Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

IoT Intrusion Detection System Using Machine Learning Classifiers

Jyoti Mante, Kishor Kolhe

School of Computer Engineering and Technology, Dr. Vishwanath Karad MIT World Peace University, Pune, India
Email: jyoti.khurpade@mitwpu.edu.in

The widespread integration of interconnected devices in the Internet of Things (IoT) has revolutionized modern lifestyle but also introduced new security vulnerabilities. Conventional security measures often struggle to defend IoT devices against a range of malicious attack. Machine learning based Intrusion Detection Systems (IDS) have emerged as a promising solution to address these challenges. The proposed work provides a comprehensive review of IDSs designed for IoT devices and evaluates the effectiveness of various Machine Learning Classifiers using the TON_IoT dataset, which encompasses four distinct device types: motion light, thermostat, fridge and garage door. Evaluation metrics such as accuracy, precision, F1 score and recall are employed to analyze and compare the performance of the IDSs in detecting the attacks.

Keywords: IoT, Intrusion Detection System (IDS), malicious attacks, machine learning, classifiers, SMOTE, imbalanced, Standard Scalar.

1. Introduction

The Internet of Things (IoT) encompasses a vast network of sensors and devices interconnected via the Internet, enabling the smooth flow of data interchange Khurpade et al., (2018). This interconnectedness, while enabling unprecedented levels of efficiency and convenience, also introduces significant security challenges. With IoT devices often linked to sensors and integrated with large cloud servers, the volume of smart city network traffic through IoT systems is escalating rapidly, presenting novel cybersecurity threats Mante et al. (2023).

In response to these challenges, intrusion detection, a crucial security technology, plays a crucial role in protecting IoT-enabled devices from a myriad of malicious threats. Intrusion Detection Systems (IDS) serve as a frontline defense, detecting and mitigating security threats effectively P. Mishra et al. (2019). The emergence of AI-based IDS represents a promising frontier in IoT security, offering innovative solutions to address evolving threats A. Mishra et al. (2023).

Khraisat et al (2019), investigated that despite advancements in IDS technology, there remains a gap in understanding about the use cases of machine learning models to diverse IoT devices. This study aims to explore Intrusion Detection Systems (IDS) for diverse IoT devices, such as fridges, thermostats, motion lights, and garage doors, using the TON_IoT dataset proposed by N. Moustafa et al. (2020). The research assesses the effectiveness of different machine learning classifiers, including Random Forest, K-Nearest Neighbors (KNN), Multi-Layer Perceptron (MLP), and Decision Trees, in identifying intrusions across a range of IoT devices, with performance evaluation focusing on different performance metrics. The performance metrics most probably used for comparison are accuracy, precision, recall and F1-score to gauge the effectiveness of IDSs in securing IoT environments.

The main contribution of this research is as follows:

- Survey of existing work related to attack detection in IoT environment
- Use of TON_IoT datasets related to IoT network
- Performance of Machine Learning Classifiers evaluated using performance metrics

The research article is organized as Literature Survey in section 2, detailed Methodology in section 3 and Results and Discussion in section 4 followed by conclusion.

2. Literature Survey

Numerous studies highlight the dynamic role of Intrusion Detection Systems (IDS) in protecting IoT environments from attacks. Traditional security measures aren't sufficient, underscoring the necessity for IDS technology. As IoT becomes more prevalent in sectors such as smart transportation and homes, it becomes more vulnerable to cyber threats. Given that standard security methods often fall short in protecting IoT devices adequately, implementing Intrusion Detection Systems (IDS) has become imperative.

G. J. Pandeeswari et.al. (2022), investigate enhancing cybersecurity threat detection by training distinct datasets for specific attacks and optimizing feature selection from various algorithms, notably Naive Bayes and QDA, achieving around 90% success in identifying threats. Through a process of combining optimal features and iterative evaluation, the research demonstrated a significant improvement in detection accuracy and efficiency, reducing both detection time and algorithm runtime. The findings suggest the potential for further exploration of more sophisticated algorithms and complex attack scenarios, indicating a progressive approach to improving cybersecurity measures.

