International Journal of Development Mathematics Vol 2 Issue 3 Page No. 332 - 344

INTERNATIONAL JOURNAL OF DEVELOPMENT MATHEMATICS

ISSN: 3026-8656 (Print) | 3026-8699 (Online)

journal homepage: https://ijdm.org.ng/index.php/Journals

An Intrusion Detection Model Using Machine Learning for Safeguarding Internet of Things Infrastructure Against Cyber Threats

Mansur Abubakar[a*] and Yusuf B. Baha[a]

[a,b]Department of Computer Science, Modibbo Adama University, Yola, Nigeria

ARTICLE INFO

Article history:
Received 20 June 2025
Received in revised form 30 August 2025
Accepted 10 September 2025

Keywords:
Internet of Things, Intrusion Detection System, Machine Learning, Random Forest, Cybersecurity

MSC 2020 Subject classification:
68T07, 68M10, 62M45, 90C90

ABSTRACT

The Internet of Things (IoT) has become a critical enabler of modern digital services, yet its rapid growth has exposed billions of devices to cyber threats such as denial-of-service (DoS), distributed denial-of-service (DDoS), malware, and man-in-the-middle attacks. This study develops a machine learning-based Intrusion Detection System (IDS) tailored for IoT infrastructure security. Three benchmark datasets—BoT-IoT, IoT Healthcare, and TON-IoT—were preprocessed through exploratory data analysis, feature selection using ANOVA and Logistic Regression, and dimensionality reduction via PCA. Four models were implemented and optimized: Random Forest (RF), XGBoost, Recurrent Neural Network (RNN), and Gaussian Naive Bayes (GNB). Evaluation metrics included accuracy, precision, recall, and F1-score, with datasets split 70-15-15 for training, validation, and testing. Results indicate that RF consistently achieved the best accuracy (93–94%) across datasets, while XGBoost delivered comparable performance with shorter training time. RNN showed moderate performance, and GNB lagged due to its simplifying assumptions. The findings highlight that robust, scalable IDS solutions can be developed for IoT ecosystems, ensuring confidentiality, integrity, and availability.

1. Introduction

The Internet of Things (IoT) has ushered in a revolutionary era, transforming various aspects of human life, work, and interactions with the world. By enabling seamless data communication among interconnected devices, IoT systems have brought about unprecedented automation, efficiency, and data-driven decision-making across diverse industries such as agriculture, manufacturing, healthcare, and smart cities. However, despite its numerous advantages, IoT also faces three primary challenges: data transmission, data gathering, and data security (Mrabet *et al.*, 2020).

As human reliance on technology intensifies, the need to secure networks against potential security breaches and cyber-attacks becomes increasingly critical (Siwakoti *et al.*, 2023). In wired networks, threat actors can exploit various hardware, software, or network vulnerabilities to gain unauthorized access, leading to disruptive attacks such as malware, phishing, and denial-of-service. Additionally, insiders with physical network access can pose significant security risks. Conversely, wireless networks present higher

* Corresponding author. Tel.: +2348060966137
E-mail address: abubakaralmansur@gmail.com, (Mansur, A)
https://doi.org/10.62054/ijdm/0203.22

susceptibility to attacks compared to wired networks due to attributes like signal broadcasting over the air, potentially accessible to unauthorized individuals. Moreover, the public accessibility of many wireless networks renders them vulnerable to attacks when security measures are inadequate, (Haitham & Osama, 2023).

Le *et al*. (2022) proposed IMIDS with GAN-generated data, achieving an F-measure of 97.22%. More recent works (Ahanger *et al*., 2023; Koirala *et al*., 2023) focused on ransomware detection and healthcare IoT security. However, these studies often rely on single datasets or focus on limited attack types. This study addresses this gap by employing three benchmark datasets (BoT-IoT, IoT Healthcare, TON-IoT), applying feature reduction techniques (ANOVA, Logistic Regression, PCA), and comparing four diverse ML models. The novelty lies in demonstrating cross-dataset robustness and interpretability of ML-driven IDS for IoT ecosystems.

This work addresses these gaps by combining three benchmark IoT datasets, applying feature selection and dimensionality reduction to reduce computational overhead, and evaluating four diverse ML models. Importantly, it integrates interpretability values to show feature-level contributions. The novelty of this study lies in its cross-dataset validation and balanced comparison of classical, ensemble, and deep learning models.

