision structure analysis can help identify intrusion patterns and identify rules that can be used to detect attacks.

iii) k nearest neighbor (KNN)

The k-nearest algorithm can be used to classify cases based on the values of variables. The foundation of the k-NN approach is the classification of an instance based on the classes of the k-most comparable training examples. Using the data supplied by the set of classified examples, the probability density function or, alternatively, the direct a posteriori probability that an instance belongs to a class may be estimated using this non-parametric classification technique. A KNN model can be trained on historical data to classify behaviors as normal or abnormal. It can be integrated as part of an intrusion detection system.

iv) Logistic Regression

Logistic regression can be used to determine the probability of a particular situation, such as a stealth attack, occurring. The model can be trained to classify logs between two categories (normal or attack) and used to predict future intrusions. Given a collection of independent variables (attributes) and a categorical distribution as the dependent variable (class), the logistic regression technique is applied to multiclass scenarios and is used to predict the likelihood of the distribution's numerous possible outcomes.

v) Multi-layer perceptron (MLP)

This is an artificial neural network can be used to learn complex patterns and analyze data interactions. Consisting of several layers, can solve non-linearly separable problems [16]. Three layers have been taken into consideration in the network: an input layer, which introduces the attribute values; a hidden layer, to which all input nodes are linked; and an output layer, which obtains the instance classification values based on the classes. MLP Classifier can be trained on history data to improve attack detection and classify data based on behavior patterns.

4. Results and Discussion

The classification results of WSN-DS dataset were obtained through a number of test cases applied variety of machine learning (ML) models (Random Forest, Decision Tree, KNN, Logistic Regression, Multi-layer Perceptron MLP). Parameters of ML models is shown in the table 1.

Table 1: Parameters of ML Algorithms

Algorithm	**Parameters**
Random Forest	Number of Estimators=20
KNN	Number of Neighbors =2
Logistic Regression	solver=liblinear multi_class= one-vs-rest
MLP (max_iter=100)	hidden_layer_sizes=(2, 4) max_iter=100 activation=relu, solver=adam

A. Performance Parameters

The following metrics are totally based on the actual and the predicted classes. There are different types of performance metrics which are used in evaluating the performance of the models:

i. True Positive (TP): The number of actual attacks that are correctly detected as attacks by a security system.

ii. True Negative (TN): The number of normal (non-attack) instances that are correctly identified as such by a security system.

iii. False Positive (FP): The number of normal instances incorrectly classified as attacks by a security system, often referred to as a "false alarm."

iv. False Negative (FN): The number of actual attacks that go undetected and are classified as normal instances by a security system.

v. Accuracy: The ratio of correctly identified instances (both normal and attacks) to the total instances. It assesses the overall effectiveness of the security system in identifying both types of instances.

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

vi. Precision: The ratio of correctly identified attacks to the total instances identified as attacks by a security system. It gauges how accurate the system is when it claims an instance is an attack.

$$Precision = \frac{(TP)}{(TP+FP)}$$

vii. Recall (Sensitivity or True Positive Rate): The ratio of correctly identified attacks to the total actual attacks. It measures the ability of the security system to identify all instances of attacks.

$$\text{Recall} = \frac{(TP)}{(TP+FN)}$$

i. F1 Score: The harmonic means of precision and recall. It's particularly useful in attack scenarios were achieving a balance between accurately detecting attacks (precision) and capturing all actual attacks (recall) is crucial.

$$\text{F1 Score } = 2 * \frac{(Precision * Recall)}{(Precision + Recall)}$$

These metrics in the context of cybersecurity give information on how well a security system detects and prevents assaults. They aid in evaluating the system's accuracy, its capacity to distinguish between attacks and typical occurrences, and its ability to balance recall and precision while responding to possible threats.

ii. Training time: Training time refers to the duration it takes for a machine-learning model to learn from the provided training data and adjust its internal parameters. During this phase, the model analyzes the input features and corresponding target labels to optimize its predictive capabilities. Training time encompasses the processes of feature extraction, model parameter adjustment, and convergence to an optimal state.

iii. Testing time: Testing time refers to the period it takes for a trained machine-learning model to make predictions or classifications on a separate set of testing data. This phase evaluates the model's performance and generalization ability by applying it to unseen data points. During testing, the model's learned patterns and relationships are utilized to generate predictions.

B. Results

The following, we will review the results related to the performance evaluation of the used machine learning models.

i) Performance Evaluation

As demonstrated in table 2, the random forest model outperforms the other ML techniques. The accuracy of the random forest model is 99.7 percent, with the precision of 99.82, recall of 99.84, and F1 score of 99.83.

