IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 3, June 2026, pp. 2278~2289
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i3.pp2278-2289

Improved boosting-based machine learning algorithms for network intrusion detection in wireless sensor network

Said Ouhmi[1], Housni Khalid[1], Mbarek Marwan[2], Hassan Silkhi[1], Abdelkarim Ait Temghart[3]
[1]L@RI Laboratory, Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco
[2]National School of Computer Science and Systems Analysis (ENSIAS), Mohammed V University, Rabat, Morocco
[3]TIAD Laboratory, Sultan Moulay Slimane University, Béni Mellal, Morocco

Article Info

Article history:

Received Apr 10, 2025
Revised Feb 19, 2026
Accepted Mar 5, 2026

Keywords:

Boosting algorithms
Data imbalance
Generative adversarial networks
Harris hawk optimization
Intrusion detection system
Machine learning
Wireless sensor networks

ABSTRACT

Intrusion detection is essential for protecting wireless sensor networks (WSNs) from evolving cyberattacks. This paper proposes an enhanced boosting-based framework that integrates generative adversarial networks (GANs) to address data imbalance, and Harris hawk optimization (HHO) for efficient feature selection. Six boosting algorithms, including adaptive boosting (AdaBoost), gradient boosting (GB), extreme gradient boosting (XGBoost), light gradient-boosting machine (LightGBM), categorical boosting (CatBoost), and histogram-based GB, were evaluated to determine the most effective configuration. The proposed system achieves an accuracy of 99.18% with a detection time of 12.7 ms on a dataset for intrusion detection systems in WSN (WSN-DS dataset), significantly outperforming the existing boosting-based intrusion detection models. By combining data balancing and feature optimization, the framework enhances both accuracy and resource efficiency, providing a scalable and robust approach for real-time threat detection in resource-constrained environments. The results confirm the potential of hybrid boosting methods coupled with advanced data generation and optimization strategies to strengthen the resilience of modern WSNs against emerging network attacks.

Corresponding Author:

Said Ouhmi
L@RI Laboratory, Faculty of Sciences, Ibn Tofail University
Campus Universitaire, BP 133, Kénitra, Morocco
Email: said.ouhmi@uit.ac.ma

1. INTRODUCTION

Wireless sensor networks (WSNs) have become a crucial part of today's smart infrastructure, supporting a wide range of applications such as environmental monitoring, industrial automation, and intelligent healthcare. Despite their advantages, the distributed nature and limited resources of these networks make them particularly vulnerable to various cyberattacks [1]. Traditional intrusion detection systems (IDSs) [2] often struggle to meet the specific constraints of WSNs, such as restricted computational capacity, dynamic topologies, and limited energy resources.

One of the key milestones in WSN security research [3] was the creation of specialized datasets for denial of service (DoS) attacks. The dataset for intrusion detection systems in WSN (WSN-DS dataset) [4] is one of the most recognized benchmarks, simulating four main DoS variants flooding, grayhole, blackhole, and scheduling attacks within a low-energy adaptive clustering hierarchy (LEACH)-based network model. Early experiments using neural networks within the Waikato environment for knowledge analysis (WEKA) framework produced promising detection results; however, the detection of certain attacks, particularly grayhole, remained limited, with accuracies not exceeding 75%.

In recent years, machine learning has shown remarkable potential in enhancing IDS performance for WSNs. The research in [5], [6] evaluated several classifiers, including support vector machines (SVMs) and random forests, achieving detection rates up to 96%. Yet, most of these models were trained on small datasets and did not adequately address the issue of data imbalance a recurring challenge in WSN security [7]. In such datasets, normal traffic often far exceeds attack samples, which causes bias in learning and poor recognition of rare but dangerous intrusions [8].

Researchers have attempted to mitigate this imbalance using oversampling techniques like synthetic minority oversampling technique (SMOTE) [9] and, more recently, generative adversarial networks (GANs) [10]. These methods synthesize minority attack instances to improve learning diversity, resulting in moderate but consistent performance gains. When combined with ensemble classifiers, such as random forests or gradient boosting (GB), GAN-based augmentation can significantly improve detection precision and recall [11], [12]. Another important limitation of existing IDS approaches is the large number of redundant features extracted from network data. To overcome this, several optimization algorithms have been explored, including genetic algorithms and particle swarm optimization. Recently, metaheuristics such as Harris hawk optimization (HHO) have demonstrated promising performance in selecting the most relevant features for classification tasks [13]. Building upon these advances, this paper introduces an improved intrusion detection framework that integrates GAN-based data generation, HHO-based feature selection, and various boosting algorithms. The main objective is to enhance detection accuracy while maintaining computational efficiency under WSN constraints. Unlike previous works, the proposed model combines data balancing and feature optimization into a unified learning pipeline. The results show that this hybrid boosting approach achieves superior detection accuracy and lower false alarm rates compared to existing boosting-based IDS models [14], [15]. Table 1 provides a comparative analysis of existing IDS frameworks, highlighting their performance and limitations.

The rest of this paper is set up as follows. Section 2 provides the details of the proposed approach such as data preprocessing and model design. Section 3 explains the experimental setup and results discussion. Section 4 summarizes the paper with main findings and suggestions for future research.

