AMERICAN JOURNAL OF AI CYBER COMPUTING MANAGEMENT
Peer Reviewed, Rferred & Indexed Journal
E-ISSN:3069-0102
VOL.6, NO. 2(2026)

An Intelligent Ensemble Learning Approach for Accurate Attack Detection in Large Scale Internet of Things Networks

E Pavithra[1], P Yaswanth [2], Syed Jeelan[3]

[1] Assistant Professor, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur,
E-mail: epavithra526@gmail.com, ORC-ID: https://orcid.org/0009-0006-8871-4551

[2] P.G Scholar, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur,
E-mail: yaswanthpyaswanth632@gmail.com , ORC-ID: https://orcid.org/0009-0000-0475-3403

[3]Assistant Professor, Department of CSE, Sri Venkatesa Perumal College of Engineering & Technology, Puttur,
E-mail: syedjeelan1971@gmail.com, ORC-ID: https://orcid.org/0009-0005-8042-9521

Abstract: The expansion of Internet of Things (IoT) technology presents significant security challenges for large-scale, heterogeneous, and resource-constrained networks. Intrusion Detection Systems are crucial for safeguarding IoT systems against advanced assaults. This research assesses the RT-IoT2022 dataset utilizing K-Nearest Neighbors, Support Vector Machine, Decision Tree, Gradient Boost, XGBoost, Random Forest, Extra Trees, and LightGBM algorithms. SMOTEENN sampling is utilized for feature selection to address class imbalance and enhance robustness. A Voting Classifier ensemble that combines Random Forest, Extra Trees, and LightGBM attains optimal performance, achieving 99.9% in accuracy, precision, recall, and F1 score, surpassing all individual algorithms. Explainable AI methodologies, such as LIME and SHAP, are utilized to elucidate feature impact and bolster confidence in forecasts. The optimized ensemble is deployed in real-time through a Flask-based web application, facilitating interactive intrusion detection. The system categorizes traffic into attack classifications including DOS_SYN_Hping, Thing_Speak, ARP_poisoning, MQTT_Publish, NMAP_UDP_SCAN, NMAP_XMAS_TREE_SCAN, NMAP_OS_DETECTION, NMAP_TCP_SCAN, DDOS_Slowloris, Wipro_bulb, Metasploit_Brute_Force_SSH, and NMAP_FIN_SCAN. The proposed architecture provides precise, interpretable, and scalable intrusion detection appropriate for next-generation IoT security monitoring in dynamic real-world threat conditions.

"Index Terms: *Accuracy, Internet of Things (IoT), intrusion detection systems (IDS), machine learning classifiers."*

1. INTRODUCTION

The Internet of Things (IoT) signifies a sophisticated advancement of the internet, amalgamating physical devices with sensing, communication, and computational functionalities to facilitate seamless connectivity and intelligent automation across diverse sectors, including smart cities, healthcare, industrial operations, and home management. This integrated ecosystem depends on a multi-tiered architecture that includes perceptual, network, and application layers, which together enable data collection, transmission, and service delivery [2]. The swift integration of IoT has markedly improved operational efficiency and convenience; yet, it has concurrently broadened the digital assault surface. The emergence of resource-limited devices with restricted computational capabilities and inadequate authentication protocols has created intricate security and privacy issues [3]. As a result, safeguarding IoT infrastructures has emerged as a global need, given that these systems are progressively targeted by cybercriminals who exploit weaknesses at all architectural levels.

Notwithstanding the increasing significance of IoT in essential applications, the security of IoT settings continues to be an unresolved issue. Every layer of the IoT architecture faces distinct security concerns, including eavesdropping, spoofing, denial-of-service attacks, and virus dissemination. Recent studies indicate a considerable increase in IoT-related assaults, resulting in the compromising of millions of devices worldwide and notable economic losses across several sectors. These threats jeopardize system integrity, user privacy, data confidentiality, and service reliability. Conventional intrusion detection systems (IDS) are utilized to monitor and identify network anomalies; however, traditional methods based on predetermined signatures are inadequate for efficiently detecting novel or advanced attacks [7]. Additionally, elevated false positive rates and scalability constraints impede the implementation of such systems in highly dynamic and diverse IoT networks [8]. Consequently, a significant research vacuum exists in the development of adaptive and effective intrusion detection algorithms specifically designed for the intricate nature of IoT ecosystems.

This paper seeks to tackle these problems through a thorough assessment of intelligent intrusion detection algorithms tailored for IoT contexts. The study highlights the evaluation of data-driven techniques that can identify anomalous traffic patterns and differentiate between benign and malicious network activities [9]. The project aims to

compare different learning paradigms by analyzing massive IoT traffic datasets to improve intrusion detection accuracy while ensuring computing efficiency. The experimental approach utilized in this study guarantees reproducibility and transparency, hence promoting further progress in IoT cybersecurity research.