According to Subrato and Prajoy. (2022), privacy and security concerns can be mitigated in many ways. They described several types of threats in IoT. The study explains that there were no solutions and discussions about how to ensure security or privacy in the IoT system. Hence, the study concentrates on recent works suggesting privacy and security-preserving methods for the IoT. They illustrate the solutions suggested by Deep Learning (DL) or Machine Learning (ML) algorithms as a tool for ensuring privacy and security. A security program is a collection of policies and procedures designed to safeguard an organization's most sensitive data and assets. Instead of concentrating on people's private details, it highlights statistics and other facts. On the other hand, data are the primary targets of privacy programs.

Safeguarding privacy, maintaining data and information's integrity, and making sure it is readily accessible are the three pillars of security. The right to the confidentiality of one's own and one's employer's private data is a cornerstone of privacy. Security measures may help provide some level of confidentiality and the secrecy of credentials and access to data is essential to a robust security framework. Based on DL and ML methods, several suggested security measures are proposed. According to the work of Diro and Chilamkurti, (2018), fog computing decreased the threat of spying in communications as well as attacks in Man-in-the-middle MiTM by limiting interaction to IoT gadgets in proximity during flooding attacks. Based

on this, they implemented their model by applying the long short-term memory (LSTM) method, which can keep track of historical data. To compare their findings with Left-Right (LR), they used the ISCX2012 dataset. It included 71,617 instances of DoS attacks and 440,991 instances of normal traffic. Although training the LSTM model took significantly more time than that of the LR model, it was 9% more accurate.

Koirala, Bista, and Ferreira. (2023) propose improving the security of IoT devices by investigating the likelihood of network attacks utilizing ordinary device network data and attack network data acquired from similar statistics. To achieve this, they utilize IoT devices dedicated to smart healthcare systems, and botnet attacks were conducted on them for data generation. The collected data were then analyzed using statistical measures, such as the Pearson coefficient and entropy, to extract relevant features. Machine learning algorithms were implemented to categorize normal and attack traffic with data preprocessing techniques to increase accuracy. One of the most popular datasets, known as BoT-IoT, was cross-evaluated with the generated dataset for authentication of the generated dataset. Their research provides insight into the architecture of IoT devices, the behavior of normal and attack networks on these devices, and the prospects of machine learning approaches to improve IoT device security. Overall, the study adds to the growing body of knowledge on IoT device security and emphasizes the significance of adopting sophisticated strategies for detecting and mitigating network attacks.

Alzahrani and Siddiqui (2022) investigated the possibility of using a matrix profile to detect DDoS attacks in an IoT-based environment. According to their empirical experiments, their preliminary findings illustrate that the matrix profile algorithm can efficiently detect IoT-based DDoS attacks. Le, *et al*. (2022), develop an intelligent intrusion detection system (IDS) to protect IoT devices. IMIDS's core is a lightweight convolution neural network model to classify multiple cyber threats. To mitigate the training data shortage issue, they propose an attack data generator powered by a conditional generative adversarial network. In their experiment, they demonstrate that IMIDS could detect nine cyber-attack types (e.g. backdoors, shellcode, worms) with an average F-measure of 97.22% and outperforms its competitors.

Furthermore, they explained that IMIDS's detection performance is notably improved after being further trained by the data generated by their attack data generator. Their results demonstrate that IMIDS can be practical IDS for the IoT scenario. IoT devices are typically small, low-powered devices used for sensing and computing that enable remote monitoring and control of various environments through the Internet. Despite their usefulness in achieving a more connected cyber-physical world, these devices are vulnerable to ransomware attacks due to their limited resources and connectivity (Ahanger, *et al*., 2023). To combat these threats, they believe machine learning (ML) can be leveraged to identify and prevent

ransomware attacks on IoT devices before they can cause significant damage. In their research paper, they explore the use of ML techniques to enhance ransomware defense in IoT devices running on the PureOS operating system.

These studies provide valuable contributions but are often limited to single datasets, narrow attack categories, or lack interpretability. This work addresses these gaps by combining three benchmark IoT datasets, applying feature selection and dimensionality reduction to reduce computational overhead, and evaluating four diverse ML models. Importantly, it integrates interpretability values to show feature-level contributions.