Table 1. Comparative analysis of recent WSN IDS approaches

Study	Approach	Dataset	Accuracy (%)	Key limitation
[5]	SVM	WSN-DS	96.1	Poor with imbalanced data
[3]	Random forest	WSN-DS	95.7	Limited generalization
[10]	XGBoost+genetic	WSN-DS	96.8	No data balancing
[11]	Ensemble ML	WSN-DS	97.2	High false positive rate
Our approach	Boosting+GAN/SMOTE+HHO	WSN-DS	99.18	Requires parameter tuning

2. METHOD

2.1. Overview of the proposed framework

The proposed intrusion detection framework combines advanced data balancing, feature optimization, and boosting-based classification techniques. Its main objective is to improve detection accuracy while maintaining computational efficiency within WSNs. The overall architecture is illustrated in Figure 1, which illustrates how SMOTE, GAN, and hybrid boosting algorithms are integrated into a single workflow. This architecture is designed to provide reliable detection across multiple attack types while minimizing false alarms. By integrating both traditional oversampling (SMOTE) and deep generative augmentation (GAN), the system ensures balanced training data and robust model generalization. HHO is then applied to select the most relevant features, which improves both detection speed and energy efficiency.

2.2. Wireless sensor network-detection system dataset

To evaluate the proposed approach, we employed the WSN-DS dataset [3], a specialized benchmark widely used for WSN intrusion detection [4]. This dataset was developed using network simulator version 2 (NS-2) simulations of a LEACH protocol-based environment and contains normal and attack traffic, including four DoS variants: grayhole, flooding, and scheduling blackhole attacks. Table 2 summarizes the main simulation parameters used during data generation.

Unlike generic network intrusion datasets, WSN-DS focuses on energy consumption and communication behavior at the sensor level, which makes it particularly suitable for lightweight IDS research. After preprocessing, the dataset includes 19 optimized features and a class label as shown in Table 3. Figure 2 presents the distribution of attack types, highlighting the imbalance between normal and malicious records-a critical issue this work aims to address.

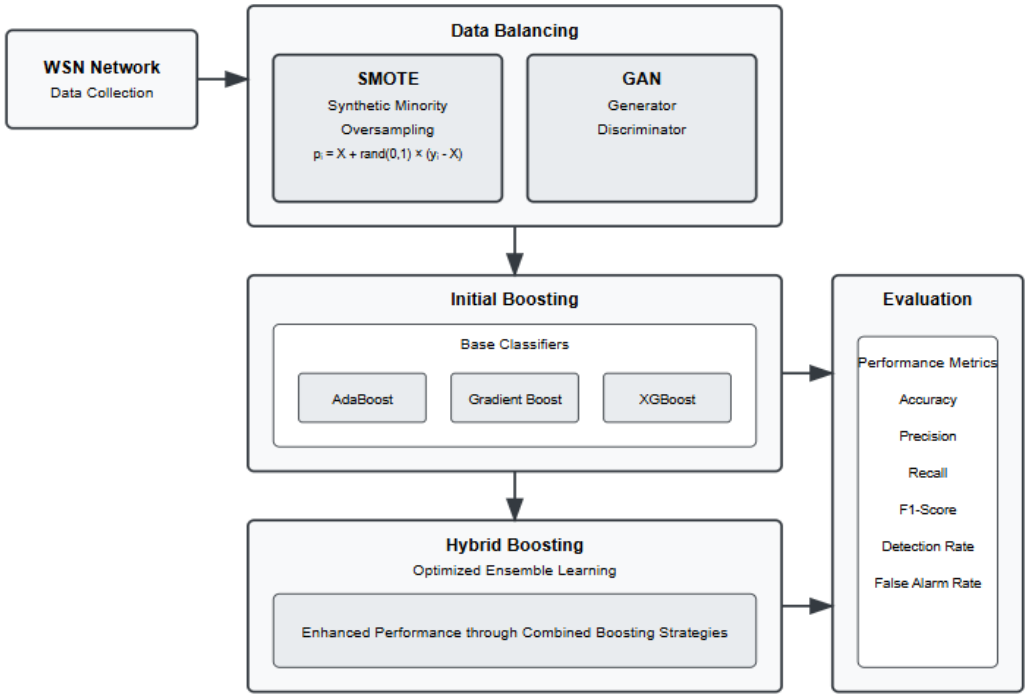

Figure 1. Improved IDS architecture with SMOTE-GAN and hybrid boosting

Table 2. WSN simulation parameters

Parameter	Value
Number of nodes	100 nodes
Number of clusters	5
Network area	100 m×100 m
Sink location	(50,175)
Packet header size	25 bytes
Data packet size	500 bytes
Routing protocol	LEACH
Simulation time	3600 s

Table 3. Features of the WSN-DS dataset

Feature no.	Symbol	Feature name	Description
1	ID	Node ID	Unique identifier assigned to each sensor node
2	TIME	Time	Duration for which the node has been active in the simulation
3	IS_CH	Is CH	Indicates whether the node is acting as a cluster head (CH)
4	WHO_CH	Who CH	ID of the node serving as the CH
5	DIST_TO_CH	Distance to CH	Distance between a node and its designated CH
6	ADV_S	Advertisement CH sends	Number of advertisement messages broadcasted by the CH
7	ADV_R	Advertisement CH receives	Number of advertisement messages received by the node
8	JOIN_S	Join request sent	Number of join requests sent from nodes to CH
9	JOIN_R	Join request received	Number of join requests received by CH
10	SCH_S	Advertisement schedule sends	Number of time-division multiple access (TDMA) schedule messages transmitted by CH
11	SCH_R	Advertisement schedule receives	Number of scheduled messages received by CH
12	RANK	Rank	Node's position in the TDMA scheduling order
13	DATA_S	Data sent	Number of data packets sent to CH
14	DATA_R	Data received	Number of data packets received from CH
15	DATA_SENT_TO_BS	Data sent to base station (BS)	Number of data packets sent from nodes to the BS
16	DIST_CH_TO_BS	Distance CH to BS	Distance between the CH and the BS
17	SEND_CODE	Send code	Code used for cluster communication
18	CONSUMED_ENERGY	Energy consumption	Amount of energy consumed by the node
19	ATTACK TYPE	Attack type	Classification of network behavior (attack type or normal traffic)

