Optimized Performance of Attacks Detection in WSN based on Machine Learning Algorithms

Amr A. Farid[†], Abeer T. Khalil[†], Mahmoud A. Shawky[‡]
Ahmed S. Samrah[†], and Hegazi Ibrahim[§]
[†] Faculty of Engineering, Mansoura University, Mansoura, Egypt
[‡] James Watt School of Engineering, University of Glasgow, UK
[§]Nile Higher Institute of Engineering and Technology, Mansoura, Egypt
Email: Merofarid31@gmail.com, abeer.twakol@mans.edu.eg, m.shawky.1@research.gla.ac.uk,
shmed@mans.edu.eg, hegazibrahim@nilehi.edu.eg

***Abstract*—Wireless sensor networks (WSN) are a type of wireless network composed of numerous sensors that collaborate to sense, collect, process, and transmit information about the physical environment within the network's geographical area. The information is ultimately received by the network owner. However, typical attacks such as Blackhole, Grayhole, Flooding, and Scheduling can pose a significant threat to the WSN, potentially causing significant damage to the system in a short period. Detection methods, such as snooping, have demonstrated low detection and high false alarm rates, and require significant computational resources. Additionally, they tend to produce redundant network data. To address these limitations, we propose a novel intervention approach called "Ensemble Bagged Trees," which employs a squared backward sequence selection (SBS) algorithm to reduce data dimensionality and computational overhead in the feature space of native traffic data. The Ensemble Bagged Trees algorithm is then utilized to detect various network attacks. Experimental results using the WSN-DS dataset demonstrate that the proposed method outperforms typical machine learning detection algorithms, with a detection rate of 99.1% for the normal black hole, gray hole, flood, and tabulation attacks.**

***Index Terms*—Denial-of-service attacks, Machine learning, Squared backward sequence selection, Wireless sensor networks.**

I. Introduction

The emergence of wireless sensor networks (WSNs) has opened up new opportunities for collecting data in various applications such as environmental monitoring, healthcare, and industrial automation. However, WSNs are often deployed in remote and harsh environments where energy resources are limited, and security is a major concern. This has led to several challenges such as limited energy, security, and data aggregation and fusion. Moreover, the growing number of cyber-attacks such as denial-of-service (DoS) attacks poses a significant threat to the performance and reliability of WSNs [1]. All of these challenges can be minimized using, artificial intelligence (AI) and machine learning (ML)-based algorithms that have emerged as promising solutions to improve the overall system performance of wireless sensors. In particular, ML algorithms can be used to optimize energy consumption, enhance security, and improve data collection and consolidation in WSNs [2]. Furthermore, ML algorithms can help in detecting and mitigating DoS attacks in WSNs. Additionally, WSNs generate massive amounts of data that require aggregation and fusion to extract meaningful information, which can be challenging and resource-intensive [3]. Finally, deploying and maintaining a WSN in remote or hostile environments requires careful planning and management to ensure the network's longevity and reliability. Breaches in security can lead to severe consequences, including privacy violations and data leakages [4], [5]. Major security threats and challenges include data confidentiality, data integrity, authentication, DoS attacks, and node compromise.

WSN are vulnerable to both passive and active attacks. Active attacks are characterized by their disruptive nature, directly impacting the system through actions that hinder normal operation. In contrast, passive attacks compromise data confidentiality by allowing attackers to obtain information transmitted from the original station to the destination station without disrupting the data communication process. Although data leakage resulting from passive attacks does not directly affect data transmission, it poses a significant risk to the security of the network. Active attacks exploit the broadcast nature of wireless communication media, making them a serious threat to the integrity of the system. In this challenging scenario, this paper aims to provide a comprehensive solution to clarify the role of AI and ML algorithms in improving the performance of wireless sensors, particularly in addressing the challenges of limited energy, security, and data aggregation and fusion, also the use of AI and ML algorithms has been highlighted in mitigating the impact of DoS attacks on WSNs.

The following summarises the contributions of this paper.

- This study involved a comprehensive evaluation and comparison of multiple ML techniques for detecting attacks. The aim was to identify the optimal ML technique that would enable efficient and accurate detection of attacks.
- To assess the accuracy of detections, we measured the performance of twenty distinct ML algorithms on five

distinct attack types.

