






Multi Cyber Attacks Detection System in Internet of Things Based on Machine Learning

Rand Nabeel Dawood¹, Neamet Akeel Fawzi^{2*}, Hind Ali Abdul Hassan³

¹ Ministry of Education, General Directorate of Public, Private and Foreign Education, Baghdad 10071, Iraq

² Computer Technique Engineering Department, Imam Al-kadhim University College, Wasit 10031, Iraq

³ Department of Computer Science and Information, Technology University of Wasit, Wasit 10031, Iraq

Corresponding Author Email: nemat.aqel@iku.edu.iq

Copyright: ©2025 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/jesa.580608>

ABSTRACT

Received: 13 May 2025

Revised: 12 June 2025

Accepted: 19 June 2025

Available online: 30 June 2025

Keywords:

sensor networks, cybersecurity, intrusion detection, Random Forest, machine learning, threat mitigation, data preprocessing, feature engineering

In this work, we propose a machine learning-based approach to improvement in cybersecurity in sensor networks (SNs). Indeed, SNs are vital to a wide range from environmental monitoring and military surveillance to underwater exploration. Due to their open and distributed nature, they inherently remain much more vulnerable to cyber-attacks. Because of unique constraints about the bandwidth, latency, and energy, traditional security approaches remain inappropriate for them. In our proposed approach, we have used a Random Forest Classifier (RFC) and Supported Vector Machine Classifier (SVM) for detecting and mitigating various kinds of network attacks. The dataset used in this work is wireless sensor network dataset (WSN-DS), which contains SN-relevant features such as attack types like normal, flooding, time division multiple access (TDMA), Grayhole, and Blackhole. Preprocessing will be done by dropping the irrelevant columns, scaling the features, and encoding the target variable. Our model worked well using the RFC approach compared to SVM, yielding an accuracy of 99.66%, with precision at 99.69%, recall at 99.66%, and an F1-score of 99.67%. Besides, we also tested our Random Forest-based IDS on the Network Security Lab-Knowledge Discovery and Data Mining (NSL-KDD) dataset that includes a number of features relevant to network traffic and intrusion detection. The obtained results for the “NSL-KDD” test set are as follows: accuracy-1.00, precision-0.87, recall-0.87, F1-score-0.87. We can understand from the confusion matrix that this model correctly identified various attack types and normal behavior. Also, a classification report shows more about the performance of the model for each class. The high performances of our Random Forest models on both datasets confirm the potentiality of machine learning-based solutions against intrusion detection in UWSNs. Indeed, robustness against imbalanced data and the ability to capture complex interactions between features explain their success. In the future, other machine learning models can be explored, including deep learning approaches, as well as real-time deployment of IDS in UWSNs. Besides, extending the datasets to more types of attacks and scenarios would improve the generalization capability of these models.

1. INTRODUCTION

Sensor networks are an important recent development in the area of wireless communication and sensing technologies due to the unparalleled capabilities they offer in a wide range of applications, including environmental monitoring, underwater exploration, and military surveillance. These networks involve the deployment of sensor nodes in underwater environments for data collection and its transmission toward surface stations or other nodes. In underwater environments, the unique challenges imposed by the underwater setting in terms of high latency, bandwidth limitation, and energy constraints make innovative solutions necessary for enabling efficient and reliable data communication [1, 2].

One of the greatest problems facing SNs is providing robust security against cyber attacks. Considering its open and distributed nature, SNs are prone to many types of attacks, such as flooding, misbehavior TDMA, Grayhole, and Blackhole [3,

4]. These types of attacks can highly weaken a network's performance, to data loss, reduced network lifetime, and compromised data integrity. Traditional security approaches designed for terrestrial networks usually fail to address the unique characteristics and constraints of underwater sensor networks. Thus, the need for specialized security solutions that are tailored to the underwater environment is becoming very crucial. With the development of an advanced Intrusion Detection System (IDS) able to detect and avoid most network attacks, machine learning can give an efficient contribution to enhancing SNs security. Machine learning (ML) algorithms can consider large volumes of data in order to detect patterns or anomalies that may define malicious activities. Among these algorithms, some of the most efficient in dealing with complex classification tasks are ensemble methods like Random Forests; therefore, they have become ideal for intrusion detection in SNs [5].

In this work, we explore the utilization of machine learning,

particularly the Random Forest Classifier, to construct an effective IDS for SNs. In this paper, we are using a rich dataset, namely the "WSN-DS" dataset, including a comprehensive set of features related to SNs. Our approach commences with the preprocessing of data, preparing the latter for machine learning through scaling features and encoding categorical variables. We train the Random Forest model to distinguish between normal network behavior and various types of attacks. In this paper, we propose a machine learning-based IDS to enhance the security of SNs in a scalable and efficient manner.

This is achieved with the help of effective detection and classification of various attacks, and it helps improve the reliability and robustness of underwater sensor networks to be resilient against any form of cyber-attack. In future work, other advanced machine learning models will be integrated together with real-time deployment scenarios that will further reinforce the security framework of SNs. The current study therefore reiterates the capability of machine learning to change the security landscape of SNs, alluding to a pathway toward more secure and reliable underwater communication networks.

2. RELATED WORKS

The security issues in WSN is an interesting field of research since it deals with a very limited resources network in computation and memory. Sadia et al. [6] presented the enhancement in security related to wireless sensor networks through the development of an advanced Network Intrusion Detection System. WSN is very important and vital in many applications; however, they are surrounded by several security threats-unauthorized access and attacks.

