Received: 2024-10-30
Accepted: 2024-12-17
Published Online: 2025-01-30
DOI: 10.70454/JRICST.2025.20108

Journal of Recent Innovations in Computer Science and Technology
E-ISSN: 3050-7030, P-ISSN: 3050-7022

Attack and Anomaly Detection in IoT Sensors Using Machine Learning Approaches

Ashish Seth

Professor, School of Computer and Information Engineering (SOCIE) Inha University in Tashkent, Uzbekistan

***Corresponding Author Email**: a.seth@inha.uz

Abstract

The extensive usage of IoT sensors significantly improved the collection and monitoring of data within various application domains, such as smart agriculture and industrial automation. On the other hand, the great dependence on IoT sensors makes systems vulnerable to hacks and anomalies. In this paper, we explore machine learning approaches that can be used to protect Internet of Things sensor networks against attacks and anomalies. Due to the limited resources available to IoT devices, traditional security measures fall short. There is, therefore, a need to develop more intelligent smart detection systems. This paper examines the capabilities of machine learning in identifying patterns of anomalies in IoT sensor data. Carried out on a dataset of simulated IoT environments, the research presents the stages of data pre-processing, exploratory data analysis, and feature engineering. In addition, three models; Logistic Regression, Decision Tree, and Random Forest were constructed and tested. The results show that it is possible to use machine learning algorithms for anomaly detection in IoT domains, thereby presenting the possibilities for improving IoT security and reliability. The findings of this study are important in that they highlight how advanced analytics can help organizations deal with IoT environments.

Keywords: Internet of Things, Cybersecurity, Machine Learning, Sensor Networks.

1. INTRODUCTION

The exponential growth of IoT devices in recent years has led to the emergence of big data in the form of sensor data, which creates problems with data processing and protection. IoT is creating opportunities to improve and innovate many organizations' operations and their customers' experiences, however, it also brings the likelihood of detection of alterations in this data is necessary so that the IoT systems can maintain security, credibility, and accuracy [1]. This research is centered on using machine learning approaches to detect any anomalous trends within the IoT sensor data set from a simulated IoT system environment. This dataset includes different IoT devices and services such as light controllers, thermostats, and smart doors within one day. Three models of machine learning namely Logistic Regression, Decision Tree, and Random Forest have been used in the research to assess their performance in terms of anomaly detection [10]. In those models, every model is then evaluated based on its capacity to distinguish normal from anomalous behavior. This assignment focuses on selecting the best way to detect anomalies in IoT systems by going through processes that involve data preprocessing, feature extraction, and model assessment. This study prepares a differential analysis of these models' performance to inform the utilitarian values of machine learning in improving IoT security and functionality [11] [12].

2. LITERATURE SURVEY

Anomaly Detection in IoT Systems

According to Abusitta, *et al*. 2023 [1], anomaly detection in the IoT context becomes challenging and important as a large number of We Smart devices are producing great volumes of data. Static methods that are normally used in anomaly detection fail to offer an optimal solution since they are based on statistical or rule-based models and are rigid when it comes to handling different data from the IoT environment. Later developments are based on the usage of machine learning approaches to improve the accuracy of the detection. Advanced machine learning approaches like clustering and classification techniques make it possible to learn from the data patterns and in the process identify signs of threats or malfunctioning. These models can consider patterns and intricate relationships within IoT, which makes them a more formal approach to addressing anomalies and enhancing the security of the system as a whole.

Machine Learning Models for Anomaly Detection

According to Elmrabit, *et al*. 2020 [2], there is no doubt that machine learning models have proved effective in most application domains as methods and techniques for detecting anomalies in IoT systems. Some of the methods that are most commonly used include Logistic Regression, Decision Trees, and Random Forests as these have a favorable capacity to handle vast datasets with many independent variables. Among the models used for binary classification, Logistic Regression is distinguished for its simplicity and interpretability; Decision Trees, in turn, offer easy access to data, in the form of points located at the decision nodes. Another type of decision tree is Random Forests explained as a more accurate method because instead of a single decision tree several of them are grouped to avoid overfitting. These models are useful in detecting anomalous data patterns by using past data hence capable of enabling real-time anomaly detection for IoT systems.

Feature Engineering for Anomaly Detection

According to Zhou, et al. 2021 [3], feature engineering is one of the most important ways that influence the effectiveness of machine learning schemes for anomaly detection. Originally acquired sensor data is high-dimensional and heavily noisy and it needs to be preprocessed and mapped with feature engineering. Pre-processing of data including normalization, categorical data encoding, and handling of missing values form the core of the data preparation process.

Selecting the features that will be most useful in the anomaly detection process is a significant step in alleviating the problem of high dimensionality and improving the algorithm's performance. Comparing the machine learning algorithms can analyze the raw data in the form of features and identify patterns and deviations which makes the Anomaly Detection System reliable in the dynamic and diverse IoT environment [13].

