Received: 15 May 2023 | Revised: 14 October 2023 | Accepted: 20 December 2023

DOI: 10.1002/spy2.366

RESEARCH ARTICLE

WILEY

Anomaly detection in IoT environment using machine learning

Harini Bilakanti | **Sreevani Pasam** | **Varshini Palakollu** | **Sairam Utukuru**

Chaitanya Bharathi Institute of Technology, Osmania University, Hyderabad, India

Correspondence
Sairam Utukuru, Chaitanya Bharathi Institute of Technology, Osmania University, Hyderabad, Telangana 500075, India.
Email: usairam_it@cbit.ac.in

Abstract

This research paper delves into the security concerns within Internet of Things (IoT) networks, emphasizing the need to safeguard the extensive data generated by interconnected physical devices. The presence of anomalies and faults in the sensors and devices deployed within IoT networks can significantly impact the functionality and outcomes of IoT systems. The primary focus of this study is the identification of anomalies in IoT devices arising sensor tampering, with an emphasis on the application of machine learning techniques. While supervised methods like one-class SVM, Gaussian Naive Bayes, and XG Boost have proven effective in anomaly detection, there has been a noticeable scarcity of research employing unsupervised methods. This scarcity is mainly attributed to the absence of well-defined ground truths for model training. This research takes an innovative approach by investigating the utility of unsupervised algorithms, including Isolation Forest and Local Outlier Factor, alongside supervised techniques to enhance the precision of anomaly detection.

KEYWORDS

anomaly detection, attacks, internet of things, machine learning

1 | INTRODUCTION

As Internet of Things (IoT) networks connect the physical environment of devices to the virtual environment through the internet, there are high chances for attacks, threats, and faults to happen. When the devices in the network are prone to attacks or malicious activities anomalous data is produced.

The cause for this anomalous data comes from different attacks on the devices in IoT networks. Sources of attacks and threats can be any of the following: Distributed denial-of-service (DDoS), Fraud, Data leakage, Intrusion, tampering, etc. An anomaly, defined as any change in usual behavior, can provide an early warning of a problem.[1] For example, anomalies in an Internet of Things (IoT) sensor's time series data can indicate a failure in a manufacturing unit. There are 2 reasons for the anomalous data. (1) Internal Failures that is, due to the software or hardware failures (2) External activities that is, the malicious events at the place of node deployment.[2] It's very important to handle the second cause as it is mostly done by the attackers to damage the organization. As the quality of data affects various decision-making processes, using such anomalous data creates a huge loss.[3] Hence there is a necessity to detect such anomalous data. So, there is a need to develop an effective anomaly detection system in the context of sensor tampering.

Training unsupervised models in IoT (Internet of Things) environments presents several challenges due to the unique characteristics of IoT data. Some of the Challenges are

1. IoT data can be highly variable, and anomalies can manifest in various forms, making it difficult to define a clear pattern. A general solution is Use anomaly detection models that can adapt to changing patterns and handle varying data distributions, such as Isolation Forests.
2. IoT data often has an imbalanced distribution, where normal data instances significantly outnumber anomalies, which can bias model training. One solution is implementing techniques like oversampling or anomaly synthesis to balance the dataset for better model performance.
3. Unlike supervised learning, labelling anomalies in IoT data can be expensive and impractical, making unsupervised learning necessary. IoT data may contain noise and missing values, which can negatively impact model performance. By Employing clustering techniques or generative models for anomaly detection that don't require labeled data.

According to the European Union Agency for Cybersecurity (ENISA), Sensor tampering in IoT is considered the second most challenging threat.[4] Sensor tampering is nothing but physically modifying the devices, and connections between the nodes in the network or manipulating the data, etc. As effects of sensor tampering are huge on IoT devices and sometimes it may lead to the fall down of the entire IoT network.[5] So, in this work, we developed an effective detection system that detects anomalies, especially for sensor tampering cases.

There are many methods available to detect anomalies. Ex Geometrical, Statistical, and Machine Learning methods, etc.[3] As machine learning methods showed better results, they gained a lot of popularity in the research community.[3] The main contributions of this work are as follows:

1. Enhancing IoT network security by detecting anomalies.
2. Enhance the efficiency of IoT devices by identifying issues that can impact performance.
3. Improving the sensor tampering detection on data with no labels by using unsupervised approaches.