The implemented NIDS is intended to safeguard the WSN against some of the well-known cyber-attacks, including impersonation, flooding, and injection. Feature selection in a proposed system is to judiciously select the most contributive features against a very large dataset, narrowing down to 76 out of 154 and further down to 13 selected features. This enables the system to give more attention to the most representative indicators of the potential breach in security.

The authors clean the dataset by removing null values and replacing unknown entries with some placeholder to prepare the data for analysis. They use feature scaling to ensure that all selected features are roughly in the same scale, which is an important preprocessing step before a machine learning model will be able to process the data.

The core of their approach is a CNN-an AI model especially suited for analyzing visual data but applied to many other kinds of data, including the security features of a WSN. This CNN needs training to recognize patterns in the data which indicate intrusion. The proposed CNN model compares various other neural networks, such as DNNs and LSTM networks, with respect to performance evaluation. The model's effectiveness is in terms of its accuracy, loss, precision, recall, support, F1, and macro-average. It yielded a very impressive result, with an accuracy rate of 97% and low loss metric of 0.14. As a result, the CNN model performed very well in identifying when the network was under an attack while keeping low the rate of false alarms.

They have developed a sophisticated NIDS with the application of CNN for the detection and evasion of intrusion by cyber threats in SNs. The system can guarantee a high detection rate with low false alarms, which has been a milestone in the security aspects of SN.

Ali et al. [7] carried out a paper related to computer network protection against intruders or malicious users. They discuss a specialized security system classified as an IDS. This system monitors both the software and the hardware of a network to ensure everything is operating correctly and no one is sneaking in where they should not be.

Since the development of IDSs, many brilliant minds have worked on enhancing them. They have made the systems very effective in raising a bar when something is amiss, such as an attempt to penetrate the network. They have also enhanced them to not give false alarms, which is when the system thinks there's a problem but there really isn't.

This article reviews a number of IDSs using something called machine learning. Machine learning may be described as teaching a computer to learn from data, so it can improve at its job over time. Similar IDSs with machine learning do an excellent job distinguishing between situations when everything is normal and when it is not. Some of them are even capable of finding new types of attacks that the computer has not seen yet.

They organized various IDSs using machine learning in a list or chart. This will make it easier for other researchers working on cybersecurity, scientists, and engineers to conceptualize all various concepts these IDSs work upon and employ them in keeping the networks safe.

This means that this paper encompasses this special kind of security system for computer networks, which uses smart computer learning in order to spot something that has gone wrong. The authors have given a helpful guide to show all the different ways this security systems can work-useful to the people who want to make the networks safer.

Francis and Sheeja [8] touched on utilizing smart computer programs called machine learning techniques to make the computer network security systems much better at catching bad people trying to break into them. The security systems are called Intrusion Detection Systems, or IDSs.

The problem with older IDS is that they sometimes get confused and think there's a problem when there isn't-so-called false positives-or they don't react fast enough to real threats. That is because the ways people try to break into networks are always changing, and older systems can't keep up.

In this paper, the method of machine learning is proposed, which teaches the computer from examples to learn and improve its job with each passing time. The computer can observe how data moves around in a network, commonly known as network traffic, and look out for any suspicious situations that might mean an attack is underway on the network.

It then puts to the test various machine learning programs, such as deep learning models, which excel in learning from lots of data. Next, it compares how well this work to the older rule-based IDS. This finds its application in the building of a new type of IDS that would be capable of recognizing old and new methods of attack against a network and halting them. The research is important because it makes computer networks a bit safer. With the new IDS able to adapt to new threats using machine learning, it works even better than those created earlier. It also goes on to highlight the importance of consideration of lots of other research, thinking of new ideas, then testing them well to ensure they work.

It is another way of saying that the paper deals with the intelligent of computer network security systems by making them learn from data about the bad guys that are trying to break into the networks in order to keep it safe. This paper

demonstrates that it is such smart learning programs that make all the difference in how the deployed security systems will work.

Is related to enhancement in WSNs' security. WSN consists of a large number of minute computers, called sensor nodes, which collect information and transmit the same to one central location [9]. Because of their small dimensions and low power in these tiny computers, usually with limited storage, ensuring safety against intruders becomes a challenging task.

To add to this, the authors have shed light on the use of a specialized kind of security mechanism called Network Intrusion Detection System (IDS). Such a system acts like a watch guard that views the network and tries to catch an intruder not supposed to be there.

It is not a new way of giving strength to this guard. They apply a mix of two smart computer programs: a multilayer perceptron and a CatBoost classifier. Those programs are a kind of brain which can learn to identify when something is wrong in the network. The authors use a special trick, with the help of feature selection, in order to render their system even smarter. In fact, this means selecting some valuable pieces of information from the great amount of data. They make use of an algorithm known as the Pelican Optimization Algorithm, POA, to assist in doing this. POA helps the computer to choose the best settings of the smart programs and the most important information to look at.

The authors tested their new system on three different sets of data from real networks. They did research on how well it could perform in spotting problems versus the number of times it was right about these incidents, missed how many times, or how many times it had found a problem when there wasn't one FAR, and DR. Also, they explored the time it took to work: complexity time.

It came out that their new system does a fantastic job in distinguishing between normal network activity and when someone is trying to cause some trouble. It had a very low false alarm rate and, therefore with high accuracy means performing good without too much confusion. More specifically, this is a paper dealing with a novel paradigm in the protection of wireless sensor networks. The authors are developing a smart security system in which specific special computer programs, so-called machine learning algorithms, will continuously learn and monitor a villain trying to break into a network. They tested it with real data and found it very good at keeping the network secure.

