

Adaptive Ensemble Learning Framework for Intrusion Detection Across IoT and IIoT Layers Using Multi-Dataset Evaluation

Ms. R Padma Sahithi¹, Ms. B Yamini Supriya²

¹Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India
Phone: +91 6302852375

Email: sahithiramayanam@gmail.com

²Assistant Professor, Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India
Phone: +91 9885732323

Email: yamini.bommiseti@gmail.com

Received: 5th January, 2026; Revised: 18th January 2026; Accepted: 25th January, 2026; Available Online: 28th January, 2026

ABSTRACT

With the explosive growth of the IoT and IIoT, the demand for lightweight, scalable and intelligent intrusion detection systems has risen. The evaluation of a comprehensive ML and DL-based intrusion detection system (IDS) presented in this study is based on the ToN_IoT, WUSTL-IIoT 2021, and Edge-IIoTset datasets. Data preparation involves outlier removal, chi-squared feature selection, and SMOTE and undersampling for handling imbalance. Algorithms used included Convolutional Neural Networks (CNN), Deep Neural Networks (DNN), Decision Tree, RF, LightGBM, Bagging, stacking ensemble with RF, LightGBM, and Decision Tree, and a voting ensemble with Boosted Decision Tree, Bagging RF, and XGBoost. The experimental results indicate that voting ensemble had a maximum accuracy of 98.4% on ToN_IoT, 98.9% on WUSTL-IIoT 2021, and 96.7% on Edge-IIoTset, while CNN, DNN, RF, Decision Tree, and LightGBM achieved an accuracy of 93.8%, 94.4%, 96.3%, 96.6%, and 96.6%, respectively. Although deployment with Flask and SQLite can provide safe, real-time intrusion detection, visualization, and monitoring of a distributed industrial network around the globe, explainable AI techniques such as SHAP and LIME provide feature-level interpretability.

Keywords: Internet of Things, Industrial Internet of Things, security, cyber-attacks, intrusion detection, ML, lightweight, data balancing, cross-dataset transfer learning.

How to cite this article: R Padma Sahithi. Adaptive Ensemble Learning Framework for Intrusion Detection Across IoT and IIoT Layers Using Multi-Dataset Evaluation. *Int J Drug Deliv Technol.* 2026;16(85s):1580-1586. DOI: 10.25258/ijddt.16.85s.168.

Source of support: Nil.

Conflict of interest: None

I. INTRODUCTION

With the growth of the IoT and IIoT automation and data-driven insights are revolutionizing industries. The magnitude of this ecosystem can be illustrated by the fact that by 2025 there will be more than 55.7 billion gadgets connected globally [1]. There are, however, significant risks to cyber security implications with this growth. The devices connected to the Internet were vulnerable, as 33% of all impacted devices were IoT devices in 2020 [2]. With the advent of Industry 4.0, which fuses IT and OT in CPS, critical infrastructures such as healthcare, energy, transportation, and manufacturing are increasingly exposed to cyber threats. As indicated, by the year 2025 cybercrime will cost \$10.5 trillion annually, of which IoT and IIoT attacks will be a major factor. Average cost of an industrial data breach is expected to be \$4.24 million [3].

Each of the layers that comprise the layered architecture of IIoT, including the Edge, Middleware, Application, IT/OT, and Cloud Services layers, are vulnerable in different ways. The applications are susceptible to virus attack, the middleware to illegal API access and the edge layer to physical manipulation. The IT/OT Layer suffers from insider threats [4] [5] whereas the Cloud Layer faces

continuous threats and serious compromises. The concerns are backed by real-world attacks, such as Stuxnet (2010), an attack on SCADA systems; Triton (2017), aimed at disrupting industrial safety procedures; and the Mirai botnet (2016), which attacked SCADA systems via IoT devices.

IDS based on ML have become more popular in order to counter these threats. Unlike traditional signature-based IDS, which can only identify attacks they have a signature for, ML can be used to identify new and previously unseen threats by learning patterns and abnormal behavior. ML is especially useful in the dynamic and diverse IoT/IIoT ecosystem because of its flexibility [6]. A number of benchmark datasets, such as UNSW-NB15, CICIDS, DS2OS, N-BaIoT, Bot-IoT [7], X-IIoTID [9], and LATAM-DDOS IOT [10], have been developed to aid in the development of IDS. TON_IoT, WUSTL-IIoT-2021 and especially Edge-IIoTset are important as they simulate real industrial scenarios. Despite their utility, there remain challenges like the high degree of imbalance in the classes (normal traffic far outweighs anomaly traffic), and limited resources on IIoT devices that requires lightweight and effective models [8].