This research enhances the resilience of IoT infrastructures against emerging cyber attacks. The work offers empirical insights into the efficacy of intelligent intrusion detection methods, hence facilitating the advancement of proactive defense mechanisms for integration into practical IoT systems [10]. The results suggest potential implications for the development of secure IoT frameworks that can support extensive deployments with minimal vulnerabilities. This work aims to inform future advancements in intelligent cybersecurity solutions that foster trustworthy, reliable, and privacy-preserving IoT ecosystems across various application domains.

2. LITERATURE REVIEW

Recent breakthroughs in machine learning (ML) have profoundly impacted the evolution of intelligent intrusion detection systems (IDS) that enhance network security in Internet of Things (IoT) environments. Numerous studies have investigated various machine learning approaches to enhance the precision and flexibility of intrusion detection system models. Dina and Manivannan [11] examined the application of machine learning approaches for intrusion detection in conventional computer networks, highlighting that data-driven models surpass signature-based systems in detecting novel attacks. Nonetheless, their research was confined to traditional network topologies, failing to assess performance in resource-limited IoT settings where fluctuating traffic patterns present further difficulties. Saranya et al. [12] conducted a comprehensive assessment of diverse machine learning methods for intrusion detection systems, evaluating their classification efficacy and computational efficiency. Their investigation provided useful insights into algorithmic behavior but neglected to address real-time adaptation and energy efficiency, which are essential for IoT-based applications.

The use of IoT-specific datasets into IDS research has increased in prominence in recent years. Nagapadma [13] presented the RT-IoT2022 dataset, a publicly accessible benchmark intended to facilitate reproducible studies in IoT attack detection. This dataset facilitates uniform assessment across studies however remains inadequately examined in comparative machine learning research. Similarly, Inuwa and Das [14] performed a comparative analysis of anomaly detection strategies for IoT networks, illustrating that hybrid models that integrate statistical and machine learning techniques improve detection efficacy. However, their findings revealed a trade-off between model complexity and computing burden, constraining scalability in extensive IoT implementations. Bacha et al. [15] mitigated this problem by developing an anomaly-based Intrusion Detection System utilizing kernel extreme learning machines, which resulted in enhanced detection accuracy and decreased training duration. Nonetheless, the model's efficacy was predominantly evaluated on limited datasets, raising concerns on its resilience in practical IoT settings.

Initiatives have been undertaken to create hybrid and intelligent IDS frameworks tailored for domain-specific IoT applications. Saif et al. [16] introduced HIIDS, a hybrid intelligent intrusion detection system for IoT-based healthcare systems, which amalgamates machine learning and metaheuristic algorithms to identify intricate assault patterns. The system exhibited significant adaptability and precision, although necessitated substantial computational resources, rendering it inappropriate for lightweight IoT devices. Ahmad et al. [17] investigated intrusion detection with supervised machine learning techniques utilizing application and transport layer features, employing the UNSW-NB15 dataset. Their research attained significant accuracy but disclosed constraints in cross-layer generalization and model scalability in diverse IoT contexts.

Recent studies have further enhanced IDS performance by tackling data imbalance and augmenting model generalization. Talukder et al. [18] presented MLSTL-WSN, a machine learning-based intrusion detection system employing SMOTETomek for wireless sensor networks, which significantly decreases false positives and enhances detection rates. Nevertheless, their methodology was customized for WSN-specific traffic patterns rather than for broader IoT systems. Saran and Kesswani [19] evaluated supervised machine learning classifiers for intrusion detection in IoT, determining that decision tree-based models are successful but susceptible to overfitting when trained on insufficient data. Awajan [20] introduced

a deep learning-based Intrusion Detection System for IoT networks, attaining high detection accuracy, however with diminished interpretability and elevated computing expenses.

3. MATERIALS AND METHODS

The proposed system seeks to establish a resilient and scalable intrusion detection framework for Internet of Things (IoT) networks, utilizing the RT-IoT2022 dataset to reliably identify and categorize hostile network activity. The workflow includes data preprocessing, feature selection, model training, and evaluation using several machine learning classifiers to guarantee dependable detection performance. To rectify data imbalance and augment generalization, the system incorporates the SMOTEENN sampling approach, while XGBoost with feature selection is utilized to optimize feature relevance and enhance model efficiency. A Voting Classifier ensemble integrating Random Forest, Extra Trees, and LightGBM is employed to harness the complimentary advantages of multiple algorithms, resulting in enhanced detection accuracy and resilience against different attack types. Explainable AI techniques, LIME and SHAP, are utilized to improve model interpretability by identifying significant factors that influence predictions. A Flask-based web interface ultimately facilitates real-time intrusion detection, enhancing practical deployment and user accessibility across diverse IoT scenarios.