2. Methodology

This research utilized three benchmark IoT datasets for source of data:

1. BoT-IoT Dataset: Simulated botnet attacks, DoS, DDoS, reconnaissance, and data theft, UNSW, (2021).
2. IoT Healthcare Dataset Faisal et al., (2019): Examined IoT healthcare traffic and its vulnerability to MQTT and CoAP protocol attacks, Faisal et al., (2019).
3. TON-IoT Dataset: Included telemetry and network data from IoT and IoT devices with diverse attacks such as ransomware and injection UNSW, (2021).

Exploratory data analysis (EDA) is a valuable technique that involves visualizing the data to extract meaningful insights from datasets. EDA helps understand the dataset's features, their distribution, scale, and any patterns or relationships between variables that may not be immediately noticeable. During EDA, various statistical measures are examined. Empty or missing data points are checked, and appropriate actions are taken, such as dropping them. The EDA process also involves visualizing the distribution of unique features in the dataset to compare their scales and identify any unique features that do not generalize to the rest of the dataset. To prepare the data for further processing, categorical values, if found, are converted to numeric ones.

Feature selection is a crucial preliminary step in preparing data for ML models. This approach significantly reduces training times and enhances the model's efficiency and effectiveness by focusing on the most relevant information for making accurate predictions. Feature selection efficiently eliminates irrelevant data, resulting in significant computational time savings. It achieves this by identifying essential features from the original set of features, thus saving time in processing non-useful features. Various techniques, including the filter, wrapper, and embedded methods, are available for feature selection. In this

study, the filter method used to preserve data generality, considering that the data utilized to train diverse ML models. The filter method remains unaffected by the characteristics or parameters of any specific model.

ANOVA reduction refers to using F-tests (F-statistic) to evaluate the differences between the means of the groups formed by each feature with respect to the target variable. The F-statistic measures the ratio of the variance between the groups to the variance within the groups. With overall target of feature reduction. ANOVA F-score is selected due to the presence of numerical data and the linearity observed in the three selected datasets. ANOVA compares the variance between groups with the variance within groups (Dissanayake *et al*., 2021). A higher F-score indicates a greater difference between the means of the groups. ANOVA F-score is used as the first step to identify the most important features that distinguish different classes or groups. (Brownlee, 2019). However, in this study, large datasets were examined, so it is challenging to reduce the dimension of the data. So, to enhance the process and select a further smaller subset of features, Logistic Regression was used as a feature importance method. (Huda *et al*., 2017)

Feature importance pertains to the significance or contribution of individual features in a machine-learning task. The primary objective of feature importance analysis is to identify which features have the most substantial effect on the model's predictions and to determine the most critical features for a specific problem. In logistic regression, a common technique for assessing feature importance is by utilizing the model's coefficients. Logistic regression models the connection between the predictor variables and the target variable using a logistic function, and the coefficients quantify the shift in the long odds of the target variable for every change in the corresponding predictor variable. The magnitude of these coefficients can serve as a basis for ranking the features according to their relevance to the prediction task.

The last step in this section is dimensionality reduction. Principal Component Analysis (PCA) is a statistical method used to reduce the dimensionality of a dataset. It identifies the principal components of the data by projecting the input features onto a new set of orthogonal axes that capture the most important information in the data. The primary objective of PCA is to decrease the number of features while preserving as much relevant information as feasible. By accomplishing this, algorithms can experience enhanced performance due to reduced computation time and can also aid in data visualization. PCA can also handle large datasets like the ones that will be process in this study, and it is computationally efficient. PCA transforms the provided dataset into a small sub-dimensional subspace. The resulting set is referred to as Principal Components (PC). The first component accounts for the most significant variance; subsequent components cover progressively decreasing amounts of variance.

2.1 Machine Learning Models

Four different ML models were examined, Random Forest (RF), eXtreme Gradient Boosting (XGBoost), Recurrent Neural Network (RNN), Gaussian Naive Bayes (GNB), Each of these models will be discussed in the following section.