S. Bhardwaj et al. (2021), explores machine learning in intrusion detection, noting the varied effectiveness of algorithms across different attack types and emphasizing the need for attack-specific approaches. They critically analyze the performance of single and multiple classifier systems, identifying a gap in detecting low frequency attacks. The study acknowledges limitations in testing across popular datasets and outlines future work aimed at improving detection methods for low-frequency attacks and adapting intrusion detection systems (IDS) for dynamic environments like cloud computing.

B. Omarov et al. (2022), presented advances in feature selection and classification using machine and deep learning techniques for intrusion detection, highlighting the effectiveness of hybrid approaches over singular algorithms. Emphasize the need for innovative methods to detect previously unknown attack patterns within datasets. The study suggests exploring the fusion or modification of existing algorithms as a promising direction for future research to improve detection accuracy and address the challenges of evolving cybersecurity threats.

G. Guo (2022), tested various classifiers, including Logistic Regression, Naive Bayesian variations, k-Nearest Neighbors, Decision Tree, Adaptive Boosting, Random Forest, Multilayer Perceptron, and Gradient Boosting on the UNSWNB15 dataset for intrusion detection. Metrics such as accuracy, precision, and F-measure were used for evaluation. Results highlighted the Random Forest classifier as superior in performance across these metrics. Future research directions include investigating the role of selective feature inclusion and employing newer datasets for enhanced intrusion detection analysis.

S. Chishakwe et al. (2022), examines machine learning algorithms for IoT anomaly detection, comparing approaches. It addresses security challenges arising from increased device connectivity and 5G network demands. Metrics like precision, recall, accuracy, F1-score, and execution times were used for assessment. Support Vector Machine showed high accuracy but slower execution, while Logistic Regression and Naïve Bayes offered acceptable accuracy with faster runtime. Future work involves developing tailored deep neural networks for IoT intrusion detection.

M. Bagaa et al. (2020), introduced an IDS tailored for IoT systems, utilizing the TON_IoT network dataset. Eight base algorithms and two ensemble models were applied, with XGB selected for anomaly detection due to its superior performance in both accuracy and time efficiency. Additionally, the study extends prior research by thoroughly assessing the quality and performance of the TON_IoT dataset. These findings serve as a crucial step towards implementing real-world IDSs in IoT environments

C. Nixon et al. (2019), presents IDS methodologies, highlighting the limitations of current machine learning techniques in detecting zero-day attacks due to outdated datasets like DARPA/KDD99. Study stresses the need for new, comprehensive datasets to reflect modern malware activities accurately. Additionally, it examines the effectiveness of IDS against evasion techniques, emphasizing the challenge of developing systems capable of accurately detecting a broad spectrum of attacks.

C. Liang et al. (2019), review IDS designs for IoT, emphasizing the need for efficient, lightweight systems tailored for smart environments. They suggest the development of an integrated IDS tested on a unified IoT database, considering placement strategies to preserve the IoT's integrity and availability. Future work includes creating a high-performance hybrid IDS, addressing IoT security vulnerabilities, and implementing the system on adaptable hardware like FPGAs to support various deployment models and detect diverse attacks.

Thus, it was noted that a critical role is played by Intrusion Detection Systems (IDS) in safeguarding Internet of Things (IoT) networks. In the proposed study, the efficacy of different Machine Learning (ML) classifiers, including K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP), Random Forest (RF), and Decision Tree algorithms, is assessed using the

TON_IoT dataset, which encompasses various IoT layers such as cloud, fog, and edge, aiming to provide insights into their comparative performance. The primary aim is to pinpoint the most accurate ML classifier for detecting cyber threats within IoT environments. This investigation yields valuable insights, contributing to the enhancement of IoT network security frameworks and laying the groundwork for future advancements towards more secure IoT systems.

3. Methodology

This section discusses the methodology adopted to assess the efficacy of various machine learning classifiers for detecting intrusions in IoT devices.

