Smart Security: AI Transformations in Wireless Sensor Networks

Vimla dangi
Computer science and engineering
IET, MLSU,Udaipur
vimupatel2009@gmail.com

Karuna Soni
Pacific institue of technology, PAHER
University, Udaipur
karunasn1@gmail.com

Kartik Joshi
Computer science and engineering
IET, MLSU,Udaipur
kartikjoshi2k2@gmail.com

Abstract: - Wireless Sensor Networks (WSNs) have seen widespread adoption as a key technology for collecting and analyzing environmental data. Their flexibility and diverse applications — including critical military operations, forest fire detection, healthcare monitoring, and civilian usage — have made them a major area of research. The ability of WSNs to deliver valuable insights and tackle real-world challenges has attracted strong interest within the scientific community.

However, WSNs are vulnerable to various security threats, especially in unattended or remote environments. To mitigate these threats, this paper integrates the widely used Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol with machine learning techniques. The study analyzes several machine learning algorithms, considering multiple parameter classes, and evaluates their suitability for WSN environments.

Among the techniques analyzed, the Random Forest algorithm proves to be the most effective for intrusion detection, capable of identifying attacks such as blackhole, grayhole, TDMA, and flooding attacks. This method uses six key extracted features and achieves an overall accuracy of 99.12%, with individual attack prediction accuracies of 87%, 89%, 92%, and 93%, respectively.

Rather than merely detecting anomalies, the integration of this system with a centralized management model transforms it into a smart governance solution. This approach not only prevents various types of intrusions but also enhances crucial network parameters such as energy efficiency, network lifetime, throughput, overall performance, and security.

Keywords— LEACH, WSN, Logistic Regression, KNN, SVM, Random Forest, Decision Tree, XGBoost, DoS.

I. Introduction

Wireless Sensor Networks (WSNs) have become a major focus of research due to their wide range of real-time applications. The design of WSNs is shaped by key factors such as scalability, fault tolerance, and energy efficiency. These networks consist of numerous autonomous sensor nodes strategically deployed across areas of interest. The nodes gather vital data and collaboratively transmit it wirelessly to a central node, often referred to as the sink node or base station.

The transmission and management of data within WSNs are largely governed by specialized network protocols. One widely adopted architecture is the clustered model, supported by the Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol at the network layer. LEACH is a hierarchical protocol that assumes each node has a radio capable of reaching either the base station or the nearest cluster head. However, continuous full-power transmission leads to unnecessary energy consumption.

To address this, LEACH uses Time Division Multiple Access (TDMA) for communication among non-cluster-head nodes, while clusters utilize Code Division Multiple Access (CDMA) to reduce interference. Despite these advantages, WSNs remain vulnerable to various security threats due to their open and distributed nature, as well as the limited computational and energy resources of sensor nodes. Protecting WSNs from such attacks presents considerable challenges given these constraints and inherent vulnerabilitie

.

II. Related Work

In response to the increasing security challenges faced by Wireless Sensor Networks (WSNs), several recent studies have explored energy efficiency, adaptive routing, and machine learning-based intrusion detection. Building upon this body of work, the proposed study aims to develop a comprehensive, intelligent security framework for WSNs that not only detects a wide range of network attacks but also ensures energy efficiency and prolonged network lifetime.

Previous research [1] has introduced reinforcement learning (RL) protocols to optimize energy consumption and network longevity. Their adaptive routing framework dynamically accommodates changes in network conditions, including nodal mobility and energy depletion, by distributing energy usage more evenly. The RL-based approach demonstrated superior performance over traditional energy-efficient protocols such as Q-Learning and LARCMS, based on key performance metrics like packet delivery rate and end-to-end reliability.

In parallel, various studies [2] have surveyed different types of attacks targeting distinct layers of WSN architecture and assessed how different machine learning (ML) techniques can be employed to counteract them. This includes a comparative tabulation of known attack types with applicable ML solutions, providing a broad landscape of existing defensive strategies.

To address specific Quality of Service (QoS) needs, another work [3] proposed LEACH-APP — an enhancement of the LEACH protocol tailored to application-specific requirements. LEACH-APP showed significant improvements in throughput and latency reduction, highlighting the potential of protocol-level customization for performance optimization.

Furthermore, a detailed study [4] developed a customized WSN dataset, WSN-DS, using NS-2 simulations to mimic real-world network behaviors and attack scenarios. With 23 extracted features, the dataset served as the basis for training various Artificial Neural Network (ANN) architectures. The study demonstrated that a single-hidden-layer ANN, validated through 10-Fold Cross Validation using the WEKA toolkit, achieved high classification accuracies for multiple attack types: 92.8% for Blackhole, 99.4% for Flooding, 92.2% for Scheduling, 75.6% for Grayhole, and 99.8% for normal traffic.