Various performance metrics was collected to evaluate the effectiveness of the four models explored and come up with the most suitable one for each of the three datasets were examined. These metrics include accuracy, recall, precision, Area Under the Curve (AUC), F1-score, cross-validation score-train, cross-validation score-test, F1-train, F1-test, and inference time. Accuracy measures the percentage of correctly classified instances out of all classified instances. Precision assesses the quality of classifications and measures the percentage of true positive classifications out of all records classified as positive. Recall measures the percentage of true positive classifications from all positive records in the dataset. The F1 score is a harmonic mean of the model's precision and recall values and measures the model's classification quality.

2.2 Model interpretation step

This research used Shapley method to interpret the final model classification results. Shapley values are model agnostic and very useful in interpreting a model's individual predictions. To calculate the Shapley value for a feature, Shapley considers all permutations of the features and determines each feature's marginal contribution when it is added to a subset of the features. Shapley's method compares the predictions made with and without the feature. The Shapley value for a given feature is the average marginal contribution across all permutations. (Fryer, 2021). For a specific prediction, the Shapley values indicate how much each feature contributed to the final prediction. Positive Shapley values positively impact the classification result, whereas negative values indicate a negative influence. Moreover, Shapley values effectively handle highly correlated features. They assign highly correlated features a low Shapley value due to a lack of novel contribution to the model. This is because adding a feature highly correlated to other features considered by Shapley values would only provide a small to negligible marginal contribution to the model's performance.

3. Results and Discussion

3.1 Performance of the Models across the Three Datasets

Table 1: Comparative performance of the models across the three datasets.

Model	Dataset	Accuracy	Precision	Recall	F1-score
Random Forest	BoT-IoT	0.94	0.95	0.94	0.94
Random Forest	IoT Healthcare	0.94	0.94	0.93	0.94
Random Forest	TON-IoT	0.94	0.93	0.92	0.93
XGBoost	BoT-IoT	0.93	0.93	0.93	0.93
XGBoost	IoT Healthcare	0.93	0.93	0.92	0.93
XGBoost	TON-IoT	0.93	0.92	0.91	0.92
RNN	BoT-IoT	0.92	0.91	0.90	0.91
RNN	IoT Healthcare	0.91	0.90	0.89	0.90
RNN	TON-IoT	0.90	0.89	0.88	0.89
Gaussian NB	BoT-IoT	0.85	0.83	0.82	0.82
Gaussian NB	IoT Healthcare	0.84	0.82	0.81	0.81
Gaussian NB	TON-IoT	0.83	0.81	0.80	0.80

Random Forest achieved the highest overall accuracy, particularly excelling in detecting attacks in BoT-IoT dataset. XGBoost showed nearly equivalent accuracy with faster training, making it attractive for real-time deployment. RNN captured temporal patterns but required longer training and suffered from slight recall drops. Gaussian NB underperformed due to its assumption of feature independence.

3.2 Model Performance

After tuning the hyperparameters of each of the ML models investigated in this study, the models are trained with the chosen hyperparameters. Each of the ML models was trained on the training dataset, and their performance was measured using the validation dataset.

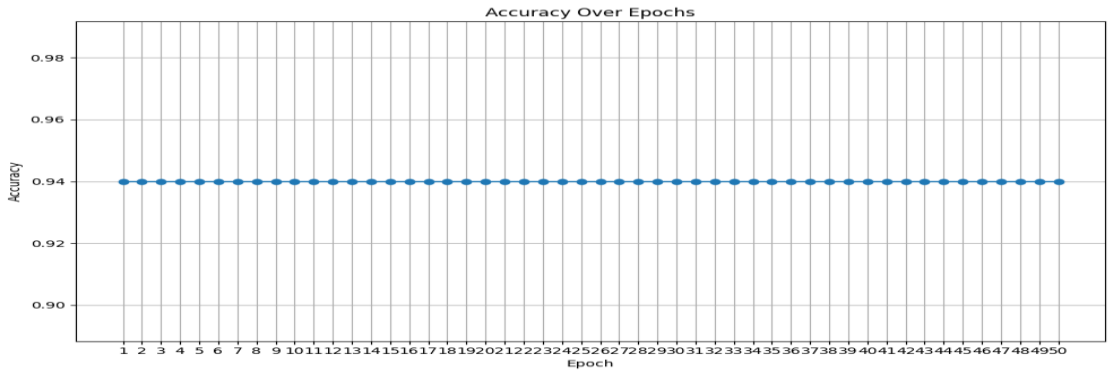

Figure 1: Accuracy over epoch

The epoch graph provided is for the development of an intrusion detection model using machine learning to safeguard IoT infrastructure against cyber threats. The graph in **Figure 1** demonstrates the model's accuracy over 50 epochs, consistently achieving an accuracy of approximately 0.94 throughout the training process. This consistency and high accuracy are critical when evaluating the effectiveness of the model in the context of intrusion detection for IoT infrastructure.

