UtilitasMathematica

ISSN 0315-3681 Volume 122, 2025

Anomaly-Aware Intrusion Detection Architecture for WSNs Leveraging Hybrid ML Ensembles

[1] Manali Pramod Shinde, [2] Rukmanna Ranchandra Khartmol, [3] Arman Mohammad Ali Undre, [4] Vinod Jagannath Kadam, [5] Munir Bashir Sayyad

[1] Researcher, Dr. Babasaheb Ambedkar Technological University, Lonere, manali.shinde2508@gmail.com

[2] Researcher, Dr. Babasaheb Ambedkar Technological University, Lonere, Khartmol.rukmanna.11et5003@gmail.com

[3] Researcher, Dr. Babasaheb Ambedkar Technological University, Lonere, armanundre@gmail.com

[4] Assistant Professor, Dr. Babasaheb Ambedkar Technological University, Lonere, vjkadam@dbatu.ac.in

[5] Assistant Vice President, Reliance JIO, Navi Mumbai, munirsayyad@gmail.com

Abstract: Wireless sensor networks (WSN) are essential for several monitoring applications; Yet they are prone to security concerns, including unauthorized approach, attacks and other harmful actions that could endanger their reliability. Accepting the intrusion detection systems (IDS) is necessary for timely identification and response to these risks. Multiple data sets such as KDD Cup, NSL KDD, UNSW-NB 15 and AWID data are often used to train and evaluate IDS models. The choice of features is an critical manner for optimizing the electricity of the version, with techniques consisting of selectkbest used with anova F-test that offers significant reduction of functions and improved accuracy. This article examines the implementation of the stacking strategy with both bagging with random forest and boosting the algorithms of the choice tree the usage of these facts units and feature choice techniques. This approach shows great accuracy in all evaluated records units and presents a long lasting hassle solution provided by using security threats in WSN. The finding emphasizes the performance of the file technique to improve IDS overall performance for good sized safety in WSN.

"Index Terms - *WSN, Wi-Fi, NIDS, WIDS attacks, security issues, network threats, feature engineering, multiclass classification, inclusive innovations".*

1. INTRODUCTION

Wireless sensor networks (WSN) are important for cutting-edge communique systems and offer a versatile and green approach of data switch across many applications, inclusive of environmental monitoring, health care and commercial automation. These networks encompass several sensor nodes running in exceptional topologies inclusive of Star, Tree or Mesh configurations, for tracking and sending statistics. The primary capabilities of those sensor nodes consist of sensing, processing, computer era and conversation, permitting them to correctly monitor and adjust extraordinary actual -time systems [1] [2]. WSN affords an efficient and financial opportunity for the transmission of wi-fi network visitors, mainly in areas restricted to power due to the fact they're supposed for low strength use [3]. This makes wi-fi sensor networks mainly beneficial for supervision and protection in remote or inaccessible regions.

However, regardless of their many advantages, WSN is susceptible to some of security worries, consisting of unauthorized technique, assaults and different risky actions which can undermine

the stableness and functioning of the community. Since wi-fi sensor networks are often decided in sensitive regions, the safety of those networks is vital for his or her performance. An unauthorized technique may come from outside and internal resources and constitute a sizable hazard of network integrity and information sent [4] [5]. Given the developing dependence on wireless networks for the essential infrastructure, it's miles vital to create strict safety protocols for the protection of sensor nodes and facts they gather.

Wi-fi-based totally sensors, frequently utilized in WSN, are without problems available because of their capability to hook up with wi-fi networks and transfer statistics to extensive distances via TCP/IP. These sensors can connect to the network using SSID and password, allowing data transfer to servers via URL or IP [6]. If the sensors are outside the Wi-Fi access point range, repetitions can be used to increase coverage and maintaining continuous data transfer. The simplicity of Wi-Fi-based sensors increases their efficiency, but also increases the susceptibility of the network to future violation.

Implementation of IDS is essential to reduce these dangers. The IDS is designed to identify and thwart an unauthorized access to a network control of traffic formulas for unusual behavior indicators. There are two main categories of disruption detection systems: host and network. The host -based ID is built into the device to supervise local processes and user activities, while the network ID ID is implemented in the network to check the traffic indications. The predominant type of ID used in WSNS is based on the network because it allows distracted supervision across the network to identify possible threats [8].

2. RELATED WORK

In recent years, the security of wireless sensor networks (WSNS) has shown as a paramount problem due to their escalating deployment in sensitive applications. The sensitivity of WSN to unauthorized approach and diverse attacks caused the creation of sophisticated IDs designed to alleviate these threats. Many research has advanced in the development of disturbance detection systems for wireless sensor networks, each focusing on various elements, including detection methodologies, energy efficiency and attack categorization.

Boahen et al. [9] They proposed a complex multi-architectural methodology to detect disturbance in online social networks. The research focused on the merger of several architectures to increase the accuracy of detection by merging the deep learning methodologies with the analysis of network behavior. This methodology underlines the need for hybrid models to deal with the complexity of identifying harmful activity in the dynamic environment and provides a significant insight into the efficiency of multi-architectural models to improve IDS performance.