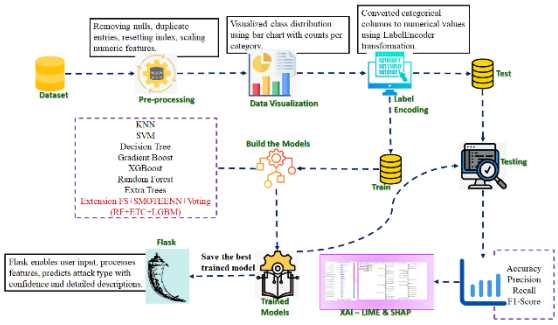


Fig.1 Proposed Architecture

Figure 1 illustrates a machine learning pathway for attack prediction, commencing with a dataset that undergoes preprocessing, data visualization, and label encoding. The data is divided into training and testing sets to construct the models, including an ensemble component. Performance is evaluated using testing metrics, analyzed with explainable artificial intelligence (XAI) methods such as LIME and SHAP, and the optimal model is implemented with Flask.

a) Dataset Collection:

The research employs the RT-IoT2022 dataset, sourced from the Kaggle repository, which functions as a thorough benchmark for intrusion detection in IoT settings. The collection comprises 123,117 network traffic records featuring 85 attributes that encapsulate various statistical, flow-based, and temporal properties derived from IoT connections. It encompasses various attack types, including DoS, DDoS, ARP poisoning, and scanning operations, mirroring authentic network actions. The dataset's variability, balanced structure, and incorporation of both benign and malicious traffic create an optimal basis for assessing machine learning models in the detection of intricate IoT intrusions.

	no	id.orig_p	id.resp_p	proto	service	flow_duration	fwd_pkts_tot	bwd_pkts_tot	fwd_data_pkts_tot	bwd_data_pkts_tot	...	active.std	idle.min
0	0	38667	1883	tcp	mqtt	32.011598	9	5	3	3	...	0.0	2.972918e+07
1	1	51143	1883	tcp	mqtt	31.883584	9	5	3	3	...	0.0	2.985528e+07
2	2	44761	1883	tcp	mqtt	32.124053	9	5	3	3	...	0.0	2.984215e+07
3	3	60893	1883	tcp	mqtt	31.961063	9	5	3	3	...	0.0	2.991377e+07
4	4	51087	1883	tcp	mqtt	31.902362	9	5	3	3	...	0.0	2.981470e+07

5 rows × 85 columns

Fig.2 RT-IoT2022 dataset

b) Pre-Processing:

The preprocessing phase readies the RT-IoT2022 dataset for effective intrusion detection by cleansing, visualizing, normalizing, selecting essential features, balancing classes, and splitting data, hence assuring precise and dependable model training.

1. Data Cleaning: The dataset was meticulously cleaned by identifying and removing missing or null items, as well as eliminating duplicate records to maintain data integrity. This stage is essential for eradicating inconsistencies and redundancies that may adversely impact model training and evaluation. Maintaining clean and unique data promotes the dependability of the learning process, diminishes noise, and strengthens the robustness and generalization capacity of the intrusion detection framework.

2. Exploratory Data Visualization: A visualization step was conducted to examine the distribution of various attack classes within the dataset. Bar charts were employed to illustrate the frequency of each class, facilitating the recognition of notable class imbalances. This stage is essential for comprehending data features, directing subsequent resampling or balancing techniques, and determining feature selection and model architecture. Visualization offers a clear overview of patterns and possible difficulties within the dataset.

3. Feature Scaling: All numerical features were normalized to a consistent scale to reduce discrepancies in their ranges. This guarantees that models sensitive to feature magnitude, such as distance-based or gradient-based methods, regard all

input variables uniformly. Standardization facilitates model convergence, promotes accuracy, and mitigates the disproportionate influence of characteristics with big numerical values on the learning process, thus fostering consistent and reliable model performance.

4. Feature Selection using XGBoost: A feature selection strategy based on XGBoost was utilized to determine the most pertinent features influencing intrusion detection. Feature significance ratings were calculated, and the top twenty characteristics were chosen to diminish dimensionality while preserving predictive efficacy. This phase optimizes model interpretability, reduces computational complexity, and improves detection accuracy by concentrating on the most informative elements of the network traffic data.