3.1 Dataset Selection

The primary dataset proposed by N. Moustafa et al. (2020), is selected for this study is the TON_IoT dataset, comprising four distinct files:

1) IoT_Fridge.csv

2) IoT_Garage_Door.csv

3) IoT_Motion_Light.csv

4) IoT_Thermostat.csv

This dataset offers a comprehensive range of IoT-related data, facilitating thorough exploration and analysis within the IoT domain. The TON_IoT dataset undergoes the following preprocessing steps:

The overall architecture diagram for intrusion detection system is shown in Figure 1. And the detailed steps are elaborated in the next sub section.

3.2 Pre-processing

The TON_IoT dataset undergoes the following preprocessing steps:

1) Data Cleaning: Systematically identifying and removing missing or null data to enhance dataset quality and reliability and give accurate insights and decision-making.

2) Encoding: Categorical data is transformed into numerical form through label encoding. This conversion is crucial for machine learning algorithms to effectively process categorical variables, thereby facilitating analysis and modeling.

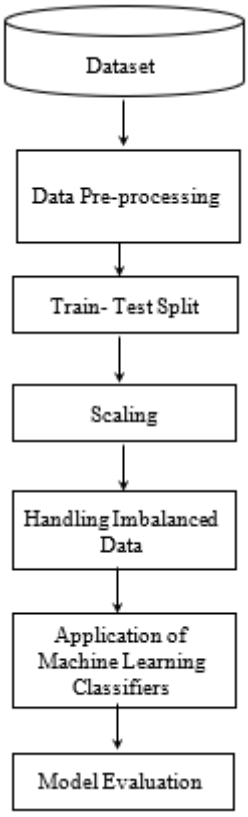

Figure 1. Architecture of IDS

3.3 Train and Test Samples

The preprocessed dataset for each IoT device is divided into training sets with 80% of data samples for training the models and testing sets with 20% of data samples for testing the models to classify the attacks, thus using a proportion of 80:20.

3.4 Scaling

Feature values for each dataset are standardized using a standard scaler. This ensures that each feature contributes equally to the model training process and enhances the performance of machine learning algorithms sensitive to feature scaling.

StandardScaler normalizes each feature by subtracting the mean and scaling it to achieve unit variance.

$$xi - mean(x)/stdev(x) \quad (5)$$

For a feature xi , the mean and standard deviation of the feature set x are computed, and then xi is scaled accordingly using equation (5).

3.5 Handling Imbalanced Data

To address imbalanced data, the Synthetic Minority Oversampling Technique (SMOTE) is

applied following standard scaling. This process augments minority class instances, promoting a balanced distribution of intrusion and normal instances in the dataset. By mitigating biases caused by data imbalance, it enhances the reliability of analysis and model performance.

3.6 Machine Learning Classifiers

Various machine learning classifiers are utilized by researchers P. Mishra et al., (2018) and Almomani et al.(2021), to predict intrusions in each type of IoT device. In this research the classifiers considered include:

1) Random Forest (RF): It is an ensemble learning technique comprising numerous decision trees, known for its versatility, scalability, and reduced susceptibility to overfitting compared to standalone decision trees.

2) K-Nearest Neighbors (KNN): An algorithm for classifying data points relies on determining the most common class among their nearest neighbors, popular for its simplicity and effectiveness across diverse classification tasks.

3) Multi-Layer Perceptron (MLP): A sophisticated artificial neural network proficient in discerning intricate patterns within data via interconnected layers of neurons, adept at handling tasks requiring high-dimensional data analysis and modeling.

4) Decision Trees (DT): Models commonly used for classification tasks, offering a straightforward and interpretable approach by dividing the feature space into decision nodes, valuable for exploratory analysis and decision-making across various fields.

3.7 Model Evaluation

The model evaluation encompasses training and testing these classifiers on the TON_IoT dataset, emphasizing diverse IoT devices to ensure applicability across the entire spectrum. Performance metrics, such as accuracy equation (1), precision equation (2), recall equation (3), and F1 score equation (4), will be employed to gauge the effectiveness of each classifier in detecting intrusions. This thorough approach seeks to offer an in-depth understanding of the effectiveness and advantages of various machine learning classifiers in the context of IoT security.

