IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 28, Issue 2, Ser. 1 (Mar. – Apr. 2026), PP 81-89
www.iosrjournals.org

Performance Evaluation Of Machine Learning Techniques For Network Intrusion Detection In Iot Systems

Abubakar Ibrahim, Nasir Musa Imam, Abba Abdullahi Wakili, Magode Emmanuel

(School Of Computer Science And Engineering, Central South University, Changsha 410083, China)
(Department Of Computer Science And Engineering, Vivekananda Global University Jaipur, Rajasthan 303012, India)
(Department Of Cyber Security Bayero University, Kano 700001, Nigeria)
(Department Of Cyber Security Mewar International University, Masaka 962001, Nigeria)

Abstract:
There has been unprecedented increase in the number of Distributed Denial of Service (DDoS) attacks and network intrusions to the extent that most nodes associated with the IoT are prone to attacks. Dynamics of the threat scenario dictate the need to develop effective and accurate intrusion detection systems. With the help of the RT-IoT2022 data set, this paper presents a deep evaluation of three machine learning models of intrusion detection in IoT. Random Forest (RF), Support Vector Machine (SVM), and Artificial Neural Network (ANN). The research involves a careful preprocessing method that incorporates class imbalance handling concepts and feature selection based on correlation in order to obtain the best performance of the model. Our experiments proved that Random Forest has the best classification performance with 98.98% accuracy as compared to SVM with 98.84% and ANN with 98.89% accuracy and it has the least compute complexity with an inference time of 2ms. Cross validation analysis shows that Random Forest has a better track record in terms of ±0.22 accuracy variations as compared to ±5.18 in ANN. Such findings show that Random Forest strikes a better balance between capability performance and computational cost on resource limited IoT installations. The study offers specific suggestions for the use of intrusion detection systems based on machine learning in edge computing systems.
Key Word: *IoT, machine learning, intrusion detection, Random Forest, artificial neural network, and support vector machine.*

Date of Submission: 04-04-2026 Date of Acceptance: 14-04-2026

I. Introduction

The Internet of Things (IoT) refers to interconnected smart devices that communicate through network protocols to enable intelligent environments in domains such as healthcare, smart homes, precision agriculture, and industrial automation. The rapid expansion of IoT has significantly improved operational efficiency and real-time monitoring capabilities. However, the resource-constrained nature of IoT devices, including limited memory, processing power, and energy capacity, makes them highly vulnerable to cyber-attacks, particularly Distributed Denial of Service (DDoS) attacks. With billions of IoT devices expected to be deployed globally, securing these networks has become a critical challenge for modern cybersecurity systems [1, 2]. Intrusion Detection Systems (IDS) play a vital role in protecting IoT infrastructures by monitoring network traffic and detecting malicious activities in real time. Traditional security mechanisms such as firewalls, encryption, and authentication protocols are often insufficient against sophisticated and evolving cyber threats [3]. IDS provides an additional security layer by analyzing network behavior and identifying abnormal or malicious patterns, thereby ensuring the confidentiality, integrity, and availability of IoT systems [4]. However, the increasing complexity of cyber-attacks and the large-scale deployment of IoT devices have made conventional IDS approaches less effective, particularly in detecting zero-day and large-scale DDoS attacks [5]. Recent advancements in machine learning (ML) and artificial intelligence (AI) have significantly improved intrusion detection capabilities. ML-based IDS can automatically learn patterns from network traffic and adapt to emerging threats, making them suitable for dynamic IoT environments. Deep learning and intelligent anomaly detection techniques further enhance detection accuracy by handling high-dimensional data and identifying complex attack patterns that traditional rule-based systems cannot detect [6]. Despite these improvements, challenges such as high false positive rates, computational overhead, and dataset imbalance remain key issues in designing efficient IDS models for IoT networks. Distributed Denial of Service (DDoS) attacks remain one of the most critical threats in IoT environments, as they overwhelm network resources using large-scale botnets composed of compromised devices. These attacks disrupt

services, degrade system performance, and cause significant economic and operational losses. Therefore, robust ML-based IDS models are required to accurately detect and mitigate DDoS attacks while maintaining computational efficiency in resource-constrained IoT systems [7]. This study focuses on the comparative evaluation of three widely used machine learning classifiers Artificial Neural Networks (ANN), Random Forest (RF), and Support Vector Machines (SVM) for IoT intrusion detection using the RT-IoT2022 dataset. The objective is to analyze their effectiveness in detecting cyber threats and provide practical recommendations for deploying ML-based IDS in real-world IoT environments.