Mahmood et al. [10] have introduced an energy - efficient data fusion methodology for scalable wireless sensor networks using a deep teaching framework. Their research emphasized the importance of energy efficiency in wireless sensor networks, especially in connection with large networks. The authors have shown the use of deep learning algorithms to effectively integrate data from several sensor nodes, which increases energy efficiency while maintaining a high degree of disruption detection accuracy. This examine emphasizes the compromise among electricity consumption and the effectiveness of detection in WSNs, that is a essential consideration in sensible applications with confined assets.

Granato et al. [11] He focused on detecting disturbances inside Wi-Fi networks using a modular and optimized classifier set. Their research has expanded the exploration of current models of disruption detection by introducing a hybrid file that integrates several classifiers to increase the efficiency efficiency. The system improved its ability to identify different attack formulas using the benefits of many classification methods. This study emphasizes the importance of file approaches in increasing the resistance and flexibility of system detection detection systems in wireless networks.

Mahmood et al. [12] They introduced the methodology of intelligent defect detection using a wireless sensor reinforcement system. The authors emphasized the use of strengthening learning to strengthen the detection process by dynamic adjustment of the system to appear and develop threats. This approach allows IDS to adapt to its surroundings and increase its performance over time and therefore ensures increased accuracy and response to new security threats. The use of learning amplification means significant progress in the development of more autonomous and intelligent system detection systems for wireless sensors.

Tao and Xueqiang [13] have added a hybrid approach that has stepped forward the algorithm of the search for sparrows for the purposes of disruption detection. Their technique integrated numerous methodologies for expanding the detection talents of detection systems of disturbance in wireless sensor networks. The hybrid technique integrates algorithms of optimization to enhance the detection system, allowing the system to correctly discover and categorize attacks. This study will increase the prevailing research of optimization techniques for ID and emphasizes its significance in growing the performance and accuracy of these structures.

Singh et al. [14] They examined the use of deep learning to are expecting the quantity of K-Barriers to detect disturbance in the circular place using WSNS. Their research focused on the use of deep learning fashions to are expecting the location of barriers in the community for efficient disruption detection. This method uses deep learning to are expecting network behavior and increase the location of the barrier, increasing the general WSN security. Research introduces an innovative method for a combination of deep getting to know to optimize network topology to improve intrusion detection.

Rajassanranran et al. [15] have delivered a safe and optimized framework to hit upon disturbance by way of LSTM-MAC ideas for underwater wireless sensor networks. The authors created a version that integrates long short term memory (LSTM) network with Medium Access Control (MAC) to increase underwater WSN security. This integration allows the gadget to discover the intrusion and at the identical time boom the network efficiency. Their research gives tremendous information approximately the safety of specialized WSNs, mainly in underwater environments where conventional IDS models won't be right away applicable.

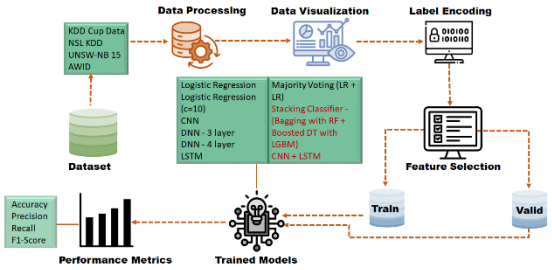
Park et al. [16] They introduced G-UDCS, a graph-oriented detection machine and classification for disturbance for the Controller Area Network (CAN). The authors used graph - based totally methodologies to version community interactions, facilitate disruption identity and category of attacks with more efficiency. This study is specially relevant to WSN, which operates inside specific communique protocols, which represents an

innovative method for detecting attacks within the protocol.

Kandhro et al. [17] They focused on identifying enemy disturbances and real -time assaults within the frames of cyber security with IoT aid. They proposed a intrusion detection structure that consists of the Internet of Things with Function Functions in actual time. The authors tried to enhance the protection of complicated structures by means of the usage of IoT and ML to stumble on harmful activities in actual time. Their studies is specially critical for wi-fi sensor networks with IoT support, wherein the dynamic environment requires instant detection and reaction to safety threats.

3. MATERIALS AND METHODS

The proposed system seeks to expand WSN security by creating improved IDs that are able to identify a variety of security threats. The system uses a number of established data sets, including KDD Cup data [20], NSL KDD, UNSW-NB 15 and AWID [21], for training and evaluation of IDS models. The selection technique uses selectkbest with anova F-test to find out the most important features for classification and therefore increase the efficiency of the model. A number of machine learning and deep learning algorithms, including logistics regression [18], logistics regression with regularization parameter (C = 10), convolution neural networks [19] (CNN), deep neural networks (DNN) with three and four layers and long short -term memory (LSTM). In addition, hybrid models such as CNN + LSTM and majority vote (integration of logistics regression models) will be evaluated. To improve the performance of the detection in the system, a stacking classifier with random forest and strengthening the decision -making tree will be used.



"Fig.1 Proposed Architecture"

The design uses machine learning to detect disturbance in wireless sensor networks. The procedure begins with the processing and visualization of data, followed by label coding and selecting functions. The data file is divided into subset of training and validation. Various machine learning models are trained and evaluated, including logistics regression, CNN, DNN and LSTM. In addition, file methods such as majority vote and stacking classifiers are used to improve the efficiency of the model. The final model is selected on the basis of performance criteria, such as accuracy, precision, recall and score F1.

i) Dataset Collection:

This study uses two important data sets, KDD, NSL KDD, UNSW-NB15 and AWID, to assess the effectiveness of detection systems in wireless sensor networks.