The rest of the paper is explained as in Section 2 related work is presented, Section 3 discusses the methodology and evaluation of two benchmark datasets, Section 4 covers the results and analysis. The conclusion is presented in Section 5.

2 | RELATED WORK

In this section literature review of anomaly detection related to sensor tampering using machine learning is presented. In this paper,[6] the entire work is carried out in an office environment. They installed 4 sensors in 3 office rooms and monitored the data that came from these 4 sensors for 28 days. On some particular days, they tampered with the sensors.

That means they either removed sensors or manipulated them in such a way that they don't work in the usual way. In their project, on days 25 and 26 they removed the wall plug sensor and this indicates the presence of anomalies. To detect the sensor tampering anomalies, they developed an (AD-ML) Anomaly Detection Machine Learning system. They used 2 approaches. First, The Isolation Forest is an unsupervised algorithm to detect anomalies in traffic patterns. They used the contamination factor value of 0.1 and silhouette coefficient value is 0.84% for better results. Second, the Decision tree which is supervised learning achieved 91.62% accuracy and a low false negative rate. They concluded that a supervised approach gave the best results.

The main goal of this research[7] is to develop an anomaly-based Intrusion Detection System. Since it differentiates the anomalies from normal flow traffic by using the underlying architecture behavior. The dataset used in this paper was uploaded from Xavier and consists of a total of 357 952 samples with 13 features out of which 2747 such tuples were there which either had null as value or missing value. They loaded each pcap file which consists of 7 features and then they applied corresponding filtering rules and separated abnormal data and normal data into different CSV files. They repeated this process for the remaining 42 pap files. They concatenated all attack files into a single file and filled all the missing values with the normal labels. They have created an extra column CLASS with 0 as a label indicating attack and 1 indicating normal.

The training and testing ratio used is 3:1. The performance of the used models is evaluated based on the values of the confusion matrix and other measures such as accuracy, recall, F1 score, max iterations, and run-time were also considered. They received the second-highest accuracy of 99% using KNN while the runtime was an average of 2 min. XG Boost showed them good results with 97% accuracy with a run time of just 10.8 s.

In this paper,[8] they have detected different kinds of anomalies based on a data set. They used two Logistic Regression and Artificial neural networks for prediction and compared the performance of the two algorithms. They experimented

with two cases. For the first case, they considered the complete dataset, for the second they applied the algorithms by removing the feature 'Value'. The dataset used was uploaded by Xavier in Kaggle and consists of 357 952 samples with 13 features. They obtained 99.3%, and 99.37% accuracy for logistic regression and ANN respectively for case 1, and case 2 99.96% for logistic regression, and 99.6 for ANN. Finally, they concluded ANN had given better results than logistic regression for predicting anomalies.

The main goal of this paper[9] is to monitor the working status and detect anomalies of sensors in IoT networks. It's not possible for traditional methods to satisfy practical requirements when detecting anomalies, they can only estimate the sensor state or environment state. In this paper, anomaly detection and identification are carried out in 2 steps. The first step consists of a customized composite distance metric and clustering of sensors, and the second step is to apply fuzzy logic based on spatio-temporal correlation. They used a real-time data set from Caltrans PeMs over 39 000 sensor stations, which consists of 228 sensors and data collected over 44 days. Their method has resulted in a high accuracy rate and identifies the source of an anomaly with the use of spatiotemporal correlations. For the future scope, they proposed to explore the association of multi-models as the association of faulty nodes are independent. This research[10] mainly focused on threats in IoT Cybersecurity in a smart city. The approach they used was intelligent enough to detect anomalies with a low false positive rate. Here a distributed fog layer monitors all IOT traffic and alerts the administrator in a smart city. They used a Random Forest machine learning algorithm and this algorithm successfully detected compromised nodes at distributed fog nodes. This research was able to build an AD-ML system with classification accuracy of 99.34% and with a very low false positive rate.

The density based & clustering Machine Learning algorithms showed some prominent results in the anomaly detection process. In Reference 11, a combination of clustering and LSTM was used and performed well in terms of accuracy and scalability but they focused only on the centralized network system. In Reference 12, an unsupervised approach with PCA (Principal Component Analysis) showed accurate results, more reliability, and less communication cost but the detection system requires 3 days to detect anomalies, which doesn't support real-time anomaly detection.