Contributions of the Paper

This study makes the following key contributions:

- A comprehensive preprocessing and feature selection framework is developed to improve IoT intrusion detection performance.
- Three state-of-the-art machine learning models (ANN, RF, and SVM) are implemented and evaluated using the RT-IoT2022 dataset.
- Performance comparison is conducted using multiple evaluation metrics, including accuracy, precision, recall, F1-score, ROC curves, and confusion matrices.
- Practical recommendations are provided for deploying ML-based IDS with improved computational efficiency and reduced false alarm rates in IoT environments.

The remainder of this paper is organized as follows. Section II presents the background on intrusion detection systems and IoT security. Section III describes the methodology, including dataset, preprocessing, and model design. Section IV discusses experimental results and performance evaluation. Section V concludes the paper and outlines future research directions.

II. Background

Intrusion Detection Systems and IoT Security

Intrusion Detection Systems (IDS) play a critical role in securing Internet of Things (IoT) environments by monitoring network and system activities to detect malicious behavior and unauthorized access. As IoT devices continue to expand across smart cities, healthcare, industrial automation, and critical infrastructure, ensuring network security has become a major concern. IDS provides real-time monitoring and threat detection by analyzing network traffic, system behavior, and communication patterns to identify potential intrusions and cyber-attacks [8]. Unlike traditional security mechanisms such as firewalls and encryption, IDS actively detects suspicious activities and generates alerts to prevent system compromise. IDS are generally categorized into anomaly-based and signature-based detection systems. Anomaly-based IDS establishes a baseline of normal network behavior and detects deviations that may indicate malicious activities. This approach is particularly useful for identifying zero-day attacks and previously unknown threats because it does not rely on predefined attack signatures [9]. However, anomaly-based IDS often suffers from high false positive rates due to dynamic network behavior and environmental variations. Signature-based IDS, on the other hand, relies on predefined attack patterns and known threat signatures to detect malicious activities. While this approach provides high accuracy for known attacks, it fails to detect new and evolving threats and requires continuous updates of signature databases. In IoT environments, IDS architecture is typically classified into centralized, distributed, and hybrid systems. Centralized IDS provides easy management and monitoring but suffers from single points of failure and scalability issues. Distributed IDS improves fault tolerance by deploying detection agents across multiple network nodes, while hybrid IDS combines both approaches to achieve improved performance and reliability [10, 11]. The integration of IDS in IoT networks is essential to ensure secure communication and protect sensitive data from cyber threats.

Machine Learning and Deep Learning for IDS

Machine learning (ML) has significantly improved the performance of intrusion detection systems by enabling automated detection of malicious patterns in network traffic. ML-based IDS analyzes large volumes of network data and learns behavioral patterns to classify normal and malicious activities with high accuracy. Supervised learning algorithms such as Random Forest (RF), Support Vector Machines (SVM), and Decision Trees are widely used due to their strong classification performance and ability to handle structured network traffic data [8]. These models rely on labeled datasets to learn attack patterns and provide reliable detection of known threats. Unsupervised learning techniques, including clustering and anomaly detection methods, are used to identify unknown attacks without requiring labeled data. Hybrid ML approaches combine supervised and unsupervised learning to improve detection accuracy and reduce false alarm rates. The integration of ML in IDS enables adaptive learning, allowing systems to respond to evolving cyber threats in dynamic IoT environments. Deep learning (DL), a subset of machine learning, has further enhanced intrusion detection capabilities by automatically extracting complex features from network traffic data. Models such as Convolutional Neural

Networks (CNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) networks have shown strong performance in detecting cyber-attacks due to their ability to capture spatial and temporal patterns in data [12, 13]. CNN models are effective in extracting spatial traffic features, while LSTM and RNN models capture sequential and temporal dependencies in network flows. Despite their high detection accuracy, deep learning models require large datasets and significant computational resources, making them challenging to deploy in resource-constrained IoT environments. Therefore, lightweight ML-based IDS models remain a practical solution for real-world IoT security applications.

Network-Based, Host-Based, and Hybrid IDS

Intrusion detection systems can also be classified based on their deployment into Network-Based IDS (NIDS), Host-Based IDS (HIDS), and Hybrid IDS. Network-Based IDS monitors network traffic across connected devices to detect malicious communication patterns and abnormal traffic flows. NIDS operates by analyzing packet headers, flow statistics, and communication protocols to identify potential threats within network infrastructure [14]. This approach is widely used in IoT environments where multiple devices communicate across distributed networks. Host-Based IDS focuses on monitoring system-level activities such as file access, system calls, application behavior, and operating system interactions. HIDS provides detailed forensic analysis and identifies attacks targeting specific host systems by analyzing audit logs and system call patterns [15]. This approach enables precise detection of system-level intrusions and provides detailed insights into attack behavior. Hybrid IDS combines the capabilities of both NIDS and HIDS to provide comprehensive security coverage. By integrating network traffic analysis with host-level monitoring, hybrid IDS improves detection accuracy and reduces false positive and false negative rates. The combination of these approaches allows better detection of sophisticated cyber-attacks and provides a more reliable security framework for IoT environments [16].