The paper is structured as follows: Section II provides a review of recently published literature, Section III introduces the proposed method, Section IV presents the performance analysis, and finally, Section V provides concluding remarks.

II. Related Works

Several studies have been conducted on improving the performance and security of WSNs through ML algorithms. Kundu et al. [6] proposed a gravitational search algorithm (GSA)-based algorithm for optimizing WSN deployment to ensure full coverage of the target area while minimizing the number of sensors required. Their approach outperforms other state-of-the-art algorithms in terms of coverage rate, connectivity rate, and the number of sensors required. Zaimen et al. [7] proposed a novel framework that employs transfer learning techniques to improve the performance of WSNs. The framework utilizes machine learning algorithms to reduce energy consumption and enhance the accuracy of classification on real-world datasets.

ML techniques can also be used to address the major challenges in machine learning software-defined wireless sensor networks (ML-SDWSNs) as proposed by Fernando et al. [8]. Their survey highlights the benefits of using ML in SDWSNs, including improved energy efficiency and increased network lifetime. However, the limited processing power and memory capacity pose challenges to implementing ML in SDWSNs. The authors suggested potential solutions to these challenges, emphasizing the importance of ML-SDWSNs in advancing the Internet of Things (IoT) and enabling smart applications. Intrusion detection systems (IDSs) can be developed for WSNs using ML techniques. Jamalipour et al. [9] presented a taxonomy of machine-learning-based IDSs for IoT, which classifies IDSs based on their type of ML algorithm and input data source. Kalnoo et al. [10] proposed a new model using hidden Markov model (HMM) to combat distributed denial of service (DDoS) attacks that violate privacy rules, achieving a 97% accuracy rate when detecting potential anomalies that indicate attacks. Gowdhaman et al. [11] proposed an IDS based on a deep neural network (DNN) approach that can detect and prevent various types of attacks, showing its effectiveness and efficiency in simulation experiments.

ML techniques can also be integrated with encryption and clustering to achieve energy-efficient and secure transmission in WSNs as proposed by Kumar et al. [12]. Sarkar et al. [13] presented a solution for efficient network intrusion detection using supervised machine learning and ensemble learning based on hyper parameter optimization. The proposed approach combines multiple classifiers and optimizes their hyperparameters to enhance the accuracy and efficiency of network intrusion detection. Finally, Waqas et al. [14] discussed the role of AI and ML in wireless network security, outlining their principles and practices in network security and examining the challenges of implementing these technologies. The paper also identifies the potential benefits of using AI and ML for wireless network security and highlights the areas where further research is needed. Sharma et al. [15] proposed a novel IDS for cyber-physical systems (CPS) that use the routing protocol for low-power and lossy networks (RPL). The proposed IDS is designed to detect and prevent various types of attacks on CPS, including DoS and tampering attacks while minimizing false alarms. In conclusion, ML techniques have shown promising results in enhancing the security of WSNs by providing efficient intrusion detection and prevention mechanisms.

III. System and Method Modelling

This section presents the system model and introduces the proposed method in detail.

A. System model

WSNs depend on intrusion detection technology to ensure the security of the system. This technology comprises three fundamental steps, namely information collection, detection model, and response module. The information collection unit is responsible for acquiring data from the surrounding environment and forwarding it to the detection unit. The analyzer present in the detection module scrutinizes the collected traffic data to detect any interference in the WSNs. In case of any anomaly detection, the response unit processes it promptly and appropriately. Fig. 1 shows the intrusion detection mode of the WSN, consisting of the sensor node (SN), the cluster head (CH), and the sink or base station node. The distributed detection mode implemented in this system helps distribute power across the network and reduces communication overheads. (WSNs), the cluster master assumes the responsibility of coordinating computing tasks across the cluster on a global level. Meanwhile, regular sensor nodes process the data to mitigate computing costs and communication overheads. In the realm of intrusion detection, numerous researchers have leveraged complex data mining algorithms to achieve exceptional detection and accuracy in WSNs. Nevertheless, the practical application and real-time deployment of these algorithms in WSN are constrained by their high computational overhead. This drawback primarily stems from the input data's high feature dimensions, abundance of redundant data, and inadequately processed data. To address these challenges, the system employs a feature selection technique called squared backward sequence selection (SBS). SBS enables the elimination of irrelevant and redundant features, thereby selecting the most optimal subset of features. By doing so, it reduces information loss, enhances the detection rate of the classification algorithm, and diminishes the computational overhead of the intrusion detection system. Additionally, the system employs the Ensemble Bagged Trees algorithm for intrusion detection purposes. This method is applied to the data analysis module in Fig. 1, where it receives event information and analyzes it to determine intrusion behavior. The algorithm flowchart is presented in Fig. 2.