In the Reference 13, Neural Network based convolutional neural network (CNN), LSTM-based Auto encoder gave a better performance on multivariate time-series data. Their work mainly focused on rare event detection and only capable of running only on a constrained network. A clustering algorithm such as DBSCAN was used in the Reference 15 and showed good performance in detection and the modeling errors in soft sensors are used as the intelligence to outlier detection process but the parameter problem of the DBSCAN should be handled carefully. The use of SNMP (SNMPv1, SNMPv2, and SNMPv3) protocol in Reference 18 provided reliability through early anomaly detection but for a different SNMP protocol, the system would fail to perform the monitoring task.

Apart from the regular machine learning models, in other research, they used different approaches such as statistical methods, edge computing, fog computing, etc. In the statistical approach,[14] methods such as Autoregressive integrated moving average, Median absolute deviation scale estimate, and Rosner statistics were used which detected and replaced anomalies but lacked decision-making support technologies. In another statistical approach[19] Hidden Markov Models (HMM) were used which detected anomalies accurately but the application they developed supports only a single user.

Edge computing technology along with Machine Learning was also used for detecting anomalies. This Reference 16 resulted in scalable, more accurate anomaly detection. And this approach cost efficient but has very average computing latency.[17] The combination of Fog computing, LPWAN, and 5G technologies with a Robust Covariance (RC) algorithm was highly scalable and had low latency in detecting anomalies in IoT. In our previous work addressing impact of loss of data in IoT application is presented in Reference 20. The recent research works on IoT anomaly detections in smart home and routing are presented in References 21–24.

Device Security, Authentication and Authorization and Data Privacy are some of the key security concerns. IoT devices are often constrained in terms of processing power and memory. This makes them vulnerable to various attacks, including malware and physical tampering. Ensuring the security of these devices is crucial. It's vital for organizations and individuals to be proactive in managing IoT security to protect against evolving threats.

In this work, we exclusively focused on detecting anomalies in IoT devices in the context of sensor tempering. As Machine Learning has gained a lot of recognition in the research and related areas, we chose the machine learning method of detecting sensor tampering and anomalies. Supervised methods are applied to address this problem and the results are good in terms of accuracy. But very few worked using unsupervised methods for detecting the anomalies so, we applied unsupervised algorithms as well as supervised ones and tried to achieve better results in terms of accuracy.

3 | METHODOLOGY

By detecting whether the sensor has been tampered with or not, we can find out the compromised device/sensor. Thus, we can avoid the spread of attacks in IoT networks. *The steps involved in our work are*:

i Data Collection: The process kicks off with the collection of data from sensors, which are connected to a gateway device via the Z-Wave communication protocol.
ii Data Pre-processing: The collected data undergoes essential pre-processing steps. This includes handling missing values, performing normalization, encoding categorical variables, and dividing the dataset into test and training subsets. These measures are taken to prepare the data for the application of machine learning models.
iii Machine Learning Application: Once the data is prepared, we proceed to apply various machine learning algorithms. The goal is to identify anomalous data effectively.
iv Model Performance Assessment: The performance of each machine learning model is rigorously assessed. We utilize key evaluation metrics such as Precision and F1 score to gauge their effectiveness in detecting anomalies. Considering the F1 score is crucial in anomaly detection because it serves as a balance metric between precision and recall. Precision reflects the ability of the model to avoid false alarms, while recall represents its capacity to detect true anomalies. An algorithm with a high F1 score signifies a harmonious equilibrium between these two aspects. In essence, it means that the algorithm can effectively identify anomalies (high recall) while maintaining a low rate of false positives (high precision). This balance is paramount for the reliability and efficiency of anomaly detection systems in real-world applications.
v Model Deployment: After thorough evaluation, the model that excels in detecting anomalies within IoT environments is selected for deployment.
vi User Notification Feature: As an additional enhancement, we have the capability to notify users when anomalies are detected, adding an extra layer of functionality to the system.

All these above-mentioned steps of the project are visually represented in the above Figure 1. In our work, we have used 2 datasets (1) the Office environment's sensor data and (2) the SWaT dataset.