DDoS Attacks in IoT Networks

Distributed Denial of Service (DDoS) attacks represent one of the most significant threats to IoT infrastructures. These attacks aim to overwhelm network resources by generating massive traffic from compromised devices, preventing legitimate users from accessing services. IoT devices are particularly vulnerable to DDoS attacks due to weak security configurations, limited processing capabilities, and lack of standard security protocols [17, 18]. Botnets such as Mirai have demonstrated the devastating impact of IoT-based DDoS attacks by exploiting vulnerable devices to launch large-scale attacks on critical infrastructure. These attacks consume bandwidth, exhaust system resources, and disrupt services, leading to significant operational and financial losses. Traditional security mechanisms are often insufficient to mitigate DDoS attacks, highlighting the need for intelligent intrusion detection systems capable of detecting high-volume and dynamic attack patterns. Machine learning-based IDS provides an effective solution for detecting DDoS attacks by analyzing traffic behavior and identifying abnormal traffic spikes. These intelligent models can detect both known and unknown attack patterns and adapt to evolving threats, making them suitable for securing IoT networks.

IDS Challenges in IoT and Cloud Environments

The integration of IoT and cloud computing introduces several challenges for intrusion detection systems. IoT devices are resource-constrained and cannot support complex IDS models, requiring lightweight and efficient detection mechanisms. Additionally, cloud-based IoT environments involve large-scale distributed networks, making real-time monitoring and threat detection more complex [19]. Other challenges include encrypted traffic, dynamic network topologies, scalability issues, and heterogeneous device communication. The lack of standardization in IoT device manufacturing and data storage further complicates security implementation. Privacy concerns, legal regulations, and data protection requirements also impact the deployment of IDS in cloud and IoT environments. These challenges highlight the need for efficient and adaptive IDS models that can operate under limited computational resources while maintaining high detection accuracy.

Limitations of Signature-Based Detection and Research Gap

Signature-based intrusion detection systems remain effective in detecting known cyber threats; however, they are limited in identifying new and evolving attacks. These systems rely on predefined attack signatures and require continuous updates to maintain detection accuracy. As cyber-attacks become more sophisticated and polymorphic, signature-based IDS struggles to detect zero-day attacks and adaptive malware [20]. Recent studies, including unified IDS frameworks developed using datasets such as UNSW-NB15, have demonstrated improved detection performance but still face challenges in handling unknown attacks and IoT-specific threats [21]. This limitation highlights the need for machine learning-based IDS models that can adapt to dynamic network environments and accurately detect DDoS attacks in IoT systems. Therefore, this study focuses on the comparative evaluation of machine learning classifiers, including ANN, Random Forest, and SVM, using the RT-IoT2022 dataset to develop an efficient and reliable intrusion detection framework for IoT environments.

III. Methodology

Dataset Description

This study utilizes the RT-IoT2022 dataset obtained from Kaggle, which is a publicly available benchmark dataset designed for intrusion detection research in IoT environments. The dataset was collected from real IoT network scenarios and contains both normal and malicious traffic, making it suitable for evaluating machine learning-based intrusion detection systems. It includes 85 network traffic features that describe communication behavior, packet characteristics, and protocol-level information. The dataset contains multiple attack categories; however, this study focuses on four major attack types: **DOS SYN Hping, MQTT Publish, NMAP UDP Scan, and ThingSpeak**. These attack types were selected because they frequently occur in real-world IoT environments and significantly impact network performance and security. The structured nature of the dataset enables detailed analysis of network traffic patterns and provides a reliable foundation for evaluating classification models.

Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) was conducted to understand the dataset structure, feature relationships, and class distribution before applying machine learning models. EDA helps identify redundant features, inconsistencies, and imbalances that may affect model performance. A correlation matrix was generated to examine relationships between features and detect multicollinearity. Features with correlation values greater than 0.9 were removed to reduce redundancy and improve classification efficiency. Additionally, class distribution analysis revealed a significant imbalance among attack categories. Figure 3 illustrates the distribution of attack types in the dataset, showing that DOS SYN Hping has the highest number of samples, while other attack types have fewer instances. This imbalance motivated the application of downsampling during preprocessing to ensure fair model training. A feature correlation heatmap was also generated (Figure 4) to visualize relationships between features and guide feature selection.