This study addresses these issues by analyzing past IDS studies conducted on these datasets, training and testing supervised and ensemble ML methods, and demonstrating real-time traffic monitoring for detecting intrusions in IoT and IIoT networks.

II. RELATED WORK

Intrusion detection in IoT and IIoT systems has been investigated extensively, with the application of ML and DL methods, to counter the severity of attacks. A reinforcement learning based IDS for SCADA systems was proposed by Mesadieu, Torre and Chennamaneni [11] which demonstrated that it can adapt to dynamic attacks with little reliance on labeled data. Eid, Soudan, Nassif and Injadat [12] investigated the importance of preprocessing and balancing methods for improving the performance of classifiers on unbalanced datasets. In a related work, Eid et al. [13] reported that minority class recognition in the related work could be improved using an optimized CNN with multi-class SMOTE, which showed that DL and synthetic oversampling improve the detection accuracy.

Lightweight IDS has also been suggested which are suitable for IoT nodes with limited resources. Farea and Küçük [14] developed a ML based IDS that focused on the practical approach in a restricted environment using the Cooja simulator. In this context, Alshehri, Saidani, Alrayes, Abbasi, and Ahmad [15] proposed a deep CNN with self-Attention mechanism to capture long-term dependencies in traffic for IIoT which outperforms traditional CNNs by a large margin.

With survey research, you get more holistic perspectives. IDS techniques used for IoT/IIoT were discussed by Bansal and Singhrova [16] who observed issues of scalability, explainability and lightweight deployment. Nuaimi, Fourati and Hamed in a systematic study [17] divided IDS into ML, DL and hybrid models, pointing to the lack of research regarding explainability and cross-dataset validation. By pointing out the shortcomings of the current IDS development, these assessments direct future study.

In future, datasets will remain important tools in the IDS research toolkit. Guo, Pan, Liu, Li, Pei and Hu [18] evaluated the models for use as a benchmark for actual IIoT traffic by using the TON_IoT dataset. Belarbi, Spyridopoulos, Anthi, Mavromatis, Carnelli and Khan [19] improved privacy-preserving IDS by introducing a federated DL system that does not require sharing of raw data for distributed training. In decentralized IoT networks, their method guaranty security and scalability.

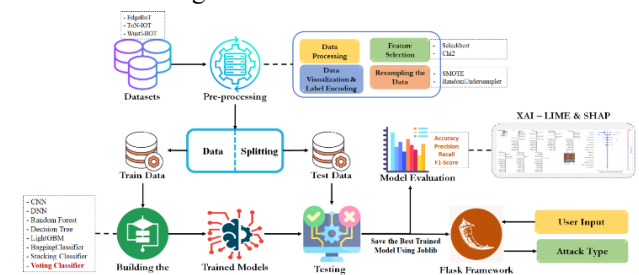
Explainability and robustness have been demonstrated to be key factors for the adoption of IDS in safety critical environments. Oseni, Moustafa, Creech, Sohrabi, Strelzoff, Tari and Linkov [20] developed an explainable DL IDS for IoT mobility. They also tackled trust and accountability concerns by incorporating interpretable AI with DL to ensure transparency in their predictions.

The literature broadly reflects a shift from conventional machine learning toward powerful deep learning, federated learning, reinforcement learning and explainable artificial intelligence. Research is geared towards data management, ensuring lightweight deployment of IoT devices, and privacy and transparency concerns. Overall, these studies

highlight the progress and challenges in developing effective IDS for IoT and IIoT networks.

III. MATERIALS AND METHODS

The proposed approach provides an efficient intrusion detection system for IoT and IIoT applications using benchmark datasets - ToN-IoT, WUSTL-IIoT-2021 and Edge-IIoTset. Apart from the baseline models for the reference evaluation, it also includes some ML and DL models like [21] Random Forest, DT, LightGBM, Bagging Classifier, Stacking Classifier, XGBoost, CNN, and DNN. To evaluate these algorithms' prediction powers in identifying anomalies and harmful activity, the framework methodically trains and tests them. An interface for real time user interactions and intrusion prediction is developed using Flask to promote usability. The system also comprises of a [22] Voting Classifier ensemble, class balance enhancement techniques, and explainable AI methods such as SHAP and LIME to ensure interpretability and provide feature-level insights.



“Fig. 1. System Architecture”

A ML pipeline for IoT security is shown in Fig. 1, where the data is fed into the ML model after it has been pre-processed, feature selected (Chi2) and resampled (SMOTE). Data is split into several models, and then they are trained and evaluated. The optimal model is selected with Joblib according to the evaluation metrics and implemented in Flask for interpreting and real-time attack detection using XAI.