Accuracy shows how well a model can tell right from wrong answers.

$$\text{Accuracy} = Tp + Tn \,/\, Tp + Tn + Fp + Fn \qquad (1)$$

It basically tells us how good the model is at getting the answers right for different types of questions.

Precision evaluates how well a model identifies true positives within all instances it predicts as positive.

$$\text{Precision} = Tp \,/\, Tp + Fp \qquad (2)$$

In essence, precision measures the model's accuracy in labeling positive cases correctly, minimizing false positives and ensuring reliability in positive predictions.

Recall, also called sensitivity or true positive rate, shows how well a model finds all the positive cases in the data.

$$\text{Recall} = Tp / Tp + Fn \quad (3)$$

Essentially, recall measures how thorough the model is at identifying all actual positive instances, making sure it doesn't miss any.

The F1 Score, which represents the harmonic mean of precision and recall, offers a balanced metric that integrates both aspects into a single value.

$$\text{F1 score} = 2 * Precision * Recall / Precision + Recall \quad (4)$$

The F1 score merges precision and recall into a unified metric, offering an assessment of a model's overall effectiveness, particularly when there is a trade-off between precision and recall.

4. Results and Discussion

Experimental Environment

The experiment was performed on a personal MacBook Pro laptop equipped with an M3 processor, 32 GB RAM, and 512 GB SSD, running the Ventura operating system. Model training and execution were carried out using Google Colab Notebook. Data loading and preprocessing utilized the Pandas and NumPy libraries, while data visualization was accomplished with Matplotlib and Seaborn. The experiment's performance is evaluated using the Scikit-learn framework.

Performance metrics such as accuracy, precision, recall and F1- score are used to define how the trained models accurately classify and detect the attacks as shown in the Tables I, II, III, IV.

Table 1: Evaluation result of Fridge Dataset

Classifiers	Metrics			
	Accuracy	Presicion	Recall	F1 Score
KNN	99.214	95.515	99.355	97.397
MLP	97.378	85.161	99.635	91.831
RF	99.502	97.096	99.612	98.338
DT	99.452	97.090	99.274	98.170

From TABLE I: K-Nearest Neighbors (KNN) showcases good performance with 99.214% Accuracy ,95.515% of Precision,99.355% of Recall, and an F1 Score of 97.397%. These figures suggest KNN is highly effective in correctly identifying positive samples and maintaining a good equilibrium between precision and recall, though it's slightly less precise than other top performers. Multi-Layer Perceptron (MLP) significantly lags in performance, having an Accuracy of 97.378%, Precision at 85.161%, high Recall at 99.635%, but a lower F1 Score of 91.831%. MLP's high recall indicates it's good at identifying positive cases, but its low precision and overall accuracy point to a high rate of false positives, reducing its effectiveness. Random Forest (RF) emerges with impressive metrics, with 99.502% of Accuracy, 97.096% of Precision,99.612% of Recall, and an F1 Score of 98.338%. RF's performance is exemplary, indicating it not only accurately classifies instances but also maintains an excellent balance between identifying all relevant instances and ensuring

minimal false positives. Decision Tree also performs good, having 99.452% Accuracy, 97.09% of Precision,99.274% of Recall, and an F1 Score of 98.17%. This suggests the Decision Tree is almost as effective as RF, with a slightly lower recall and maintaining less precision and accuracy, creating it a good choice for classification of attacks.