Preprocessing and Feature Selection

Data preprocessing and feature selection were performed to enhance model accuracy and reduce computational complexity. The following steps were applied:

- Removal of non-informative and redundant features to reduce dimensionality
- Standardization of numerical features using StandardScaler to normalize feature values
- Handling class imbalance using downsampling to ensure equal representation of attack classes
- Selection of relevant features based on correlation analysis and data distribution

After preprocessing, the dataset was properly structured and optimized for training machine learning models and conducting performance evaluation.

Model Architectures

Three machine learning models were implemented in this study: Random Forest (RF), Support Vector Machine (SVM), and Artificial Neural Network (ANN).

Random Forest: The Random Forest classifier consists of 100 decision trees and uses the entropy criterion for optimal feature splitting. This ensemble method improves classification accuracy and reduces overfitting by combining multiple decision trees.

Support Vector Machine: The SVM model employs a Radial Basis Function (RBF) kernel to handle nonlinear decision boundaries. The regularization parameter C was set to 1, and gamma was configured as *scale* to optimize classification performance.

Artificial Neural Network: The ANN model consists of a three-layer architecture with 128 neurons in the first hidden layer, 64 neurons in the second hidden layer, and a softmax output layer for multiclass classification. ANN models are effective in learning complex network traffic patterns and improving intrusion detection performance each model was trained using the preprocessed dataset and evaluated using standard classification metrics.

E. Experimental Setup

The experimental setup was designed to ensure reliable and reproducible results.

- The dataset was divided into **80% training and 20% testing sets**
- **5-fold stratified cross-validation** was applied to improve generalization
- Hyperparameter tuning was performed for all models to achieve optimal performance
- Grid search was used to optimize Random Forest parameters (*n_estimators* and *max_depth*)
- SVM parameters (*C* and *gamma*) were optimized using cross-validation

- ANN utilized **dropout regularization (0.2)** to reduce overfitting
- Evaluation metrics included **Accuracy, Precision, Recall, F1-score, and ROC-AUC**
- All experiments were conducted on a system equipped with **16 GB RAM, Intel i7 processor, and NVIDIA RTX 3060 GPU**, ensuring efficient model training and evaluation.

Figure 2 presents the overall research workflow, illustrating the data preprocessing, model training, and evaluation process.

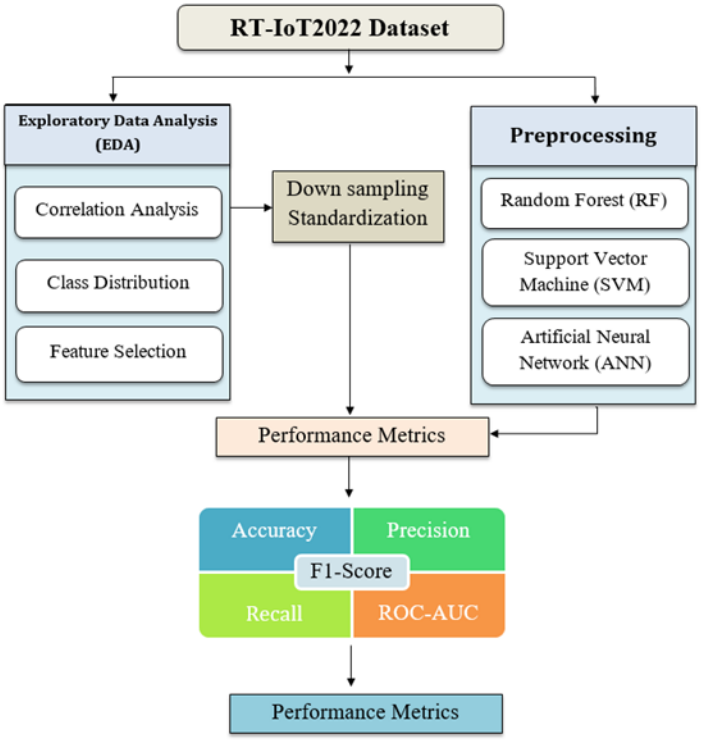

Fig. 2. The flowchart represents the technique of this research approach.