Table 2: Performance Metrics

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Random Forest	99.7	99.82	99.84	99.83
Decision Tree	99.55	99.76	99.75	99.75
KNN	98.47	99.46	99.15	99.15
Logistic Regression	97.41	98.24	98.58	98.58
MLP	91.76	92.76	100	95.15

According to comparison results in figure 3, It is noticed that random forest algorithm has best performance metrics compared to all other algorithms, next comes the Decision tree algorithm, then the KNN algorithm, followed by the Logistic Regression algorithm and finally the MLP model.

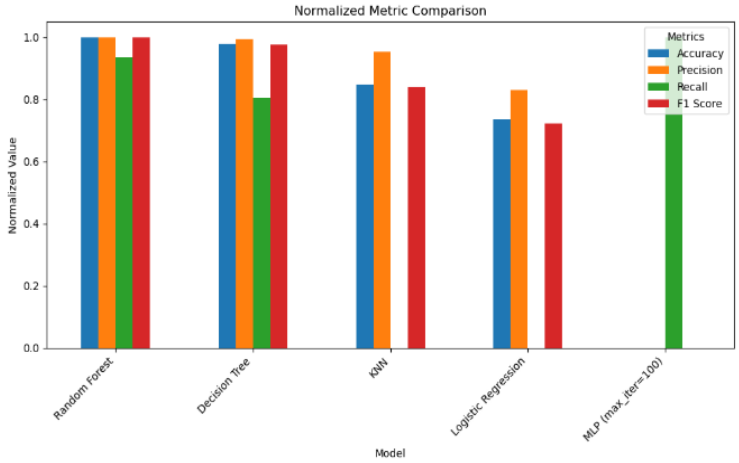
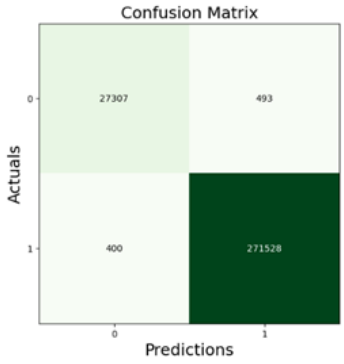
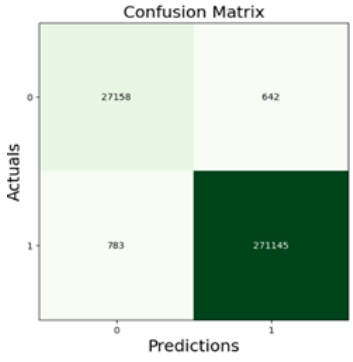


Figure 3. Performance metircs for ML models

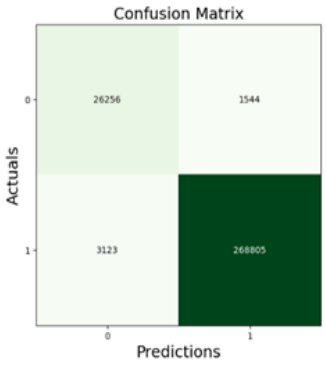
Figure 4 demonstrate the values of TP, TN, FP, FN for each matrix, and figure 5 displays comparison between confusion matrixes for all machine learning algorithms. Consistent with the previous results, the confusion matrix of Random Forest algorithm has best values, as its values of TP, TN are the largest while the values of FP, FN are the smallest. Then the results of Decision Tree, KNN, Logisticc Regression, and MLP comes sequentially.



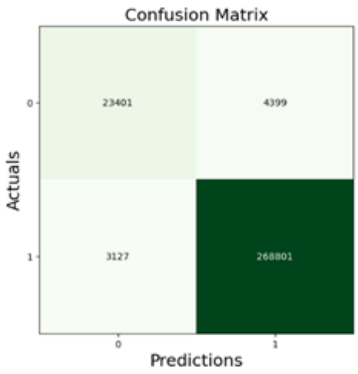
(a)Random Forest



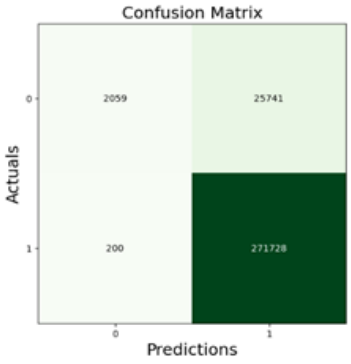
(b) Decision Tree



(c) KNN



(d) Logistic Regression



(e)MLP Classifier

Figure 4. Confusion Matrix for ML algorithms.