From TABLE II: K-Nearest Neighbors (KNN) showcases strong performance with 99.571% Accuracy, 97.265% Precision, 99.798% Recall, and an F1 Score of 98.515%. Multi-Layer Perceptron (MLP) significantly lags in performance with context to the other models, having an Accuracy of 98.378%, Precision at 89.949%, high Recall at 99.783%, but a lower F1 Score of 94.611%. MLP's high recall indicates it's good at identifying positive cases, but its low precision and overall accuracy point to a high rate of false positives, reducing its effectiveness. Random Forest (RF) emerges with impressive metrics, with 99.533% Accuracy, 97.488% Precision,99.278% Recall, and an F1 Score of 98.379%. RF's performance is exemplary, indicating it not only accurately classifies instances but also maintains an excellent balance between identifying all relevant instances and ensuring minimal false positives. Decision Tree also performs well, having 99.569% Accuracy, 97.528% Precision, 99.504% Recall, and an F1 Score of 98.506%. This suggests the Decision Tree is almost as effective as KNN, with a KNN having slightly lower precision but maintaining high recall, accuracy and F1-score, making it a reliable choice for classification tasks.

Table II: Evaluation result of Motion Light Dataset

Classifiers	Metrics			
	Accuracy	Precision	Recall	F1 score
KNN	99.571	97.265	99.798	98.515
MLP	98.378	89.949	99.783	94.611
RF	99.533	97.488	99.278	98.379
DT	99.569	97.528	99.504	98.506

From TABLE III: K-Nearest Neighbors (KNN) showcases strong performance with 99.415% of an Accuracy, 95.95% Precision, 99.602% Recall, and an F1 Score of 97.742%. Multi-Layer Perceptron (MLP) significantly lags in performance, having an Accuracy of 99.029%, Precision at 93.148%, high Recall at 99.691%, but a lower F1 Score of 96.309%. MLP's high recall indicates it's good at identifying positive cases, but its low precision and overall accuracy point to a high rate of false positives, reducing its effectiveness. Random Forest (RF) emerges with impressive metrics, with 99.468% of Accuracy, 96.410% of Precision,99.519% of Recall, and an F1 Score of 97.940%. RF's performance is exemplary, indicating it not only accurately classifies instances but also maintains an excellent balance between identifying all relevant instances and ensuring minimal false positives. Decision Tree also performs well, having 99.329% Accuracy, 95.865% Precision, 98.99% Recall, and an F1 Score of 97.403%. This suggests the RF is almost as effective as DT, with a slightly lower precision and recall but maintaining high accuracy, making it a reliable choice for classification of the attacks.

Table III: Evaluation result of the Thermostat Dataset

Classifiers	Metrics			
	Accuracy	Precision	Recall	F1 score
KNN	99.415	95.95	99.602	97.742
MLP	99.029	93.148	99.691	96.309
RF	99.468	96.410	99.519	97.940
DT	99.329	95.865	98.990	97.403

Table IV: Evaluation result for Garage Door dataset

Classifiers	Metrics			
	Accuracy	Precision	Recall	F1 score
KNN	99.633	97.374	99.837	98.591
MLP	81.628	40.983	97.466	57.703
RF	99.602	97.507	99.452	98.47
DT	99.625	97.568	99.57	98.559

From TABLE IV: K-Nearest Neighbors (KNN) showcases strong performance having 99.633% Accuracy, 97.374% Precision, 99.837% Recall, and an F1 Score of 98.591%. Multi-Layer Perceptron (MLP) significantly lags in performance compared to the other models, having an Accuracy of 81.628%, Precision at 40.983%, high Recall at 97.466%, but a lower F1 Score of 57.703%. MLP's high recall indicates it's good at identifying positive cases, but its low precision and overall accuracy point to a high rate of false positives, reducing its effectiveness. Random Forest (RF) emerges with impressive metrics, with an Accuracy of 99.602%, Precision at 97.507%, Recall at 99.452%, and an F1 Score of 98.47%. RF's performance is exemplary, indicating it not only accurately classifies instances but also maintains an excellent balance between identifying all relevant instances and ensuring minimal false positives. Decision Tree also performs well, having 99.625% Accuracy, 97.568% Precision, 99.57% Recall, and an F1 Score of 98.559%. This suggests the Decision Tree is almost as effective as RF, with a slightly lower recall but maintaining high precision and accuracy making it more reliable for classification of the attack.