B. The proposed method

This subsection describes the proposed scheme through the following Stages.

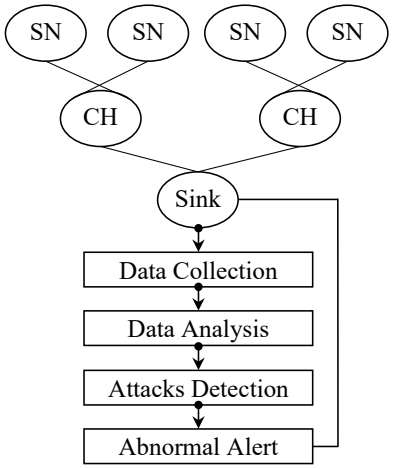

Fig. 1: Workflow of intrusion detection scheme for WSNs [3].

1) Stage 1 (Data modeling): The wireless sensor network data can be obtained through the utilization of the Network Simulator-2 (NS-2) software. In this network, the routing protocol employed is the low energy adaptive clustering hierarchy (LEACH), which operates by selecting cluster head nodes in a random, circular fashion. The LEACH protocol also ensures the fair allocation of energy load to each sensor node, thus reducing the overall network energy consumption. The status of each node in the network is then determined by analyzing normal and DoS attacks [3]. Moreover, a comprehensive analysis of current IDSs has been implemented and compared based on the classification of IDSs, such as anomaly-based, signature-based, hybrid, or cross-layer [16]. The proposed methodology based on latch protocol has been assigned based on the following steps:

- *Step 1*: The TDMA schedule assigns each node $s \in S$ to a specific time slot $t(c, s) \in T$ in the TDMA schedule of its cluster $c \in C$:

$$T(C, S) = \mathrm{k} \times \mathrm{T} + \mathrm{S} \tag{1}$$

  where k is a positive integer and T is the duration of each time slot in the TDMA schedule [17].
- *Step 2*: At each time slot $t \in T$, a node $s \in S$ can transmit its data if and only if it is a member of a cluster $c \in C$ and the current time slot corresponds to its assigned time slot in the TDMA schedule:

$$T = t(c, s) \tag{2}$$

- *Step 3*: If the node s has data to transmit in its assigned time slot, it can transmit the data to its cluster head using a transmission probability $p(t)$:

$$p(t) = f(E(s)) \tag{3}$$

  where $E(s)$ is the remaining energy of the node s and f is a decreasing function of energy, such that higher energy nodes have a higher probability of transmitting their data [17].
- *Step 4*: The cluster head $ch(c)$ acknowledges receipt of the data by sending an ACK message to the node s. If the ACK is not received by the node s within a specified time interval, the node will attempt to retransmit the data in a subsequent time slot.
- After transmitting its data, the node s can enter a low-power sleep mode until its next assigned time slot. By using the TDMA schedule and transmission probabilities based on node energy, the LATCH protocol helps to optimize energy consumption and ensure reliable data transmission in WSNs [17].