5. Class Balancing with SMOTEENN: A hybrid resampling strategy was employed to mitigate severe class imbalance, integrating SMOTE oversampling with Edited Nearest Neighbors (ENN) undersampling. This procedure produces synthetic samples for minority classes while eliminating noisy cases from the majority class. Balancing the dataset guarantees equitable representation of all attack types, augments the model's capacity to identify infrequent incursions, and mitigates bias towards majority classes, resulting in enhanced overall detection performance and reliability.

c) Training and Testing:

The curated feature dataset was divided into training and testing subsets to enable model assessment. A conventional 80-20 division was employed to train the models on a representative segment of the data, while a distinct subset was kept aside for impartial performance evaluation. This stage is crucial for validating model generalization, mitigating overfitting, and precisely estimating the real-world prediction capabilities of the intrusion detection system.

d) Algorithms:

K-Nearest Neighbors (KNN): K-Nearest Neighbors is a non-parametric supervised learning approach utilized for classification and regression, depending on the proximity of cases to deduce outcomes. [25] It forecasts the category of a data point by analyzing the nearest nearby items based on a distance metric, facilitating intuitive decision-making. KNN enhances system performance by adeptly managing irregular decision boundaries and identifying patterns in data without the need for significant model training. Its simplicity and interpretability render it very useful for detecting anomalies in intricate feature spaces.

$$distance(x, X_i) = \sqrt{\sum_{j=1}^{d} \left(x_j - X_{i_j}\right)^2} \quad (1)$$

Support Vector Machine (SVM): Support Vector Machine is a supervised learning method that aims to distinguish classes by establishing an ideal hyperplane that maximizes the margin between data points. It adeptly manages high-dimensional and non-linearly separable feature spaces, rendering it appropriate for intricate classification tasks. SVM enhances robustness and generalization by reducing misclassification and attaining high prediction accuracy. The method entails mapping features into a higher-dimensional space and delineating boundaries that clearly classify occurrences, hence ensuring accurate pattern recognition across various datasets.

$$minimize\ \frac{1}{2}\ ||W||^2 + C\sum_{i=1}^{n} \xi_i \quad (2)$$

Decision Tree: A Decision Tree is a supervised learning model that systematically divides data according to feature values, creating a tree structure comprised of decision nodes and terminal leaves that signify outcomes. The workflow assesses features in succession, choosing qualities that effectively partition data to minimize ambiguity. Decision Trees improve interpretability and offer hierarchical decision logic, aiding in the comprehension of feature significance. They enhance accuracy and efficiency in classification tasks by methodically modeling relationships between characteristics, rendering them helpful for identifying patterns and anomalies without necessitating considerable preprocessing.

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

Gradient Boosting: Gradient Boosting is an ensemble learning method that incrementally constructs decision trees, where each successive model rectifies the faults of prior rounds. It enhances predictive performance using gradient-based error minimization, facilitating the modeling of intricate, non-linear interactions among features. Gradient Boosting enhances classification accuracy, robustness, and generalization by amalgamating several weak learners into a formidable prediction model. The methodical sequential learning process adeptly mitigates imbalances and noise in data, rendering it particularly useful for discerning

complex patterns in high-dimensional or heterogeneous datasets.

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \quad (4)$$

XGBoost: XGBoost is a high-performance gradient boosting technique that constructs an ensemble of decision trees utilizing a regularized objective function to mitigate overfitting. The workflow entails the progressive training of trees that rectify the faults of their predecessors while enhancing prediction performance. XGBoost enhances system efficiency, accuracy, and scalability by managing extensive, high-dimensional datasets and facilitating concurrent calculations. Its good management of absent data and feature interactions improves model generalization and dependability, rendering it especially suitable for real-time classification applications that demand rapid and accurate predictions inside intricate feature spaces.

$$\hat{y}_i = \sigma\left(\sum_{k=1}^{K} f_k(x_i)\right), f_k \in F \quad (5)$$

Random Forest: Random Forest is an ensemble learning technique that generates numerous decision trees and consolidates their predictions to deliver a majority-vote categorization. The procedure entails training various trees with random subsets of features and amalgamating the outputs to mitigate variation and overfitting. Random Forest improves accuracy, resilience, and generalization through collective decision-making, facilitating reliable and consistent predictions across diverse datasets. Its capacity to describe intricate interactions and support many feature types guarantees dependable performance for classification tasks, while preserving interpretability and robustness against noisy or high-dimensional data.

$$Gini = 1 - \sum_{i=1}^{C} (Pi)^2 \quad (6)$$

Extra Trees (Extremely Randomized Trees): Extra Trees is an ensemble learning technique akin to Random Forest, except it incorporates greater randomness by setting split thresholds at random. The method generates numerous randomized decision trees and consolidates their predictions to enhance variance reduction and model generalization. Extra Trees improves performance, resilience, and speed by reducing overfitting and effectively managing high-dimensional data. The randomization approach facilitates swift predictions while ensuring correctness, rendering it appropriate for situations necessitating real-time detection and dependable classification over varied and intricate feature spaces.