Device and system health, performance, and security may all be monitored with the help of anomaly detection in the IoT. Problems like equipment breakdowns, security breaches, and inefficiencies can be identified early on with its help, allowing for fast interventions and reducing the likelihood of major failures. In order to identify unexpected events that could offer valuable insights in domains like healthcare, anomaly detection in IoT networks is crucial. The Internet of Things (IoT) anomaly detection process incorporates multiple approaches, including machine learning, probability distributions, and time series analysis [14] [15].

Table 1: Internet of Things (IoT) sensor networks employ various machine learning techniques for attack and anomaly detection.

Method/Approach	Contribution	Limitations
Supervised Learning (SVM, Decision Trees)	Developed a supervised ML-based intrusion detection system for IoT.	Struggles with unknown attacks and requires labeled datasets.
Unsupervised Learning (K-Means Clustering)	Proposed an unsupervised approach for anomaly detection in IoT traffic.	High false-positive rate and limited scalability in large networks.
Deep Learning (Autoencoders, LSTM)	Used deep learning models to detect anomalies in sensor data.	Computationally expensive for resource-constrained IoT devices.
Ensemble Learning (Random Forest, XGBoost)	Combined multiple ML algorithms for better detection accuracy.	Higher computation cost and complexity.
Semi-Supervised Learning (SVM)	Focused on detecting novel attacks in IoT using semi-supervised models.	Performance drops in highly dynamic environments.
Anomaly-Based Detection (K-Nearest Neighbors)	Applied k-NN for real-time anomaly detection in IoT networks.	Performance declines with large datasets and complex attacks.

3.0 Materials and Methods

The material and Methods section explains the broad framework used in the attacking and anomaly detection process in the Internet of Things (IoT) network. This broad is divided into four main steps, including data acquisition, data preprocessing, feature engineering, and machine learning model construction. First, an IoT emulation scenario is used to gather data including various sensor and service interactions. Several processes are carried out usually data preprocessing, to deal with missing values, normalize features, and decode categorical features to numerical form to suit the analysis. Feature engineering is done to extract and select the attributes that will improve the performance of the model. Logistic Regression, Decision Trees, and Random Forests are then trained to identify between normal and abnormal behavior patterns. All the models are trained and tested on the processed data set to identify the success rate of the deviations. The results are then compared, to determine which of the models is most reliable for anomaly detection to facilitate a better understanding of the resilience of each in matters that concern IoT security.

Received: 2024-10-30
Accepted: 2024-12-17
Published Online: 2025-01-30
DOI: 10.70454/JRICST.2025.20108