In the study [10] introduced the work that has been done on how deep learning, which is a type of smart computer learning, can be used to help keep computer networks safe from bad people trying to break in. They refer to this type of security system as an Intrusion Detection System (IDS).

The paper calls for the significant importance of having good data to train these smart security systems. They enumerate and classify 35 different data sets useful for this task. These datasets originate from various types of networks, including those which carry internet traffic, those that are part of the electrical grid, or those used by Android apps and Internet of Things (IoT) devices. They then consider seven variant types of deep learning models-like different ways with which a computer learns to spot problems. Examples of these models include, among others, a recurrent neural network, deep neural networks, and convolutional neural networks, all of which are implemented in different fashions through which a computer is supposed to learn patterns in data.

These models are evaluated on two new sets of real network

traffic data: the CSE-CIC-IDS2018 dataset and the Bot-IoT dataset. They want to see the capability of each model in distinguishing between normal network activity and when someone is trying to launch an attack on the network.

They use three important measures as a means of gauging how well these models work: precision, or how often the model is right; false alarm rate, or how often it thinks there's a problem when there isn't; and the detection rate, which is just how often the model catches real attacks.

The paper is on how different smart computer programs can be used to watch the networks of computers and to catch bad people trying to break in. Many such variants of learning from data for these programs are discussed and tested on real network traffic to see which methods better maintain the safety of the network.

Focused on big challenges of keeping WSNs energy-efficient and secure [11]. WSNs are made up of small devices that collect information and send it wirelessly; however, all these devices are not extremely powerful since their power supply is derived from a very small-sized battery.

The problem is, trying to make these networks more secure usually means using more energy - and hence drains batteries faster. Traditional security approaches, such as encryption and key management, are too power hungry for use in such small devices.

In their opinion, this could be resolved with the use of machine learning algorithms. Those are smart kinds of computer programs that can learn from data in order to improve their job over time. They would assist in network monitoring and security decision-making without power-hungry computations.

The use of machine learning in WSNs is not an easy task; it requires large amounts of data to be provided for the training of algorithms, which is hard for already power- and data storage-limited networks.

The paper discussed how machine learning can help minimize the security costs of WSNs while improving the ability of sensors to spot threats, attacks, and malicious nodes. It also examines challenges regarding the good performance of machine learning with the limited capabilities of sensors in WSNs.

3. METHODOLOGY

The methodology is designed in such a way that it could effectively utilize machine learning techniques to improve cybersecurity in UWSNs. This process is structured to ensure thorough data preparation, robust model training, and comprehensive evaluation of the Random Forest Classifier. The dataset used for this work, "WSN-DS.csv," was first preprocessed to make it suitable for machine learning modeling [12-14]. In this respect, the features for this study are selected from the NSL-KDD dataset by considering a more comprehensive feature set of network traffic and intrusion detection. It thus illustrates that the selected features have been very appropriate in order to present the performance of the IDS in SNs. NSL-KDD has given primary importance to resolving some of the inherent problems within it, such as the distribution of attack types and duplicate records that lead to biased models. The NSL-KDD dataset contains a wide variety of features such as protocol type, service, flag, several connection statistics, and the target variable for the type of attack or normal behavior. Such attributes match well with the

characteristics and security challenges of UWSNs, which are usually deployed in underwater environments with stringent resources and give way to increased vulnerability against cyber-attacks. Therefore, the detection and mitigation capability of a wide range of attacks of the proposed Random

Forest (RF) or Supported Vector Machine (SVM) based IDS can be effectively evaluated and optimized using NSL-KDD dataset in UWSNs to develop secure and reliable underwater communication networks [15, 16].