Voting Classifier: The Voting Classifier is an ensemble method that amalgamates predictions from various base models to generate a final conclusion by majority or weighted voting. The workflow consolidates many model outputs to utilize their complementing strengths, thereby minimizing individual model biases and misclassifications. The Voting Classifier improves accuracy, precision, and robustness by integrating classifiers with diverse inductive biases, yielding stable and dependable predictions. It guarantees enhanced generalization across datasets and feature spaces, rendering it especially suitable for intricate classification problems where individual models may display varying strengths or weaknesses.

$$\hat{y} = argmax_c \left(\sum_{i=1}^{n} II\ (\hat{y}_i = c)\right) \quad (7)$$

e) Integration of XAI and Flask Framework

The amalgamation of Explainable Artificial Intelligence (XAI) with the Flask framework offers a holistic methodology for predictive modeling and interpretability inside a web-based context. XAI approaches facilitate comprehension of model conclusions by the generation of visual explanations, insights into feature relevance, and interpretable reasoning for specific predictions. This transparency is essential in applications such as network intrusion detection, where comprehending the reasoning behind classifications fosters confidence, accountability, and informed decision-making.

Flask is a lightweight web framework that enables the deployment of XAI-enabled machine learning models via interactive web interfaces. It enables users to provide input data, obtain predictions, and access visual elucidations produced by XAI methodologies in real time. The integration guarantees an accessible platform that connects intricate AI models with end-users, facilitating effective monitoring, analysis, and decision-making assistance.

4. EXPERIMENTAL RESULTS

Accuracy: The accuracy of a test refers to its capacity to correctly distinguish between patient and healthy cases. To assess the accuracy of a test, one must compute the ratio of true positives and true negatives across all assessed cases. This can be expressed mathematically as:

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} (8)$$

Precision: Precision assesses the proportion of accurately classified cases among those identified as positive. Consequently, the formula for calculating precision is expressed as:

$$Precision = \frac{\text{True Positive}}{\text{True Positive } + \text{False Positive}} (9)$$

Recall: Recall is a metric in machine learning that assesses a model's capability to identify all pertinent instances of a specific class. It is the ratio of accurately predicted positive observations to the total actual positives, offering insights into a model's efficacy in identifying occurrences of a specific class.

$$Recall = \frac{\text{TP}}{\text{TP } + \text{ FN}} (10)$$

F1-Score: The F1 score is a metric for evaluating the accuracy of a machine learning model. It integrates the precision and recall metrics of a model. The accuracy metric quantifies the frequency of true predictions generated by a model throughout the entire dataset.

$$F1\ Score = 2 * \frac{Recall \text{ X } Precision}{Recall + Precision} * 100 \quad (11)$$

Table.1 Performance Evaluation Table

ML Model	Accuracy	Precision	Recall	F1-Score
KNN	0.996	0.996	0.996	0.996
SVM	0.974	0.978	0.974	0.973
Decision Tree	0.998	0.998	0.998	0.998
GradientBoost	0.993	0.993	0.993	0.993
XGBoost	0.998	0.998	0.998	0.998
RandomForest	0.999	0.999	0.999	0.999
ExtraTrees	0.999	0.999	0.999	0.999
Extension FS+SMOTEENN+Voting	**0.999**	**0.999**	**0.999**	**0.999**

Table 1 presents a performance comparison of different machine learning models, indicating that Random Forest, Extra Trees, and the ensemble method attain the maximum accuracy.

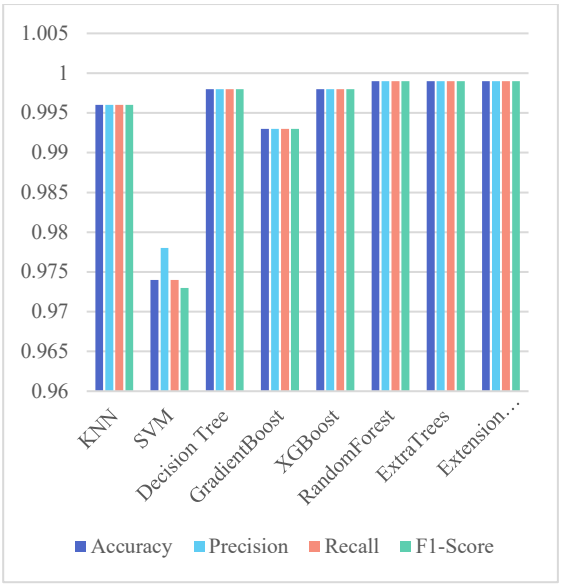


Fig.3 Comparison Graph

Figure 3 illustrates that Random Forest is the most effective method. Accuracy is denoted in purple, precision in blue, recall in orange, and F1-score in green, illustrating distinct metric differentiations.