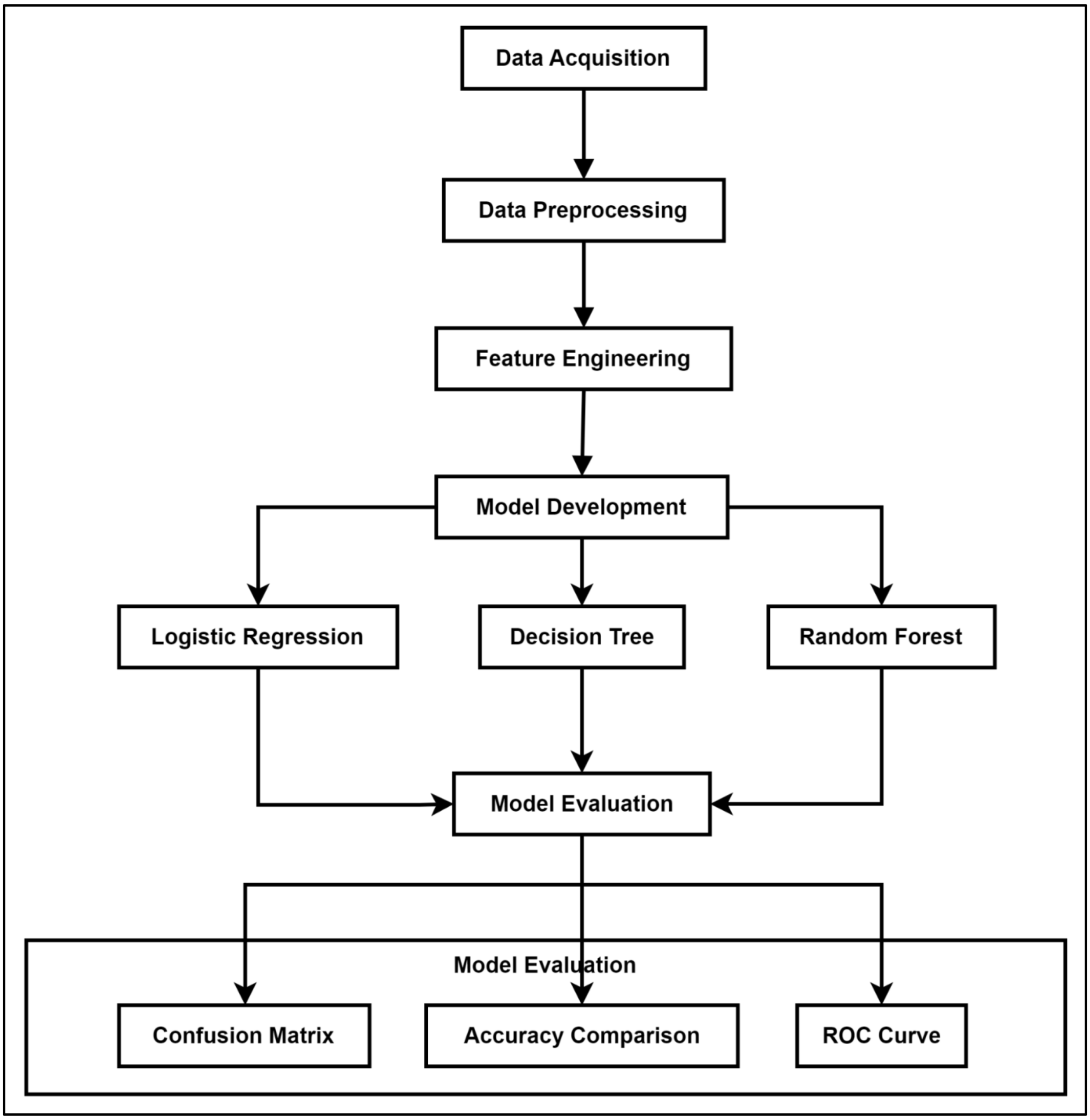


Figure 1: Framework of the Attack and Anomaly Detection in IoT Sensors Using Machine Learning Approaches

3.1. Dataset collection and description

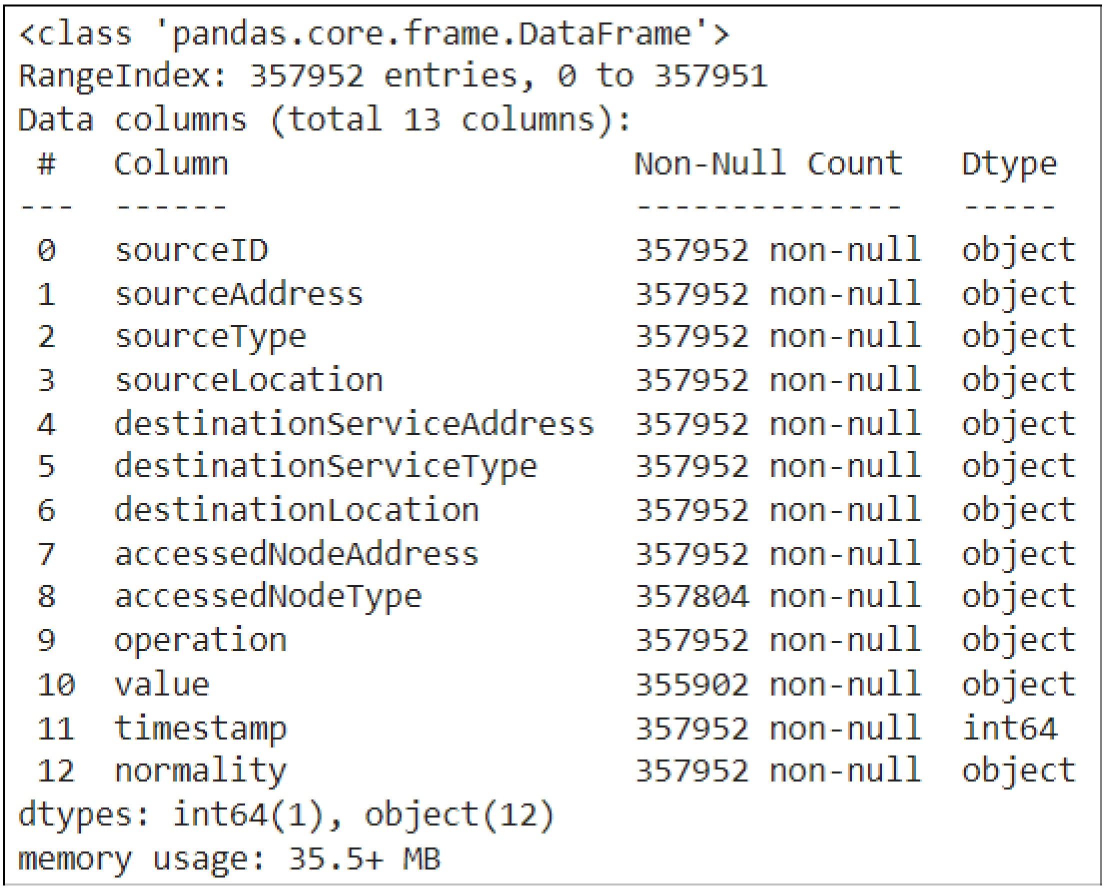
The data used in this study is obtained from Kaggle, a site that undertakes data science, and machine learning challenges. This kind of secondary data collection involves IoT traffic traces which are obtained from emulating IoT devices and services. The dataset is made of specific data obtained from several IoT locations consisting of light regulates, thermostats, movement detectors, and appliances among other objects.