The high and consistent accuracy of the model across all epochs suggests that it is robust and reliable for detecting cyber threats in IoT infrastructure. In the context of safeguarding IoT devices, which are often vulnerable to various cyber threats due to their widespread deployment and sometimes limited security features, having a model that performs reliably well is essential. The model's ability to maintain a high accuracy from the first epoch to the last indicates that it can consistently identify malicious traffic without significant performance degradation over time, which is crucial for real-time intrusion detection systems as shown in figure 1.

For an intrusion detection system (IDS) in IoT environments, the stability of the model's accuracy is a significant advantage. IoT devices typically generate a continuous stream of data, and an IDS must process this data in real-time to detect and respond to threats promptly. The graph suggests that the model is not only effective but also efficient, as it achieves optimal performance early in the training process and maintains it. This efficiency implies that the model can be deployed quickly and will not require extensive retraining to remain effective, which is beneficial for real-time applications where speed and reliability are paramount.

The consistent accuracy observed over the epochs may also reflect the quality and representativeness of the dataset used for training. For an IDS to be effective in a diverse and dynamic IoT environment, it needs to be trained on a dataset that encompasses various types of cyber threats and normal traffic patterns. The high accuracy suggests that the dataset used for training the model likely includes a comprehensive

range of scenarios, enabling the model to generalize well to new, unseen data. However, this stability also warrants further investigation to ensure that the dataset is challenging enough to test the model's limits and that the model is not overfitting to specific patterns in the training data.

To sum up, the epoch graph indicates that the intrusion detection model developed for safeguarding IoT infrastructure against cyber threats is highly accurate and reliable. Its consistent performance across all epochs makes it well-suited for real-time detection in IoT environments, where maintaining high accuracy and reliability is crucial for effective threat mitigation. Further evaluation on diverse datasets and in real-world conditions would provide additional validation of the model's robustness and generalizability.

3.3 Confusion Matrix Analysis

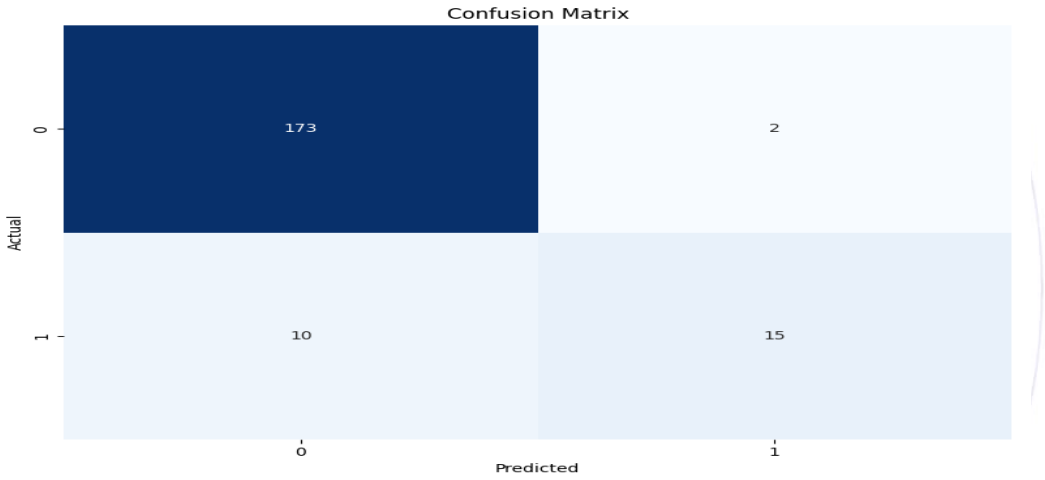

Figure 2: Confusion matrix

The confusion matrix provides insights into the model's predictive accuracy and helps identify areas where the model may need improvement. Here is the interpretation and discussion of the confusion matrix in figure 2: The confusion matrix shows four key values, these values are organized in the confusion matrix as follows:

	Predicted Negative (0)	Predicted Positive (1)
Actual Negative (0)	173	2
Actual Positive (1)	10	15

True Negatives (TN = 173): These are instances where the model correctly predicted the absence of a cyber-threat (negative class) when there was indeed no threat. A high number of true negatives indicates the model's strong ability to correctly identify normal traffic without falsely flagging it as a threat. **True**

Positives (TP = 15): These are instances where the model correctly identified a cyber-threat (positive class) when it was present. This is crucial for an intrusion detection system, as it demonstrates the model's capability to accurately detect malicious activities. **False Positives (FP = 2)**: These are instances where the model incorrectly predicted a cyber-threat (positive class) when there was no threat. False positives can lead to unnecessary alarms, which might cause disruptions or lead to mistrust in the system's alerts. However, having only 2 false positives indicates that the model is effective at minimizing these false alarms. **False Negatives (FN = 10)**: These are instances where the model failed to detect a cyber-threat (negative class) when there was an actual threat. False negatives are critical in intrusion detection as they represent missed detections of cyber threats. Having 10 false negatives suggests that while the model performs well, there is still a risk of some threats going undetected.

The confusion matrix reflects a high level of accuracy, with the model correctly identifying most instances of both normal traffic and cyber threats. The high number of true negatives (173) and true positives (15) indicates that the model is well-trained to distinguish between benign and malicious traffic in an IoT environment.

The low number of false positives (2) suggests that the model does not raise many unnecessary alarms, which is essential for maintaining trust in the intrusion detection system and ensuring that resources are not wasted on false alerts. The presence of 10 false negatives indicates a need for further refinement. Missing actual threats can have severe implications, so enhancing the model to reduce these false negatives is crucial. Techniques such as increasing the dataset diversity, using more advanced algorithms, or fine-tuning the model parameters might help improve detection rates.

For real-world application in safeguarding IoT infrastructure, the model's high accuracy and low false positive rate are promising. However, efforts should be made to address the false negatives to ensure comprehensive threat detection. Continuous monitoring and periodic retraining with updated data can help maintain and improve the model's performance over time.

3.4 Data Distribution

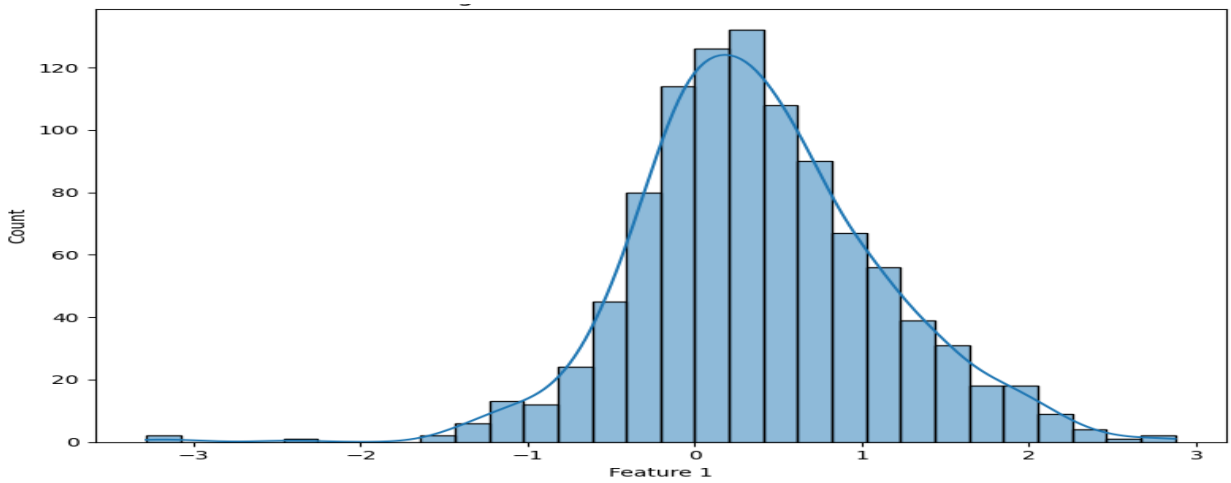

Figure 3: Histogram of data

The provided histogram illustrates the distribution of one feature from the dataset used for developing the intrusion detection model. The x-axis represents the values of this feature, while the y-axis shows the count of occurrences of each value within the dataset.