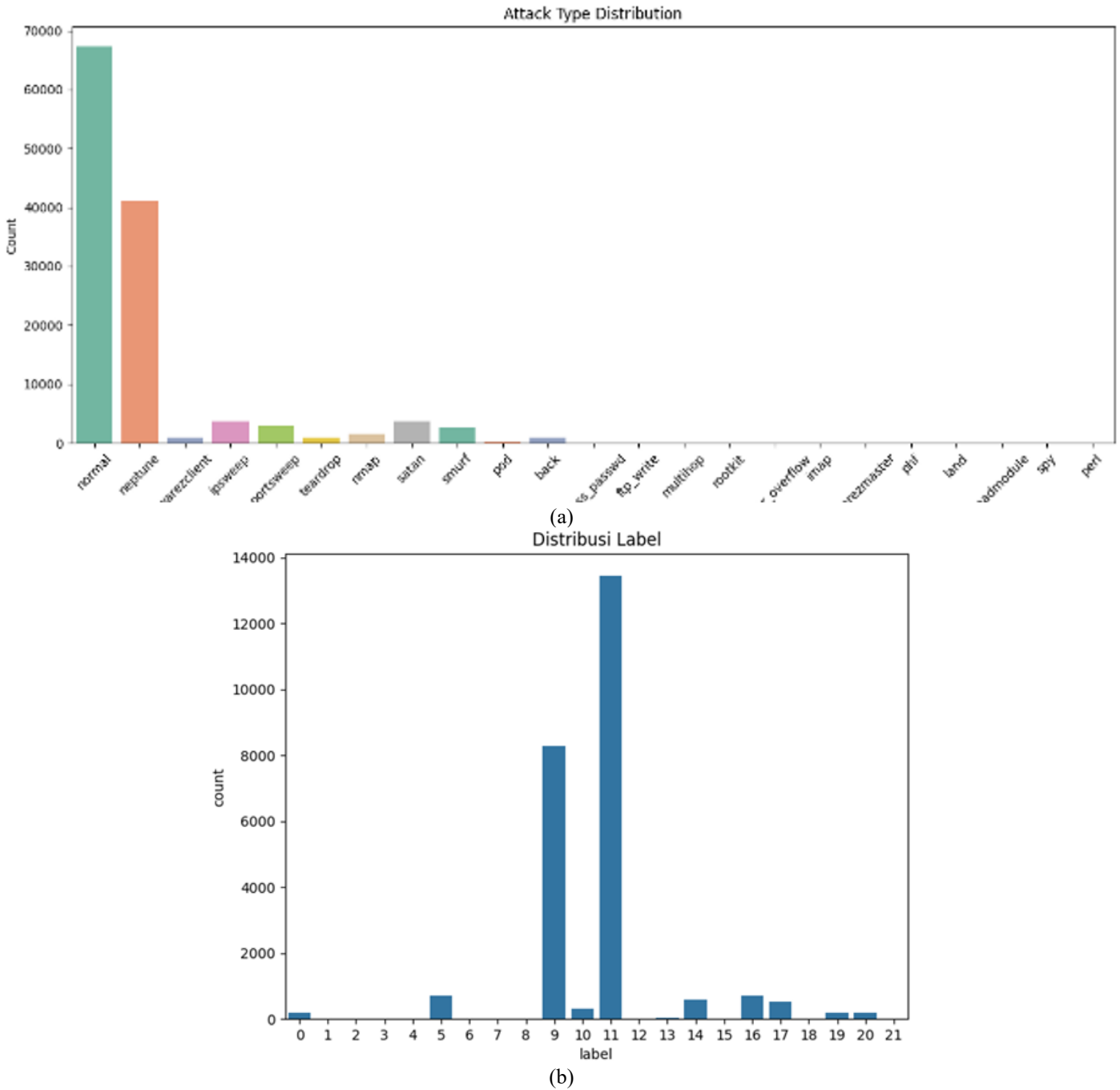


Figure 1. WSN-DS attack distribution

A Random Forest Classifier (RFC) is an ensemble machine learning technique employed for predictive modeling, applicable to both classification and regression applications. It amalgamates the results from many decisions rendered by distinct decision trees. The final stage involves determining the verification of the mode (most common value) for classification or the mean forecast from these trees for regression. The method commences by dividing the dataset into a training set and a test set, followed by the random selection of all samples from the training set. Each category employs a decision tree to delineate divisions, aiming for

precise segmentation of data points into groups or the prediction of values. This process involves sequentially selecting samples and subsequently constructing decision trees. The Random Forest aggregates the individual predictions of all decision trees and ultimately performs a majority vote. The forecasts receiving the majority of votes constitute the definitive output values [17].

Support Vector Classifiers (SVC) are the most prevalent systems utilized in machine learning. Support Vector Classifiers are supervised learning models in machine learning that utilize an algorithm to analyze data for classification and

regression analysis. Advancing from their ability to do linear classification, Support Vector Classifiers (SVCs) have demonstrated non-linear classification via the method known as "the kernel trick." This approach generates the requisite k-dimensional feature spaces that delineate specific classes according to characteristic traits linked to each class. The design is intricate and requires careful consideration of the equilibrium between accuracy and coverage; each boundary is traversed to minimize the distance between them and the distinct classes, hence mitigating misclassification mistakes [18].

Figure 1(a) shows the attacks distribution which has been classified in this work while Figure 1(b) shows the attacks classes. The dataset has a 23 different attacks, this require a powerful classifier to detect and classify these attacks with acceptable accuracy.

4. RESULTS AND DISCUSSIONS

Two machine learning classifier have been used in this work; RF and SVM classifiers. Many metrics have used to evaluate the classification ability for both methods such as accuracy, macro avarage, weighted avarage, precision, recall, etc. Exploratory Data Analysis (EDA), is basically a critical preliminary step in data analysis, especially in machine learning and predictive modeling. It mainly consists of the investigation and visualization of data to uncover the patterns, anomalies, trends, and relationships that exist within the data. This process helps with understanding the data, and finding potential problems with missing values, outliers, and inconsistencies that will affect a machine learning model. A good EDA might provide important insight that guides further steps with regard to data preprocessing, feature selection, and model development.

The "WSN-DS.csv" used in obtaining the dataset includes feature records of sensor readings and network parameters, touching aspects related to wireless sensor networks. The 'Attack type' will be the target variable, classifying the different types of network attacks. It is expected that the diversity of the dataset reflects the complex nature of network security threats.