The dataset consists of 357952 records with 13 attributes including source id, destination id, types of service, node address, operation id, and time stamp. This will afford a full understanding of the regular and non-standard behaviors in the IoT context. This rich dataset is used in the training and

Received: 2024-10-30
Accepted: 2024-12-17
Published Online: 2025-01-30
DOI: 10.70454/JRICST.2025.20108

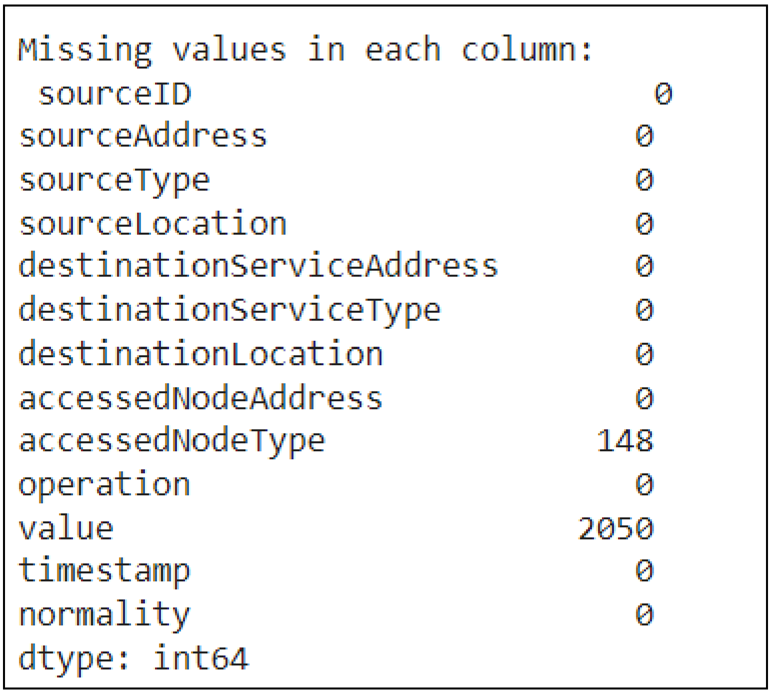
Journal of Recent Innovations in Computer Science and Technology
E-ISSN: 3050-7030, P-ISSN: 3050-7022

testing of the machine learning models with a view of attributing correct anomalies and attacks in IoT systems. With this data in mind, this study seeks to build strong models to detect drifts and improve IoT security.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 357952 entries, 0 to 357951
Data columns (total 13 columns):
 #   Column                     Non-Null Count   Dtype 
---  ------                     --------------   ----- 
 0   sourceID                   357952 non-null  object
 1   sourceAddress              357952 non-null  object
 2   sourceType                 357952 non-null  object
 3   sourceLocation             357952 non-null  object
 4   destinationServiceAddress  357952 non-null  object
 5   destinationServiceType     357952 non-null  object
 6   destinationLocation        357952 non-null  object
 7   accessedNodeAddress        357952 non-null  object
 8   accessedNodeType           357804 non-null  object
 9   operation                  357952 non-null  object
 10  value                      355902 non-null  object
 11  timestamp                  357952 non-null  int64 
 12  normality                  357952 non-null  object
dtypes: int64(1), object(12)
memory usage: 35.5+ MB
```

Figure 2: Dataset Info