Building on these insights, the proposed work introduces a novel security model that integrates the LEACH protocol with a machine learning-based intrusion detection mechanism, specifically using the Random Forest algorithm. This approach not only identifies attacks such as Blackhole, Grayhole, TDMA, and Flooding with high accuracy but also incorporates a centralized management system to implement smart governance. The model aims to enhance key WSN performance metrics — including energy efficiency, throughput, and overall network resilience — by intelligently managing both routing and security responses.

This integrated system aspires to offer a dual benefit: mitigating intrusion risks while maintaining the core operational integrity of the network. Future extensions may involve incorporating deep learning methods, expanding the dataset with more complex attack types, and testing the framework under real-world deployment scenarios to further validate its adaptability and robustness.

III. Ease of Use

The proposed work collects environmental data from the network which is treated to obtain the components of an intrusion detection system, such as feature extraction and modelling algorithm. Based on the obtained best system we deploy the machine learning model. Once it is done, when the parameters chosen through feature extraction from the network are given the model detects and classifies the type of attack. This methodology will prevent intrusion instances to a larger extent and enhance the characteristics of a wireless sensor network.

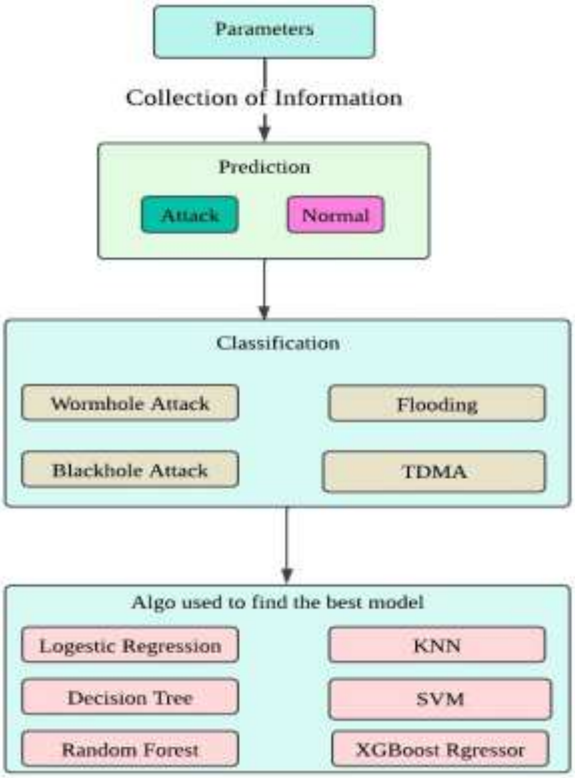

Fig. 1. Block diagram of the proposed model

This model aims to pick the optimized and most accurate solution for an intrusion instance. Through progressive selection of machine learning algorithms used for classification of top 6 algorithms were chosen and used for this work.

a) Logistic Regression: It is a supervised learning algorithm that utilizes a more sophisticated cost function known as Sigmoid or Logit function. It predicts a binary outcome either if something happens or doesn't not happen by analyzing the relationship between variables. It is efficient to train and leads to overfitting if the number of features is more than the number of observations.

b) K-Nearest Neighbors: It is a non-parametric algorithm that relies on measuring distances to determine the category of an unknown entity. It identifies the k nearest neighbors to the target instance and assigns it to the category that appears most frequently among those neighbors. KNN can be particularly effective when the training data is large. However, determining the optimal value of k, which represents the number of neighbors to consider, can be a time-consuming process. Finding the right balance for k is crucial to ensure accurate classification results.

c) Support Vector Machine: It is a model that represents different classes by constructing a hyperplane in a multidimensional space. SVM performs well in cases where the dataset is not excessively large, but it is known to be less effective on big datasets. However, one advantage of SVM is its robustness against outliers, as it is not highly sensitive to their presence. SVM aims to find the optimal hyperplane that maximally separates different classes, leading to effective classification performance.

d) Decision Tree: It is a supervised learning technique that organizes data into precise classes by recursively splitting it into similar categories. It follows a flowchart-like structure, starting from the trunk and branching out to leaves, where categories become increasingly specific. While decision trees consider all possible outcomes, they can face overfitting issues. To address this, the Random Forest algorithm combines multiple decision trees, mitigating overfitting and improving generalization.

e) Random Forest: It is an ensemble learning method that offers a solution to complex problems. Multiple decision trees that have been trained on various subsets of the dataset are what make up this classification system, which serves as a classifier. Random Forest is able to improve the accuracy of predictions by generating an average of the predictions made by these trees. It effectively addresses the overfitting problem encountered in individual decision trees. However, it should be noted that Random Forest may require additional training time due to its ensemble nature.

f) XGBOOST Regressor: Extreme Gradient Boost is a machine learning library that offers scalable and distributed gradient-boosted decision tree algorithms. It employs a sequential approach to build decision trees, boosting their performance through parallel tree boosting. XGBoost is known for its strong performance, scalability, and interpretability, as well as its ability to handle missing values in data. However, it's important to note that XGBoost's computational complexity can be high, and in certain cases, it may be prone to overfitting.