The AWID data file [21] has 313 248 items and 84 attributes, mostly aimed at detecting wireless disruption. After selecting the functions, the data file contains the following attributes: 'frame.offset_shift', 'frame.time_epoch', 'frame.time_delta', 'frame.time_delta_displayed', 'frame.time_relative', 'frame.len', 'frag.len', ' 'wlan.fc.ds' and 'wlan.fc.frag'. These features include critical attributes of wireless images, including timing, frame length, and designated symptoms that offer significant knowledge to identify network inconsistencies and likely to enter wireless networks.

	frame.interface_id	frame.dlt	frame.offset_shift	frame.time_epoch
0	?	0.0	1.393668e+09	0.000000
0	?	0.0	1.393668e+09	0.001056
0	?	0.0	1.393668e+09	0.000141
0	?	0.0	1.393668e+09	0.000156
0	?	0.0	1.393668e+09	0.001842

5 rows × 154 columns

"Fig.2 Dataset Collection Table – AWID-CLS-R-Tst"

The KDD data file [20], which concerns the network penetration, has 125 973 items and 42 characteristics. Selection of functions after function, data file contains the following selected features: "logged_in", 'root_shell', 'serror_rate', 'srv_serror_rate', 'Same_srv_rate', 'dst_host_srate', 'dst_host_same_src_port_srame' and 'dst_host_srv_serror_rate'. These properties are necessary to identify attacks and network disruption and offer a targeted number of properties to increase the efficiency of disruption detection systems.

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment
0	0	tcp	ftp_data	SF	491	0	0	0
1	0	udp	other	SF	146	0	0	0
2	0	tcp	private	S0	0	0	0	0
3	0	tcp	http	SF	232	8153	0	0
4	0	tcp	http	SF	199	420	0	0

5 rows × 42 columns

"Fig.3 Dataset Collection Table – KDDCUP"

NSL KDD data set (which is focused on intrusion detection) contains 125,972 entries and 43 attributes. After selecting the functions, the dataset contains the following features that have been selected: 'logged in', 'root shell', 'serror rate', 'srv serror rate', 'same srv rate', 'dst host srv count', 'dst host same srv rate', 'dst host same src port rate', 'dst host serror rate' and 'dst host srv serror rate'. These homes are important in terms of identifying community intrusions and intrusions and assaults and such an array of variables offers advanced proposed settings in improving the accuracy in the models of intrusion.

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot
0	0	udp	other	SF	146	0	0	0	0	0
1	0	tcp	private	S0	0	0	0	0	0	0
2	0	tcp	http	SF	232	8153	0	0	0	0
3	0	tcp	http	SF	199	420	0	0	0	0
4	0	tcp	private	REJ	0	0	0	0	0	0

5 rows × 43 columns

"Fig.4 Dataset Collection Table – NSL-KDD"

UNSW-NB15 dataset contains 82,332 gadgets and 45 traits which are community-oriented gadget intrusion detection. After typical selection, the data set is updated with the following basic properties: such as rate, sttl, swin, dwin, ct_srv_src, ct state ttl, ct_src dport ltm, ct_dst sport ltm, ct_dst src ltm and ct_srv dst. These features are essential in determining the attacks in the community, the aggregation of other numerous visitor characteristics, and the provision of the glide dynamics in the community to improve the accuracy of intrusion surveying diversion models.

	id	dur	proto	service	state	spkts	dpkts	sbytes	dbytes	rate	...	ct_dst_sport_ltm
0	1	0.000011	udp	-	INT	2	0	496	0	90909.0902	...	1
1	2	0.000008	udp	-	INT	2	0	1762	0	125000.0003	...	1
2	3	0.000005	udp	-	INT	2	0	1068	0	200000.0051	...	1
3	4	0.000006	udp	-	INT	2	0	900	0	166666.6608	...	1
4	5	0.000010	udp	-	INT	2	0	2126	0	100000.0025	...	1

5 rows × 45 columns

"Fig.5 Dataset Collection Table – UNSW-NB15"

ii) Pre-Processing:

a) Data Processing: It enhances its data file by removing those values that are zero and overlapping constituents to ensure that the data is of quality and reliability. The steps correct the abnormalities, such as the absent data, and redundant items and reset the index of the data file that allows continued analysis. The model-friendly curatorial data file is saved by keeping the predominantly relevant items and normalizing class proportions, and thus offer us the powerful and objective data to be used in machine learning.

b) Data Visualization: The data visualization also entails analysis of a distribution of classes with the use of column graphs and provides knowledge of category balance within the data

file. Its thermal map helps to demonstrate the dependencies on the numerical features, identify potential dependencies and addictions. These graphical tools allow understanding the data files structure better, identify the formulas and a direction of the element selection in creating the models.

c) Label Encoding: labels that are categorical data such as the classes are labeled into a numeric form by coding the labels in order to present conformity to the machine learning algorithms. Every one of the unique classes is awarded a certain integer value which enables the model to conceptualize and accommodate input efficiently. This stage ensures an effective incorporation of categorical features in training process.