```
Missing values in each column:
 sourceID                       0
sourceAddress                  0
sourceType                     0
sourceLocation                 0
destinationServiceAddress      0
destinationServiceType         0
destinationLocation            0
accessedNodeAddress            0
accessedNodeType             148
operation                      0
value                       2050
timestamp                      0
normality                      0
dtype: int64
```

Figure 3: Missing Values in the dataset

3.2. Data pre-processing

Preprocessing of data is a very important stage in the organization of the dataset for its use in machine learning. Starting with pre-processing, first, rows or columns with missing values are examined and in this process, missing values are handled through some techniques known as imputation. The 'accessedNodeType' column is imputed with the first mode and the 'value' column is imputed with the median to keep the data consistent (Alam and Yao, 2019) [4]. Second, categorical variables are encoded using a method known as Label Encoding so that they are in numerical form that can be used in the model. Some of the fields include the source ID, source address, source type, opening date, as well as others. In use Label encoding assigns an integer to each category hence allowing the machine learning algorithms to interpret these features well. After encoding the data passes through the feature scaling step of data preparation. On this note, standardization is conducted by employing the `StandardScaler` to facilitate equal feature value scaling since the feature space's variance could stabilize the learning process (Ozsahin, *et al*. 2022) [5].

Also, feature selection is used again to reject the features that are less important or even not relevant in the analysis. Irrelevant features with columns that offer low variation impacts on the predictive model are usually omitted in favor of features that improve the model's efficiency (Zhang, *et al*. 2020) [6]. Such a method of data preprocessing minimizes cleaning issues and preempts them for the next steps of constructing and comparing ML algorithms.

3.3. Theoretical considerations

3.3.1. Logistic Regression (LR)

Logistic Regression is a supervised learning algorithm that is preferably used for binary classification problems. Is a technique used to predict the probability of occurrence of a given binary dependent variable using one or more independent variables. Logistic regression is based on the logistic function also known as logistic transform that nonlinearly maps the linear equation into the range from 0 to 1 as a probability score (Noureen, *et al*. 2019) [7]. This is realized through the use of the logistic function to the linear weighted sum of the inputs. The model finds out the value of the features that have a direct relationship with the coefficients by using the likelihood function whose goal is to test how accurate predicted probabilities are to binary outcomes. Logistic Regression is highly used because of its simplicity and interpretability furthermore; it provides high efficiency for linearly separable data.

The mathematical equation of the LR model is:

$$P(Y = 1 \mid X) = 1/(1 + e - (\beta 0 + \beta 1X))$$

3.3.2. Decision Tree (DT)

Decision Trees are one of the non-parametric supervised learning techniques that is used for classification and regression. They use decision-tree-like structures where nodes inside the tree correlate with decisions and features/attributes, edges of the tree stand for decision rules, and end nodes stand for outcomes. The tree is constructed by iteratively splitting the data based on features that have the highest information gain or the

Received: 2024-10-30
Accepted: 2024-12-17
Published Online: 2025-01-30
DOI: 10.70454/JRICST.2025.20108

Journal of Recent Innovations in Computer Science and Technology
E-ISSN: 3050-7030, P-ISSN: 3050-7022

least entropy or Gini index (Aguilar, *et al*. 2022) [8]. Another advantage associated with Decision Trees is their interpretability since it resembles the decision-making process of human beings. But it undergoes over fitting, more especially with large trees which can be corrected by pruning or placing a condition.

The mathematical equation of the DT model is

$$\text{Information Gain}(\mathrm{P}_d,x) = \mathrm{I}(\mathrm{P}_d) - \frac{\mathrm{LC}_n}{\mathrm{P}_n}\mathrm{I}(\mathrm{LC}_d) - \frac{\mathrm{RC}_n}{\mathrm{P}_n}\mathrm{I}(\mathrm{RC}_d)$$

3.3.3. Random Forest (RF)

To improve the accuracy and reliability of classifications, Random Forest, another tree-based learning technique, uses several Decision Trees. (Biswas and Samanta, 2021) [9] It learns a huge number of Decision Trees during training and then returns the class with the most common categorization or mean regression. To increase tree variety and decrease over-fitting, Random Forest uses decision trees built with bootstrapped training data samples and a random selection of features at each split.

3.4. Evaluation criteria

3.4.1. Confusion matrix

The Confusion matrix is another performance measurement for the models of classification where the model results are defined by the counts of true positive, false positive, true negative, and false negative. This is made possible since it shows how well the classifier is performing to determine the sources of errors and improvement.