IV. RESULTS AND DISCUSSION

The research was conducted using Jupyter Notebook, and multiple sets of results were generated and analyzed.

A. Data Pre-processing

Data pre-processing plays a pivotal role in the development of a robust machine learning model, serving as the initial and essential phase in transforming raw data into a format that aligns seamlessly with the model's requirements. This crucial step lays the foundation for subsequent stages in the machine learning pipeline, ensuring that the input data is appropriately organized and refined for optimal model performance.

TABLE I. DATA DESCRIPTION

	count	mean	std	min	25%	50%	75%
id	374661.0	274969.325879	389898.554898	101000.0	107093.00000	116071.00000	215072.00000
Time	374661.0	1064.748712	899.646164	50.0	353.00000	803.00000	1503.00000
Is_CH	374661.0	0.115766	0.319945	0.0	0.00000	0.00000	0.00000
who CH	374661.0	274980.411108	389911.221734	101000.0	107096.00000	116072.00000	215073.00000
Dist_To_CH	374661.0	22.599380	21.955794	0.0	4.73544	18.37261	33.77600
ADV_S	374661.0	0.267698	2.061148	0.0	0.00000	0.00000	0.00000
JOIN_S	374661.0	0.779905	0.414311	0.0	1.00000	1.00000	1.00000
SCH_S	374661.0	0.288984	2.754746	0.0	0.00000	0.00000	0.00000
Rank	374661.0	9.687104	14.681901	0.0	1.00000	3.00000	13.00000
DATA_S	374661.0	44.857925	42.574464	0.0	13.00000	35.00000	62.00000
dist_CH_To_BS	374661.0	22.562735	50.261604	0.0	0.00000	0.00000	0.00000
send_code	374661.0	2.497957	2.407337	0.0	1.00000	2.00000	4.00000
Expaned_Energy	374661.0	0.305661	0.669462	0.0	0.05615	0.09797	0.21776

B. Evaluation Criterion

When it comes to classification, the major goal of parameter grouping is to carefully investigate and achieve the maximum possible accuracy while making use of the ideal amount of parameters that are required for efficient classification. In order to ensure that the chosen parameters make a significant contribution to the classification job without causing needless duplication or overcomplication, this strategic approach seeks to establish a balance between the complexity of the model and its performance. The goal is to develop an ideal configuration that promotes accuracy while also increasing efficiency in the classification process. This will be accomplished by grouping parameters in a reasonable manner.

By delineating parameter groups inside the architectural framework of our network research, we are able to methodically discover and explain the complexities of network dynamics. During the current cycle, the first set, which consists of FOUR PARAMETERS, includes crucial structural insights. The determination of whether or not a node occupies the crucial position of a cluster head, the identification of the cluster head that corresponds to the node, and the distances between the cluster head, the node, and the base station are all factors that contribute to these insights. This foundation is the basis upon which the SIX PARAMETERS grouping is developed as we go forward. It not only provides core network statistics, but it also incorporates crucial metrics such as the count of data packets that have been received from the cluster head and transferred to the base station during the present cycle.

The third grouping, which is referred to be EIGHT PARAMETERS, broadens the scope of the analysis by include particulars such as the number of channel advertising messages that were received from the cluster head and the cluster sending code. This set is a supplement to the information that was obtained from the classes of six parameters that came before it. These classes included fundamental network information as well as insights into the packet transfer database. There are fifteen parameters that are involved in further investigation. These parameters include all of the factors that are associated with the network, with the exception of the expanded energy quantity, which can only be quantified after the network has been evaluated. In the current round, the major objective is to determine whether or not this comprehensive set is capable of maximizing the accuracy of forecasting the sorts of attacks for that round. Last but not least, the SIXTEEN PARAMETERS class incorporates the extended energy component. The purpose of this class is to cross-verify and investigate any differences in the accuracy of predicting attack types when compared to the class of 15 parameters.

C. Experimental Analysis

Experiment-1: A performance comparison of six machine learning algorithms, with a focus on their accuracy metrics. In the experiment, only four features were used to train the models. While the Random Forest algorithm outperforms others with the highest accuracy of 0.9511, the indication that the feature count is not optimal suggests that the model's predictive power could potentially be improved. This implies that the current feature set may be too limited to capture the complexity of the underlying data fully. By exploring more features or employing feature engineering techniques, there could be an opportunity to enhance the model's accuracy further, as some algorithms can benefit from a richer feature set to detect more subtle patterns in the data.