3.1 | Dataset description

The first dataset contains sensor data collected from 2 rooms in an office IoT environment and 454 total numbers of records. They installed 3 different sensors in each lab room. The sensors they used are 4-in-1 motion sensors (2), door sensors (2), and wall plugs (2). This dataset contains packets captured from previously mentioned sensors for a period of 18 days. On two days (day 25 and day 26) the experiment was continued without the wall plug as shown in Figures 2–4

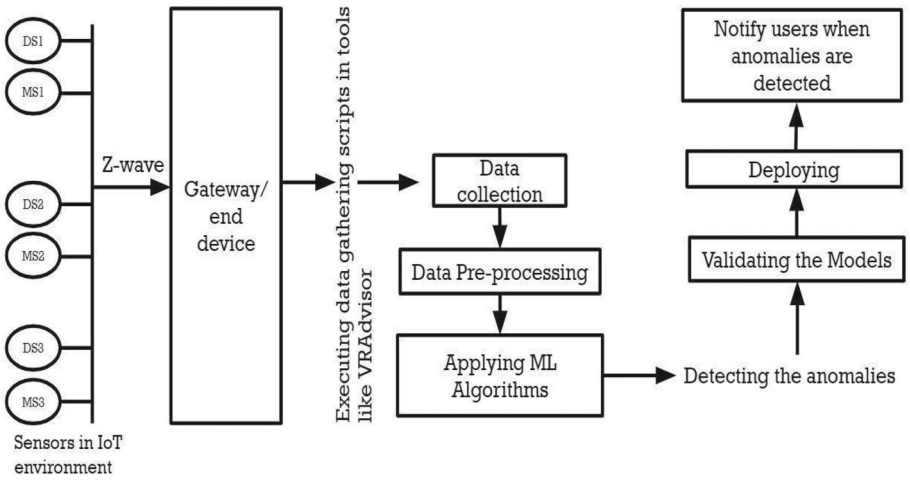

FIGURE 1 System architecture.

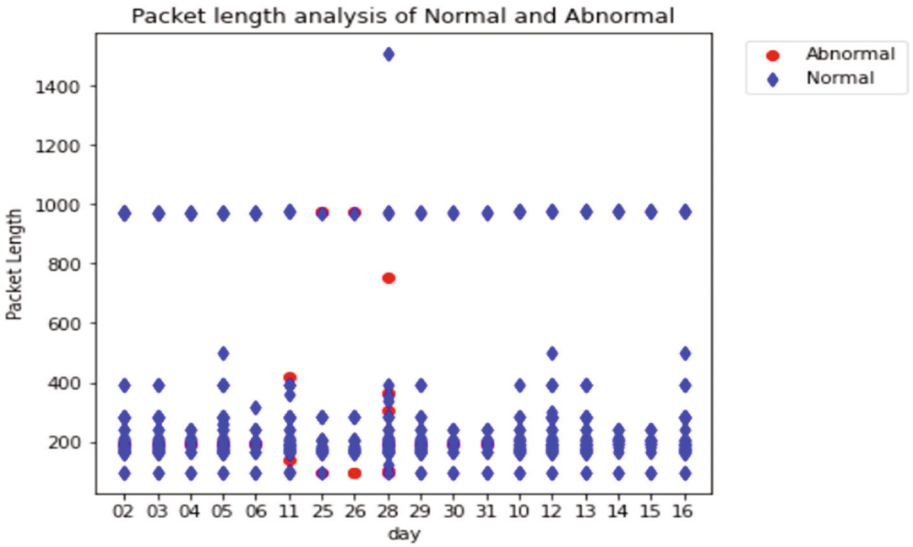

FIGURE 2 Packet length analysis of normal and abnormal using local outlier factor.

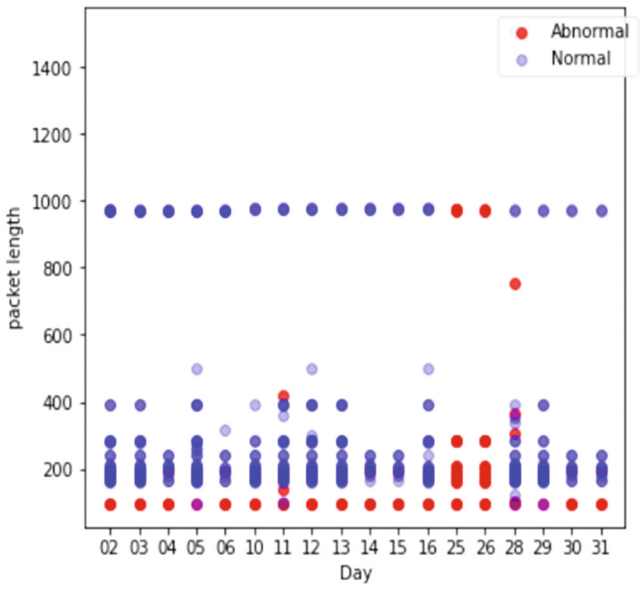

FIGURE 3 Packet length analysis of normal and abnormal using isolation factor.

which indicates anomalies. The abnormal day dataset also included with weekend data and sensor-triggered data is not available.