3.4.2. Accuracy

Accuracy is the ratio of the number of instances that were predicted by the model as belonging to a particular category divided by the total number of instances. It is determined by the total (true positive + true negative)/total number of examples. Accuracy gives one complete picture of the performance of a model despite being highly misleading in the case of imbalanced datasets.

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative}}$$

3.4.3. Precision

Precision or positive predictive value is one of the statistical measures of the validation study; it shows the accuracy of the positive predictions. The Precision can be defined as the ratio of true positives and false positives to the power of true positives. The chance that sources are not accurate is an issue in scenarios whereby the implications of false positives are expensive.

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

3.4.4. Recall

Recall is productivity of the model and quantity of the relevant instances which was used, Sensitivity checks whether all the possibilities are identified. It refers to the proportion of actual positives to the number of actual positives that got detected in addition to the number of actual negatives that got misclassified. This kind of approach is important any time the costs associated with a false negative are high.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

3.4.5. F1 score

F1 is the measure of accuracy to have both the aspects of precision and recall as it implements the F1 score which is the average of both precision and recall. The F1 score comes in handy when you are interested in both precision and recall especially if dealing with many few classes.

$$\text{F1 Score} = \frac{2^*\text{True Positive}}{2^*\text{True Positive} + \text{False Positive} + \text{False Negative}}$$

4.0 Implementation and Result:

4.1. Experimental setup

The experiments are done in Google COLAB which is a cloud-based notebook with GPU support that helps in faster training of models and their evaluation. Internet connection in the computer is a requirement due to the access and downloading of the dataset from Kaggle and access to the libraries and tools in Colab. The IoT traffic traces form the dataset that is imported into the environment to undergo preprocessing and be used to train the model. Python libraries like `pandas`, `scikit-learn`, and TensorFlow are used to build, train and test Logistic Regression, Decision Trees, and Random Forests. They are further examined using different performance measures and data visualization techniques.

4.2. Result analysis

In this section, the evaluation of the experimental outcomes is discussed while implementing Logistic Regression, Decision Tree, and Random Forest on the IoT attack and anomaly detection problem. To assess each model, the standard measures of accuracy, precision, recall, and F1 score are used. Moreover, apply the use of confusion matrices and receiver operating characteristic (ROC) curves to evaluate the model's capabilities in classifying different activities as normal or anomalous. The analysis also involves a comparison of these models to know how well they perform and if it is suitable for detecting anomalies in IoT. Some of the findings obtained from these evaluations are highlighted with a focus on the advantages and disadvantages of each approach alongside their practical applicability in the theme of IoT security.

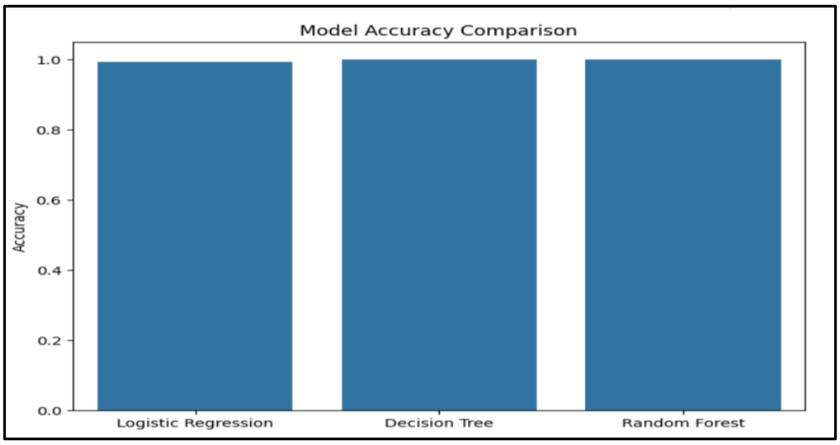


Figure 4: Accuracies of the Models

This bar chart compares the accuracy of the Logistic Regression model with the Decision Tree and Random Forest model. Hence, it can be stated that all three models demonstrate a high degree of accuracy, the values

Received: 2024-10-30
Accepted: 2024-12-17
Published Online: 2025-01-30
DOI: 10.70454/JRICST.2025.20108

of which are very close to 99%). From the figures depicted above, Decision Tree and Random Forest models seem to give slightly better results compared to Logistic Regression though the difference is very small. This implies that all three models have high accuracy in classifying the IoT sensor data with a little bit of even a margin between tree-based models such as Decision Tree and the Random Forest.

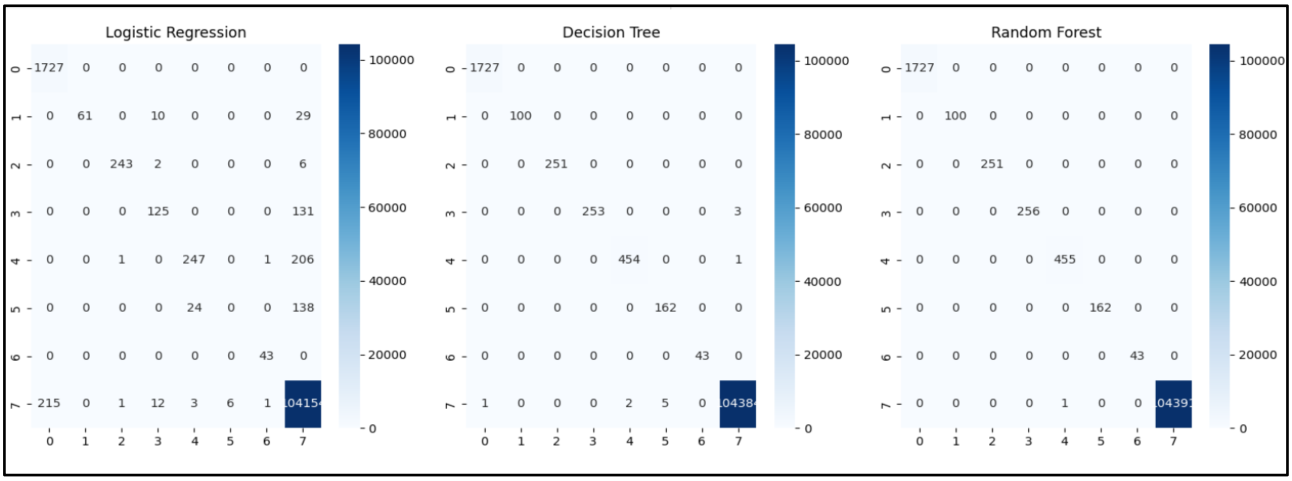
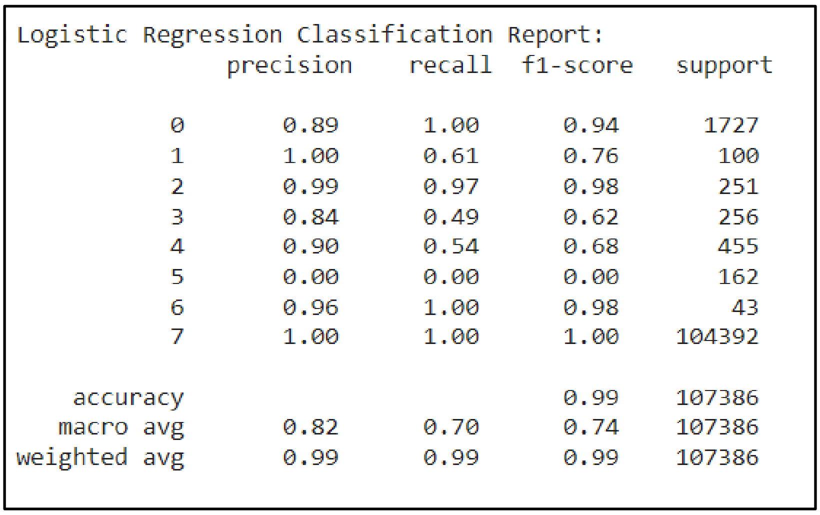


Figure 5: Confusion Matrices of the Models

These are heat maps of confusion matrices for Logistic Regression, Decision Trees, and Random Forest techniques. On the diagonal, the number of correct predictions is presented, and off the diagonal means that misclassification occurred. There are some misclassified points according to the Logistic Regression matrix this is most probably because most of the points belong to the majority class (likely class 7). The Decision Tree as well as Random Forest matrices give almost 99.9% accuracy across all classes and most of the observations lie in the diagonal in both. This means that all tree-based models yield better prediction accuracy as compared to Logistic Regression in the classification of divergence, non-divergence, and all data points.