Table 1. Random forest classification report

Class	Precision	Recall	F1-Score	Support
0	1	1	1	36
1	1	1	1	1
3	1	1	1	2
5	1	0.99	1	147
7	0	0	0	1
9	1	1	1	1694
10	0.98	1	0.99	51
11	1	1	1	2674
13	1	1	1	10
14	0.99	0.98	0.99	117
16	0.98	0.98	0.98	130
17	1	1	1	114
18	0	0	0	1
19	1	1	1	31
20	0.96	0.93	0.95	28
21	1	1	1	2
accuracy		1		5039
macro avg	0.87	0.87	0.87	5039
weighted avg	1	1	1	5039

Table 2. SVM classification report

Class	Precision	Recall	F1-Score	Support
0	0	0	0	36
1	0	0	0	1
3	0	0	0	2
5	0	0	0	147
7	0	0	0	1
9	0	0	0	1694
10	0	0	0	51
11	0.53	1	0.69	2674
13	0	0	0	10
14	0	0	0	117
16	0	0	0	130
17	0	0	0	114
18	0	0	0	1
19	0	0	0	31
20	0.8	0.14	0.24	28
21	0.67	1	0.8	2
accuracy			0.53	5039
macro avg	0.12	0.13	0.11	5039
weighted avg	0.29	0.53	0.37	5039

The classification reports (Tables 1 and 2) show that the Random Forest model outperforms the SVC model by a substantial margin. Random Forest: Classifies most classes effectively with high precision, recall, and F1-scores. It is an excellent classifier for this dataset because to its near-perfect accuracy and weighted average. It cannot classify the exceedingly rare classes (7 and 18) due to data shortages, its single small drawback. SVC: Most classes exhibit 0.00 for all metrics, making classification difficult. It has some success with classes 11, 20, and 21, but its accuracy and macro average are low. The reports suggest the Random Forest model is best for this classification assignment. SVC seems useless for numerous classes.

Figure 2 ranks features by importance, with the most significant at the top and the least important at the bottom. The x-axis shows importance scores from 0.00 to 0.12. The top five features are `src_bytes`, `same_srf_rate`, `flag`, `dst_host_srv_error_rate`, and `error_rate`. Moderately important characteristics are `dst_host_same_srv_rate`, `dst_bytes`, `count`, and `difficulty`, with scores above 0.04 but below the top features. Less Important Features: The importance score drops substantially as you go down.

`Dst_host_error_rate`, `logged_in`, `wrong_fragment`, `srv_diff_host_rate`, and `num_compromised` are less important. `Num_guest_login`, `right_rhost`, `num_failed_logins`, `num_file_creations`, `root_shell`, `num_access_files`, `urgent`, `num_outbound_cmds`, `is_host_login`, and `num_shells` have almost minimal relevance scores, close to 0.00. The graphic shows that `src_bytes` and `same_srf_rate` are the most important features for model predictions.

The confusion matrix is perhaps the simplest diagnostic tool in machine learning that details the performance of any model by breaking it down into TP, TN, FP, and FN categories. From this matrix, analysis can be made to reach a conclusion on its behavior about correct and wrong classification. That is, a high number of true positives and true negatives will mean that the model is truly good at correctly identifying instances of the respective classes, while for low false positives and false negatives, this will mean the model is good at keeping errors as low as possible. These four components balance each other, hence, are very important to understand the model's precision and recall along with overall correctness. Confusion matrix analysis can find out possible biases or aspects a model may

underperform, which may direct further refinement and optimization of the model. The confusion matrix, at its core, is a detailed diagnostic tool that provides actionable insights into a model's predictive capabilities and its readiness for deployment in real-world scenarios.

A confusion matrix categorizes data points incorrectly when a machine learning algorithm errs in classification. The model

exhibits an error. Examine the off-diagonal elements of the confusion matrix to comprehend misclassification. The anticipated class of these cells diverges from their true class. By rows: Numerals outside the diagonal column (denoting the actual class) signify misclassifications when the class was misrepresented as an alternative.