A) Dataset Collection

i) *EdgeIIoT*: A benchmark dataset for intrusion detection in IoT and IIoT networks is called EdgeIIoT. It stores the network traffic and protocol-level details such as TCP, UDP, DNS, MQTT, Modbus connections, etc. 55 characteristics, 32404 occurrences. The dataset contains seven classes of attacks: DDoS (UDP, ICMP, TCP), port scan, vulnerability scan, password attack, ransomware and port scan. It has unique characteristics of dangerous traffic patterns, making it a useful reference for real time intrusion detection system to test ML and DL models.

	frame.time	ip.src.host	ip.dst.host	arp.dst.proto.ipv4	arp.opcode	arp.hw.size	arp.src.proto.ipv4	icmp.checksum	icmp.seq.le
0	1.707159e+09	0	0	0	0	0	0	0	0
1	1.707159e+09	0	0	0	0	0	0	0	0
2	1.707159e+09	0	0	0	0	0	0	0	0
3	1.707159e+09	0	0	0	0	0	0	0	0
4	1.707159e+09	0	0	0	0	0	0	0	0

5 rows × 10 columns

Fig. 2. EdgeIIoT Dataset

ii) *ToN-IoT*: With 190,474 instances and 20 features, the ToN-IoT dataset is a large-scale benchmark dataset for intrusion detection in IoT systems. It contains a range of network features such as DNS settings, web traffic features, ports, protocols, packet statistics and connection states. Ten categories—normal, scanning, DDoS, injection, password, DoS, backdoor, XSS, ransomware, and MITM attacks—are included in the dataset. ToN-IoT has been a crucial tool

for training and testing intrusion detection models, offering a balanced representation of diverse and heterogeneous IoT traffic and attacks.

	src_ip	src_port	dst_ip	dst_port	proto	service	duration	src_bytes	dst_bytes	conn_state	...	http_response_body_len	http_status_code
0	192.168.1.37	4444	192.168.1.193	49178	tcp	-	290.371539	101568	2592	OTH	...	0	0
1	192.168.1.193	49180	192.168.1.37	8080	tcp	-	0.000102	0	0	REJ	...	0	0
2	192.168.1.193	49180	192.168.1.37	8080	tcp	-	0.000148	0	0	REJ	...	0	0
3	192.168.1.193	49180	192.168.1.37	8080	tcp	-	0.000113	0	0	REJ	...	0	0
4	192.168.1.193	49180	192.168.1.37	8080	tcp	-	0.000130	0	0	REJ	...	0	0

5 rows × 44 columns

Fig. 3. ToN-IoT Dataset

iii) *WUSTL-IIoT*: The WUSTL-IIoT 2021 contains 1,194,464 records and 45 attributes, making it a comprehensive reference for intrusion detection in IIoT systems. It stores specific network flow parameters such as packet counts, byte counts, jitter, loss rates, protocol behavior, and application-level metrics. The data consists of 5 types of traffic: regular traffic, DoS traffic, reconnaissance traffic, command injection traffic, and backdoor attacks. With a broad spectrum of genuine IoT network traffic, WUSTL-IIoT 2021 provides a solid ground to evaluate ML and DL models used for intrusion detection research.

StartTime	EndTime	SrcAddr	DestAddr	Mean	Spdrt	SrcPktS	DestPktS	TotPktS	...	SrcBytes	DestBytes	TotBytes	SpdRct	SpdDct	SpdTot	...	SrcRct	DestRct	TotRct
2019-08-18 12:23:28	2019-08-18 12:23:28	192.168.0.20	192.168.0.2	0	59834	502	10	8	18	...	24	35	44	0.001176	0.02007	0	0.000000	0.0	normal
2019-08-18 15:15:34	2019-08-18 15:15:34	192.168.0.20	192.168.0.2	0	55641	502	10	8	18	...	24	35	44	0.001388	0.02081	0	0.000000	0.0	normal
2019-08-18 15:41:31	2019-08-18 15:41:31	192.168.0.20	192.168.0.2	0	62778	502	10	8	18	...	24	35	44	0.000890	0.01703	0	0.000000	0.0	normal
2019-08-18 15:45:32	2019-08-18 15:45:32	205.246.215.82	192.168.0.2	0	61771	83	4	0	4	...	0	0	0	0.000000	0.88955	0	0.47810813	0.0	DoS
2019-08-18 14:48:44	2019-08-18 14:48:44	192.168.0.20	192.168.0.1	3	0	0	14	0	14	...	476	0	476	0.000000	3.50005	0	0.525146582	0.0	normal