```
Logistic Regression Classification Report:
              precision    recall  f1-score   support

           0       0.89      1.00      0.94      1727
           1       1.00      0.61      0.76       100
           2       0.99      0.97      0.98       251
           3       0.84      0.49      0.62       256
           4       0.90      0.54      0.68       455
           5       0.00      0.00      0.00       162
           6       0.96      1.00      0.98        43
           7       1.00      1.00      1.00    104392

    accuracy                           0.99    107386
   macro avg       0.82      0.70      0.74    107386
weighted avg       0.99      0.99      0.99    107386
```

Figure 6: Logistic Regression Classification Report

The following is the classification report for the Logistic Regression model with the detailed performance measurement of each class (0-7). Given the fact that the model is very clean for the majority class (most probably class 7), the precision and recall are 99 percent. Nevertheless, it is not satisfactory when dealing with some minority classes most especially class 5 where the model did not correctly predict any instance (0). The overall accuracy of the model is good, it is equal to 0.99, although the model performs great in the majority class, it struggles in some of the minority classes.

The Decision Tree classification report reveals reasonable accuracy within each class ranging from 0 to 7. As to all the obtained precisions, recalls and F1-scores, all of them are equal to .99 or less than one cent from it (0.99). This shows that the Decision Tree model is exceptionally good in classifying normal and anomalous activities in the IoT sensor data whether in class 1 or class 2. The first analysis shows that the overall accuracy is .99, which means the test set of the movie classification performance is 99.9 percent perfect.

Like in Decision Tree, the Random Forest classification report provides an exemplary performance with one being closer to 1 for all classes (0-7). Each of the related accuracy, recollective, and F1-score coefficients is identified as 1. Thus, the value of loss in the scale of recalls that the model has correctly detected each example of the test set. This means that the Random Forest model is very efficient in identifying normal as well as anomalous behaviors in the IoT sensor data with no instances of misclassification in any of the classes.

Table 2: Performance Metrics of the Models

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Logistic Regression	0.99	0.996576	0.996479	0.996378	0.991623
Decision Tree	0.99	0.999885	0.999885	0.999885	0.999998
Random Forest	0.99	0.999999	0.999999	0.999999	0.999999

Table 3: Cross-Validation(CV) Results

Model	CV Accuracy
Logistic Regression	0.871473
Decision Tree	0.766149
Random Forest	0.989819

Table 2: Performance Metrics of the Models and Table 3: Cross-Validation(CV) Results,the evaluation criteria of the models reveal very high performance, with Logistic Regression, Decision Tree, and Random Forest presenting near-perfect scores of accuracies, precision, recall, F1-score, and AUC-ROC. In detail, the Random Forest model gets the highest result among all the indices which means that this model has high accuracy when trained on the data. The performance of the model is also depicted by the Decision Tree as good as the Random Forest with almost perfect accuracy, precision, and recall. Similar to the previous models, Logistic Regression has slightly lower precision and recall values yet indicates rather high accuracy as well as AUC-ROC.

The cross-validation results have a different story to tell, as shown in the above tables the Random Forest model despite having an excellent training accuracy of more than 99% has a cross-validation accuracy of around 98%. Of the test results, it has 98% accuracy, which is slightly worse than its training accuracy. This means that although the training data that it has learned from it does a good job of mapping, it’s not perfect when it comes to unseen data. Logistic Regression and Decision Tree are lower in terms of cross-validation accuracy with an accuracy of 87.15 %, 76.61%, and 70% of the total data respectively which indicates signs of overfitting of these models or these models may not perform as well on the new unseen data set as the Random Forest.

This leads to a further study into the model overfitting in training as the performance achieved in the training phase is significantly higher than the performance in the cross-validation phase. Hence, one of the ways that could have prevented this is by upgrading the size and nature of features, as well as optimizing the use of hyperparameters. Besides, it is more accurate to perform stratified k-fold cross-validation to get more accurate results while estimating the model performance. Hence, the results observed imply that the models are effective tools, yet their fine-tuning and calibration would make them less sensitive to deviations in real-life uses.