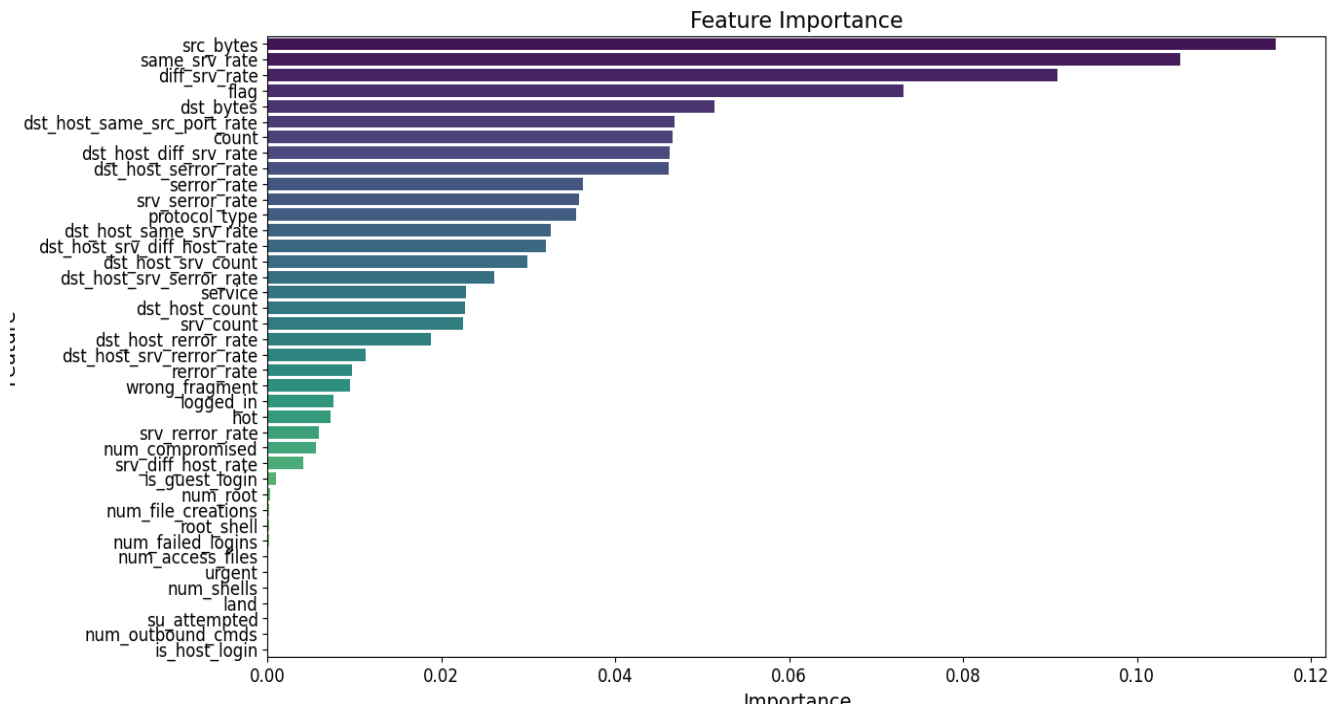


Figure 2. Feature importance analysis

Figure 3 demonstrates that Random Forest (RF) consistently surpasses Support Vector Machine (SVM) across all assessed criteria. Accuracy: RF attains an accuracy of 1.0 (or nearly so), signifying perfect or near-perfect classification, whereas SVM's accuracy is approximately 0.53. Precision: Random Forest exhibits a precision of roughly 0.88, markedly surpassing the precision of Support Vector Machine, which is approximately 0.12. Recall: Likewise, the recall of RF is approximately 0.88, far surpassing SVM's recall of roughly 0.13. The F1-Score for RF is roughly 0.88, far surpassing SVM's F1-Score of around 0.11. The image unequivocally illustrates that the Random Forest model significantly outperforms the Support Vector Machine model in the assessed job, as indicated by the classification performance measures.

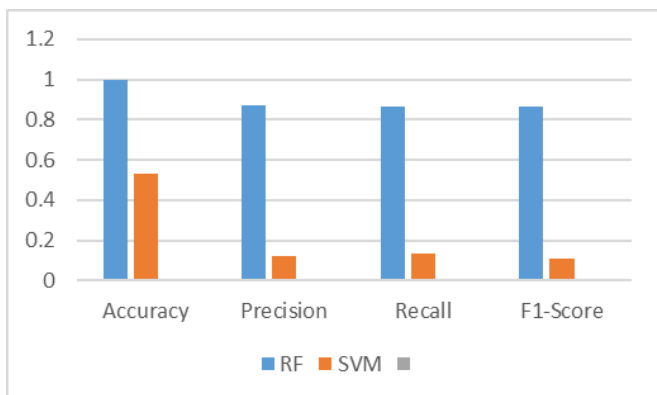


Figure 3. RF and SVM evaluation metrics

Results of intrusion detection system on NSL-KDD dataset

This confusion matrix depicts the performance of a Random Forest Classifier and supported vector machine based on various attack classes of the NSL-KDD dataset. The colorbar shows the number of predictions, its highest intensity indicating a larger number of correct or incorrect classifications. Diagonal elements correspond to correct classifications, while the off-diagonal ones represent misclassifications. Figure 4 shows the confusion matrices for both RF and SVM.

In terms of classes 5 and 7, both models appear to do exceptionally well, as seen by the high number of right classifications they achieve. In comparison to Random Forest, Support Vector Machines (SVM) have 147 correct predictions for class 3, while Random Forest only has 146. When compared to Random Forest, SVM has 1694 valid predictions for class 5, while Random Forest only has 1693. Random Forest has 2671 correct predictions for class 7, whereas Support Vector Machines have 2672 correct predictions. When compared to Random Forest, SVM has 117 valid predictions for class 9, while Random Forest only has 115. Comparatively, Random Forest only has 127 valid predictions for class 10, whereas SVM has 130 correct predictions. The apparent flaws include the fact that RF successfully identified ten occurrences of class 1, whereas SVM incorrectly classified one instance (which was anticipated to be 7) and shown that there were no valid classifications achieved on the diagonal for class 1. It appears from here that RF is more effective in classifying Class 1. Classes 4, 6, and 12b The diagonal values of True Label 4, 6, and 12 are either extremely low or entirely

absent, which makes it difficult for both models to correctly categorize instances of these labels. As a result, it may be deduced that these classes provide difficulties for both algorithms or have a limited number of instances. The following is a condensed version of the misclassifications that were given for both models: A case of True Label 3 is incorrectly identified as True Label 6 by RF. A case of True Label 1 is incorrectly identified as True Label 7 by SVM. There are 51 instances of True Label 5 being incorrectly classified as True Label 6 by both models. Misclassifications from True Label 7 to other classes are observed in both models, albeit the specific classes presented by each model are slightly

different (RF: 9, 13; SVM: 13, 14). Each of the models incorrectly assigns the value 10 to instances of True Label 12.

According to the supplied confusion matrices, the Support Vector Machine (SVM) marginally surpasses the Random Forest in the accuracy of classifications for certain classes (3, 5, 7, 9, 10). Nonetheless, Random Forest demonstrates superior performance in classifying Class 1. Both models exhibit difficulties with classes 4, 6, and 12, indicating that they may be minority classes or particularly challenging to differentiate. The elevated values on the diagonal for classes 5 and 7 signify that both algorithms are highly proficient in classifying these particular classes.