d) Feature Selection: The most pertinent properties are identified with the help of statistical methods. SelectKBest method involves the analysis of the variance (ANOVA) and scoring functions based on their significance and determining the 10 best predictors of the intended variable. This makes the dimension, improves the efficiency in the calculation and strengthens the accuracy of the model by focusing on the simple features.

iii) Training & Validation:

The data file is divided into subset training and testing to evaluate the efficiency of machine learning models. The data segment is used for training, allowing the model to recognize formulas and correlation between characteristics. Residual data is assigned to testing and facilitates the assessment of the accuracy, accuracy and generality of the model. This division guarantees a comprehensive evaluation of the skills of the model's prediction on unknown data.

iv) Algorithms:

DNN-3Layer: a deep neural network with three layers is used to detect complicated patterns in the data using the advantage of multiple layers of neurons. It is also effective at capturing the non linear relationship and very suitable in explaining complexities of relationships between the features giving high accuracy of predictive modeling.

DNN-4Layer: An extra layer is added to the results of the study called the four-layer deep neural network that will offer more feature extraction. It is applicable to the problems that demand a high degree of representation learning and addressing the difficulties in an ultra-high space of data.

CNN: Convolutional Neural Networks work best at extracting features particularly structured/ grid gallery data. Their convolutional layers [19] find spatial hierarchy, whereas the pooling layers diminish dimensionality, and are thus very useful in capturing localised features.

LSTM: Long Short-Term Memory networks are networks that deal with sequential data preserving long-term dependencies. Pi-kernels Their gating devices avoid gradient challenges, making them very accurate at studying time-series data or time-sensitive datasets.

CNN+LSTM: The model is a combination of the spatial representations learned with a CNN with the capability of capturing temporal dependences gathered by LSTM. The architecture is best suited with the mixed types of tasks which deal with spatial and sequential information which brings in additional performance of improving predictive performance and feature convergence.

Logistic Regression: This is a linear model that analyses the relationships that exist between features, and binary outcomes. Its interpretability and simplicity enables the use of

it to comprehend the importance of the features and attain the baseline predictive performance.

Regularization (C=10): modification of the strength of the regularizer by setting C=10 minimizes overfitting on feature weights. It has a good trade off between complexity and generalization of models and makes predictions reliably.

Majority Voting (LR + LR (C=10)): The ensemble method integrates the outcome of two logistic regressions, which are trained with different regularization parameters. It also improves decision accuracy and consistency as it uses the complementary advantages of each of the models.

Stacking Classifier: This is an ensemble method that combines the bagging feature of the Random Forest and the boosting feature of Decision Tree. It learns multiple features relationships enhancing the performance of prediction with complementary learning.

4. RESULTS & DISCUSSION

Accuracy: A test capacity towards create a proper difference between healthy & sick cases is a measure of accuracy. We can determine accuracy of a test through calculating proportion of cases undergoing proper positivity & genuine negative. It is possible towards express this mathematically:

$$\text{"Accuracy"} = \frac{\text{"TP + TN"}}{\text{"TP + FP + TN + FN"}} (1)$$

Precision: Precision quantifies the percentage of efficiently identified positive cases or samples. Precision is decided by using the components:

$$\text{"}Precision\text{"} = \frac{\text{"True Positive"}}{\text{"True Positive + False Positive"}} (2)$$

Recall: ML recall assesses a model's potential to choose out all relevant times of a class. It demonstrates a version's efficacy in encapsulating times of a class by using comparing nicely anticipated high satisfactory observations to the general variety of positives.

$$\text{"}Recall\text{"} = \frac{\text{"TP"}}{\text{"TP + FN"}} (3)$$

F1-Score: The accuracy of a system ML of model is classed the usage of the F1 score. Integrating the precision and do not forget metrics of the model. The accuracy metric quantifies the frequency of proper predictions made through a model at some level inside the dataset.

$$\text{"}F1\ Score\text{"} = \text{"2"} * \frac{\text{"}Recall\ \text{X}\ Precision\text{"}}{\text{"}Recall + Precision\text{"}} * \text{"100"} (4)$$

Tables 1 to 4 assess the "performance metrics—accuracy, precision, recall, and F1-Score"—for each method. The "Stacking Classifier" routinely surpasses all other algorithms across all measures. The tables provide a comparative examination of the metrics for the alternative methods.

"Table.1 Performance Evaluation Metrics for AWID-CLS-R-Tst"

ML Model	Accuracy	Precision	Recall	F1_score
DNN-3Layer	0.977	0.767	0.998	0.861
DNN-4Layer	0.959	0.759	0.999	0.856
CNN	0.973	0.974	0.968	0.970
LSTM	0.971	0.765	0.975	0.845
CNN+LSTM	0.976	0.976	0.974	0.974
Logistic Regression	0.344	1.000	0.344	0.511
Logistic Regression(C=10)	0.976	0.978	0.976	0.976
Ensemble Model	0.976	0.978	0.976	0.976
Stacking Classifier	**0.989**	**0.989**	**0.989**	**0.989**

"Table.2 Performance Evaluation Metrics for KDDCUP"