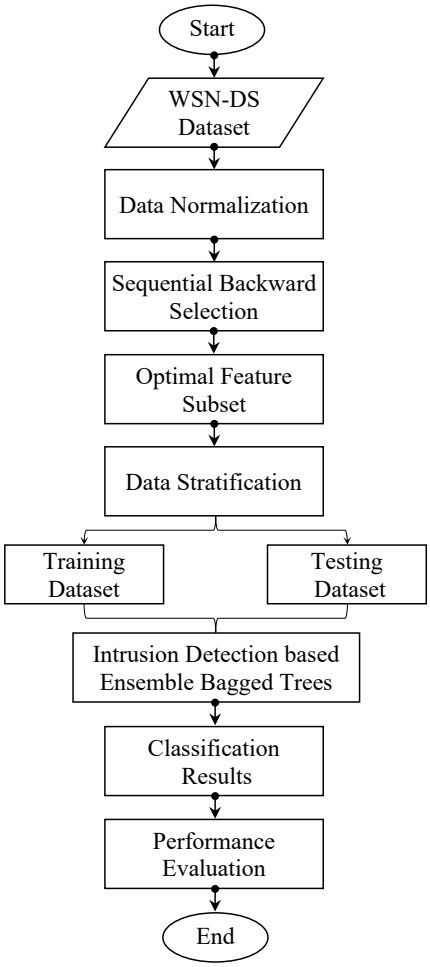

Fig. 2: The flowchart of the proposed method.

2) Stage 2 (Data Pre-processing): To ensure that the label feature in the sample data does not affect the algorithm, we must convert the characters into numerical values. The attack type data is of five types: normal, black hole, gray hole, flood, and TDMA. Since these types cannot be counted, we assign the values of 0, 1, 2, 3, and 4 to sort them.

3) Stage 3 (Sequence backward features selection): The (SBS) algorithm [18] is a greedy search technique that reduces the dimensions of the original feature space with minimal impact on classification performance to enhance computational efficiency. SBS sequentially removes features from the feature set until the new subspace contains the required number of features. To determine which features need to be removed at each stage, we use the standard scaling function J that needs

to be minimized. The computation criterion for this function is to compare the performance of the classifier before and after removing a particular feature. Therefore, we remove features at each step that increases the value of the standard scaling function, or more simply, that result in minimal loss of model performance after features are removed. Given the definition of SBS, we can summarize the algorithm in four simple steps:

- *Step 1*: Set k to d to initialize the algorithm, where d is the feature space dimension X_d [3].
- *Step 2*: Define x' as a feature that satisfies the criterion $x' = \text{argmax } J(X_k - x)$ to maximize $x \in X_k$.
- *Step 3*: Remove feature x' from the feature set: $X_{k-1} = X_k - x'$, $k = k - 1$.
- *Step 4*: If k is equal to the number of features in the objective function, terminate the algorithm; Otherwise, go to step 2.

4) Stage 4 (Data Satisfaction): In this stage, a range of machine learning (ML) algorithms were employed to ensure data congruency and enable the classification of distinct attack categories denoted by the values 0, 1, 2, 3, 4, and 5. A comprehensive comparison of diverse ML algorithms was conducted, including complex tree, medium tree, and sample tree models, support vector machine (SVM) models, encompassing linear, quadratic, cubic, fine Gaussian, medium Gaussian, and Coarse Gaussian SVMs, K-nearest neighbor (KNN) models, including fine, medium, and coarse KNN, cosine KNN, cubic KNN, and weighted KNN, as well as ensemble models such as ensemble budget trees, ensemble sub-space discriminate, ensemble boosted trees, ensemble subspace KNN, and ensemble RUS boosted trees.

IV. Performance Analysis

This section presents the experimental settings and elaborates on the corresponding experimental results.

A. Experimental settings

We used the public data set WSN-DS in [19], which was developed for the WSN and collected using the LEACH protocol from NS-2. The dataset contains four types of DoS attacks: black hole, gray hole, flood, and scheduling. The training data set and test data set were randomized, with 224,796 samples (60%) and 149,865 samples (40%), respectively. Table I provides the dataset for the proposed algorithms. In recent simulations, Ensemble Bagged trees have been identified as the best type of machine learning algorithm to improve the performance of wireless sensors. Ensemble Bagged trees are a form of ensemble learning, where multiple decision trees are combined to make a prediction. This approach helps to reduce overfitting and improve the accuracy of the predictions. The algorithm framework is shown in Fig. 2.