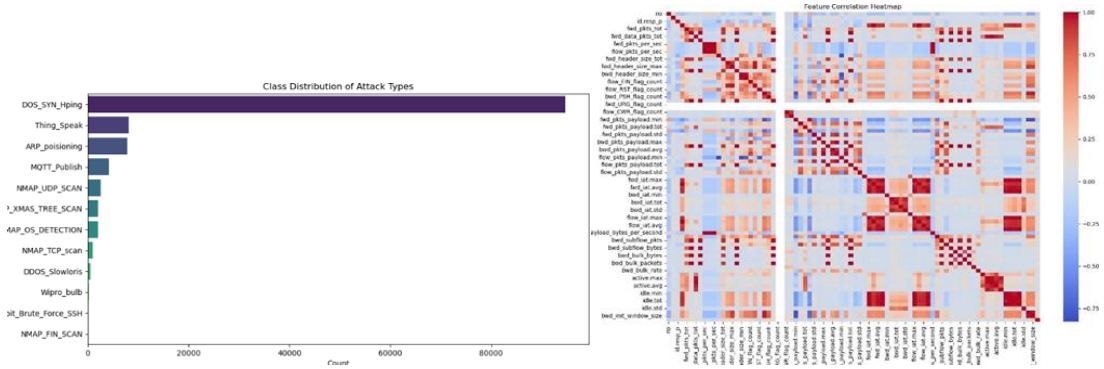

Fig. 3. Class Distribution of Attack Types
Fig. 4. Feature Correlation Heatmap

IV. Results And Discussion

A. Random Forest Decision Mechanism

The Random Forest (RF) classifier operates by aggregating the predictions of multiple decision trees to improve classification accuracy and reduce overfitting. For an input feature vector **x**, each tree h_t produces a class prediction, and the final predicted label $\hat{y}$ is obtained through majority voting among all trees:

$$\hat{y} = \text{mode}\{h_t(\mathbf{x})\}_{t=1}^{T}. \quad (1)$$

To support probabilistic evaluation such as ROC-AUC analysis, the Random Forest also estimates class membership probabilities. The probability of class k for a given input **x** is computed as the proportion of trees that classify the input into class k :

$$p_k(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T} I(h_t(\mathbf{x}) = k), \quad (2)$$

Where $I(\cdot)$ represents the indicator function that returns 1 when the condition is true and 0 otherwise.

Within each decision tree, node splitting is guided by minimizing the Gini impurity, which measures the homogeneity of the dataset at each node:

$$\text{Gini}(D) = 1 - \sum_{k=1}^{K} p_k^2, \quad (2)$$

Where p_k denotes the proportion of samples belonging to class k in dataset D. This impurity reduction strategy ensures that nodes become increasingly pure, contributing to the strong generalization capability of the Random Forest classifier.

B. Random Forest Classifier

The Random Forest classifier achieved the highest classification accuracy of **0.9898**, demonstrating superior performance in detecting IoT network intrusions. The confusion matrix in **Figure 5** shows that most traffic instances were correctly classified, with only minimal misclassification across attack categories.

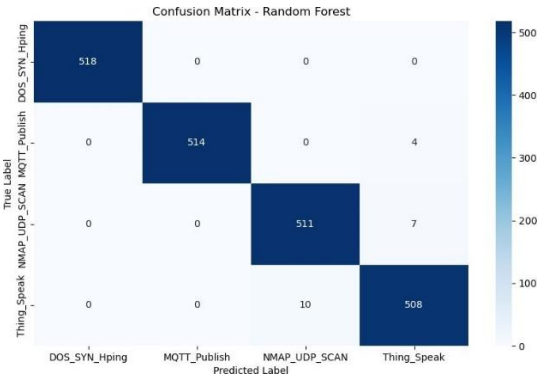

Fig. 5. Confusion Matrix- Random Forest

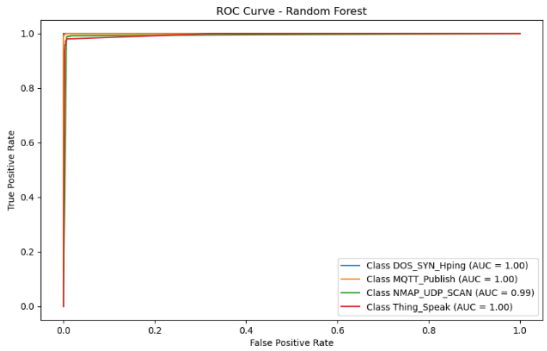

Fig. 6. ROC Curve- Random Forest

This indicates that the model effectively distinguishes between normal and malicious network traffic. The ROC curve presented in **Figure 6** further confirms the robustness of the RF classifier, with an Area under the Curve (AUC) value close to **1.0**, indicating excellent class separability and reliable intrusion detection performance. The near-perfect ROC curve suggests that the model maintains high sensitivity and specificity across different threshold values, making it suitable for real-time intrusion detection environments.

C. Support Vector Machine (SVM) Classifier

The Support Vector Machine (SVM) classifier achieved an accuracy of **0.9884**, performing closely to the Random Forest model. The confusion matrix shown in **Figure 7** indicates strong classification capability with only a small number of misclassified instances, demonstrating the model's effectiveness in handling IoT intrusion detection tasks.