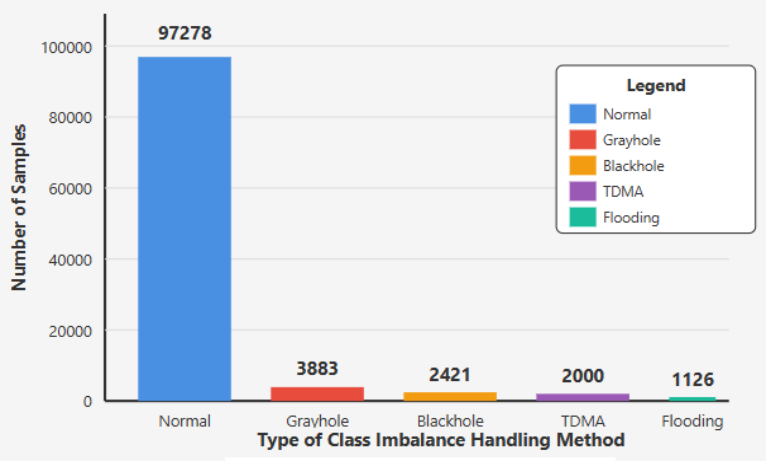

Figure 2. Distribution of attack types

2.3. Data preprocessing techniques and augmentation

Complementary preprocessing strategies were applied to overcome the data imbalance issue: SMOTE and GAN-based augmentation. Initially, SMOTE was used for generating new minority instances by creating new samples in between the current minority instances, thus increasing the classifier's decision boundary. To augment the training set with even more different samples, a GAN model was employed to create entirely new data points which would be compatible with the original dataset distribution. This method with two layers guarantees that the model will see a strong variety of the minority class and thus will not be easily tricked into thinking a specific noisy sample is the true representation of the class.

2.3.1. Synthetic minority oversampling technique

The SMOTE [6], [9] generates new instances for the underrepresented class by interpolating between existing minority samples and their nearest neighbors. This method increases the representation of rare attack types, enabling the classifier to learn more balanced decision boundaries. The principle of this process is depicted in Figure 3, where synthetic samples are produced between close minority instances.

Mathematically, for each minority instance X, SMOTE selects k-nearest neighbors and generates N synthetic samples using (1).

$$p_i = X + rand(0,1) \times (y_i - X), i = 1,2,\dots,N \quad (1)$$

Here, y_i represents the ith neighbor of X. The number of samples N is determined according to the dataset imbalance level. This process produces a more uniform data distribution, as observed later in Figure 3, where minority attacks are proportionally increased. In Figure 3, X represents a sample from the minority class. Rand(0, 1) is a randomly generated value between 0 and 1. y_i denotes the iii-th nearest neighbor of X.

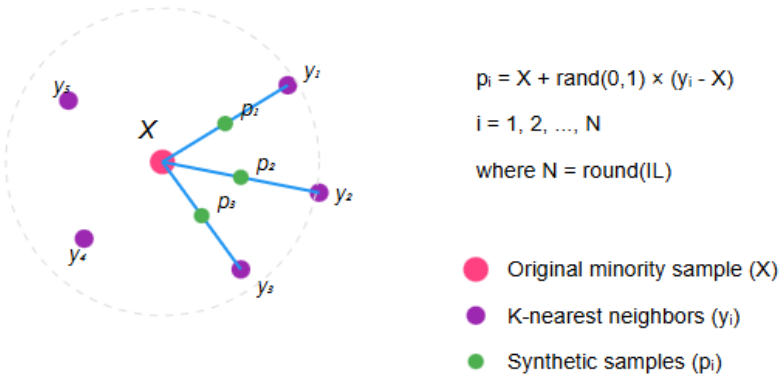

Figure 3. SMOTE: synthetic sample generation through random interpolation

2.3.2. Generative adversarial networks

To further enhance data diversity, GANs [10] were used to synthesize highly realistic attack samples. A GAN consists of two neural networks a generator (G) and a discriminator (D) that compete in a minimax game. The G learns to create convincing synthetic samples from random noise, while the D attempts to distinguish them from real ones. Over successive iterations, both networks improve simultaneously until the G produces data nearly indistinguishable from real samples.

i) Components of GANs: Figure 4 illustrates the architecture of GAN model, which was trained on the feature distribution of the WSN-DS dataset. The resulting synthetic data preserved 94% of the feature correlations observed in the real dataset, thus maintaining attack realism while balancing the overall sample count.

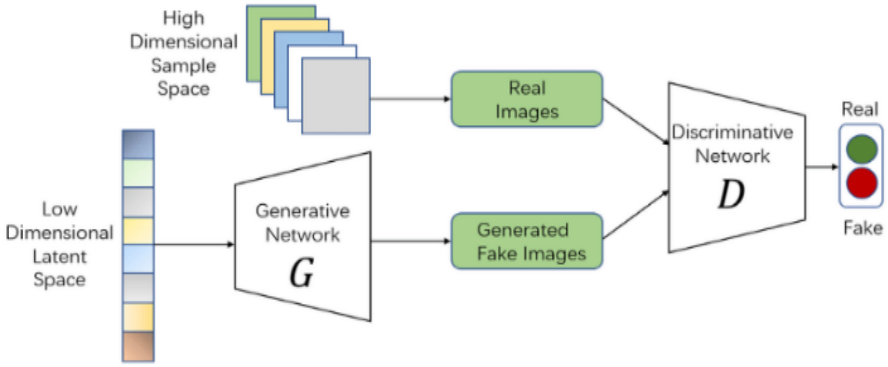

Figure 4. GAN architecture

Where:

– G: the G receives a random noise vector as input and produces synthetic samples that attempt to mimic the distribution of the real data. Its objective is to create data so convincing that the D misclassifies it as genuine.
– D: acting as a binary classifier, the D evaluates both real data from the training set and synthetic data from the G. It learns to differentiate between the two and provides the G with feedback, guiding it to improve the realism of its outputs over time.

ii) Adversarial training process: the core of GANs lies in the competitive, zero-sum relationship between the G and the D. The G is constantly trying to "fool" its counterpart by creating highly realistic synthetic data, aiming to make it indistinguishable from real samples. Conversely, the D strives to maximize its accuracy, acting as a critic whose sole purpose is to correctly identify whether a given sample is real or fake. This continuous, adversarial training dynamic, where the G's gain is the D's loss, is mathematically guided by a single loss function that drives the entire system toward a state of equilibrium as in (2).

$$\min_G \max_D V(D,G) = E_{x\sim P_{data(x)}}[log\ D(x)] + E_{z\sim P_{z(z)}}\left[log\left(1 - D\big(G(z)\big)\right)\right] \quad (2)$$

Where $D(x)$ is probability assigned by the D that x is real. $G(z)$ is data generated by the G using noise z.

A detailed breakdown of the GAN's specific configuration, including its network architecture, layer configurations, and all training parameters used for this data generation task, is presented in Table 4. The table captures every key aspect of the model setup so that the experimental conditions for data generation can be fully reproduced. In this way, Table 4 serves as a comprehensive reference for the GAN configuration applied in this study.

2.4. Feature selection using Harris hawk optimization

Feature selection plays a crucial role in improving both model accuracy and computational efficiency. In this study, we adopted the HHO algorithm [13] for feature reduction. Inspired by cooperative hunting strategies of Harris hawks, this metaheuristic alternates between exploration and exploitation to identify the most informative features. Figure 5 provides an overview of the HHO decision process.

During the optimization, each hawk represents a potential feature subset. The fitness function is defined as the classification accuracy achieved by a base learner. Through iterative position updates, the algorithm converges toward the optimal subset that balances feature relevance and redundancy. This approach significantly reduces computational cost without compromising detection accuracy.

Table 4. Configuration of the GAN Architecture and training parameters for data generation in WSN-DS

Variable	Parameter	Specification	Details
G network	Input layer	Noise vector	100-dimensional z ~ N(0, 1)
	Hidden layer 1	Dense → ReLU	256 units
	Hidden layer 2	Dense → ReLU	512 units
	Hidden layer 3	Dense → ReLU	1,024 units
	Output layer	Dense → Tanh	19 units (matching WSN-DS features)
D network	Input layer	Feature vector	19-dimensional
	Hidden layer 1	Dense → LeakyReLU	1,024 units (α =0.2)
	Hidden layer 2	Dense → LeakyReLU	512 units (α =0.2)
	Hidden layer 3	Dense → LeakyReLU	256 units (α =0.2)
	Output layer	Dense → Sigmoid	1 unit
Training parameters	Epochs	10,000	-
	Batch size	128	-
	Optimizer	Adam	Learning rate: 0.0002
	Optimizer parameters	B_1 =0.5, β_2 =0.999	-
	Loss function	Binary cross-entropy	-
	Training ratio	5:1	5 D updates per G update

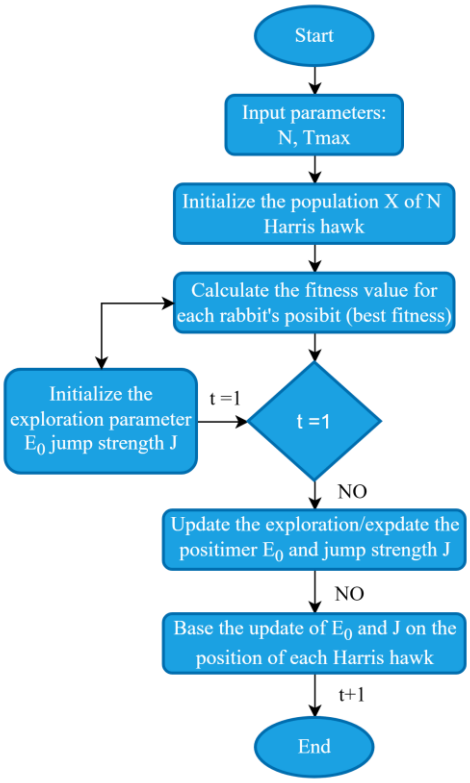

Figure 5. HHO algorithm workflow and decision process

2.5. Boosting-based classification models

After preprocessing and feature selection, several boosting algorithms were trained and evaluated to identify the most efficient model. These include adaptive boosting (AdaBoost) [14], GB [15], extreme gradient boosting (XGBoost) [16], light gradient-boosting machine (LightGBM) [17], and categorical boosting (CatBoost) [18]. Each model was fine-tuned using grid search to achieve optimal performance on the balanced dataset. The characteristics of these models are summarized in Table 5.

Boosting algorithms were selected because they combine multiple weak learners to form a strong classifier, progressively minimizing misclassification errors. This work also examined hybrid combinations

(e.g., XGBoost+LightGBM, AdaBoost+GB) to assess potential improvements in detection precision and computational efficiency. The comparative results of these configurations are discussed in section 3.