Best Classifier

Among these, Random Forest (RF) stands out as the best classifier for this dataset. It not only shows the highest accuracy but also the best balance between precision and recall as reflected in its F1 Score. This balance is crucial for effective classification, especially in datasets where both identifying all relevant instances (high recall) and ensuring high precision are important. RF's ensemble approach, leveraging multiple decision trees, likely contributes to its superior performance by reducing the risk of overfitting and increasing predictive accuracy.

5. Conclusion and Future Scope

Based on comprehensive evaluation results spanning multiple datasets and classifiers, it is clear that the Random Forest (RF) classifier consistently achieves superior performance across all datasets examined within the TON_IoT environment, which includes data from Fridge, Thermostat, Motion Light, and Garage Door sensors. Specifically, RF outperforms other classifiers such as K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP), and Decision

Trees (DT) in most cases, offering robustness and reliability in detecting anomalies or attacks in IoT environments. Therefore, for the TON_IoT framework, Random Forest emerges as the most suitable classifier for effectively detecting and responding to potential threats or malicious activities. Figure 2 represents the accuracy of various classifiers on the four distinct datasets in diagrammatic format.

In future the main focus will be utilizing ensemble and deep learning models for attack classification and detection using TON_IoT dataset.

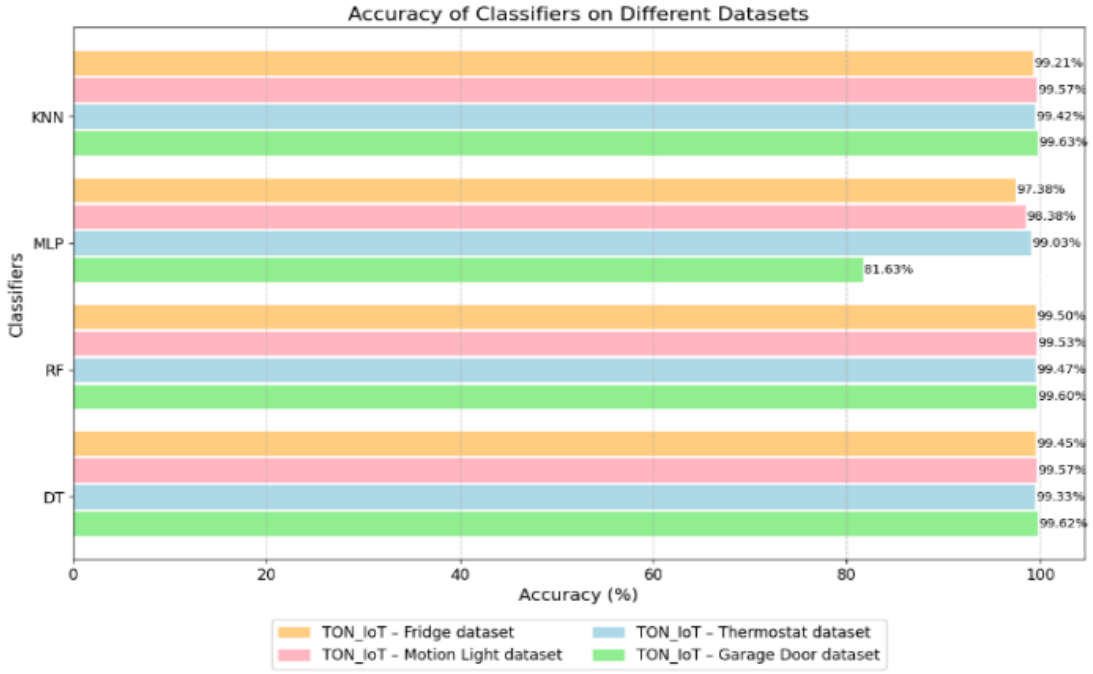

Figure 2: Accuracy of Classifiers on different TON_IoT datasets

References

1. A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood and A. Anwar, "TON_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems," in IEEE Access, vol. 8, pp. 165130-165150, 2020, doi:10.1109/ACCESS.2020.3022862
2. A. Mishra, J. Soni, P. Jaiswal, S. Chavan, K. Kolhe and J. Mante, "Critical Feature Selection (CFS) Techniques for Early DDoS Attack Detection in the Internet of Things Environment," 2023 7th International Conference On Computing, Communication, Control and Automation (ICCUBEA), Pune, India, 2023, pp. 1-6, doi: 10.1109/ICCUBEA58933.2023.10392251.
3. B. Omarov, O. Auelbekov, T. Koishiyeva, R. Sadybekov, Y. Uxikbayev and A. Bazarbayeva, "IoT Network Intrusion Detection Using Machine Learning Techniques," 2022 International Conference on Smart Information Systems and Technologies (SIST), Nur-Sultan, Kazakhstan, 2022, pp. 1-6, doi: 10.1109/SIST54437.2022.9945769
4. C. Liang, B. Shanmugam, S. Azam, M. Jonkman, F. D. Boer and G. Narayansamy, "Intrusion Detection System for Internet of Things based on a Machine Learning approach," 2019