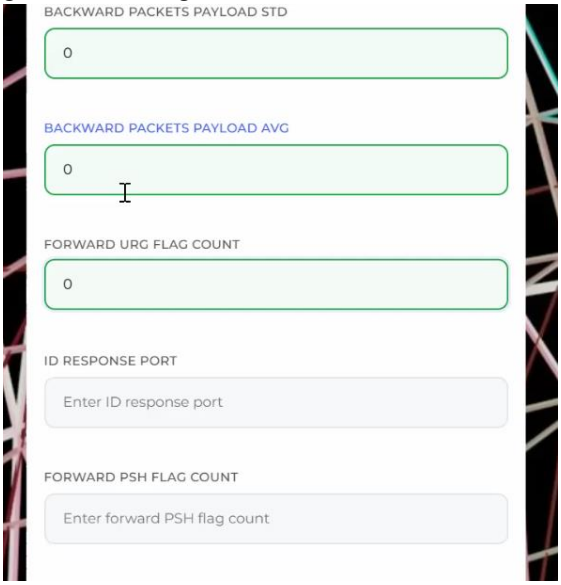


Fig.4 Enter Input Data

Figure 4 depicts an input interface that allows users to input network traffic characteristics, including packet payloads, flag counts, and response ports for examination.

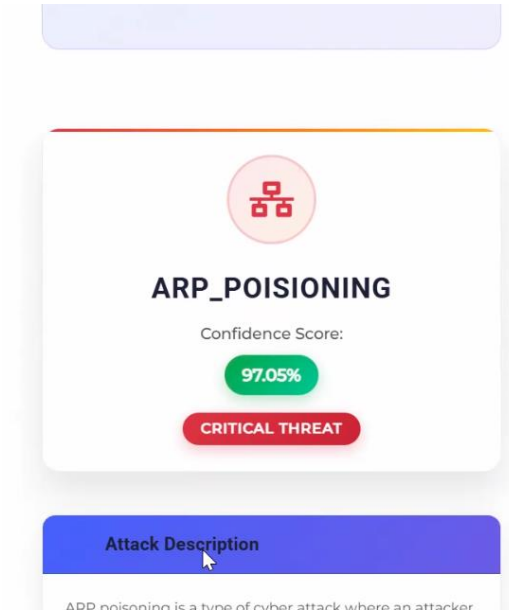


Fig.5 Predicted Results

Figure 5 presents the detection result interface indicating "ARP_POISONING" with a confidence score of 97.05%, classified as a critical danger.

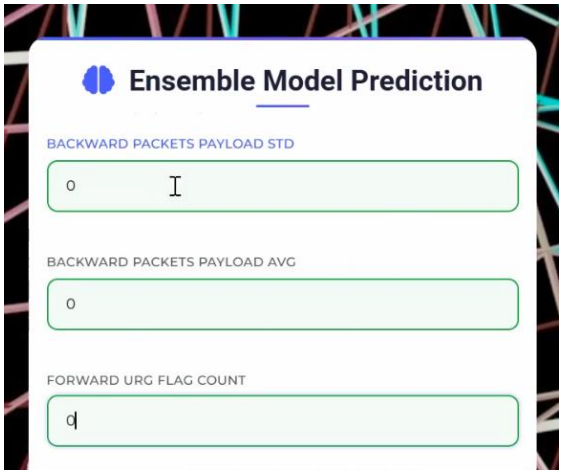


Fig.6 Enter Input Data

Figure 6 depicts an input interface that allows users to input network traffic details, encompassing packet information, flags, and port numbers, for analytical purposes.

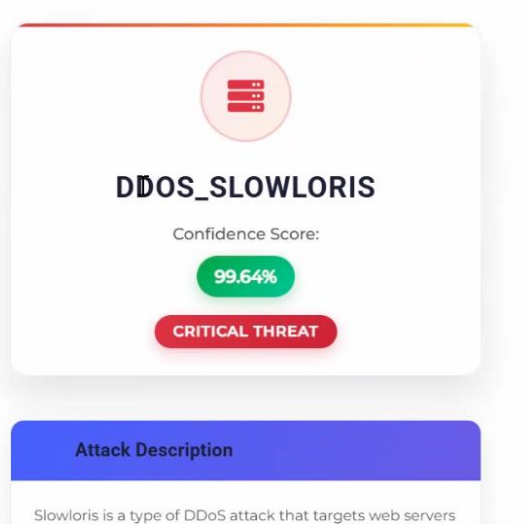


Fig.7 Predicted Results

Figure 7 presents the analysis results for the submitted network traffic, identifying "DDOS_SLOWLORIS" as a significant threat with 99.64% confidence.