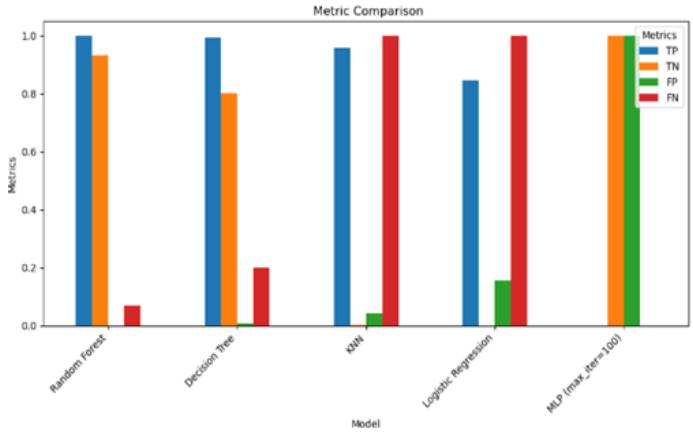


Figure 5. Comparision between Confusion Matrix metrics for ML algorithms

ii) Executing Times of Models

Table 3 show the training time and the testing time for each model, we notice that training with lowest value for KNN algorithm while with highest value for MLP. Conversely, testing time for all models is small and less than 1 excluding the KNN algorithm because its testing time is about 380 seconds, which is unacceptable time when attack detection must be performed in real time. Where as we know, testing time for machine learning in wireless sensor Networks (WSN) is more important, especially when required react quickly to security challenges and threats. Consequently, the best algorithm in terms of test testing time is the Logistic Regression algorithm, followed by the Decision Tree algorithm, then the MLP algorithm, and finally the Random Forest algorithm.

Table 3: Time for ML Algorithms

Model	**Training Time (Sec)**	**Testing Time (Sec)**
Random Forest	6.5362	0.1249
Decision Tree	2.9373	0.016
KNN	0.033	376.4968
Logistic Regression	2.758	0.006
MLP	11.3927	0.019

5. Conclusion

This research studied the using machine learning techniques based intrusion detection systems for improving the security of wireless sensor networks, with a focus on identifying the model that achieves the highest accuracy and lowest testing time in detecting attacks. Simulation results showed that studied machine learning models achieved accuracy ranging between 91% and 99.7%. Random Forest model achieves highest performance metrics among the studied models. Executing time for training and testing these models was also measured, and it was found that it ranges between 0.006 and 0.1249 seconds, with the exception of the KNN algorithm in which its test time is about 380 seconds, logistic regression model has lowest testing time among the studied models. As a result, random forest algorithm has the best performance in terms of accuracy and other performance metrics, whereas logistic regression algorithm has lowest testing time. On the other hand, the decision tree algorithm achieves balancing between high performance and the responding to security threats in real-time contexts.

Funding: “This research received no external funding”

Conflicts of Interest: “The authors declare no conflict of interest.”

References

[1] R. Wazirali, R. Ahmad, A. Al-Amayreh, M. Al-Madi, and A. Khalifeh, “Secure watermarking schemes and their approaches in the IoT technology: An overview,” *Electronics*, vol. 10, p. 1744, 2021.

[2] D. Kandris et al., “Applications of wireless sensor networks: An up-to-date survey,” *IEEE Internet Things J.*, vol. 7, no. 3, pp. 577–602, Mar. 2020.

[3] V. C. Gungor, B. Lu, and G. P. Hancke, “Opportunities and challenges of wireless sensor networks in smart grid,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3557–3564, 2010.

[4] M. Bouaziz and A. Rachedi, “A survey on mobility management protocols in wireless sensor networks based on 6LoWPAN technology,” *Comput. Commun.*, vol. 74, pp. 3–15, 2016.

[5] U. Ghugar and J. Pradhan, “NL-IDS: Trust-based intrusion detection system for network layer in wireless sensor networks,” in *Proc. 5th Int. Conf. Parallel, Distrib. Grid Comput. (PDGC)*, 2018.

[6] P. Laskov, D. Patrick, and C. Sch, “Learning intrusion detection: Supervised or unsupervised?” in *Proc. Sept. 2005*, pp. 50–57, 2014.

[7] H. Liu and B. Lang, “Machine learning and deep learning methods for intrusion detection systems: A survey,” *Appl. Sci.*, vol. 9, p. 4396, 2019.

[8] R. Ahmad, R. Wazirali, and T. Abu-Ain, “Machine learning for wireless sensor networks security: An overview of challenges and issues,” *Sensors*, vol. 22, no. 13, p. 4730, 2022.

[9] Z. Liu, G. Mohiuddin, Z. Jiangbin, M. Asim, and S. Wang, “Intrusion detection in wireless sensor network using enhanced empirical-based component analysis,” *Future Gener. Comput. Syst.*, vol. 135, pp. 181–193, 2022.