TABLE II. ACCURACY REPORT EXPERIMENT-1

Algorithm	Accuracy
Random Forest	0.9854
Support Vector Machine	0.9662
Decision Tree	0.9636
Logistic Regression	0.9535
K-Nearest Neighbor	0.9521
Xgboost Regressor	0.9421

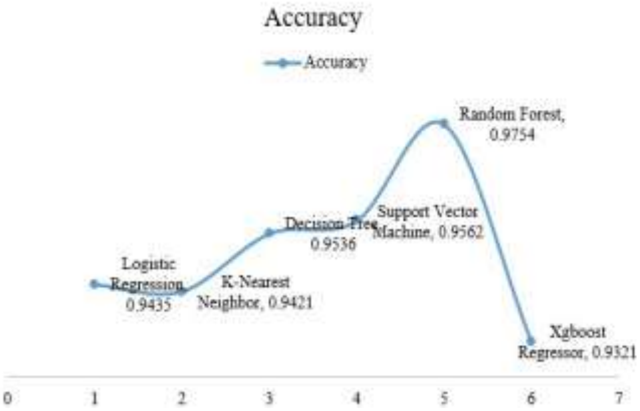

Fig. 2. Graphically representation accuracy report Experiment-1

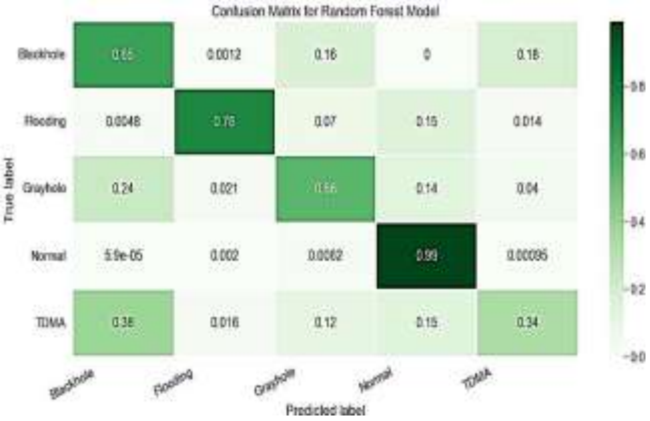

Fig. 3. Confusion matrix of random forest for 4 parameters

Experiment-2: In below table listing the accuracy of six different machine learning algorithms, suggesting an experiment designed to evaluate their performance on a dataset with six extracted features. Random Forest emerges as the top-performing algorithm with an accuracy of 0.9845. Despite this high accuracy, the commentary on the number of features not being optimal implies that the current feature set may not fully capture the complexity of the data or may include redundant or irrelevant features. This indicates that there may be potential to either refine the feature selection further or to engineer additional features that could lead to improved model performance. Fine-tuning the feature set could help other algorithms achieve better accuracy or even lead to a new best-performing algorithm.

TABLE III. ACCURACY REPORT OF EXPERIMENT-2

Algorithm	Accuracy
Random Forest	0.9945
Support Vector Machine	0.9649
K-Nearest Neighbor	0.9749
Decision Tree	0.9841
Xgboost Regressor	0.9658
Logistic Regression	0.9521