The second Dataset is Swat, It is a water plant that produces 5 gallons/min of double-filtered water. It contains Six – stage filtration processes (P1, P2, P3, P4, P5, P6). A total of 51 sensors and actuators are used in the six-stage process. Swat is run for a total of 11 days in the first 7 days system is operated normally and for the next four days, certain attacks are launched. They have performed and launched physical attacks and network traffic attacks respectively.

Physical attacks are performed by simply tampering with the devices, network traffic is launched by hijacking the network and by altering the packets during transmission. A total of 36 attacks were launched during 4 days. This dataset contains both categorical and numerical data with the data shape of 4 491 953 samples and 53 features. Also, the dataset is labeled, unlike the Office environments dataset.

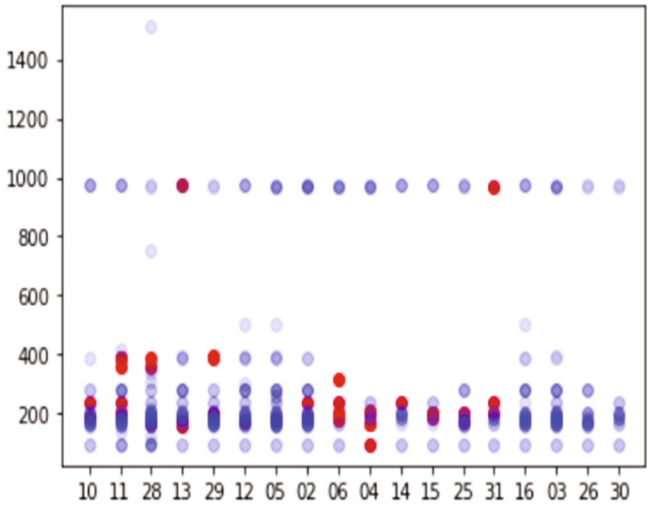

FIGURE 4 Packet length analysis of normal and abnormal using one class SVM.

3.2 | Data preprocessing

To make the datasets ready for analysis, missing data is removed by the method data. Isnull ().values. any (). As dataset 2 contains both categorical data and numerical data, the categorical data should be transformed by one hot encoding. Get_ dummies method is used to create the dummies for one hot encoding. To eliminate the undesired features for getting better results, Feature selection is done. Sequential forward selection is one of the best methods that perform iteratively to select the features but, in our work, all the features are given importance after considering the output of sequential forward selection.

3.3 | Models description

Before applying the machine learning models, the steps described in Figure 1 were performed. We applied 3 unsupervised algorithms namely Local outlier Factor, Isolation Forest, and One-class SVM on dataset 1.

LOF is an unsupervised technique that does not require prior examples. Local outlier factor is the most common technique for anomaly detection. This algorithm works on the concept of local density. It compares the local density of an object with that of its neighboring data points. If a data point has a lower density than its neighbors, then it is considered an outlier.

Isolation Forest is an unsupervised algorithm that is based on the principle that anomalies are few and different and are easy to isolate from the rest of the data. General anomaly detection approaches first define the normal behavior and filter everything that doesn't conform to normal definition as anomalies because of which we may get more false positives.

The isolation forest algorithm overcomes this by "isolating" anomalies by creating decision trees over random attributes. And it is computationally faster when compared to other algorithms as it isolates anomalies quickly and easier.

One class SVM (support vector machine) is an unsupervised approach and a variation of SVM that can be used for anomaly detection. For the model-building process, one-class SVM doesn't require output labels but regular SVM requires output variables. One class SVM learns the boundary of the normal data points and the data points which do not fall into this normal boundary are categorized as anomalies. Upon dataset 2, we have applied two models called XG Boost and Gaussian Naïve Bayes. G Boost is an effective algorithm for the detection of anomalies in time series data.