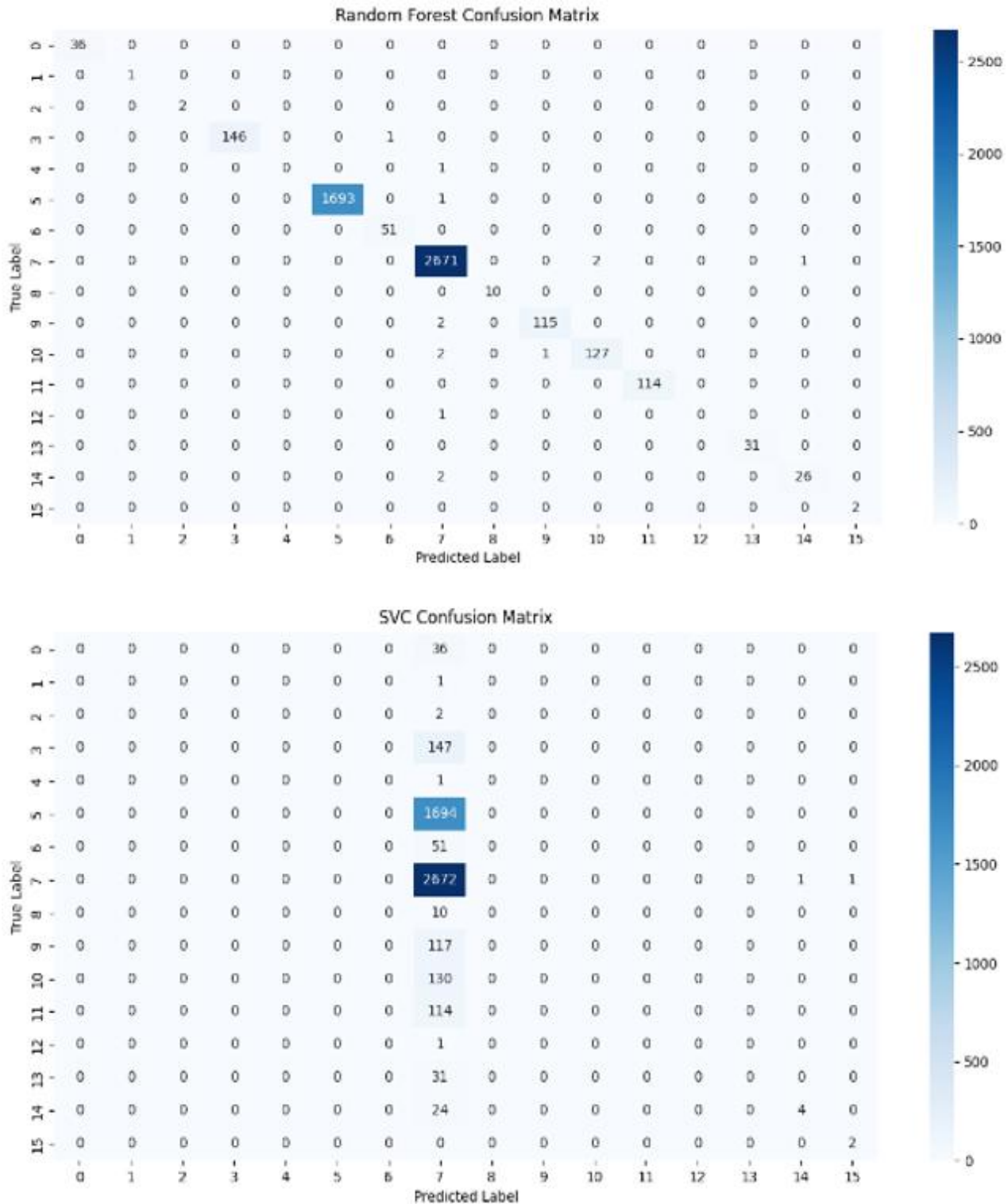


Figure 4. Confusion matrix of proposed model on NSL-KDD dataset

5. CONCLUSIONS

This study highlights the essential requirement for strong cybersecurity in underwater wireless sensor networks (UWSNs) because of their vulnerability to several cyberattacks. The research effectively illustrates the efficacy of machine learning, particularly the Random Forest (RF) classifier, in creating an Intrusion Detection System (IDS) for underwater wireless sensor networks (UWSNs). The proposed Random Forest-based Intrusion Detection System demonstrated exceptional performance on both the WSN-DS and NSL-KDD datasets. In the WSN-DS dataset, the RF model attained an accuracy of 99.66%, with precision at 99.69%, recall at 99.66%, and an F1-score of 99.67%. Upon evaluation using the NSL-KDD dataset, the RF model achieved an accuracy of 1.00, a precision of 0.87, a recall of 0.87, and an F1-score of 0.87. These findings underscore the efficacy of machine learning systems for intrusion detection in underwater wireless sensor networks, crediting their success to resilience against skewed data and the capacity to discern intricate feature associations.

In contrast, the Support Vector Machine (SVM) classifier had markedly inferior performance, with the majority of classes reflecting 0.00 for precision, recall, and F1-score in its classification report. Although SVM demonstrated considerable efficacy with specific classes (11, 20, and 21), its overall accuracy and macro average were subpar, rendering it less appropriate for this classification assignment in comparison to Random Forest. The confusion matrix study indicated that although SVM slightly outperformed RF in categorizing specific classes (3, 5, 7, 9, 10) within the NSL-KDD dataset, Random Forest exhibited greater efficacy in classifying Class 1. Both models encountered difficulties with grades 4, 6, and 12.