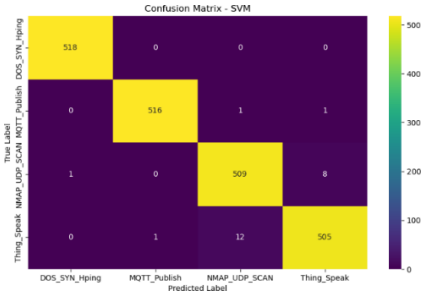

Fig. 7. Confusion Matrix- SVM

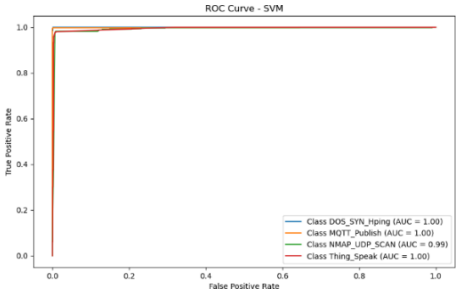

Fig. 8. ROC Curve- SVM

The ROC curve in **Figure 8** shows high AUC values across all classes, confirming that SVM can effectively separate normal and attack traffic. Despite its strong performance, SVM requires careful parameter tuning and higher computational resources due to the kernel-based optimization process, which may limit its scalability in large IoT environments.

D. Artificial Neural Network (ANN) Classifier

The Artificial Neural Network (ANN) classifier achieved an accuracy of **0.9889**, demonstrating competitive performance compared to RF and SVM. The confusion matrix in **Figure 9** shows minimal classification errors, indicating that the ANN model successfully captures complex patterns within the IoT traffic data.

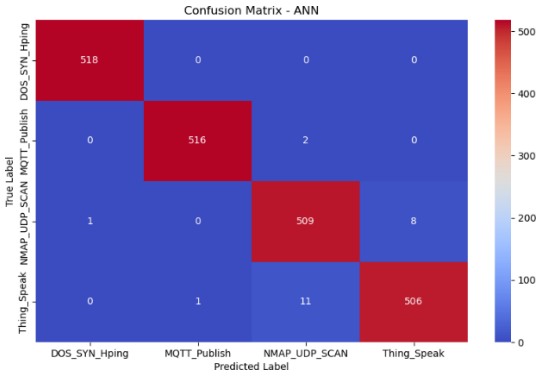

Fig. 9. Confusion Matrix- ANN

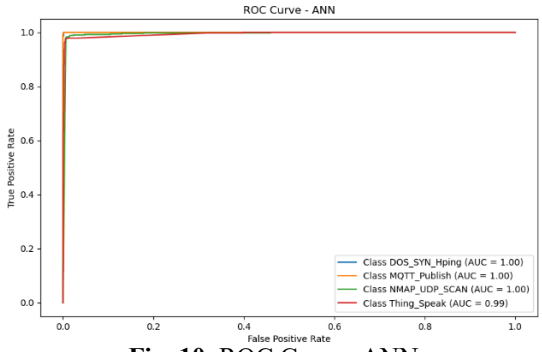

Fig. 10. ROC Curve- ANN

The ROC curve presented in **Figure 10** further validates the model's performance, with near-perfect AUC values across all classes. This highlights the ANN's strong ability to learn non-linear relationships and generalize well to unseen data. However, neural network models typically require longer training time and more computational resources, which may impact real-time deployment in resource-constrained IoT environments.

E. Comparison of Models

Table I presents the performance comparison of the three classifiers based on accuracy, precision, recall, and F1-score.

TABLE I: PERFORMANCE METRICS OF CLASSIFIERS

Model	Accuracy	Precision	Recall	F1-Score
RF	0.9898	0.99	0.99	0.99
SVM	0.9884	0.99	0.99	0.99
ANN	0.9889	0.99	0.99	0.99

The results show that **Random Forest achieved the highest accuracy**, followed closely by ANN and SVM. All models demonstrated strong precision, recall, and F1-score values, confirming their effectiveness in IoT intrusion detection. The slight advantage of Random Forest can be attributed to its ensemble learning capability, which improves robustness and handles non-linear data efficiently. SVM provides strong classification performance but requires higher computational resources and careful kernel tuning. ANN demonstrates strong generalization ability but demands longer training time and larger datasets for optimal performance.

Trade-offs and Computational Cost

- **Random Forest:** Computationally efficient, robust, and suitable for real-time intrusion detection, but less interpretable due to ensemble decision-making.
- **SVM:** Effective for structured datasets but scales poorly with large IoT traffic data and requires intensive computation.
- **ANN:** Provides strong feature learning and generalization but requires longer training time and higher computational resources.

These findings suggest that Random Forest is the most suitable model for real-time IoT intrusion detection, while ANN and SVM can be used in scenarios requiring deeper feature learning or advanced classification capability.