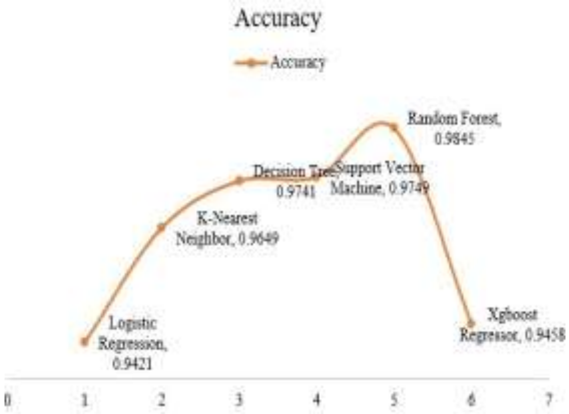

Fig. 4. Graphically representation accuracy report Experiment-2

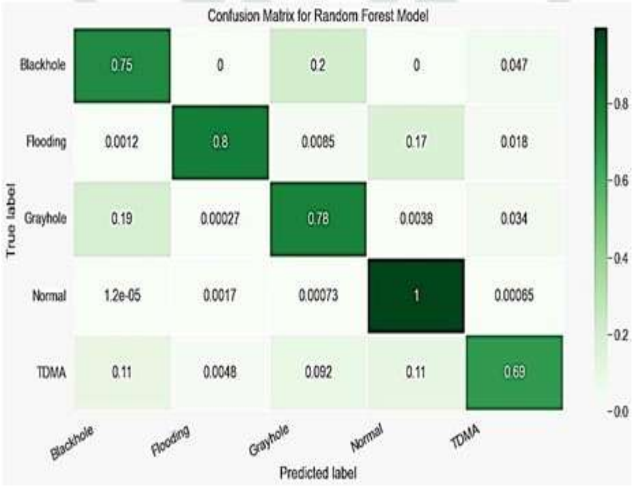

Fig. 5. Confusion matrix of random forest for 6 parameters

Experiment-3: In below table listing the accuracy of six different machine learning algorithms, suggesting an experiment designed to evaluate their performance on a dataset with six extracted features. Random Forest emerges as the top-performing algorithm with an accuracy of 0.9845. Despite this high accuracy, the commentary on the number of features not being optimal implies that the current feature set may not fully capture the complexity of the data or may include redundant or irrelevant features. This indicates that there may be potential to either refine the feature selection further or to engineer additional features that could lead to improved model performance. Fine-tuning the feature set could help other algorithms achieve better accuracy or even lead to a new best-performing algorithm.

TABLE IV. ACCURACY REPORT OF EXPERIMENT-3

Algorithm	Accuracy
Xgboost Regressor	0.9986
Random Forest	0.9966
Support Vector Machine	0.9884
Decision Tree	0.9784
K-Nearest Neighbor	0.9774
Logistic Regression	0.9424

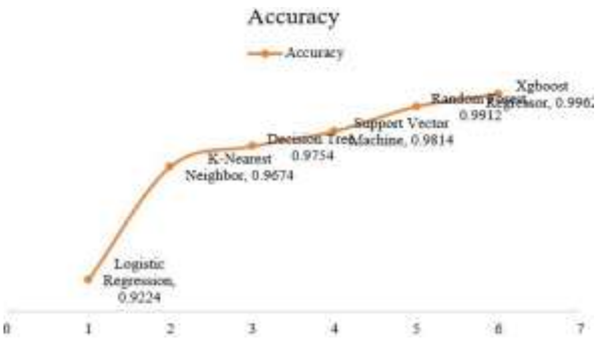

Fig. 6. Graphically representation accuracy report Experiment-3

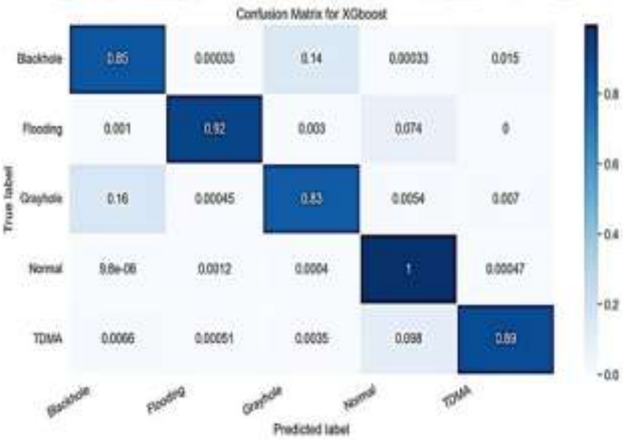

Fig. 7. Confusion matrix of xgboost for 8 parameters

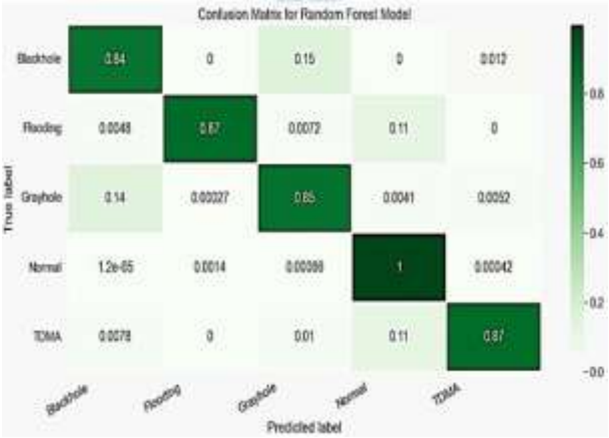

Fig. 8. Confusion matrix of random forest for 8 parameters

Experiment-4: In below table detailing the accuracy scores of six machine learning algorithms tested with a set of four extracted features. The Random Forest algorithm tops the list with an accuracy of 0.9925, closely followed by the XGBoost Regressor at 0.9949. The statement about the number of features not being optimal suggests that despite the high accuracy scores, there might be room for improvement. This could mean that the current features may not capture all the nuances of the underlying dataset, or that some features may be redundant or not entirely relevant for the predictive task. Optimizing the feature set could involve adding more informative features, removing irrelevant ones, or creating new features through transformations, with the goal of further improving the model's performance.