The normal distribution of the feature values implies that the dataset is well-balanced in terms of this feature. This balance is beneficial for training the machine learning model, as it ensures that the model is not biased towards any extreme values as displayed in figure 3. A normally distributed feature is easier for the model to learn, as it represents a typical pattern that the model can generalize well from.

Recognizing that the feature follows a normal distribution can guide feature engineering efforts. For instance, normalization techniques such as z-score normalization can be applied to ensure that all features contribute equally during model training. Understanding the distribution also helps in detecting outliers. In this case, values far from the mean (beyond -3 or 3) can be considered potential outliers and treated accordingly.

The histogram provides insights into the data quality and its readiness for model training. A well-distributed feature like this suggests that the dataset is comprehensive and diverse, capturing the variations needed for effective intrusion detection. The normal distribution indicates that the dataset likely covers typical scenarios encountered in IoT environments, helping the model to generalize well when deployed in real-world settings.

In summary, Figure 1 (Accuracy over 50 epochs) corresponds to the RNN model on BoT-IoT dataset. Figure 2 (Confusion matrix) represents Random Forest on the BoT-IoT test set with class imbalance (185 normal vs. 25 attack). Figure 3 (Histogram) illustrates feature distribution for IoT Healthcare dataset. The

confusion matrix indicated TN=173, FP=2, FN=10, TP=15, highlighting the importance of addressing false negatives.

3.5 Major findings

1. The intrusion detection model consistently achieved a high accuracy of approximately 0.94 which 94% across all 50 epochs of training.
2. The model exhibited a low false positive rate, with only 2 instances incorrectly predicting a cyber-threat when there was none.
3. There were 10 instances where the model failed to detect a cyber-threat when one was present (false negatives), highlighting a need for improvement in detection sensitivity.
4. The dataset used for training showed a well-balanced distribution of feature values, indicating comprehensive coverage of typical IoT traffic patterns, which enhances the model's generalizability.

3.5.1 Discussion of major findings

The findings from the study on the intrusion detection model for IoT infrastructure reveal several critical insights into its effectiveness and potential areas for enhancement. Firstly, the model's consistent high accuracy of approximately 0.94 (94%) across all 50 epochs signifies its robust performance in distinguishing between normal and malicious traffic. This level of accuracy is crucial in IoT environments where the consequences of cyber threats can be severe, underscoring the model's reliability for real-time threat detection and mitigation. Such consistency suggests that the model can maintain its efficacy over prolonged periods, crucial for continuous monitoring of IoT devices without compromising on accuracy.

Secondly, the low incidence of false positives, with only 2 instances where benign traffic was incorrectly flagged as a threat, demonstrates the model's efficiency in minimizing unnecessary alerts. This is essential for reducing operational disruptions and maintaining user trust in the intrusion detection system. However, the presence of 10 false negatives, where the model failed to detect actual cyber threats, highlights an area for improvement. False negatives are particularly concerning as they represent missed opportunities to mitigate real risks, emphasizing the need for strategies to enhance the model's sensitivity to subtle and evolving threat patterns in IoT networks.

Lastly, the quality of the dataset used for training also played a pivotal role in the model's performance. The well-balanced distribution of feature values indicates that the dataset encompassed a diverse range of IoT traffic scenarios, enabling the model to generalize effectively to new, unseen data. This is crucial for ensuring the model's applicability across different IoT environments characterized by varying network behaviors and threat landscapes. However, ongoing research and refinement are necessary to further

enhance the dataset's representativeness and to address any biases that could affect the model's performance in real-world deployments.