5. CONCLUSION

The research indicated that the application of ensemble learning techniques markedly improves intrusion detection efficacy in IoT settings. The RT-IoT2022 dataset was utilized by a Voting Classifier that combines Random Forest, Extra Trees, and LightGBM to effectively assess various real-time assault situations. The ensemble demonstrated exceptional performance, attaining 99.9% in accuracy, precision, recall, and F1-score, much surpassing individual classifiers. These findings validate that integrating several learners leverages complimentary abilities, resulting in a more robust and dependable detection system. Feature selection methods combined with SMOTEENN sampling enhanced learning stability by mitigating class imbalance and diminishing bias from skewed data distributions. To promote transparency and trust, Explainable AI techniques like LIME and SHAP were integrated, facilitating straightforward interpretation of feature contributions and decision-making logic, which is vital for important IoT infrastructures. The optimized ensemble model was deployed using a Flask-based web framework, facilitating real-time intrusion detection and interactive monitoring. This interface enables the effective classification of incoming traffic into distinct attack categories, facilitating prompt response actions. The amalgamation of ensemble learning, explainability, and lightweight Flask deployment yields a scalable, precise, and reliable

intrusion detection solution appropriate for intricate and dynamic IoT network environments.

Future research may concentrate on enhancing the intrusion detection system to tackle emerging and complex attack patterns in IoT networks. Incorporating deep learning models, such as recurrent and attention-based architectures, may improve the capacity to identify temporal and contextual patterns throughout extensive data streams. The exploration of real-time deployment on resource-constrained IoT devices is essential for assessing computational efficiency and scalability. Another approach entails enhancing ensemble models through dynamic weighting techniques that adjust according to attack kinds and network conditions. Moreover, integrating cross-domain validation with various IoT datasets might enhance generalization and resilience. Collaborative defensive solutions utilizing federated learning can be explored to guarantee privacy-preserving and decentralized intrusion detection within dispersed IoT infrastructures.

REFERENCES

[1] G, Viswanath., N, Madhvik., K, Bhaskar., K, Supriya. (2024). Machine-Learning-Based Cloud Intrusion Detection. International Journal of Mechanical Engineering Research and Technology, 16(9), 38–52.

[2] Almotairi, A., Atawneh, S., Khashan, O. A., & Khafajah, N. M. (2024). Enhancing intrusion detection in IoT networks using machine learning-based feature selection and ensemble models. Systems Science & Control Engineering, 12(1), 2321381.

[3] Kaddour, H., Das, S., Bajgai, R., Sanchez, A., Sanchez, J., Chiu, S. C., ... & Fouda, M. M. (2024, April). Evaluating the performance of machine learning-based classification models for iot intrusion detection. In 2024 IEEE Opportunity Research Scholars Symposium (ORSS) (pp. 84-87). IEEE.

[4] Amouri, A., Al Rahhal, M. M., Bazi, Y., Butun, I., & Mahgoub, I. (2024, October). Enhancing Intrusion Detection in IoT Environments: An Advanced Ensemble Approach Using Kolmogorov-Arnold Networks. In 2024 International Symposium on Networks, Computers and Communications (ISNCC) (pp. 1-6). IEEE.

[5] Sharma, S. B., & Bairwa, A. K. (2025). Leveraging AI for Intrusion Detection in IoT Ecosystems: A Comprehensive Study. IEEE Access.

[6] J.Fox,TopCybersecurityStatistics for 2024. USA: Cobalt, 2023. [Online]. Available: https://www.cobalt.io/blog/cybersecurity-statistics-2024

[7] A. Marton and S. Systems, IoT Malware Attacks up by 37% in the First Half of 2023. IoTAC Association: EU Research and Innovation Programme, 2023.

[8] S. M. Muzammal, R. K. Murugesan, and N. Z. Jhanjhi, ‘‘A comprehen sive review on secure routing in Internet of Things: Mitigation methods and trust-based approaches,’’ IEEE Internet Things J., vol. 8, no. 6, pp. 4186–4210, Mar. 2021.

[9] S. A. Hamad, Q. Z. Sheng, W. E. Zhang, and S. Nepal, ‘‘Realizing an Internet of Secure Things: A survey on issues and enabling technologies,’’ IEEE Commun. Surveys Tuts., vol. 22, no. 2, pp. 1372–1391, 2nd Quart., 2020.