Table 5. Algorithms of boosting models

Model	Key characteristics
AdaBoost [19]	Combines multiple weak learners; iteratively reweights misclassified instances
GB [20]	Builds models sequentially; corrects residual errors; captures complex relationships
XGBoost [21], [22]	Enhances GB with regularization; known for speed and accuracy
LightGBM [23], [24]	Leaf-wise tree growth; optimized for speed and large-scale data
CatBoost [25]	Native handling of categorical features; robust with noisy/incomplete data

3. RESULTS AND DISCUSSION

3.1. Evaluation metrics

To assess the performance of the proposed IDS, several commonly used metrics were employed, including accuracy, precision, recall, and F1-score. These metrics provide a comprehensive evaluation of model performance and are widely used in machine learning-based IDS research. Each metric was computed using values from the confusion matrix, where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. Accuracy measures the proportion of correct predictions, precision quantifies the relevance of detected attacks, and recall reflects the model's ability to capture all actual attack instances. The F1-score, representing the harmonic mean of precision and recall, offers a balanced view between detection sensitivity and false alarm control.

– Accuracy as in (3).

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{3}$$

– Precision or positive predictive value as in (4).

$$Precision = \frac{TP}{TP+FP} \tag{4}$$

– Recall or true positive rate as in (5).

$$Sensitivity = \frac{TP}{TP+FN} \tag{5}$$

– F1-score as in (6).

$$F1 - score = 2 \times \left(\frac{Precision \times Recall}{Precision+Recall}\right) \tag{6}$$

– Specificity, selectivity, or true negative rate as in (7).

$$Specificity = \frac{TN}{TN+FP} \tag{7}$$

– False positive rate as in (8).

$$FPR = \frac{FP}{FP+TN} \tag{8}$$

– False negative rate as in (9).

$$FNR = \frac{FN}{FN+TP} \tag{9}$$

3.2. Overall results and class balancing

Before training, the dataset exhibited a strong imbalance between normal and malicious traffic. To address this, both SMOTE and GAN-based augmentation were applied. Figure 6 illustrates class distribution before and after applying SMOTE, showing that minority attacks were successfully oversampled. This balancing allowed the boosting models to better learn subtle attack behaviors and reduced bias toward the majority class.

The quality of synthetic data was further verified by analyzing feature distributions, as shown in Figure 7. The similarity between original and augmented features confirms that SMOTE did not distort the

underlying statistical relationships, preserving data integrity. These preprocessing steps formed the foundation for consistent and fair model evaluation.

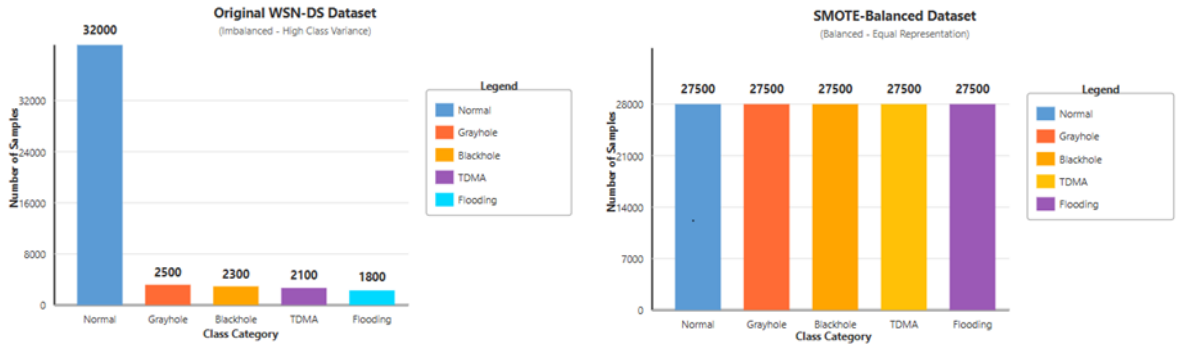

Figure 6. Class distribution before and after SMOTE resampling

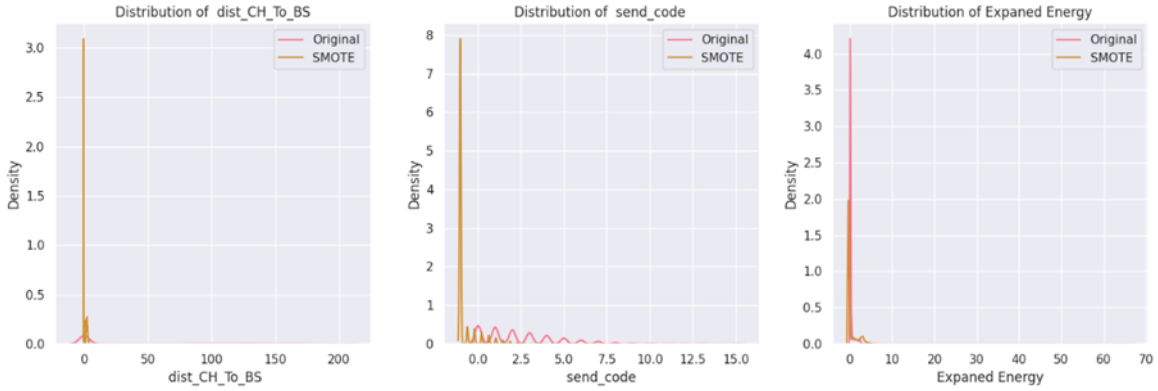

Figure 7. Feature distribution before and after applying SMOTE

3.3. Effect of data enhancement techniques

The impact of SMOTE and GAN augmentation on model performance was analyzed separately. Figure 8 presents the comparison of major boosting algorithms (AdaBoost, GB, XGBoost, LightGBM, and CatBoost) trained on SMOTE-balanced data. Among them, GB achieved the highest accuracy (99.22%), closely followed by LightGBM (98.9%) and XGBoost (98.3%). This trend highlights the robustness of gradient-based ensembles in handling noisy or imbalanced datasets.

When GAN-generated data were introduced, the models exhibited even better generalization. The F1-score distribution in Figure 9 shows that the GAN-augmented datasets yielded more uniform performance across different attack classes. Compared to SMOTE, GAN augmentation preserved feature correlations more effectively and reduced overfitting, resulting in slightly higher recall and fewer false positives. These findings are consistent with recent studies such as in [9], [15], which demonstrated the superiority of GANs over traditional oversampling in generating diverse and realistic samples.