ML Model	Accuracy	Precision	Recall	F1_score

DNN-3 Layer	0.864	0.483	0.955	0.637
DNN-4 Layer	0.860	0.562	0.973	0.704
CNN	0.844	0.790	0.668	0.701
LSTM	0.826	0.788	0.885	0.816
CNN+LSTM	0.888	0.794	0.594	0.660
Logistic Regression	0.830	0.849	0.830	0.831
Logistic Regression(C=10)	0.832	0.851	0.832	0.833
Ensemble Model	0.831	0.850	0.831	0.832
Stacking Classifier	**0.938**	**0.943**	**0.938**	**0.940**

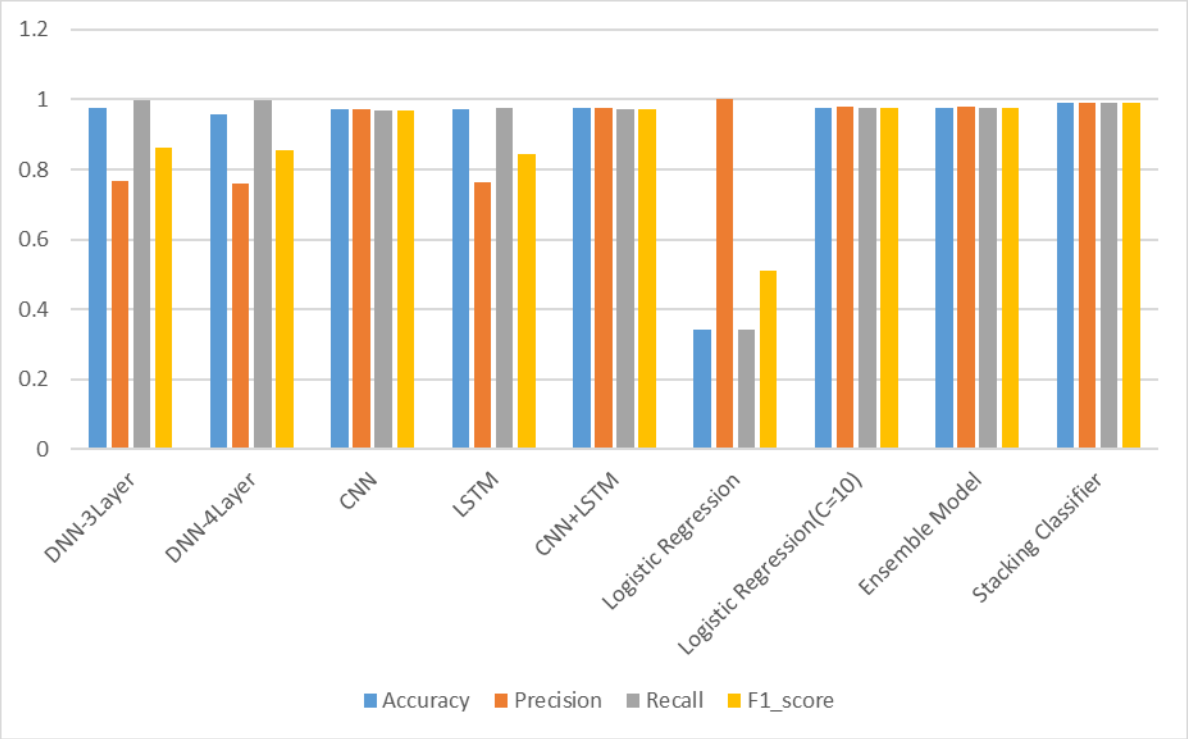
"Table.3 Performance Evaluation Metrics for NSL KDD"

ML Model	Accuracy	Precision	Recall	F1_score
DNN-3 Layer	0.858	0.512	0.982	0.666
DNN-4 Layer	0.819	0.495	0.959	0.645
CNN	0.833	0.769	0.574	0.639
LSTM	0.812	0.685	0.914	0.769
CNN+LSTM	0.859	0.772	0.557	0.628
Logistic Regression	0.817	0.835	0.817	0.818
Logistic Regression(C=10)	0.826	0.843	0.826	0.828
Ensemble Model	0.820	0.839	0.820	0.821
Stacking Classifier	**0.932**	**0.935**	**0.932**	**0.933**