TABLE V. ACCURACY REPORT OF EXPERIMENT-4

Algorithm	Accuracy
Xgboost Regressor	0.9989
Random Forest	0.9985
Support Vector Machine	0.9982
Decision Tree	0.9871
K-Nearest Neighbor	0.9797
Logistic Regression	0.9721

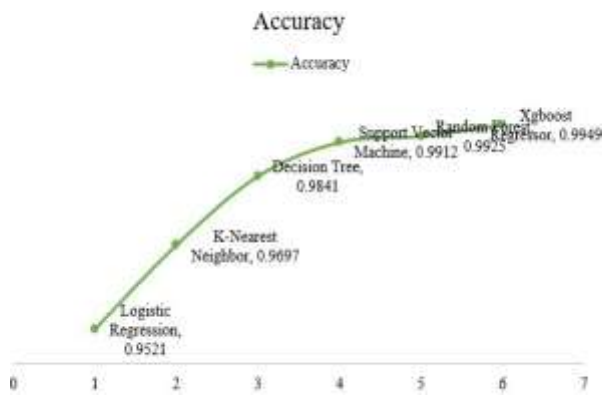

Fig. 9. Graphically representation accuracy report Experiment-4

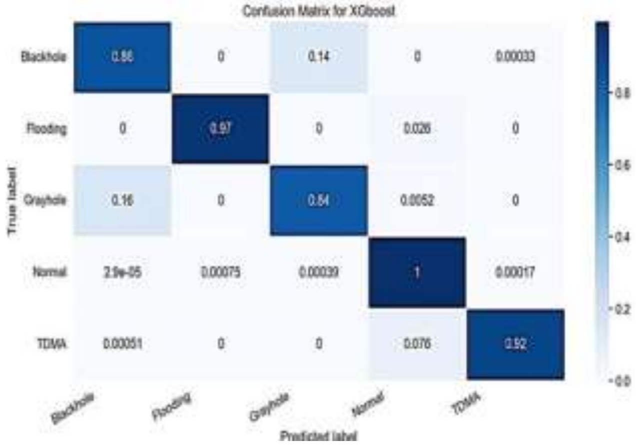

Fig. 10. Confusion matrix of xgboost for 15 parameters

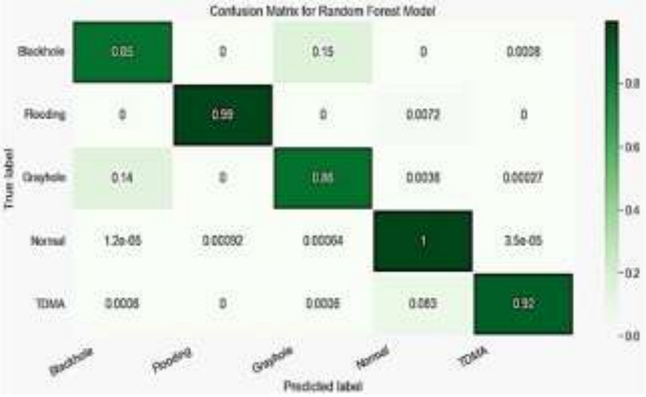

Fig. 11. Confusion matrix of random forest for 15 parameters

Experiment-5: In below table table summarizing the accuracy of six machine learning algorithms when applied to a dataset with four extracted features. According to the table, the Random Forest algorithm outperforms the others with an accuracy of 0.9847, closely followed by the XGBoost Regressor at 0.9947. Despite the high accuracy reported, the comment that the number of features may not be optimal suggests that the predictive capability could be limited by the current feature set. This implies that the inclusion of additional features, or the refinement of existing ones, might yield a more nuanced model, potentially increasing accuracy and the ability to generalize. It also raises the possibility that some algorithms could benefit from a more complex feature set, while others might need feature reduction to prevent overfitting and enhance performance.

TABLE VI. ACCURACY REPORT OF EXPERIMENT-5

Algorithm	Accuracy
Xgboost Regressor	0.9977
Random Forest	0.9887
Support Vector Machine	0.9885
K-Nearest Neighbor	0.9784
Decision Tree	0.9785
Logistic Regression	0.9621