5 rows × 49 columns

Fig. 4. Wustl-IIoT Dataset

B) Pre-Processing:

i) *Data preprocessing*: Data preparation ensures data is clean, consistent, and reliable by removing null values, duplicates, and outliers. Outliers are removed to preserve realistic traffic patterns, duplicate entries are removed to prevent bias and null and corrupted entries are removed. Noise, dimensionality and computational costs are minimized by removing redundant columns such as IP addresses, or time stamps, so that models can focus on the important factors when detecting and preventing intrusions.

ii) *Data Visualization*: With the assistance of visualization of data, one may evaluate the link between features and traffic patterns of IoT/IIoT data sets. Outcome visualizations show distributions of attack types and traffic imbalances, and correlation matrices show mutable dependencies, multicollinearity and redundancy. These insights are useful for effective feature engineering, selection and preprocessing to guarantee models learn from meaningful and representative data patterns for reliable performance in intrusion detection.

iii) *Label Encoding*: An ML models technique used to encode categories into numbers is called label encoding. They are given a unique integer to allow efficient processing by algorithms including identifying the attributes of the protocol, service or attack category. This transformation ensures that the categorical features are used in an appropriate manner, enabling the correct learning and prediction in IoT/IIoT intrusion detection systems.

iv) *Feature Selection*: Feature Selection minimizes the dimensionality by keeping only relevant characteristics and removing extraneous features. Strong feature-label relationships are identified using, for example, SelectKBest and chi-squared testing. This ensures that the intrusion

detection models will stay accurate, effective and efficient in detecting advanced attack patterns. It tackles the challenges of improved performance and shortened training time and renders the models more interpretable.

v) *Data Resampling*: Data resampling is used to address the unequal distribution of intrusion and normal data in intrusion data sets. The RandomUnderSampler is used to balance the distribution by undersampling the majority classes and methods such as SMOTE over-sample the minority classes. This minimizes biases, enhances sensitivity to infrequent threats and enhances F1-scores, precision and memory. Balanced data for fair learning and improving performance of the model in a number of attack areas.

C) Training and Testing:

In an effort to have a fair model evaluation the dataset is split into a training and a testing part. The majority of the expenditure is generally on training, which enables algorithms to discover links and patterns in IoT and IIoT data. When the testing set isn't observed in the training set, then it can be used as an objective assessment of prediction ability. This process ensures that the intrusion detection models will perform well on real world events by evaluating various metrics such as accuracy, recall, precision, F1 score.

D) Algorithms

Convolutional Neural Network (CNN): CNN learns the hierarchies of the data and finds complex patterns and irregularities in the network traffic. It enhances the real-time responsiveness and robustness of IoT/IIoT networks, enables high accuracy intrusion detection, is suitable for dealing with a large number of data, and has the ability to detect subtle attack fingerprints.

$$S(i, j) = \sum_m \sum_n I(i + m, j + n) \cdot K(m, n) \quad (1)$$

Deep Neural Network (DNN): The DNN model has multiple hidden layers, which enable it to learn intricate nonlinear relationships among the features of IoT and IIoT. It allows for complex feature interactions, helps classify benign and malicious behavior, supports various datasets and offers reliable anomaly detection with higher accuracy and faster processing speed in real time.

RF (RF): RF has achieved stable and reliable intrusion detection through multiple predictions with different underlying decision tree models trained with random subsets of features, thus alleviating the problem of overfitting, processing high-dimensional data, class imbalance, and interpretable feature importance.

$$Gini = 1 - \sum_{i=1}^C (P_i)^2 \quad (2)$$

Decision Tree (DT): A decision tree provides simple and accurate classification of valid and malicious communications, by hierarchical division of the data to generate easy-to-understand classification rules. It gives foundation for ensemble technique that is able to handle both numerical and categorical features efficiently.

$$I(i) = 1 - \sum_{i=1}^k p_i^2 \quad (3)$$

LightGBM: LightGBM is a good solution for unbalanced and high dimensional network data, as it

produces series of succeeding gradient-boosted trees to fix the prior trees. It delivers fast, scalable and reliable intrusion detection, improves the detection accuracy as well as feature importance evaluation.

Bagging Classifier: Bagging is a method of using multiple random subsets of the data to train various base classifiers that helps to decrease variance and overfitting. It improves the generalization, handles unbalanced classes, strengthens detection of different kinds of threats and ensures scalable, understandable and reliable network traffic classification.