In order to evaluate the performance of our proposed method on a given dataset, we employ a confusion matrix (CM) to measure the accuracy, recall, precision, and F-measurement. The evaluation metrics are calculated using the following equations

TABLE I: Dataset partitioned 60% training set and 40% testing sets using ML algorithms after data normalization

Attack type	Training set (60 %)	Testing set (40%)
Normal	3000	2000
Scheduling	3000	2000
Gravhole	3000	2000
Blackhole	3000	2000
Flooding	1988	1324
Sum	13988	9324

$$\begin{gathered} Accuracy = TP + TN/(TP + TN + FP + FN), \\ Recall = TP/(TP + FN), \\ Accuracy = TP/(TP + FP), \\ MeasureF = 2 \times Accuracy \times Recall/(Accuracy + Recall) \end{gathered} \tag{4}$$

where TP refers to the count of true positives, TN represents the count of true negatives, FP denotes the count of false positives, and FN signifies the count of false negatives [20], [21]. Accuracy represents the ratio of correctly classified cases, while Retrieval represents the proportion of correctly classified positive items out of all positive items. The F scale represents the average sum of the detection rate and retrieval rate. Detection time denotes the duration it takes for the test sample to complete the test.

B. Experimental results

In this section, we evaluate the effectiveness of the proposed model. We conducted a series of experiments including 1) using different ML classification algorithms, 2) comparing multiple ML classification approaches with Ensemble Bagged Trees, and 3) evaluating the performance of Ensemble Bagged Trees against four types of attacks in terms of accuracy rate and precision. Our goal is to meet network requirements and integrate the model into a network intrusion detection system. In Table II, we compare the effectiveness of the traditional classification algorithms such as SVM, KNN, ensemble RUS boosted trees, Gaussian SVM, complex tree, ensemble boosted trees, and Ensemble Bagged Trees. It is noteworthy that among all the algorithms, Ensemble Bagged Trees demonstrates superior accuracy, accuracy, recall, and F-scaling indices.

The significant features employed for precise classification based on the SPS algorithm were organized in the following sequence:

- ADV_S: This parameter represents the number of advertisement messages sent by the current node.
- Is_{CH}: This parameter indicates whether the node is currently acting as a cluster head or not.
- SCH_S: This parameter represents the number of scheduled messages sent by the current node.
- $DATA_S$: This parameter represents the number of data packets sent by the current node.
- SCH_R: This parameter represents the number of scheduled messages received by the current node.

TABLE II: Comparison of multiple ML classification algorithms

ML Technique	Accuracy	Precision	Recall	F-measure
complex tree	0.9822	0.9836	0.9834	0.9834
medium tree	0.9481	0.9542	0.9515	0.9515
simple tree	0.7414	0.6779	0.7588	0
linear SVM	0.8884	0.9155	0.8955	0.8926
quadratic SVM	0.9326	0.9393	0.9370	0.9370
cubic SVM	0.9494	0.9540	0.9527	0.9528
fine Gaussian SVM	0.9426	0.9482	0.9457	0.9460
medium Gaussian SVM	0.8929	0.9156	0.8998	0.8978
coarse Gaussian SVM	0.8750	0.9001	0.8826	0.8798
fine KNN	0.9689	0.9711	0.9706	0.9708
medium KNN	0.9602	0.9639	0.9619	0.9622
coarse KNN	0.9179	0.9249	0.9215	0.9221
cosine KNN	0.9604	0.9640	0.9624	0.9625
cubic KNN	0.9587	0.9623	0.9606	0.9608
weighted KNN	0.9712	0.9741	0.9724	0.9728
Ensemble Boosted trees	0.9114	0.9292	0.9173	0.9185
Ensemble Bagged trees	0.9915	0.9921	0.9920	0.9920
Ensemble subspace Discriminant	0.8514	0.8809	0.8577	0.8566
Ensemble subspace KNN	0.9624	0.9624	0.9629	0.9625
Ensemble RUS Boosted trees	0.3564	0	0.3998	0

- $Data - Sent - to - BS$:This parameter represents the distance between the current cluster head and the base station.
- ADV_R: This parameter represents the number of advertisement messages received by the current node.
- $DATA_R$: This parameter represents the number of data packets received by the current node.
- $Attack$: This parameter represents the type of attack (if any) that is being simulated in the network.

When we selected approximately 9 features, the accuracy of the classification algorithm stabilized. However, as the number of features increases, the accuracy of the model decreases. recent simulation results have shown that Ensemble RUS Boosted trees may not be the best choice for improving the performance of wireless sensors that work with the LEACH protocol. In fact, it has been found to be the worst type of ML algorithm for this purpose. Table II and Fig. 3 present the experimental results, and the classification performance of our method (Ensemble Bagged Trees) is evaluated using the CM. The CM results can be found in Fig. 4. In addition, Table III shows that our proposed model performs better.