XG Boost is an ensemble method that combines predictions of multiple weak models to produce a stronger prediction. XG Boost has a large range of parameters. To take advantage, parameters need to be tuned. Grid search CV is the technique that can be used to search for the best parameters from the grid of parameters. Best parameters are extracted from grid search CV and predictions are made.

Gaussian Naive Bayes is a probabilistic classification algorithm in anomaly detection, the goal is to identify data points that deviate significantly from the normal behavior or patterns of the system. Gaussian Naive Bayes is particularly useful for anomaly detection in cases where the data is continuous and normally distributed. It assumes that the probability distribution of each attribute of the data follows a Gaussian (or normal) distribution.

The algorithm learns the parameters of the normal distribution for each attribute of the data from a training dataset. It then uses these parameters to calculate the likelihood of a new data point belonging to each class. The class with the highest likelihood is considered the most probable class for the data point. These algorithms are particularly well-suited for IoT data because they excel at handling high-dimensional data, are robust to data imbalances, and can detect anomalies in local or global contexts. IoT data often exhibits these characteristics, making Isolation Forest and LOF valuable tools for effective anomaly detection.

On the SWaT dataset, data pre-processing steps like handling missing values, categorical encoding, and test train split with test size 0.2 are performed. Later ML models as shown in Figure 5 are created using Isolation Forest unsupervised algorithms and XG Boost, Gaussian Naive Bayes supervised algorithms were also performed and the following are the evaluation metrics of each algorithm.

4 | RESULTS AND ANALYSIS

For dataset 1, the model local outlier factor was trained with the following parameters, n_estiamators = 70 and contamination factor = 0.1. The total number of anomalies predicted by the local outlier factor is 46 and the normal labels are 408. The accuracy is 0.69. Whereas the Isolation Forest algorithm gave 0.94 accuracy when the contamination factor is 0.05 and One-class SVM gave 0.722 accuracies with outlier fraction 0.1 and nu = 0.95.

According to the description of the dataset, sensors were tampered with on days 25 and 26. The above Figures 3 and 4 visually represent anomalies in the color red and in Figures 2–4 we could see that on days 25 and 26 anomalies were detected by all the chosen ml models. The other anomalies represent the weekend data and breaks where the absence of people creates abnormal situations other than sensor tampering.

As per Figure 5 accuracy, both the local outlier factor and one-class SVM are performing better. But when we consider both accuracy and F1 score which is the harmonic mean of precision and recall only one class SVM performed better than the other Machine learning models but still does not meet the levels of real-world anomaly detection. So, we applied a decision tree which is a supervised model and the following Table 1 describes the evaluation metrics. The decision tree

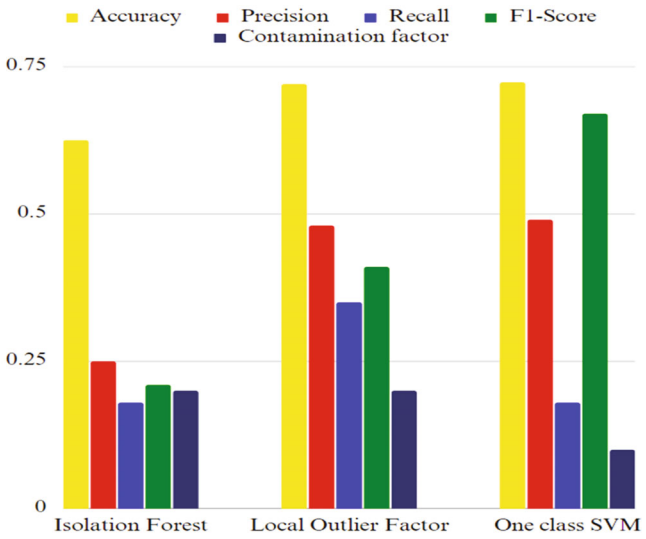

FIGURE 5 Performance comparison among used models for office environment dataset.

TABLE 1 Supervised model accuracy.