Stacking Classifier: The stacking method combines multiple base classifiers and a meta-learner to achieve the benefits of many classifiers. It ensures robust and high accuracy classification on heterogeneous IoT and IIoT data sets, reduces bias and variance, increases the ability for detection of complex incursions, and increases the generalization ability.

$$\hat{y} = g(Y_{base}) = g(f_1(x), f_2(x), \dots, f_m(x)) \quad (4)$$

Voting Classifier: Voting boosts accuracy and robustness by fusing multiple classifiers with weights or voting. It resolves class imbalance, takes advantage of complementary capabilities of models, and offers precise, understandable and scalable real-time classification of both normal and malicious network activities.

$$\hat{y} = \operatorname{argmax}_c \left(\sum_{i=1}^n II(\hat{y}_i = c) \right) \quad (5)$$

E) Integration of XAI and Flask Framework:

It is a system consisting of intuitive graphical interface with Flask framework and features of XAI. Users can safely interact with the system thanks to the frontend's registration and login configuration, which is backed by a SQLite database. The values of the features in the network that are provided by the user are preprocessed before they are passed to the developed ML models. The models analyze the data and come up with predictions for intrusion detection. This is then clearly displayed on the front end, ensuring accessibility and use in real-time.

As an improvement over the base study, the Voting Classifier ensemble is introduced in the system, which aims at enhancing the robustness and accuracy of the prediction. Furthermore, the concept of feature dependency and relevance is discussed and explained with the help of XAI methods, such as LIME and SHAP, that give interpretable insights into how the models decide. This powerful ensemble learning and feature explainability combination ensures clear reasoning and reliable detection, enhancing user confidence and understanding in IoT/IIoT network security.

IV. EXPERIMENTAL RESULTS

Accuracy: The ability of a test to identify correctly healthy cases and patients is called accuracy. The number of true positive and true negative cases should be determined as a percentage of all cases analyzed to provide an estimation of the accuracy of a test. This can be represented mathematically as:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (6)$$

Precision: Precision is the percentage of samples (or incidents) that are correctly classified as positive. So, the equation below can be employed to find the precision:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \quad (7)$$

Recall: Recall is a measure of a ML model's ability to identify all relevant examples of a class. It gives details on the ability of a model to explain a given class of observations, and is defined as the proportion of the positive observations predicted correctly.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (8)$$

F1-Score: F1 score is a ML metric for assessing the accuracy of a model. Combines the recall and precision scores of a model. The accuracy measure returns the percentage of the whole number of times a model has correctly predicted the whole dataset.

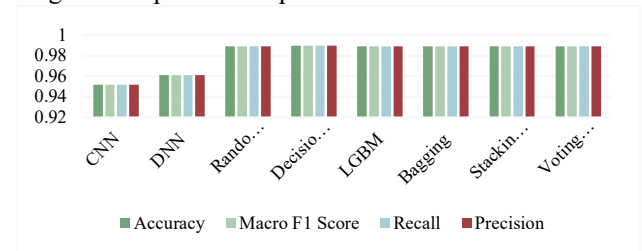
$$\text{F1 Score} = 2 * \frac{\text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}} * 100 \quad (9)$$

“Table.1 Performance Evaluation – Wustl-IIoT 2021 Dataset”

ML Model	Accuracy	Macro F1 Score	Recall	Precision
CNN	0.952	0.952	0.952	0.952
DNN	0.961	0.961	0.961	0.961
RF	0.989	0.989	0.989	0.989
Decision Tree	0.990	0.990	0.990	0.990
LGBM	0.989	0.989	0.989	0.989
Bagging	0.989	0.989	0.989	0.989
Stacking Classifier	0.989	0.989	0.989	0.989
Voting Classifier	0.989	0.989	0.989	0.989

Table 1 presents an overall performance for each algorithm, and DT in IoT and IIoT intrusion detection seems to be the most accurate in prediction.

“Fig. 5. Comparison Graph – Wustl-IIoT 2021 Dataset”



The maximum values obtained are in decision tree with 0.990 as shown in Fig. 5. The values of green, light green, blue and red indicate accuracy, precision, recall and macro F1 score respectively.

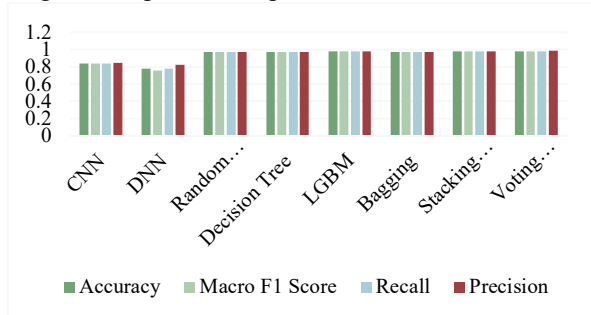
“Table.2 Performance Evaluation – ToN-IoT dataset”

ML Model	Accuracy	Macro F1 Score	Recall	Precision
CNN	0.837	0.836	0.837	0.845
DNN	0.779	0.755	0.779	0.825
RF	0.976	0.976	0.976	0.976
Decision Tree	0.972	0.972	0.972	0.973
LGBM	0.977	0.977	0.977	0.977
Bagging	0.973	0.973	0.973	0.973

Stacking Classifier	0.978	0.978	0.978	0.978
Voting Classifier	0.984	0.984	0.984	0.985

Voting Classifier is the best model in Table 2 as it is the most accurate and resilient in detecting network intrusions in IoT and IIoT environments.