V. Conclusions

To summarize, WSNs are commonly used in environmental monitoring, security systems, and industrial automation, but their performance can be limited by factors such as signal strength, interference, and battery life. To address these limitations, machine learning algorithms have been employed to optimize resource usage and predict events. While the LEACH protocol is commonly used in wireless sensor networks to conserve energy, Ensemble Bagged trees have shown promising results as a more efficient and accurate alternative for improving the performance of wireless sensors in the LEACH protocol-based systems. Therefore, for optimal performance in these types of networks, Ensemble Bagged trees should be considered a better option than other ML techniques. Our study intends to explore the potential of enhancing attack classification by employing various combinations of ML algorithms and utilizing different data normalization techniques. This future work aims to identify the most efficient combination of normalization methods and ML algorithms to improve

TABLE III: Classification performance of the Ensemble Bagged Trees algorithm.

	Normal	Gxayhole	Blackhole	TMDA	Flooding
Accuracy	0.9921	0.9813	0.9894	1	1
precision	0.9823	0.9985	0.9901	0.9893	1
recall	0.9918	0.9810	0.9894	0.9986	0.9990
F measure	0.9870	0.9897	0.9897	0.9939	0.9995

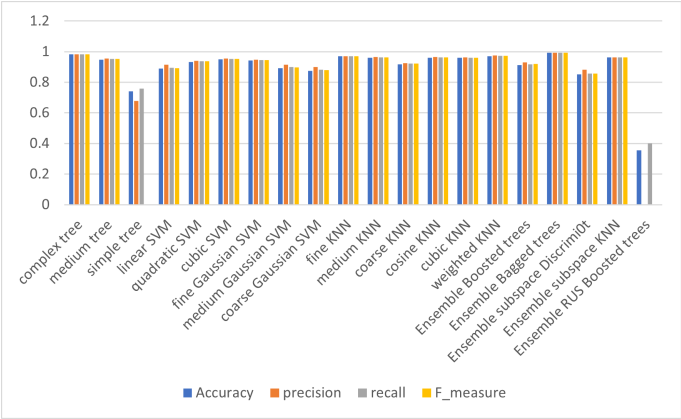

Fig. 3: Comparison results according to precision, Accuracy, Recall, and F-measure.

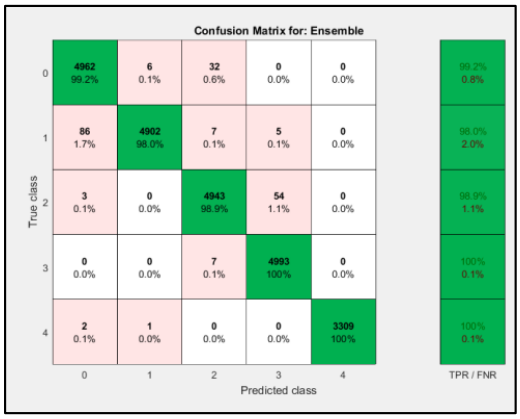

Fig. 4: The confusion matrix was utilized for the detection of various types of DoS attacks.

the classification accuracy of attacks. By integrating diverse normalization and ML techniques, we can potentially achieve more robust and accurate attack classification models.

REFERENCES

[1] Y. Du, J. Xia, J. Ma, and W. Zhang, "An optimal decision method for intrusion detection system in wireless sensor networks with enhanced cooperation mechanism", *IEEE Access*, vol. 9, pp. 69498-69512, 2021.

[2] S. Subbiah, K. S. M. Anbananthen, S. Thangaraj, S. Kannan, and D. Chelliah, "Intrusion detection technique in wireless sensor network using grid search random forest with Boruta feature selection algorithm", *Journal of Commun. and Net.*, vol. 24, no. 2, pp. 264-273, 2022.

[3] S. Jiang, J. Zhao, and X. Xu, "SLGBM: An intrusion detection mechanism for wireless sensor networks in smart environments", *IEEE Access*, vol. 8, pp. 169548-169558, 2020.

[4] 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.

[5] T. Saba, K. Haseeb, A. A. Shah, A. Rehman, U. Tariq, and Z. Mehmood, "A machine-learning-based approach for autonomous IoT security", *IT Professional*, vol. 23, no. 3, pp. 69-75, 2021.