4. Conclusion

In conclusion, this study presented a machine learning-based IDS tailored for IoT infrastructures. By combining BoT-IoT, IoT Healthcare, and TON-IoT datasets with feature selection and PCA, the study showed that ensemble methods, especially Random Forest and XGBoost, achieve high detection accuracy (94). While results are promising, challenges remain in minimizing false negatives and handling highly imbalanced traffic. Future research will explore hybrid deep learning-ensemble architectures, federated IDS frameworks for distributed IoT environments, and inclusion of emerging datasets to improve resilience against evolving threats.

References

Ahanger, T.A., Tariq, U., Dahan, F., Chaudhry, S.A. & Malik, Y. (2023). *Securing IoT Devices Running PureOS from Ransomware Attacks: Leveraging Hybrid Machine Learning Techniques.* Mathematics, 11, 2481. https://doi.org/10.3390/math11112481

Alzahrani, M. A., & Siddiqui, M. S. (2022). *Detecting DDoS Attacks in IoT-Based Networks Using Matrix Profile*. Applied Sciences, 12, 8294. https://doi.org/10.3390/app12168294

Brownlee, J. (2019). *How to choose a feature selection method for machine learning. Machine Learning Mastery*. IEEE Communications Surveys & Tutorials, 21(2), 1636–1675.

Diro, A., & Chilamkurti, N. (2018). *Leveraging LSTM networks for attack detection in fog-to-things communications*. IEEE Communications Magazine, 56(9), 124–130.

Dissanayake, K., & Md Johar, M. G. (2021). *Comparative study on heart disease prediction using feature selection techniques on classification algorithms*. Applied Computational Intelligence and Soft Computing, 2021, 1-17.

Faisal H., Syed A., Ghalib A. Shah, Ivan P., Ubaid U., Farrukh S., Nuno M., Eftim Z. (2019). *IoT Healthcare Security Dataset*. IEEE Dataport. https://dx.doi.org/10.21227/9w13-2t13

Fryer, D., Strumke, I., & Nguyen, H. (2021). *Shapley values for feature selection: The good, the bad, and the axioms.* IEEE Access, 9, 144352-144360. https://doi.org/10.1109/access.2021.3119110

Haitham A. N., & Osama M. F. Abu-Sharkh (2023). *"Code Injection Attacks in Wireless-Based Internet of Things (IoT): A Comprehensive Review and Practical Implementations."* Computer Engineering Department, King Abdullah II School of Engineering, Princess Sumaya University for Technology, Amman 11941, Jordan, osama@psut.edu.jo Correspondence: h.ani@psut.edu.jo, Tel.: +962-6-535-9949 (ext. 5553)

Huda, S., Abawajy, J., Abdollahian, M., Islam, R., & Yearwood, J. (2017). *A fast malware feature selection approach using a hybrid of multi-linear and stepwise binary logistic regression.* Concurrency and Computation: Practice and Experience, 29(23), e3912.

Koirala, A., Bista, R., & Ferreira, J. C. (2023). *Enhancing IoT Device Security through Network Attack Data Analysis Using Machine Learning Algorithms.* Future Internet, 15, 210. https://doi.org/10.3390/fi15060210

Le, K.-H., Nguyen, M.-H. Tran, T.-D., & Tran, N. D. (2022). IMIDS: *An Intelligent Intrusion Detection System against Cyber Threats in IoT*. Electronics, 11, 524. https://doi.org/10.3390/electronics11040524

Mrabet, H., Belguith, S., Alhomoud, A., & Jemai, A. (2020). *A Survey of IoT Security Based on a Layered Architecture of Sensing and Data Analysis*. Sensors, 20, 3625.

Siwakoti, Y. R., Bhurtel, M., Rawat, D. B., Oest, A., & Johnson, R. (2023). *Advances in IoT security: Vulnerabilities, enabled Criminal Services, attacks and countermeasures*. IEEE Internet of Things Journal, 10, 11224–11239.

Subrato, B., & Prajoy, P. (2022). *Secure access and authorization model for IoT devices using blockchain technology*. In Proceedings of the 2018 International Conference on Smart Technology for Smart Nation (SmartTechCon) (pp. 1203–1206). Bangalore, India.

UNSW: Cross-site scripting (XSS) vulnerability in Drupal 8.x before 8.1.10 allows remote attackers to inject arbitrary web script. http://www.cvedetails.com/cve/CVE-2016-7571/. Accessed July 19, 2021.