“Fig. 6. Comparison Graph – ToN-IoT Dataset”



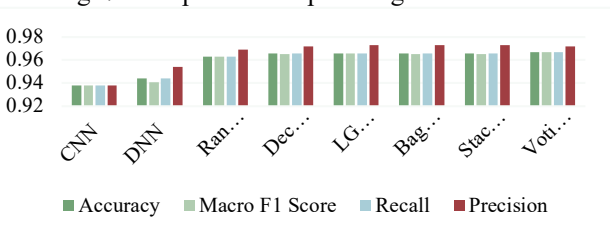
Voting Classifier shows the best results in Figure 6. The differences between the models are clearly displayed with accuracy being displayed in green, precision in red, recall in blue, and Macro F1 Score in light green.

“Table.3 Performance Evaluation – Edge-IIoT dataset”

ML Model	Accuracy	Macro F1 Score	Recall	Precision
CNN	0.938	0.938	0.938	0.938
DNN	0.944	0.941	0.944	0.954
RF	0.963	0.963	0.963	0.969
Decision Tree	0.966	0.965	0.966	0.972
LGBM	0.966	0.966	0.966	0.973
Bagging	0.966	0.965	0.966	0.973
Stacking Classifier	0.966	0.965	0.966	0.973
Voting Classifier	0.967	0.967	0.967	0.972

The Voting Classifier model has the best performance out of the models studied in Table 3.

“Fig. 7. Comparison Graph – Edge-IIoT Dataset”



The Voting Classifier is superior in performance as shown in figure 7. The accuracy, macro f1 score, recall, and precision are highlighted green, light green, blue, and red respectively, for all the assessed models.

Fig. 8. Enter Input data for EdgellIoT

The data values entered by users in EdgellIoT in Fig. 8 are used to identify and classify specific types of network attacks.

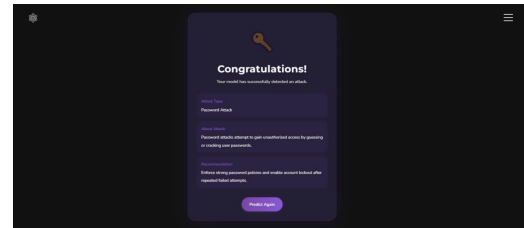


Fig. 9. Predicted Results for EdgellIoT

After processing the EdgellIoT input, the system correctly identifies the attack type of ‘Password Attack’ as shown in Fig. 9.

Fig. 10. Enter Input Data for ToN-IoT

The user provides ToN-IoT data sets in Fig. 10 to the system, which analyzes the activity of the network and detects specific types of attacks.

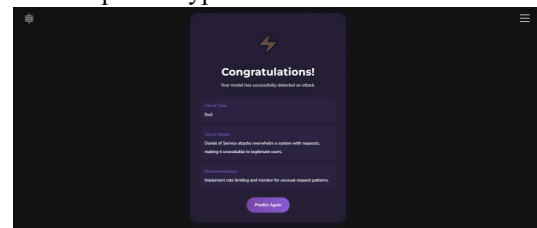


Fig. 11. Predicted Results for ToN-IoT

After processing the ToN-IoT input into the system, the system effectively identifies the intrusion in the network as a DoS attack in Fig. 11.

DistBytes	508	SrcBytes	644
TotBytes	1152	SrcLoad	83712.20313
DistLoad	64227.46875	Load	147939.6719
SrcRate		DistRate	
Rate		dstpid	

Fig. 12. Enter Input Data for Wustl-IIOT

Users can add numbers in the WUSTL-IIoT dataset shown in Fig. 12 for the system to analyze network traffic and detect specific types of attacks.

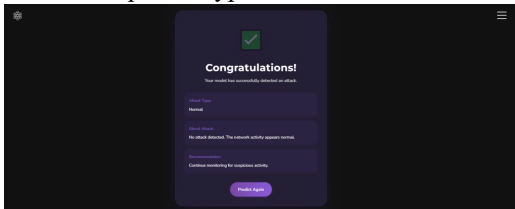


Fig. 13. Predicted Results for Wustl-IIOT

After examining the WUSTL-IIoT input in Fig. 13, the algorithm is able to correctly determine that the network activity is normal, which implies that no assaults are detected.