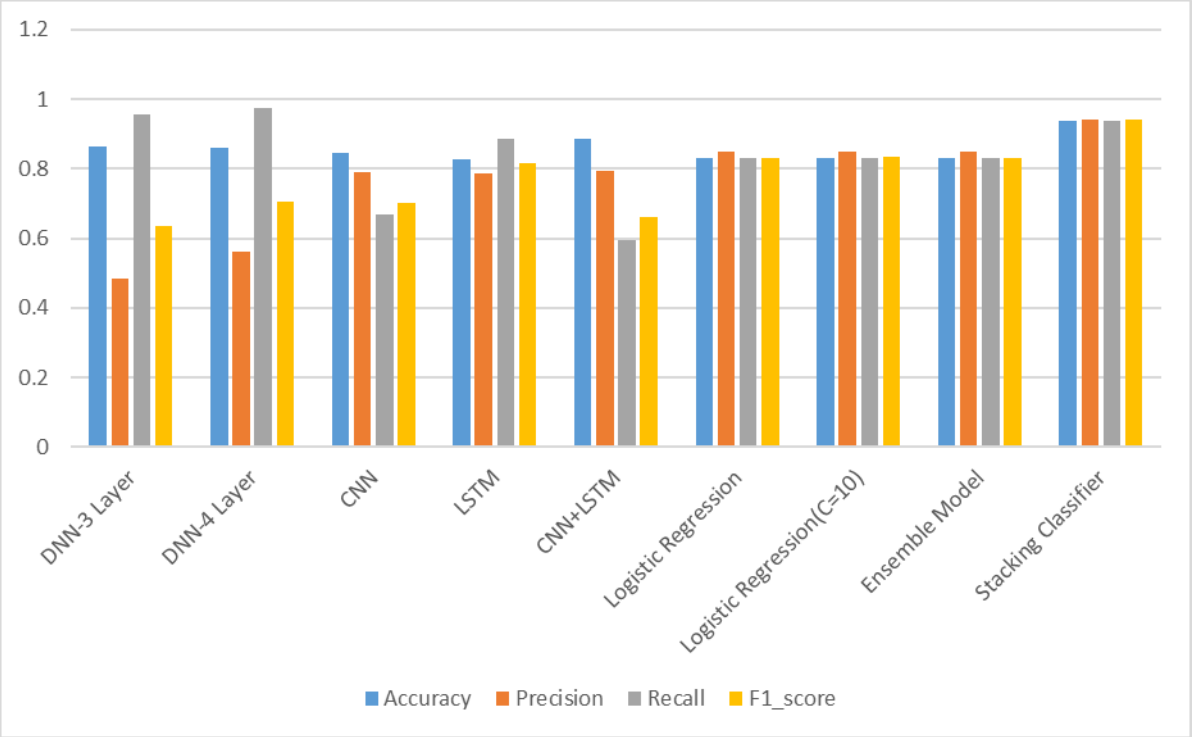
"Table.4 Performance Evaluation Metrics for UNSW-NB15"

ML Model	Accuracy	Precision	Recall	F1_score
DNN-3 Layer	0.833	0.712	0.581	0.608
DNN-4 Layer	0.834	0.624	0.644	0.606
CNN	0.860	0.867	0.886	0.874
LSTM	0.825	0.846	0.779	0.801
CNN+LSTM	0.869	0.869	0.871	0.870
Logistic Regression	0.720	0.722	0.720	0.720
Logistic Regression(C=10)	0.823	0.823	0.823	0.823
Ensemble Model	0.818	0.819	0.818	0.818
Stacking Classifier	**0.938**	**0.938**	**0.938**	**0.938**

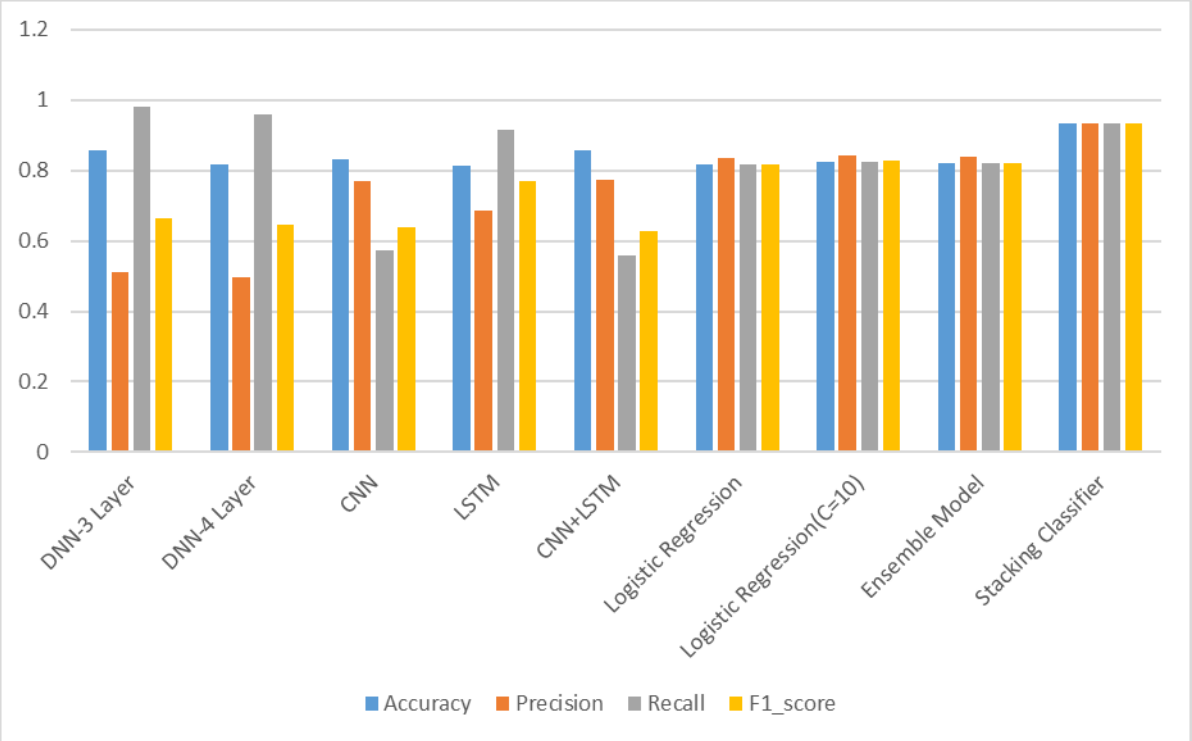
"Graph.1 Comparison Graphs for AWID-CLS-R"