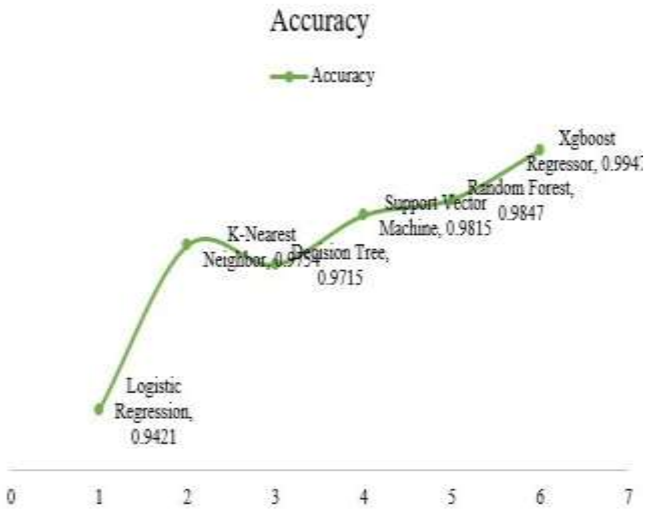

Fig. 12. Graphically representation accuracy report Experiment-5

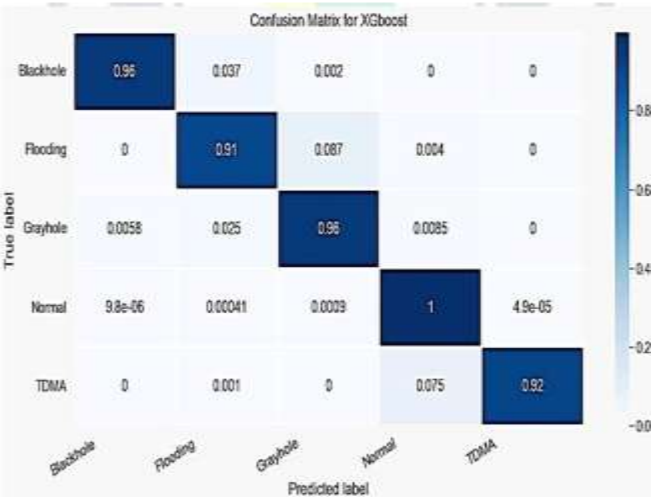

Fig. 13. Confusion matrix of xgboost for 16 parameters

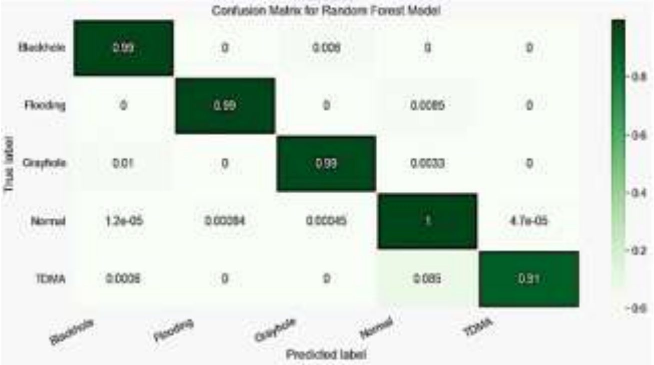

Fig. 14. Confusion matrix of random forest for 16 parameters

D. Best-fit Algorithm

Typically, the model that gets the greatest accuracy or the lowest error on this dataset is the one that is considered to be the "best fit" model. But what really defines the "best fit" might vary depending on a number of different aspects, such as the application and the intended trade-offs between the needs for speed, accuracy, and resources. On the basis of the experiment analysis that was carried out for several classes of grouped parameters and taking into consideration their accuracy, precision, recall, and confusion matrix, we are able to determine that the eight parameters class is the most ideal. It is possible for us to draw the conclusion that, among the two regression models and four classification models with eight parameters class, RANDOM FOREST is the most suitable option that is capable of meeting the objective that we have established for the intrusion detection model. Based on the data presented above, it is possible to see that the accuracy scores of XGBOOST and RANDOM FOREST are almost identical to one another. When additional aspects of the models are taken into consideration, the training score provided by XGBOOST is lower than that provided by Random Forest. In addition, since the amount of power and time required for calculation is more in the XGBOOST Regressor, we choose "RANDOM FOREST" for intrusion detection rather than XGBOOST.

TABLE VII. EVALUATION OF VARIOUS ALGORITHMS AND THEIR CORRESPONDING ACCURACY SCORES

ALGORITHM	**ACCURACY**				
Parameters	**4**	**6**	**8**	**15**	**16**
Logistic regression	0.9535	0.9521	0.9324	0.9581	0.9481
K-nearest neighbor	0.9521	0.9749	0.9674	0.9797	0.9754
Decision tree	0.9636	0.9841	0.9754	0.9841	0.9815
Support vector machine	0.9762	0.9849	0.9874	0.9912	0.9815
Random forest	0.9854	0.9945	0.9912	0.9925	0.9847
Xgboost regressor	0.9421	0.9558	0.9962	0.9949	0.9947