V. CONCLUSION

In this work, the usefulness of using ensemble methods with lightweight ML models for reliable intrusion detection in IoT and IIoT environments is demonstrated. Through comprehensive preprocessing, thorough feature selection using chi-squared, and class balancing techniques, SMOTE and undersampling, dealing with high-dimensional, unbalanced network traffic across numerous data sets was made possible. The detection reliability and generalization were greatly improved by combining individual classifiers with ensemble techniques. The Voting Classifier was resilient against heterogeneous attacks and achieved the following accuracy on the aforementioned datasets: ToN_IoT: 98.4%, WUSTL-IIoT 2021: 98.9%, Edge-IIoTset: 96.7%. Transparency was enhanced by explainable AI techniques like SHAP and LIME, and real-time, interpretable, and secure intrusion detection was made possible by Flask deployment with SQLite, making it appropriate for real-world IoT and IIoT cybersecurity monitoring.

Future work plans to use the microcontroller platforms of IoT/IIoT with the ML-based intrusion detection system to enable practical testing. Latency, energy efficiency, flexibility and computational optimizations will be considered. The integration with edge intelligence and light-weight security protocols will enhance scalability, resilience, and practical use, ensuring reliable, accurate, and effective intrusion detection in large-scale and industrial IoT systems.

REFERENCES

[1] Alam, K., Monir, M. F., Hassan, Z., & Habib, M. T. (2024, October). Optimizing IoT Network Intrusion Detection: A DL Approach. In 2024 7th Conference

on Cloud and Internet of Things (CIoT) (pp. 1-5). IEEE.

- [2] Lazzarini, R., Tianfield, H., & Charissis, V. (2023). A stacking ensemble of DL models for IoT intrusion detection. *Knowledge-Based Systems*, 279, 110941.
- [3] Kim, Y. G., Ahmed, K. J., Lee, M. J., & Tsukamoto, K. (2022, August). A Comprehensive Analysis of ML-Based Intrusion Detection System for IoT-23 Dataset. In *International Conference on Intelligent Networking and Collaborative Systems* (pp. 475-486). Cham: Springer International Publishing.
- [4] Morshedi, R., Matinkhah, S. M., & Sadeghi, M. T. (2024). Intrusion detection for IoT network security with DL. *Journal of AI and Data Mining*, 12(1), 37-55.
- [5] Eren, K. K., Küçük, K., Özyurt, F., & Alhazmi, O. H. (2025). Simple Yet Powerful: ML-Based IoT Intrusion System With Smart Preprocessing and Feature Generation Rivals DL. *IEEE Access*.
- [6] J. A. Beauty Angelin and C. Priyadharsini, "DL based network based intrusion detection system in industrial Internet of Things," in *Proc. 2nd Int. Conf. Intell. Data Commun. Technol. Internet Things (IDCIoT)*, Jan. 2024, pp. 426–432.
- [7] M. Ramaiah and M. Y. Rahamathulla, "Securing the industrial IoT: A novel network intrusion detection models," in *Proc. 3rd Int. Conf. Artif. Intell. For Internet Things (AIIoT)*, May 2024, pp. 1–6.
- [8] S. I. Popoola, A. L. Imoize, M. Hammoudeh, B. Adebisi, O. Jogunola, and A. M. Aibinu, "Federated DL for intrusion detection in consumer-centric Internet of Things," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1610–1622, Feb. 2024.
- [9] Z. Ahmed and S. S. Askar, "EdgeGuard: ML for proactive intrusion detection on edge networks," *Artif. Intell. Cybersecurity*, vol. 1, pp. 37–43, Jun. 2024.
- [10] R. S. Tiwari, D. Lakshmi, T. K. Das, A. K. Tripathy, and K.-C. Li, "A lightweight optimized intrusion detection system using ML for edge-based IIoT security," *Telecommun. Syst.*, vol. 87, no. 3, pp. 605–624, Nov. 2024.
- [11] F. Mesadieu, D. Torre, and A. Chennamaneni, "Leveraging deep reinforcement learning technique for intrusion detection in SCADA infrastructure," *IEEE Access*, vol. 12, pp. 63381–63399, 2024.
- [12] A. M. Eid, B. Soudan, A. B. Nassif, and M. Injadat, "Comparative study of ML models for IIoT intrusion detection: Impact of data preprocessing and balancing," *Neural Comput. Appl.*, vol. 36, no. 13, pp. 6955–6972, May 2024.
- [13] A. M. Eid, B. Soudan, A. B. Nassif, and M. Injadat, "Enhancing intrusion detection in IIoT: Optimized CNN model with multi-class SMOTE balancing," *Neural Comput. Appl.*, vol. 36, no. 24, pp. 14643–14659, Aug. 2024.
- [14] A. H. Farea and K. Küçük, "ML-based intrusion detection technique for IoT: Simulation with cooja," *Int. J. Comput. Netw. Inf. Secur.*, vol. 16, no. 1, pp. 1–23, Feb. 2024.
- [15] M. S. Alshehri, O. Saidani, F. S. Alrayes, S. F. Abbasi, and J. Ahmad, "A self-attention-based deep convolutional neural networks for IIoT networks