5.0 Conclusion

Hence, it can be concluded that this research offers a critical evaluation of attack and anomaly detection within IoT systems based on multiple machine-learning models. The study then uses Logistic Regression, Decision Tree, and Random Forest models to show that all models are equally effective in performance classification and are on the verge of ~99.9% accuracy in identifying IoT-Sensor. H2 is that Random Forest and Decision Tree models are slightly more accurate than Logistic Regression based on the accuracy and near-perfect classification depicted above. Analyzing confusion matrices and classification reports shows that even though the Logistic Regression is effective, it generally underperforms with minority classes an aspect that could be worked on. In contrast, Decision Tree, and Random Forests Models are quite stable across all classes and thus very suitable for the dataset in question. In sum, the studies highlighted in the work confirm the effectiveness of tree-based methodologies in attacking and anomaly detection and thus support the workability of the methods in strengthening IoT security.

Author Contributions

All authors endorsed the article that was submitted and made contributions to it.

Funding

External funding was not provided for this study.

Conflict of Interest

The authors declare no conflict of interest.

Data availability

https://www.kaggle.com/datasets/francoisxa/ds2ostraffictraces

Reference

[1] Abusitta, A., de Carvalho, G.H., Wahab, O.A., Halabi, T., Fung, B.C. and Al Mamoori, S., 2023. Deep learning-enabled anomaly detection for IoT systems. Internet of Things, 21, p.100656.

[2] Elmrabit, N., Zhou, F., Li, F. and Zhou, H., 2020, June. Evaluation of machine learning algorithms for anomaly detection. In 2020 international conference on cyber security and protection of digital services (cyber security) (pp. 1-8). IEEE.

[3] Zhou, Y., Song, X., Zhang, Y., Liu, F., Zhu, C. and Liu, L., 2021. Feature encoding with autoencoders for weakly supervised anomaly detection. IEEE Transactions on Neural Networks and Learning Systems, 33(6), pp.2454-2465.

[4] Alam, S. and Yao, N., 2019. The impact of preprocessing steps on the accuracy of machine learning algorithms in sentiment analysis. Computational and Mathematical Organization Theory, 25, pp.319-335.

[5] Ozsahin, D.U., Mustapha, M.T., Mubarak, A.S., Ameen, Z.S. and Uzun, B., 2022, August. Impact of feature scaling on machine learning models for the diagnosis of diabetes. In 2022 International Conference on Artificial Intelligence in Everything (AIE) (pp. 87-94). IEEE.

[6] Zhang, Z., Wen, J., Zhang, J., Cai, X. and Xie, L., 2020. A many objective-based feature selection model for anomaly detection in cloud environment. Ieee Access, 8, pp.60218-60231.

[7] Noureen, S.S., Bayne, S.B., Shaffer, E., Porschet, D. and Berman, M., 2019, February. Anomaly detection in cyber-physical system using logistic regression analysis. In 2019 IEEE texas power and energy conference (TPEC) (pp. 1-6). IEEE.

[8] Aguilar, D.L., Medina-Pérez, M.A., Loyola-Gonzalez, O., Choo, K.K.R. and Bucheli-Susarrey, E., 2022. Towards an interpretable autoencoder: A decision-tree-based autoencoder and its application in anomaly detection. IEEE transactions on dependable and secure computing, 20(2), pp.1048-1059.

[9] Biswas, P. and Samanta, T., 2021. Anomaly detection using ensemble random forest in wireless sensor network. International Journal of Information Technology, 13(5), pp.2043-2052.

[10] Sahu, N.K. and Mukherjee, I., 2020, June. Machine learning based anomaly detection for IoT network:(Anomaly detection in IoT network). In 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184) (pp. 787-794). IEEE.

[11] Kovács, G., Sebestyen, G. and Hangan, A., 2019. Evaluation metrics for anomaly detection algorithms in time-series. Acta Universitatis Sapientiae, Informatica, 11(2), pp.113-130.

[12] A. E. F. Alfalahi, S. R. A. Alhebsi and T. Murugan, "Recent Research Solutions on Deep Learning-based Anomaly Detection in Internet of Things," *2024 15th Annual Undergraduate Research Conference on Applied Computing (URC)*, Dubai, United Arab Emirates, 2024, pp. 1-6, doi: 10.1109/URC62276.2024.10604622.

[13] Y. Y. Ghadi *et al.*, "Machine Learning Solutions for the Security of Wireless Sensor Networks: A Review," in *IEEE Access*, vol. 12, pp. 12699-12719, 2024, doi: 10.1109/ACCESS.2024.3355312.

[14] U. Zukaib, X. Cui, C. Zheng, M. Hassan and Z. Shen, "Meta-IDS: Meta-Learning-Based Smart Intrusion Detection System for Internet of Medical Things (IoMT) Network," in *IEEE Internet of Things Journal*, vol. 11, no. 13, pp. 23080-23095, 1 July1, 2024, doi: 10.1109/JIOT.2024.3387294.

[15] A. H. Farea, O. H. Alhazmi, R. Samet and M. S. Guzel, "AI-Powered Integrated with Encoding Mechanism Enhancing Privacy, Security, and Performance for IoT Ecosystem," in *IEEE Access*, vol. 12, pp. 121368-121386, 2024, doi: 10.1109/ACCESS.2024.3449630