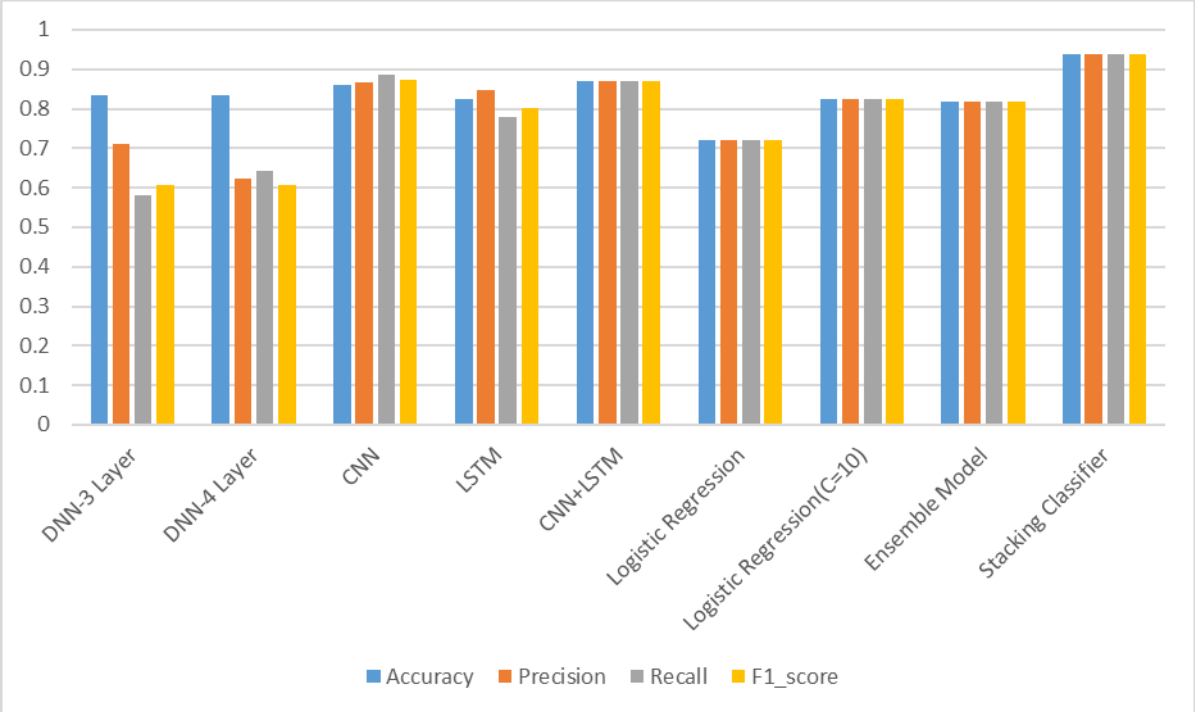
"Graph.2 Comparison Graphs for KDDCUP"



"Graph.3 Comparison Graphs for NSL KDD"



"Graph.4 Comparison Graphs for UNSW-NB15"



The accuracy is displayed in blue, orange precision, a recall in gray, F1-score in yellow and AUC in blue in graphs 1 to 4. Due to other models, the stacking classifier shows increased performance in all measures and achieves the highest values. Graphs above graphically these results graphically represent.

5. CONCLUSION

The Machine Learning IDS for WSN significantly increases safety by precise identification and attacking attacks. The IDS, using WSN vulnerability to attack and unauthorized access, uses various data sets-KDD Cup Data, NSL-KDD, UNSW-NB15 and AWID-train models for detection. The key element of its implementation

is the selection of elements made by methods such as Selectkbest and Anova F-test, which optimize data for increased model efficiency. Research emphasizes the use of stacking classifier, integration of bagging with random forest and boosted decision tree algorithms to increase the efficiency of detection. This file technique achieves a remarkable accuracy of 93% on three data sets and demonstrates exceptional performance with an accuracy of 98.9% in the AWID data file, emphasizing its reliability and efficiency. By integrating sophisticated machine learning methodologies with refined data sets, IDS offers a resistant solution for solving security problems in WSN, which guarantees rapid detection, efficient reaction and improved reliability for basic applications. The results illustrate the transformation ability of the file to protect WSN settings from malicious events.

In the future, the effectiveness of IDS for WSN may be extended by integrating the methods of modern machine learning, including the approaches of deep learning to increase accuracy and facilitate real -time detection. Furthermore, exploring hybrid models that combine detection techniques based on anomaly and signature - based detection can increase the efficacy of threat identification. Increasing the capacity of the system to manage larger and more diverse data sets and the integration of adaptive learning mechanisms for the development of offensive formulas will retain its relevance and effectiveness in WSN protection over time.

REFERENCES

[1] M. Mittal, R. P. de Prado, Y. Kawai, S. Nakajima, and J. E. Muñoz-Expósito, ‘‘Machine learning techniques for energy efficiency and anomaly detection in hybrid wireless sensor networks,’’ Energies, vol. 14, no. 11, p. 3125, May 2021.