F. Cross-Validation Results

To evaluate the robustness and generalization ability of the models, **5-fold cross-validation** was conducted on all classifiers. The cross-validation accuracy across different folds is presented in **Figure 11**, while Table II summarizes the mean accuracy and standard deviation.

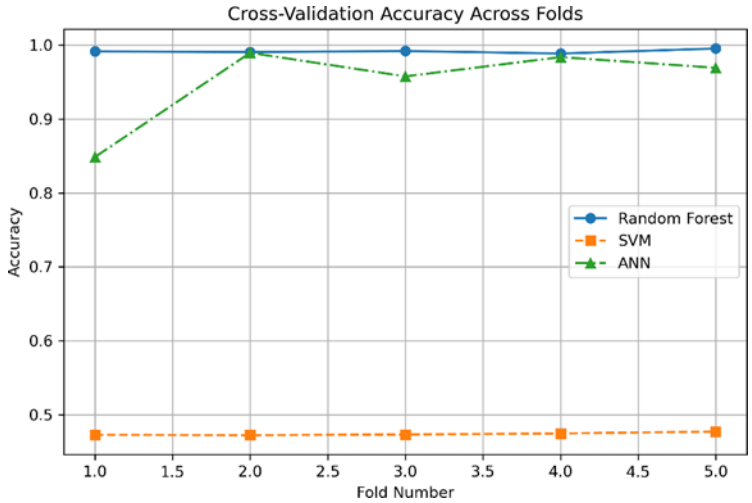

Fig. 11. Cross-Validation Accuracy across Different Folds for RF, SVM, and ANN

TABLE II: 5-FOLD CROSS-VALIDATION RESULTS

Model	Mean Accuracy	Standard Deviation
Random Forest	0.9914	±0.0022
SVM	0.4742	±0.0017
ANN	0.9496	±0.0518

Random Forest Performance

Random Forest achieved the highest mean accuracy of **99.14%** with a very low standard deviation of **±0.0022**, indicating excellent stability and strong generalization across different data folds. This demonstrates that RF effectively captures decision boundaries and maintains consistent performance across various subsets of the dataset.

SVM Performance

SVM showed significantly lower performance with a mean accuracy of **47.42%**. The low variance indicates consistent but poor performance across folds. This may be due to high data dimensionality, lack of optimal feature separation, and insufficient hyperparameter tuning, particularly for the **C** and **gamma** parameters.

ANN Performance

The ANN model achieved a mean accuracy of **94.96%**, demonstrating strong predictive capability. However, the higher standard deviation (**±5.18%**) suggests performance instability across different folds. This variation may result from sensitivity to training data, potential overfitting, and the need for further hyperparameter optimization such as activation functions, batch normalization, and optimizer selection.

Overall, Random Forest proved to be the most stable and accurate model, followed by ANN, while SVM underperformed significantly in cross-validation. The results indicate that **tree-based ensemble methods are more suitable for IoT intrusion detection**, as they effectively handle high-dimensional and non-linear traffic data. Random Forest is therefore recommended as the primary classifier for IoT intrusion detection due to its **high accuracy, low variance, and strong generalization performance**.

V. Conclusion

This study demonstrates that machine learning-based Intrusion Detection Systems (IDS) can effectively detect cyber threats in IoT networks. Random Forest (RF), Support Vector Machine (SVM), and Artificial Neural Networks (ANN) were evaluated using the RT-IoT2022 dataset, and all models achieved high accuracy, with RF performing the best overall. While RF achieved the highest accuracy (0.9898), ANN showed stable learning and high classification reliability. SVM performed competitively but had higher computational costs. The findings suggest that RF is well-suited for real-time deployment, while ANN and SVM could be leveraged in advanced security applications requiring deep pattern recognition.

Limitations and Future Work: This study primarily focused on four attack types, and future work would include more attack categories to further generalize IDS models. Additionally, real-time testing on IoT edge devices could assess the feasibility of deploying ML-based IDS in production environments. Exploring hybrid deep learning models (e.g., CNN-LSTMs) could also improve long-term IoT security solutions.

Competing Interest: The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

References

[1] M. B. Bankó Et Al., "Advancements In Machine Learning-Based Intrusion Detection In Iot: Research Trends And Challenges," Algorithms, Vol. 18, No. 4, P. 209, 2025. [Online]. Available: Https://Www.Mdpi.Com/1999-4893/18/4/209.

[2] N. M. Imam, M. Ali, And A. Bhatnagar, "Investigation Of Machine Learning Models For Multi-Class Detection And Classification Of Ddos Attack," In 2025 International Conference On Communication And Smart Devices (Iccosd), 2025, Vol. 1: Ieee, Pp. 1-6.