[6] S. Kundu and N. Das, "A study on boundary detection in wireless sensor networks", *Innovations in Systems and Software Engineering*, pp. 1-9, 2022.

[7] K. Zaimen, L. Moalic, A. Abouaissa, and L. Idoumghar, "A Survey of Artificial Intelligence Based WSNs Deployment Techniques and Related Objectives Modeling", *IEEE Access*, vol. 10, pp. 21658-21684, 2022.

[8] F. F. Jurado-Lasso, L. Marchegiani, J. F. Jurado, A. M. Abu-Mahfouz, and X. Fafoutis, "A survey on machine learning software-defined wireless sensor networks (ml-SDWSNS): Current status and major challenges", *IEEE Access*, vol. 10, pp. 23560-23592, 2022.

[9] A. Jamalipour and S. Murali, "A Taxonomy of Machine-Learning-Based Intrusion Detection Systems for the Internet of Things: A Survey", *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9444-9466, 2021.

[10] G. Kalnoor and S. Gowrishankar, "A model for intrusion detection system using hidden Markov and variational Bayesian model for IoT based wireless sensor network", *Int. J. Inf. Technol.*, pp. 1-13, 2021.

[11] V. Gowdhaman and R. Dhanapal, "An intrusion detection system for wireless sensor networks using deep neural network", *Soft Comput.*, pp. 1-9, 2021.

[12] N. M. Saravana Kumar, E. Suryaprabha, and K. Hariprasath, "Machine learning based hybrid model for energy efficient secured transmission in wireless sensor networks", *J. Ambient Intell. Humaniz. Comput.*, pp. 1-16, 2022.

[13] A. Sarkar, H. S. Sharma, and M. M. Singh, "A supervised machine learning-based solution for efficient network intrusion detection using ensemble learning based on hyperparameter optimization", *Int. J. Inf. Technol.*, pp. 1-12, 2022.

[14] M. Waqas, S. Tu, Z. Halim, S. U. Rehman, G. Abbas, and Z. H. Abbas, "The role of artificial intelligence and machine learning in wireless networks security: Principle, practice and challenges", *Artif. Intell. Rev.*, vol. 55, no. 7, pp. 5215-5261, 2022.

[15] M. Sharma, H. Elmiligi, and F. Gebali, "A novel intrusion detection system for RPL-based cyber–physical systems", *IEEE Can. J. Electr. Comput. Eng.*, vol. 44, no. 2, pp. 246-252, 2021.

[16] M. M. Ata, K. M. Elgamily, and M. A. Mohamed, "Robust features fusion utilization for supervised palmprint recognition", *Concurrency and Computation: Practice and Experience*, vol. 34, no. 10, Oct. 2022.

[17] S. L. O'Reilly, E. C. O'Brien, D. McGuinness, J. Mehegan, B. Coughlan, D. O'Brien, M. Szafranska, et al., "Latch On: A protocol for a multicentre, randomised controlled trial of perinatal support to improve breastfeeding outcomes in women with a raised BMI", *Contemp. Clin. Trials Commun.*, vol. 22, 2021.

[18] F. J. Ferri, P. Pudil, M. Hatef, and J. Kittler, "Comparative study of techniques for large-scale feature selection", in Machine intelligence and pattern recognition, vol. 16, pp. 403-413, 1994.

[19] I. Almomani, B. Al-Kasasbeh, and M. Al-Akhras, "WSN-DS: A dataset for intrusion detection systems in wireless sensor networks", *Journal of Sensors*, vol. 2016, 2016.

[20] J. Su, Z. Sheng, A. X. Liu, Z. Fu, and Y. Chen, "A time and energy saving-based frame adjustment strategy (TES-FAS) tag identification algorithm for UHF RFID systems", *IEEE Trans. on Wireless Communications*, vol. 19, no. 5, pp. 2974-2986, May 2020.

[21] J. Su, Z. Sheng, A. X. Liu, Y. Han, and Y. Chen, "A group-based binary splitting algorithm for UHF RFID anti-collision systems", *IEEE Trans. on Communications*, vol. 68, no. 2, pp. 998-1012, Feb. 2019.