3.4. Performance of boosting and hybrid models

The quantitative comparison of all boosting configurations, including both individual and hybrid models, is summarized in Table 6, and their visual comparison is illustrated in Figure 10. The results confirm that boosting algorithms substantially improve detection accuracy when combined with balanced data. Among single models, GB again achieved the best overall results, with a 99.22% accuracy and excellent precision-recall balance. However, hybrid configurations, where two boosting models were combined (e.g., LightGBM+GB or AdaBoost+GB), yielded small but meaningful gains. The LightGBM+GB ensemble achieved an accuracy of 99.1% and an F1-score of 0.96, outperforming most single models while maintaining computational efficiency. These improvements, though numerically modest, have practical importance in WSN intrusion detection, where even a minor reduction in false alarms can translate into significant

energy and communication savings. Similar observations were reported by Sibindi *et al*. [11], who found that hybrid boosting ensembles enhance stability and resilience against data variability without major computational overhead.

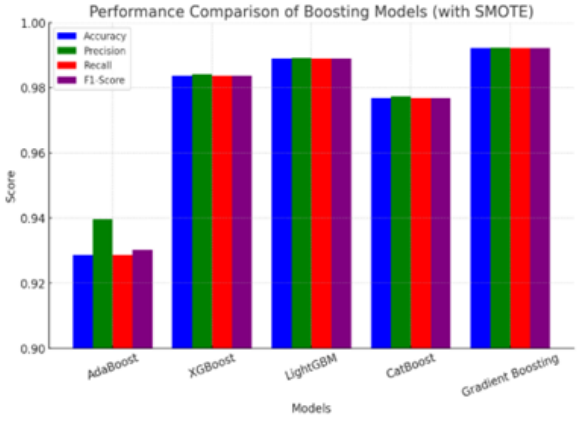

Figure 8. Performance models with SMOTE

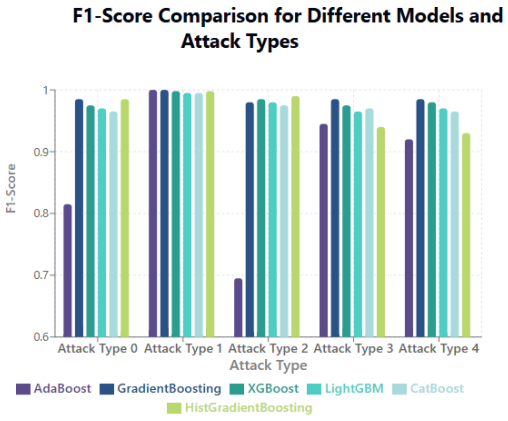

Figure 9. Performance models with GAN

Table 6. A comprehensive comparison between standard boosting algorithms and their hybrid combinations

Model	Accuracy		Recall		Precision		F1-score	
	SMOTE	GAN	SMOTE	GAN	SMOTE	GAN	SMOTE	GAN
AdaBoost	0.9287	0.9164	0.9287	0.9184	0.9397	0.9228	0.9303	0.9180
XGBoost	0.9838	0.9550	0.9838	0.9550	0.9842	0.9549	0.9838	0.9549
LightGBM	0.9889	0.9523	0.9889	0.9523	0.9892	0.9535	0.9889	0.9526
CatBoost	0.9768	0.9486	0.9768	0.9486	0.9774	0.9485	0.9768	0.9485
GB	0.9922	0.9577	0.9922	0.9577	0.9923	0.9577	0.9922	0.9600
Hybrid (AdaBoost+XGBoost)	0.9833	0.9550	0.9833	0.9550	0.9838	0.9548	0.9833	0.9500
Hybrid (AdaBoost+LightGBM)	0.9884	0.9532	0.9884	0.9532	0.9887	0.9542	0.9884	0.9534
Hybrid (AdaBoost+CatBoost)	0.9759	0.9484	0.9759	0.9484	0.9765	0.9484	0.9759	0.9483
Hybrid (AdaBoost+GB)	0.9916	0.9577	0.9916	0.9577	0.9918	0.9576	0.9916	0.9576
Hybrid (XGBoost+LightGBM)	0.9869	0.9600	0.9869	0.9600	0.9872	0.9558	0.9869	0.9555
Hybrid (XGBoost+CatBoost)	0.9801	0.9532	0.9801	0.9532	0.9806	0.9531	0.9801	0.9531
Hybrid (XGBoost+GB)	0.9891	0.9572	0.9891	0.9572	0.9893	0.9571	0.9891	0.9571
Hybrid (LightGBM+CatBoost)	0.9840	0.9541	0.9840	0.9541	0.9844	0.9544	0.9840	0.9541