[2] K. Haseeb, N. Islam, T. Saba, A. Rehman, and Z. Mehmood, ‘‘LSDAR: A light-weight structure based data aggregation routing protocol with secure Internet of Things integrated next-generation sensor networks,’’ Sustain. Cities Soc., vol. 54, Mar. 2020, Art. no. 101995.

[3] R. Guetari, H. Ayari, and H. Sakly, ‘‘Computer-aided diagnosis systems: A comparative study of classical machine learning versus deep learning based approaches,’’ Knowl. Inf. Syst., vol. 65, no. 10, pp. 3881–3921, Oct. 2023.

[4] R. Ramadan and K. Medhat, ‘‘Intrusion detection based learning in wireless sensor networks,’’ PLOMS AI, vol. 2, no. 1, pp. 1–20, 2022.

[5] W.Wang, S.Jian, Y. Tan, Q. Wu, and C.Huang, ‘‘Representation learning based network intrusion detection system by capturing explicit and implicit feature interactions,’’ Comput. Secur., vol. 112, Jan. 2022, Art. no. 102537.

[6] S. Mujeeb, T. A. Alghamdi, S. Ullah, A. Fatima, N. Javaid, and T. Saba, ‘‘Exploiting deep learning for wind power forecasting based on big data analytics,’’ Appl. Sci., vol. 9, no. 20, p. 4417, Oct. 2019.

[7] S. M. Kasongo and Y. Sun, ‘‘A deep gated recurrent unit based model for wireless intrusion detection system,’’ ICT Exp., vol. 7, no. 1, pp. 81–87, Mar. 2021.

[8] A. Wajahat, J. He, N. Zhu, T. Mahmood, A. Nazir, F. Ullah, S. Qureshi, and S. Dev, ‘‘Securing Android IoT devices with GuardDroid transparent and lightweight malware detection,’’ Ain Shams Eng. J., vol. 15, no. 5, May 2024, Art. no. 102642.

[9] E. K. Boahen, S. A. Frimpong, M. M. Ujakpa, R. N. A. Sosu, O. Larbi-Siaw, E. Owusu, J. K. Appati, and E. Acheampong, ‘‘A deep multi-architectural approach for online social network intrusion detection system,’’ in Proc. IEEE World Conf. Appl. Intell. Comput. (AIC), Jul. 2022, pp. 919–924.

[10] T. Mahmood, J. Li, T. Saba, A. Rehman, and S. Ali, ‘‘Energy optimized data fusion approach for scalable wireless sensor network using deep learning-based scheme,’’ J. Netw. Comput. Appl., vol. 224, Apr. 2024, Art. no. 103841.

[11] G. Granato, A. Martino, L. Baldini, and A. Rizzi, ‘‘Intrusion detection in Wi-Fi networks by modular and optimized ensemble of classifiers: An extended analysis,’’ Social Netw. Comput. Sci., vol. 3, no. 4, p. 310, Jul. 2022.

[12] T. Mahmood, J. Li, Y. Pei, F. Akhtar, S. A. Butt, A. Ditta, and S. Qureshi, ‘‘An intelligent fault detection approach based on reinforcement learning system in wireless sensor network,’’ J. Supercomput., vol. 78, no. 3, pp. 3646–3675, Feb. 2022.

[13] L. Tao and M. Xueqiang, ‘‘Hybrid strategy improved sparrow search algorithm in the field of intrusion detection,’’ IEEE Access, vol. 11, pp. 32134–32151, 2023.

[14] A. Singh, J. Amutha, J. Nagar, and S. Sharma, ‘‘A deep learning approach to predict the number of k-barriers for intrusion detection over a circular region using wireless sensor networks,’’ Expert Syst. Appl., vol. 211, 2023, Art. no. 118588.

[15] S. Rajasoundaran, S. V. N. S. Kumar, M. Selvi, K. Thangaramya, and K. Arputharaj, ‘‘Secure and optimized intrusion detection scheme using LSTM-MAC principles for underwater wireless sensor networks,’’ Wireless Netw., vol. 30, no. 1, pp. 209–231, 2024.

[16] S. B. Park, H. J. Jo, and D. H. Lee, ‘‘G-IDCS: Graph-based intrusion detection and classification system for CAN protocol,’’ IEEE Access, vol. 11, pp. 39213–39227, 2023.

[17] I. A. Kandhro, S. M. Alanazi, F. Ali, A. Kehar, K. Fatima, M. Uddin, and S. Karuppayah, "Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures,’’ IEEE Access, vol. 11, pp. 9136–9148, 2023.

[18] M. A. Rahman, A. T. Asyhari, O. W. Wen, H. Ajra, Y. Ahmed, and F. Anwar, ‘‘Effective combining of feature selection techniques for machine learning-enabled IoT intrusion detection,’’ Multimedia Tools Appl., vol. 80, no. 20, pp. 31381–31399, Aug. 2021, doi: 10.1007/s11042 021-10567-y.

[19] B. Alenazi and H. E. Idris, ‘‘Wireless intrusion and attack detection for 5G networks using deep learning techniques,’’ Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 7, pp. 1–6, 2021.

[20] KDD Dataset, Intrusion detection dataset, Available at: https://www.kaggle.com/datasets/toobajamal/kdd99-dataset

[21] Zhiqing Cui, AWID-CLS-R, Available at: https://www.kaggle.com/datasets/zhiqingcui/awidclsr