Algorithm	Accuracy	Precision	Recall	F1-score
Decision tree	0.91	0.88	0.93	0.74

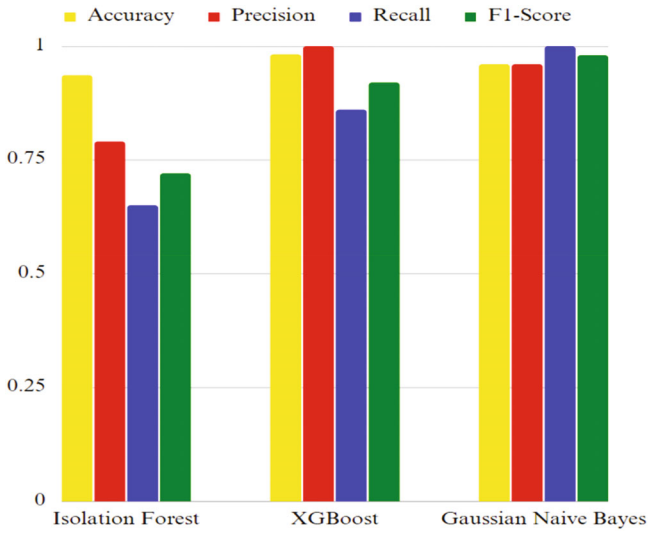

FIGURE 6 Performance comparison among chosen models for Swat dataset.

algorithm performed better with an accuracy of 91% and recall of 0.9 which is almost to 1 and describes the model as performing well as shown in Table 1.

For dataset 2 the results are shown in Figure 6. When choosing the best algorithm for detecting anomalies, it is important to consider the F1-score, precision, and recall. Precision measures the proportion of true positives among all the positive predictions, while recall measures the proportion of true positives among all the actual positives.

An algorithm with high precision is good at detecting true positives and avoiding false positives, while an algorithm with high recall is good at detecting all the actual positives. As F1-score is a harmonic mean of precision and recall, which measures the balance between the two. An algorithm with a high F1 score indicates that it has a good balance between precision and recall, which is important for detecting anomalies.

5 | CONCLUSION

In conclusion, our study has explored the realm of anomaly detection through the utilization of both density-based and time series algorithms. Our experimentation encompassed unsupervised methods, primarily featuring Isolation Forest, Local Outlier Factor, and One-Class SVM, on dataset1, which represents an office environment's network dataset. These methods displayed commendable accuracy; however, it's important to recognize that accuracy alone doesn't always serve as the ultimate criterion for effective outlier detection. In numerous scenarios, the paramount objective is to identify all genuine outliers (maximizing recall) rather than solely focusing on the reduction of false positives (maximizing precision). Hence, we took into account both accuracy and the F1 score as our evaluation metrics. On dataset1, the One-Class SVM outperformed Local Outlier Factor and Isolation Forest, which underscores its efficacy in outlier detection. Transitioning to the SWaT dataset, we adopted a different approach. Here, supervised algorithms, including Isolation Forest, XGBoost, and Gaussian Naive Bayes, were employed. Notably, these algorithms exhibited impressive performance, achieving accuracy levels exceeding 90% and F1 scores approaching unity. This outcome signifies that, for our specific study, supervised algorithms prove to be more adept at anomaly detection.

DATA AVAILABILITY STATEMENT

These data were derived from the following resources available in the public domain: https://itrust.sutd.edu.sg/itrust-labsdatasets/datasetinfo.