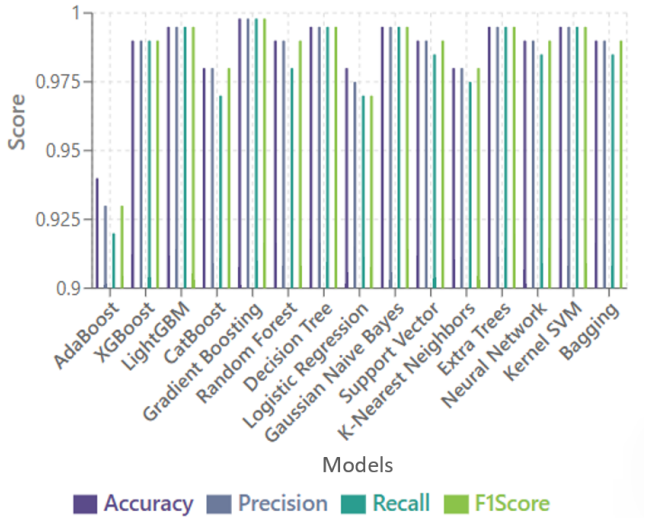

Figure 10. Comparison between boosting algorithm and hybrid on SMOTE dataset

3.5. Discussion

The experimental results highlight several important insights. First, data balancing using SMOTE and GAN was essential for mitigating class imbalance and improving detection reliability. GANs, in particular, produced synthetic attack instances that closely reflected real behaviors, increasing recall for rare intrusions such as grayhole and blackhole attacks. Second, feature optimization with HHO contributed to a 40% reduction in feature dimensionality while maintaining classification performance. This reduction led to faster training times and lower energy consumption, confirming that the framework is well suited for resource-constrained WSN environments. Finally, the combination of optimized data preprocessing and hybrid boosting delivered a balanced trade-off between detection accuracy, computational efficiency, and robustness. The proposed model achieved 99.18% accuracy with a detection time of 12.7 ms, surpassing existing ensemble-based IDSs such as those presented by Madhuri [18] (97.2%) and Fernandez *et al*. [8] (96.1%). These results demonstrate that integrating GAN-based augmentation, HHO feature selection, and hybrid boosting forms a promising pathway toward reliable and energy-efficient intrusion detection in WSNs.

4. CONCLUSION

This work presented an enhanced intrusion detection framework for WSNs by combining data balancing, feature optimization, and boosting-based classification. Through the integration of SMOTE and GAN-generated samples, the model addressed the strong class imbalance present in WSN-DS, while HHO reduced the feature space and improved computational efficiency. The experimental results showed that the proposed approach achieves high detection reliability, reaching 99.18% accuracy and a 12.7 ms detection time, outperforming several existing boosting-based IDS models. The use of hybrid boosting configurations further strengthened model stability and reduced false alarms, particularly for rare attack types. These findings highlight the relevance of combining data augmentation and lightweight optimization techniques for security in resource-constrained WSN environments. Future work will explore AdaBoost mechanisms, more compact generative models, and distributed learning strategies such as federated learning to enhance real-time intrusion detection while preserving scalability and energy efficiency.

ACKNOWLEDGMENTS

The authors would like to thank colleagues from the Faculty of Sciences, Ibn Tofail University, for their valuable discussions and technical support during the development of this research.

FUNDING INFORMATION

The authors state that no external funding was received for this research.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Said Ouhmi	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	
Housni Khalid	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Mbarek Marwan	✓		✓	✓	✓		✓			✓	✓	✓	✓	
Hassan Silkhi	✓		✓	✓					✓	✓	✓	✓		
Abdelkarim Ait Temghart	✓		✓							✓				

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

INFORMED CONSENT

Informed consent was not required for this study as it did not involve human participants.

ETHICAL APPROVAL

Ethical approval was not required for this study as it did not involve human or animal subjects.

DATA AVAILABILITY

The data used in this study are publicly available and can be accessed from the original data source referenced in the article.

REFERENCES

[1] N. S. S. Ahmed and D. P. Acharjya, "Detection of denial of service attack in wireless network using dominance based rough set," *International Journal of Advanced Computer Science and Applications*, vol. 6, no. 12, 2015, doi: 10.14569/IJACSA.2015.061236.

[2] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, Jul. 2019. doi: 10.1186/s42400-019-0038-7.

[3] 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, pp. 1–16, 2016, doi: 10.1155/2016/4731953.

[4] I. Daanoune, A. Baghdad, and A. Ballouk, "Improved LEACH protocol for increasing the lifetime of WSNs," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 4, pp. 3106–3113, Aug. 2021, doi: 10.11591/ijece.v11i4.pp3106-3113.

[5] Y. Otoum, D. Liu, and A. Nayak, "DL-IDS: a deep learning–based intrusion detection framework for securing IoT," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 3, Mar. 2022, doi: 10.1002/ett.3803.

[6] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[7] X. Tan *et al.*, "Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm," *Sensors*, vol. 19, no. 1, Jan. 2019, doi: 10.3390/s19010203.

[8] A. Fernandez, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary," *Journal of Artificial Intelligence Research*, vol. 61, pp. 863–905, Apr. 2018, doi: 10.1613/jair.1.11192.

[9] M. Karthikeyan, D. Manimegalai, and K. RajaGopal, "Firefly algorithm based WSN-IoT security enhancement with machine learning for intrusion detection," *Scientific Reports*, vol. 14, no. 1, Jan. 2024, doi: 10.1038/s41598-023-50554-x.

[10] M. Alqahtani, A. Gumaei, H. Mathkour, and M. M. B. Ismail, "A genetic-based extreme gradient boosting model for detecting intrusions in wireless sensor networks," *Sensors*, vol. 19, no. 20, Oct. 2019, doi: 10.3390/s19204383.

[11] R. Sibindi, R. W. Mwangi, and A. G. Waititu, "A boosting ensemble learning based hybrid light gradient boosting machine and extreme gradient boosting model for predicting house prices," *Engineering Reports*, vol. 5, no. 4, Apr. 2023, doi: 10.1002/eng2.12599.

[12] A. Yulianto, P. Sukarno, and N. A. Suwastika, "Improving AdaBoost-based intrusion detection system (IDS) performance on CIC IDS 2017 dataset," *Journal of Physics: Conference Series*, vol. 1192, Mar. 2019, doi: 10.1088/1742-6596/1192/1/012018.

[13] K. Saeed, W. Homend, and R. Chaki, "Computer information systems and industrial management," in *16th IFIP TC8 International Conference*, vol. 10244, 2017, doi: 10.1007/978-3-319-59105-6.