[10] O. H. Abdulganiyu, T. A. Tchakoucht, and Y. K. Saheed, ‘‘A systematic literature review for network intrusion detection system (IDS),’’ Int. J. Inf. Secur., vol. 22, no. 5, pp. 1125–1162, Oct. 2023.

[11] Viswanath, G., Abirami, V., & Prathyusha, G. (2024). Hybrid Feature Extraction With Machine Learning To Identify Network Attacks. International Journal Of HRM And Organizational Behavior, 12(3), 217–228.

[12] T.Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, ‘‘Per formance analysis of machine learning algorithms in intrusion detection system:Areview,’’Proc.Comput.Sci.,vol.171,pp. 1251–1260,Jan.2020.

[13] B.S.A.R.Nagapadma,RT-IoT20222024:UCIMachineLearningRepos itory, USA, 2024.

[14] M. M. Inuwa and R. Das, ‘‘A comparative analysis of various machine learning methods for anomaly detection in cyber attacks on IoT networks,’’ Internet Things, vol. 26, Jul. 2024, Art. no. 101162.

[15] S. Bacha, A. Aljuhani, K. B. Abdellafou, O. Taouali, N. Liouane, and M. Alazab, ‘‘Anomaly-based intrusion detection system in IoT using ker nel extreme learning machine,’’ J. Ambient Intell. Humanized Comput., vol. 15, no. 1, pp. 231–242, Jan. 2024.

[16] Kumar, K., Udaya Suriya Rajkumar, D., Viswanath, G., & Mahalakshmi, J. (2024). A Hybrid Particle Swarm Optimization and C4.5 for Network Intrusion Detection and Prevention System. International Journal of Computing, 23(1), 109–115. https://doi.org/10.47839/ijc.23.1.3442

[17] M. Ahmad, Q. Riaz, M. Zeeshan, H. Tahir, S. A. Haider, and M. S. Khan, ‘‘Intrusion detection in

Internet of Things using supervised machine learn ing based on application and transport layer features using UNSW-NB15 data-set,'' EURASIPJ.WirelessCommun.Netw.,vol.2021,no.1, pp. 1–23, Dec. 2021.

[18] 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,'' Int. J. Inf. Secur., vol. 23, no. 3, pp. 2139–2158, Jun. 2024.

[19] N. Saran and N. Kesswani, ''A comparative study of supervised machine learning classifiers for intrusion detection in Internet of Things,'' Proc. Comput. Sci., vol. 218, pp. 2049–2057, Jan. 2023.

[20] A. Awajan, ''A novel deep learning-based intrusion detection system for IoT networks,'' Computers, vol. 12, no. 2, p. 34, Feb. 2023.

[21] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, ''Intrusion detection system using feature extraction with machine learn ing algorithms in IoT,'' J. Sensor Actuator Netw., vol. 12, no. 2, p. 29, Mar. 2023.

[22] Poojari, R. INTELLIGENT SYSTEMS+B108 AND APPLICATIONS IN ENGINEERING.

[23] B. S. Sharmila and R. Nagapadma, ''Quantized autoencoder (QAE) intru sion detection system for anomaly detection in resource-constrained IoT devices using RT-IoT2022 dataset,'' Cybersecurity, vol. 6, no. 1, p. 41, Sep. 2023.

[24] T. S. Othman, K. R. Koy, and S. M. Abdullah, ''Intrusion detection systems for IoT attack detection and identification using intelligent techniques,'' Networks, vol. 5, p. 6, Jan. 2023.

[25] S. Yaras and M. Dener, ''IoT-based intrusion detection system using new hybrid deep learning algorithm,'' Electronics, vol. 13, no. 6, p. 1053, Mar. 2024.

[26] H. Nandanwar and R. Katarya, ''Deep learning enabled intrusion detec tion system for industrial IoT environment,'' Exp. Syst. Appl., vol. 249, Sep. 2024, Art. no. 123808.

[27] Naresh, M., Gudditti., M., Viswanath, & SunilKumarReddy, M.T. (2014). Distributed Utility-Based Energy Efficient Cooperative Medium Access Control in MANETS.

[28] V.Choudhary, S. Tanwar, T. Choudhury, and K. Kotecha, ''Towards secure IoT networks: A comprehensive study of metaheuristic algorithms in conjunction with CNN using a self-generated dataset,'' MethodsX, vol. 12, Jun. 2024, Art. no. 102747.

[29] V. Choudhary et al., ''Towards secure IoT networks: A comprehensive study of metaheuristic algorithms in conjunction with CNN using a self generated dataset,'' MethodsX, vol. 12, 2024, Art. no. 102747.

[30] A. R. Gad, A. A. Nashat, and T. M. Barkat, ''Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset,'' IEEE Access, vol. 9, pp. 142206–142217, 2021.