- intrusion detection,” *IEEE Access*, vol. 12, pp. 45762–45772, 2024.
- [16] K. Bansal and A. Singhrova, “Review on intrusion detection system for IoT/IIoT-brief study,” *Multimedia Tools Appl.*, vol. 83, no. 8, pp. 23083–23108, Aug. 2023.
- [17] M. Nuaimi, L. C. Fourati, and B. B. Hamed, “Intelligent approaches toward intrusion detection systems for industrial Internet of Things: A systematic comprehensive review,” *J. Netw. Comput. Appl.*, vol. 215, Jun. 2023, Art. no. 103637.
- [18] G. Guo, X. Pan, H. Liu, F. Li, L. Pei, and K. Hu, “An IoT intrusion detection system based on TON IoT network dataset,” in *Proc. IEEE 13th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Mar. 2023, pp. 0333–0338.
- [19] O. Belarbi, T. Spyridopoulos, E. Anthi, I. Mavromatis, P. Carnelli, and A. Khan, “Federated DL for intrusion detection in IoT networks,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, vol. 3125, Dec. 2023, pp. 85–99.
- [20] A. Oseni, N. Moustafa, G. Creech, N. Sohrabi, A. Strelzoff, Z. Tari, and I. Linkov, “An explainable DL framework for resilient intrusion detection in IoT-enabled transportation networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 1, pp. 1000–1014, Jan. 2023.
- [21] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, “Intrusion detection system using feature extraction with ML algorithms in IoT,” *J. Sensor Actuator Netw.*, vol. 12, no. 2, p. 29, Mar. 2023.
- [22] M. M. Shtayat, M. K. Hasan, R. Sulaiman, S. Islam, and A. U. R. Khan, “An explainable ensemble DL approach for intrusion detection in industrial Internet of Things,” *IEEE Access*, vol. 11, pp. 115047–115061, 2023.
- [23] M. Wang, N. Yang, and N. Weng, “Securing a smart home with a transformer-based IoT intrusion detection system,” *Electronics*, vol. 12, no. 9, p. 2100, May 2023.
- [24] S. Li, G. Chai, Y. Wang, G. Zhou, Z. Li, D. Yu, and R. Gao, “CRSF: An intrusion detection framework for industrial Internet of Things based on pretrained CNN2D-RNN and SVM,” *IEEE Access*, vol. 11, pp. 92041–92054, 2023.
- [25] M. M. Rashid, S. U. Khan, F. Eusufzai, M. A. Redwan, S. R. Sabuj, and M. Elsharief, “A federated learning-based approach for improving intrusion detection in industrial Internet of Things networks,” *Network*, vol. 3, no. 1, pp. 158–179, Jan. 2023.
- [26] A. M. Eid, A. B. Nassif, B. Soudan, and M. N. Injadat, “IIoT network intrusion detection using ML,” in *Proc. 6th Int. Conf. Intell. Robot. Control Eng. (IRCE)*, Aug. 2023, pp. 196–201.
- [27] T. Gaber, J. B. Awotunde, S. O. Folorunso, S. A. Ajagbe, and E. Eldesouky, “Industrial Internet of Things intrusion detection method using ML and optimization techniques,” *Wireless Commun. Mobile Comput.*, vol. 2023, pp. 1–15, Apr. 2023.
- [28] B. Xu, L. Sun, X. Mao, R. Ding, and C. Liu, “IoT intrusion detection system based on ML,” *Electronics*, vol. 12, no. 20, p. 4289, Oct. 2023.
- [29] Z. Benamor, Z. A. Seghir, M. Djeddar, and M. Hemam, “A comparative study of ML algorithms for intrusion detection in IoT networks,” *Revue d’Intell. Artificielle*, vol. 37, no. 3, pp. 567–576, Jun. 2023.
- [30] M. Al-Hawawreh, E. Sitnikova, and N. Aboutorab, “X-IIoTID: A connectivity-agnostic and device-agnostic intrusion data set for industrial Internet of Things,” *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3962–3977, Mar. 2022.