[10] V. Gowdhaman and R. D. Dhanapal, “An intrusion detection system for wireless sensor networks using deep neural network,” *Soft Comput.*, 2021. DOI: 10.1007/s00500-021-06473-y.

[11] W. Wang, H. Huang, Q. Li, F. He, and C. Sha, “Generalized intrusion detection mechanism for empowered intruders in wireless sensor networks,” *IEEE Access*, vol. 8, pp. 25170–25183, 2020.

[12] Cateruccio et al., “Short-long term anomaly detection in wireless sensor networks based on machine learning and multi-parameterized edit distance,” *Inf. Fusion*, vol. 52, Dec. 2019, pp. 13–30.

[13] V. Sivagaminathan, M. Sharma, and S. K. Henge, “Intrusion detection systems for wireless sensor networks using computational intelligence techniques,” *Cybersecurity*, vol. 6, p. 27, 2023.

[14] S. Ifzarne, H. Tabbaa, I. Hafidi, and N. Lamghari, “Anomaly detection using machine learning techniques in wireless sensor networks,” *J. Phys. Conf. Ser.*, vol. 1743, no. 1, 2021.

[15] I. Almomani, B. Al-Kasasbeh, and M. AL-Akhras, “WSN-DS: A dataset for intrusion detection systems in wireless sensor networks,” *J. Sensors*, vol. 2016, p. 4731953, 2016.

[16] M. Kubat, *An Introduction to Machine Learning*, Berlin/Heidelberg, Germany: Springer, 2021.

[17] M. J. Khudair, H. A. Abd Ali, and S. M. Darwish, “A quantum-inspired ant colony optimization approach for exploring routing gateways in mobile ad hoc networks,” *Electronics*, vol. 12, no. 5, p. 1171, 2023.

[18] Z. H. Noori, S. K. Ebis, and D. Saad, “An information security engineering framework for modeling packet filtering firewall using neutrosophic petri nets,” *Computers*, vol. 12, no. 10, 2022.

[19] Y. Y. Ghadi et al., “Machine learning solution for the security of wireless sensor network,” *IEEE Access*, 2024.

[20] V. N. N. Tam and C. T. Thanh, “Enhancing wireless sensor network security with machine learning,” in *Proc. Comput. Sci. Online Conf.*, Cham: Springer Nature Switzerland, Apr. 2024, pp. 604–626.

[21] O. A. Khashan, “Dual-stage machine learning approach for advanced malicious node detection in WSNs,” *Ad Hoc Netw.*, vol. 166, p. 103672, 2025.

[22] O. Ahmed, “Enhancing intrusion detection in wireless sensor networks through machine learning techniques and context awareness integration,” *Int. J. Math. Stat. Comput. Sci.*, vol. 2, pp. 244–258, 2024.

[23] P. R. Jayanthi et al., “An improved dynamic traffic routing protocols for WSNs using machine learning,” in *Proc. 15th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, 2024, pp. 1–6.

[24] A. Darabseh and M. Faizan, “Outlier detection in wireless sensor networks using machine learning and statistical-based approaches,” *Revue d’Intell. Artif.*, vol. 38, no. 4, 2024.

[25] Z. Hao, M. Li, W. Yang, and X. Li, “Evaluation of UAV spraying quality based on 1D-CNN model and wireless multi-sensors system,” *Inf. Process. Agric.*, vol. 11, no. 1, pp. 65–79, 2024.

[26] T. Roy and S. K. Shome, “Optimization of RNN-LSTM model using NSGA-II algorithm for IoT-based fire detection framework,” *IETE J. Res.*, vol. 70, no. 7, pp. 6239–6254, 2024.

[27] B. C. Sengodan et al., “Variational autoencoders for network lifetime enhancement in wireless sensors,” *Sensors*, vol. 24, no. 17, p. 5630, 2024.

[28] Q. Li, Y. Ma, and Y. Wu, “Utilize DBN and DBSCAN to detect selective forwarding attacks in event-driven wireless sensors networks,” *Eng. Appl. Artif. Intell.*, vol. 126, p. 107122, 2023.

[29] S. P. Kumar, S. Garg, E. Alabdulkreem, and A. B. Miled, “Advanced generative adversarial network for optimizing layout of wireless sensor networks,” *Sci. Rep.*, vol. 14, no. 1, p. 32139, 2024.

[30] K. Ramu et al., “Deep learning-infused hybrid security model for energy optimization and enhanced security in wireless sensor networks,” *SN Comput. Sci.*, vol. 5, no. 7, p. 848, 2024.