In real world, oil pipelines spill detection could be an excellent UWSN application also, Internet under water cables maintenance. Future research may investigate other machine learning models, encompassing deep learning methodologies, as well as the real-time implementation of Intrusion Detection Systems in underwater wireless sensor networks. Moreover, augmenting the datasets to encompass a broader range of attack types and scenarios would enhance the generalization capacity of these models. This study emphasizes the potential of machine learning to improve the security framework of underwater wireless sensor networks, facilitating the development of more secure and dependable underwater communication systems.

REFERENCES

- [1] Mittal, M., De Prado, R.P., Kawai, Y., Nakajima, S., Muñoz-Expósito, J.E. (2021). Machine learning techniques for energy efficiency and anomaly detection in hybrid wireless sensor networks. *Energies*, 14(11): 3125. <https://doi.org/10.3390/en14113125>
- [2] Haseeb, K., Islam, N., Saba, T., Rehman, A., Mehmood, Z. (2020). LSDAR: A light-weight structure based data aggregation routing protocol with secure internet of things integrated next-generation sensor networks. *Sustainable Cities and Society*, 54: 101995. <https://doi.org/10.1016/j.scs.2019.101995>
- [3] Guetari, R., Ayari, H., Sakly, H. (2023). Computer-aided diagnosis systems: A comparative study of classical machine learning versus deep learning-based approaches. *Knowledge and Information Systems*, 65(10): 3881-3921. <https://doi.org/10.1007/s10115-023-01894-7>
- [4] Ramadan, R., Medhat, K. (2022). Intrusion detection based learning in wireless sensor networks. *PLOMS AI*, 2(1): 1-20.
- [5] Wang, W., Jian, S., Tan, Y., Wu, Q., Huang, C. (2022). Representation learning-based network intrusion detection system by capturing explicit and implicit feature interactions. *Computers & Security*, 112: 102537. <https://doi.org/10.1016/j.cose.2021.102537>
- [6] Sadia, H., Farhan, S., Haq, Y.U., Sana, R., Mahmood, T., Bahaj, S.A.O., Rehman, A. (2024). Intrusion detection system for wireless sensor networks: A machine learning based approach. *IEEE Access*, 12: 52565-52582. <https://doi.org/10.1109/ACCESS.2024.3380014>
- [7] Ali, A., Naeem, S., Anam, S., Ahmed, M.M. (2022). Machine learning for intrusion detection in cyber security: Applications, challenges, and recommendations. *UMT Artif. Intell. Rev*, 2(2): 41-64. <https://doi.org/10.32350/icr.22.03>
- [8] Francis, E., Sheeja, S. (2024). An optimized intrusion detection model for wireless sensor networks based on MLP-CatBoost algorithm. *Multimedia Tools and Applications*.
- [9] Ferrag, M.A., Maglaras, L., Moschogiannis, S., Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*, 50: 102419. <https://doi.org/10.1016/j.jisa.2019.102419>
- [10] Ahmad, R., Wazirali, R., Abu-Ain, T. (2022). Machine learning for wireless sensor networks security: An overview of challenges and issues. *Sensors*, 22(13): 4730. <https://doi.org/10.3390/s22134730>
- [11] Mujeeb, S., Alghamdi, T.A., Ullah, S., Fatima, A., Javaid, N., Saba, T. (2019). Exploiting deep learning for wind power forecasting based on big data analytics. *Applied Sciences*, 9(20): 4417. <https://doi.org/10.3390/app9204417>
- [12] Kasongo, S.M., Sun, Y. (2021). A deep gated recurrent unit based model for wireless intrusion detection system. *ICT Express*, 7(1): 81-87. <https://doi.org/10.1016/j.ict.2020.03.002>
- [13] Wajahat, A., He, J., Zhu, N., Mahmood, T., Nazir, A., Ullah, F., Qureshi, S., Dev, S. (2024). Securing Android IoT devices with GuardDroid transparent and lightweight malware detection. *Ain Shams Engineering Journal*, 15(5): 102642. <https://doi.org/10.1016/j.asej.2024.102642>
- [14] Boahen, E.K., Frimpong, S.A., Ujakpa, M.M., Sosu, R.N.A., et al. (2022). A deep multi-architectural approach for online social network intrusion detection system. In 2022 IEEE World Conference on Applied Intelligence and Computing (AIC), Sonbhadra, India, pp. 919-924. <https://doi.org/10.1109/AIC55036.2022.9848865>
- [15] Mahmood, T., Li, J., Saba, T., Rehman, A., Ali, S. (2024). Energy optimized data fusion approach for scalable wireless sensor network using deep learning-based scheme. *Journal of Network and Computer Applications*, 224: 103841. <https://doi.org/10.1016/j.jnca.2024.103841>
- [16] Shaukat, Z., Zulfiqar, A.A., Xiao, C., Azeem, M., Mahmood, T. (2020). Sentiment analysis on IMDB using lexicon and neural networks. *SN Applied Sciences*, 2: 1-

10. <https://doi.org/10.1007/s42452-019-1926-x>
- [17] Mohammed, N.A., Abdulateef, O.F., Hamad, A.H., Abdullah, O.I. (2024). Performance analysis of different machine learning algorithms for predictive maintenance. *Al-Khwarizmi Engineering Journal*, 20(2): 26-38. <https://doi.org/10.22153/kej.2024.11.003>
- [18] Hamad, A.H. (2021). Smart campus monitoring based video surveillance using Haar like features and k-nearest neighbour. *International Journal of Computing and Digital Systems*, 10(1): 863-871. <https://doi.org/10.12785/ijcds/100179>