V. CONCLUSION AND FUTURE SCOPE

With the purpose of determining which model is the most successful for autonomous governance, this study endeavors to conduct an exhaustive assessment of a variety of machine learning models, each of which is trained on a different collection of extracted data. A strong, self-regulating system that serves as both a central command entity and an intelligent intrusion detection framework is the end product of this process. While concurrently increasing essential network qualities such as lifetime, throughput, overall performance, security, and energy efficiency, its deployment in wireless sensor networks considerably strengthens the network's resistance against a variety of assaults that target the network layer. Among the many advantages that the suggested methodology has, one of its most important advantages is its adaptability to a broad variety of protocols, which allows it to be scaled up. Modifying certain parameters inside the algorithmic code in order to cater to particular requirements is the means by which this flexibility is accomplished. In addition, the process of monitoring network nodes might be simplified by incorporating the credentials of the devices that are part of the wireless sensor network into an interface that is linked to the cloud. It is possible to reduce the amount of computational pressure placed on the central controlling interface by putting the machine learning model onto the cloud. This will ensure that the system operates well. This system is meant to be future-proof, meaning that it can be upgraded and deployed in a seamless manner to accommodate the ever-increasing and ever-changing requirements of technology ecosystems. This is because technological breakthroughs are always evolving, and the system is built to be future-proof.

REFERENCES

[1] Simon, J. (2022). An Energy Efficient Routing Protocol based on Reinforcement Learning for WSN. IRO Journal on Sustainable Wireless Systems, 4(2), 79-89. doi:10.36548/jsws.2022.2.002.

[2] R. Kaur and J. Kaur Sandhu, "A Study on Security Attacks in Wireless Sensor Network," 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2021, pp. 850-855, doi: 10.1109/ICACITE51222.2021.9404619.

[3] Alba Rozas, Alvaro Araujo, "An Application-Aware Clustering Protocol for Wireless Sensor Networks to Provide QoS Management", Journal of Sensors, vol. 2019, Article ID 8569326, 11 pages, 2019. https://doi.org/10.1155/2019/8569326.

[4] Iman Almomani, Bassam Al-Kasasbeh, Mousa AL-Akhras, "WSN-DS: A Dataset for Intrusion Detection Systems in Wireless Sensor Networks", Journal of Sensors, vol. 2016, Article ID 4731953, 16 pages, 2016. https://doi.org/10.1155/2016/4731953.

[5] A. Alsadhan and N. Khan, (2013) "A proposed optimized and efficient intrusion detection system for wireless sensor network," International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering, vol. 7, no. 12, pp. 1621–1624.

[6] A. Braman and G. R. Umapathi, (2014) "A comparative study on advances in LEACH Routing protocol for wireless sensor networks: a survey," International Journal of Advanced Research in Computer and Communication Engineering, vol. 3, no. 2, pp.5883–5890.

[7] Bendigeri KY, Mallapur JD, Kumbalavati SB (2021) Direction Based Node Placement in Wireless Sensor Network. In 2021, International Conference on Artificial Intelligence and Smart Systems (ICAIS), pp 1306–1313.

[8] I. Butun, S. D. Morgera, and R. Sankar, (2014) "A survey of intrusion detection systems in wireless sensor networks," IEEE Communications Surveys & Tutorials, vol. 16, no. 1, pp. 266–282.

[9] M. Tripathi, M. S. Gaur, and V. Laxmi, (2013) "Comparing the impact of black hole and gray hole attack on LEACH in WSN," Procedia Computer Science, vol. 19, pp. 1101–110.

[10] S. Khan and K.-K. Loo, (2009) "Real-time cross-layer design for a largescale flood detection and attack trace-back mechanism in IEEE 802.11 wireless mesh networks," Network Security, vol. 2009, no.5, pp. 9–16.

[11] A. Kumar, S. K. Pandey, S. Prakash, K. U. Singh, T. Singh and G.Kumar, "Enhancing Web Application Efficiency: Exploring Modern Design Patterns within the MVC Framework," 2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), Greater Noida, India, 2023, pp. 43-48, doi: 10.1109/CISES58720.2023.10183582.

[12] G. Kumar, S. K. Pandey, D. P. Yadav, K. U. Singh, T. Singh and A. Kumar, "Machine Learning based model to Detect Anomaly in the Water," 2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), Greater Noida, India, 2023, pp. 49-54, doi: 10.1109/CISES58720.2023.10183555.

[13] A. Chauhan, I. Kumar, K. U. Singh and P. Singh, "Deep Architecturefor Breast Cancer Detection: Using Optimized VGG16 Model and Transfer Learning," 2023 International Conference on Sustainable Emerging Innovations in Engineering and Technology (ICSEIET), Ghaziabad, India, 2023, pp. 853-858, doi 10.1109/ICSEIET58677.2023.10303623.

[14] G. Kumar, S. K. Pandey, N. Varshney, R. R. Janghel, K. U. Singh and A. Kumar, "Arrhythmia Detection from ECG Signals using CNN Model," 2023 6th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2023, pp. 1-6, doi: 10.1109/ISCON57294.2023.10112173.