[3] A. Zadeh And A. Jeyaraj, "A Multistate Modeling Approach For Organizational Cybersecurity Exploration And Exploitation," Decision Support Systems, Vol. 162, P. 113849, 2022.

[4] B. Yang, M. H. Arshad, And Q. Zhao, "Packet-Level And Flow-Level Network Intrusion Detection Based On Reinforcement Learning And Adversarial Training," Algorithms, Vol. 15, No. 12, P. 453, 2022.

[5] M. S. El Sayed, N.-A. Le-Khac, M. A. Azer, And A. D. Jurcut, "A Flow-Based Anomaly Detection Approach With Feature Selection Method Against Ddos Attacks In Sdns," Ieee Transactions On Cognitive Communications And Networking, Vol. 8, No. 4, Pp. 1862-1880, 2022.

[6] J. G. Almaraz-Rivera, J. A. Perez-Diaz, J. A. Cantoral-Ceballos, J. F. Botero, And L. A. Trejo, "Toward The Protection Of Iot Networks: Introducing The Latam-Ddos-Iot Dataset," Ieee Access, Vol. 10, Pp. 106909-106920, 2022.

[7] A. Ibrahim, S. Saxena, And R. A. A. Al Rabeei, "Interdisciplinary On Digital Privacy."

[8] A. Shah, S. Clachar, M. Minimair, And D. Cook, "Building Multiclass Classification Baselines For Anomaly-Based Network Intrusion Detection Systems," In 2020 Ieee 7th International Conference On Data Science And Advanced Analytics (Dsaa), 2020: Ieee, Pp. 759-760.

[9] S. Sadhwani, B. Manibalan, R. Muthalagu, And P. Pawar, "A Lightweight Model For Ddos Attack Detection Using Machine Learning Techniques," Applied Sciences, Vol. 13, No. 17, P. 9937, 2023.

[10] N. Ibrahim, S. Shehmir, A. Yadav, And R. Kashef, "Intrusion Detection: Architecture, Classification Heads, And Transformer," In Proceedings Of Iemtronics 2024: International Iot, Electronics And Mechatronics Conference, Volume 1, 2025, Vol. 1: Springer Nature, P. 149.

[11] S. K. R. Mallidi And R. R. Ramisetty, "Advancements In Training And Deployment Strategies For Ai-Based Intrusion Detection Systems In Iot: A Systematic Literature Review," Discover Internet Of Things, Vol. 5, No. 1, P. 8, 2025.

[12] M. M. Rahman, S. Al Shakil, And M. R. Mustakim, "A Survey On Intrusion Detection System In Iot Networks," Cyber Security And Applications, Vol. 3, P. 100082, 2025.

[13] Y. Liu, Z. Pang, M. Karlsson, And S. Gong, "Anomaly Detection Based On Machine Learning In Iot-Based Vertical Plant Wall For Indoor Climate Control," Building And Environment, Vol. 183, P. 107212, 2020.

[14] G. Kalnoor And S. Gowrishankar, "Retracted Article: Iot-Based Smart Environment Using Intelligent Intrusion Detection System," Soft Computing, Vol. 25, No. 17, Pp. 11573-11588, 2021.

[15] M. Di Mauro, G. Galatro, G. Fortino, And A. Liotta, "Supervised Feature Selection Techniques In Network Intrusion Detection: A Critical Review," Engineering Applications Of Artificial Intelligence, Vol. 101, P. 104216, 2021.

[16] G. Singh And N. Khare, "A Survey Of Intrusion Detection From The Perspective Of Intrusion Datasets And Machine Learning Techniques," International Journal Of Computers And Applications, Vol. 44, No. 7, Pp. 659-669, 2022.

[17] C. Kolias, G. Kambourakis, A. Stavrou, And J. Voas, "Ddos In The Iot: Mirai And Other Botnets," Computer, Vol. 50, No. 7, Pp. 80-84, 2017.

[18] C. Wei, G. Xie, And Z. Diao, "A Lightweight Deep Learning Framework For Botnet Detecting At The Iot Edge," Computers & Security, Vol. 129, P. 103195, 2023.

[19] M. Ammar, G. Russello, And B. Crispo, "Internet Of Things: A Survey On The Security Of Iot Frameworks," Journal Of Information Security And Applications, Vol. 38, Pp. 8-27, 2018.

[20] O. A. Okpe, O. A. John, And S. Emmanuel, "Intrusion Detection In Internet Of Things (Iot)," International Journal Of Advanced Research In Computer Science, Vol. 9, No. 1, 2018.

[21] V. Kumar, A. K. Das, And D. Sinha, "Uids: A Unified Intrusion Detection System For Iot Environment," Evolutionary Intelligence, Vol. 14, No. 1, Pp. 47-59, 2021.