International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Vellore, India, 2019, pp. 1-6, doi: 10.1109/ViTECoN.2019.8899448

5. C. Nixon, M. Sedky and M. Hassan, "Practical Application of Machine Learning based Online Intrusion Detection to Internet of Things Networks," 2019 IEEE Global Conference on Internet of Things (GCIoT), Dubai, United Arab Emirates, 2019, pp. 1-5, doi: 10.1109/GCIoT47977.2019.9058410
6. Ge Guo. "An Intrusion Detection System for the Internet of Things Using Machine Learning Models." 2022 3rd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE) (2022): 332-335. https://doi.org/10.1109/ICBAIE56435.2022.9985800
7. G. J. Pandeeswari and S. Jeyanthi, "Analysis of Intrusion Detection Using Machine Learning Techniques," 2022 Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing & Communication Engineering (ICATIECE), Bangalore, India, 2022, pp. 1-5, doi: 10.1109/ICATIECE56365.2022.10047057
8. Jyoti Mante Khurpade, D. Nageswara Rao and Parth Sanghavi. “A Survey on IOT and 5G Network.” 2018 International Conference on Smart City and Emerging Technology (ICSCET) (2018): 1-3.
9. Khraisat, A., Gondal, I., Vamplew, P. et al. Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity 2, 20 (2019). https://doi.org/10.1186/s42400-019-0038-7
10. Mante, J., Kolhe, K. (2023). Attack Detection in Internet of Things: A Systematic Literature Review. In: Fong, S., Dey, N., Joshi, A. (eds) ICT Analysis and Applications. Lecture Notes in Networks and Systems, vol 517. Springer, Singapore. https://doi.org/10.1007/978-981-19-5224-1_24. 7
11. M. Bagaa, T. Taleb, J. B. Bernabe and A. Skarmeta, "A Machine Learning Security Framework for Iot Systems," in IEEE Access, vol. 8, pp. 114066-114077, 2020, doi: 10.1109/ACCESS.2020.2996214
12. P. Mishra, V. Varadharajan, U. Tupakula and E. S. Pilli, "A Detailed Investigation and Analysis of Using Machine Learning Techniques for Intrusion Detection," in IEEE Communications Surveys & Tutorials, vol. 21, no. 1, pp. 686-728, Firstquarter 2019, doi: 10.1109/COMST.2018.2847722
13. S. Bhardwaj, P. Kumar and H. B. Maringanti, "Intrusion Detection in Internet of Things using Machine Learning Classifiers," 2021 International Conference on Technological Advancements and Innovations (ICTAI), Tashkent, Uzbekistan, 2021, pp. 571-575, doi: 10.1109/ICTAI53825.2021.9673449.
14. S. Chishakwe, N. Moyo, B. M. Ndlovu and S. Dube, "Intrusion Detection System for IoT environments using Machine Learning Techniques," 2022 1st Zimbabwe Conference of Information and Communication Technologies (ZCICT), Harare, Zimbabwe, 2022, pp. 1-7, doi: 10.1109/ZCICT55726.2022.10045992