ORCID

Sairam Utukuru https://orcid.org/0000-0001-9639-2923

REFERENCES

1. Hastie T, Tibshirani R, Friedman J. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. 2nd ed. Springer; 2009.
2. Wei Z, Wang F. Detecting anomaly data for IoT sensor networks. *Sci Program*. 2022;2022:4671381. doi:10.1155/2022/4671381
3. Pathak AK, Saguna S, Mitra K, Åhlund C. Anomaly Detection using machine learning to discover sensor tampering in IoT Systems, ICC 2021: IEEE International Conference on Communications. 2021, 1–6. doi:10.1109/ICC42927.2021.9500825
4. https://www.trendmicro.com/vinfo/mx/security/news/internet-of-things/the-iot-attack-surface-threats-and-security-solutions
5. Chatterjee Y, Ahmed BS. IoT anomaly detection methods and applications: a survey. *Internet Things*. 2022;19:100568. doi:10.1016/j.iot.2022.100568
6. Liu Z, Thapa N, Shaver A, Roy K, Yuan X, Khorsandroo S. Anomaly Detection on IoT Network Intrusion Using Machine Learning," 2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD). 2020, 1–5. doi:10.1109/icABCD49160.2020.9183842
7. Sahu NK, Mukherjee I. Machine Learning based anomaly detection for IoT Network: (Anomaly detection in IoT Network), 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184). 2020, 787–794. doi:10.1109/ICOEI48184.2020.9142921
8. Cui Y, Bao J, Wang J, Zhang Q, Jiang X. Spatio-Temporal Correlation based Anomaly Detection and Identification Method for IoT Sensors, 2019 International Conference on Control, Automation and Information Sciences (ICCAIS). 2019, 1–6. doi: 10.1109/ICCAIS46528.2019.9074607
9. Band SS, Ardabili S, Sookhak M, et al. When smart cities get smarter via machine learning: an in-depth literature review. *IEEE Access*. 2022;10:60985-61015.
10. Tran DH, Nguyen VL, Utama IBKY, Jang YM. An Improved Sensor Anomaly Detection Method in IoT System using Federated Learning, 2022 Thirteenth International Conference on Ubiquitous and Future Networks (ICUFN). 2022, pp. 466–469. doi:10.1109/ICUFN55119.2022.9829561
11. Teh HY, Wang KI-K, Kempa-Liehr AW. Expect the unexpected: unsupervised feature selection for automated sensor anomaly detection. *IEEE Sensors J*. 2021;21(16):18033-18046. doi:10.1109/JSEN.2021.3084970
12. Nizam H, Zafar S, Lv Z, Wang F, Hu X. Real-time deep anomaly detection framework for multivariate time-series data in industrial IoT. *IEEE Sensors J*. 2022;22(23):22836-22849. doi:10.1109/JSEN.2022.3211874
13. ElMenshawy D, Helmy W. Detection techniques of data anomalies in IoT: a literature survey. International journal of civil. *Eng Technol*. 2018;9:794-807.
14. Tian H-X, Liu X-J, Han M. An outliers detection method of time series data for soft sensor modeling, 2016 Chinese Control and Decision Conference (CCDC). 2016, 3918–3922. doi:10.1109/CCDC.2016.7531669
15. Antonini M, Pincheira M, Vecchio M, Antonelli F. A TinyML approach to non-repudiable anomaly detection in extreme industrial environments, 2022 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0&IoT). 397–402 2022.
16. Silva MF, Pacini A, Sgambelluri A, Valcarenghi L. Learning long- and short-term temporal patterns for ML-driven fault Management in Optical Communication Networks. *IEEE Trans Netw Serv Manag*. 2022;19(3):2195-2206.
17. Tang S, Zhaochen G, Yang Q, Song F. Smart Home IoT Anomaly Detection based on Ensemble Model Learning From Heterogeneous Data, 2019 IEEE International Conference on Big Data (Big Data). 4185–4190 2019.
18. Wang R, Jiang S, Ma D, Sun Q, Zhang H, Wang P. The energy Management of Multiport Energy Router in smart home. *IEEE Trans Consum Electron*. 2022;68(4):344-353.
19. Sairam U, Voruganti S, Prakash MVB, Reddy RG. A study on IoT applications towards impact of loss of data, 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India. 2021, pp. 440–445. doi:10.1109/ICOEI51242.2021.9452935
20. Gaurav A, Gupta BB, Panigrahi PK. A comprehensive survey on machine learninapproaches for malware detection in IoT-based enterprise information system. *Enterp Inf Syst*. 2023;17(3):2023764.
21. Cvitić I, Peraković D, Periša M, Gupta B. Ensemble machine learning approach for classification of IoT devices in smart home. *Int J Mach Learn Cybern*. 2021;12(11):3179-3202.
22. Tiwari A, Garg R. Adaptive ontology-based IoT resource provisioning in computing systems. *Int J Semantic Web Inf Syst*. 2022;18(1):1-18.
23. Raj MG, Pani SK. Chaotic whale crow optimization algorithm for secure routing in the IoT environment. *Int J Semantic Web Inf Syst*. 2022;18(1):1-25.
24. Wassan S, Suhail B, Mubeen R, et al. Gradient boosting for health IoT federated learning. *Sustainability*. 2022;14(24):16842.

How to cite this article: Bilakanti H, Pasam S, Palakollu V, Utukuru S. Anomaly detection in IoT environment using machine learning. *Security and Privacy*. 2024;7(3):e366. doi: 10.1002/spy2.366