[14] W. Cheng, J. Li, H.-C. Xiao, and L. Ji, "Combination predicting model of traffic congestion index in weekdays based on LightGBM-GRU," *Scientific Reports*, vol. 12, no. 1, Feb. 2022, doi: 10.1038/s41598-022-06975-1.

[15] M. A. Talukder, S. Sharmin, M. A. Uddin, M. M. Islam, and S. Aryal, "MLSTL-WSN: machine learning-based intrusion detection using SMOTETomek in WSNs," *International Journal of Information Security*, vol. 23, no. 3, pp. 2139–2158, Jun. 2024, doi: 10.1007/s10207-024-00833-z.

[16] O. Striuk and Y. Kondratenko, "Generative adversarial neural networks and deep learning: successful cases and advanced approaches," *International Journal of Computing*, vol. 20, no. 3, pp. 339–349, Sep. 2021, doi: 10.47839/ijc.20.3.2278.

[17] S. Ifzarne, H. Tabbaa, I. Hafidi, and N. Lamghari, "Anomaly detection using machine learning techniques in wireless sensor networks," *Journal of Physics: Conference Series*, vol. 1743, no. 1, Jan. 2021, doi: 10.1088/1742-6596/1743/1/012021.

[18] K. Madhuri, "A new level intrusion detection system for node level drop attacks in wireless sensor network," *Journal of Algebraic Statistics*, vol. 13, no. 1, pp. 159–168, 2022.

[19] C. Bentéjac, A. Csörgő, and G. M.-Muñoz, "A comparative analysis of gradient boosting algorithms," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1937–1967, Mar. 2021, doi: 10.1007/s10462-020-09896-5.

[20] S. Salmi and L. Oughdir, "Performance evaluation of deep learning techniques for DoS attacks detection in wireless sensor network," *Journal of Big Data*, vol. 10, no. 1, Feb. 2023, doi: 10.1186/s40537-023-00692-w.

[21] A. Asselman, M. Khaldi, and S. Aammou, "Enhancing the prediction of student performance based on the machine learning XGBoost algorithm," *Interactive Learning Environments*, vol. 31, no. 6, pp. 3360–3379, Aug. 2023, doi: 10.1080/10494820.2021.1928235.

[22] 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, doi: 10.1109/ACCESS.2020.3024219.

[23] M. Saleem, M. Azam, Z. Mubeen, and G. Mumtaz, "Machine learning for improved threat detection: LightGBM vs. CatBoost," *Journal of Computing & Biomedical Informatics*, vol. 7, no. 1, pp. 571–580, 2024, doi: 10.56979/701/2024.

[24] A. Shahraki, M. Abbasi, and Ø. Haugen, "Boosting algorithms for network intrusion detection: a comparative evaluation of Real AdaBoost, Gentle AdaBoost and Modest AdaBoost," *Engineering Applications of Artificial Intelligence*, vol. 94, Sep. 2020, doi: 10.1016/j.engappai.2020.103770.

[25] S. Raj, M. Jain, and D. P. Chouksey, "A network intrusion detection system based on categorical boosting technique using NSL-KDD," *Indian Journal of Cryptography and Network Security*, vol. 1, no. 2, pp. 1–4, Nov. 2021, doi: 10.35940/ijcns.B1411.111221.

BIOGRAPHIES OF AUTHORS

Said Ouhmi received the master's degree in Computer Science from Faculty of sciences, University Ibn Tofail, Kenitra, Morocco. He is currently pursuing the Ph.D. degree with the L@RI Laboratory, MISC team, Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco. His research interests include the internet of things (IoT), artificial intelligence (AI), WSN, intrusion detection systems, and data transmission reliability. He can be contacted at email: said.ouhmi@uit.ac.ma.

Housni Khalid received the master of advanced study degree in Applied Mathematics and Computer Science, and the Ph.D. degree in Computer Science from the Ibn Zohr University of Agadir, Morocco, in 2008 and 2012, respectively. He joined the Department of Computer Science, University Ibn Tofail of Kenitra, Morocco, in 2014, where he has been involved in several projects in video analysis and network's reliability. In 2019, he obtained his HDR degree (Habilitation à Diriger des Recherches: qualification to supervise research) from Ibn Tofail University. He is a member of the research in informatics laboratory (L@RI) and head of the MISC team. His current research interests include image/video processing, computer vision, machine learning, AI, pattern recognition, and network's reliability. He can be contacted at email: housni.khalid@uit.ac.ma.

Mbarek Marwan received his engineer degree from ENIM, Morocco, and his Ph.D. in Mathematics and Computer Science from ENSA, Chouaib Doukkali University, Morocco. His research interests primarily focus on the latest advances in cloud computing, big data, and the IoT. He is currently conducting research on the application of these technologies, with particular emphasis on security, encryption, and data protection. He can be contacted at email: marwan.mbarek@gmail.com.

Hassan Silkhi received the master's degree in Computer Science from Faculty of sciences, University Ibn Tofail, Kenitra, Morocco. He is currently pursuing the Ph.D. degree with the L@RI Laboratory, Faculty of Sciences, Ibn Tofail University, Kénitra, Morocco. His research interests include AI, educational guidance, large language models chatbot, and recommender systems. He can be contacted at email: silkhi@gmail.com.

Abdelkarim Ait Temghart received his Ph.D. degree in 2024 from the University Sultan Moulay Slimane, Morocco. His research interests primarily focus on game theory, cloud computing, AI, and the IoT. He is currently conducting research at the Ibn Tofail University, where he explores the application of game-theoretic models to optimize resource allocation in cloud computing environments and IoT networks, with a particular emphasis on edge computing, AI-driven decision-making, and network management. He can be contacted at email: abdelkarim.aittemghart@uit.ac.ma.