IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 38, No. 4, August, 2024, pp. 1109-1118
Journal homepage: http://iieta.org/journals/ria

Outlier Detection in Wireless Sensor Networks Using Machine Learning and Statistical Based Approaches

Alaa Darabseh*, Mohammad Faizan

Department of Mathematics, Engineering and Computer Science, LaGuardia Community College, CUNY, NY 11101, USA

Corresponding Author Email: adarabseh@lagcc.cuny.edu

https://doi.org/10.18280/ria.380406

Received: 31 January 2024
Revised: 5 April 2024
Accepted: 22 June 2024
Available online: 23 August 2024

Keywords:
wireless sensor networks, outlier detection, machine learning, statistical approaches, anomaly detection, energy-aware detection

ABSTRACT

Outliers in wireless sensor networks (WSNs), stemming from harsh environmental conditions and limited processing and communication capacities of sensor nodes, pose a significant challenge to data reliability and quality collected by the network. Energy-efficient outlier detection methods are crucial for prolonging network lifespan. This study introduces a two-phase approach to address this challenge. At sensor nodes, a lightweight statistical method based on mean and standard deviation detects and removes outliers, conserving energy. Early outlier filtering reduces data transmission, saving substantial energy due to the high energy cost of communication in WSNs. At the base station, several unsupervised Machine Learning algorithms, including One Class Support Vector Machine (OCSVM), Histogram Based Outlier Score (HBOS), Isolation Forest (IForest), K-Nearest Neighbor (KNN), and Cluster Based Local Outlier Factor (CBLOF), identify remaining outliers. The base station, with greater computational power and energy resources, can handle these tasks without the constraints faced by sensor nodes. Evaluation using real-world datasets demonstrates the effectiveness of our approach, achieving a 77.59% outlier removal rate at the node level while maintaining over 90% detection accuracy at the base station. By employing computationally light statistical methods at sensor nodes, reducing data transmission, and shifting complex tasks to the base station, our approach optimizes energy efficiency, minimizing consumption and reducing the need for frequent recalibration or maintenance, thereby extending the lifespan of the network.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) that we know today are used in various domains including healthcare, agricultural industry, military, and environmental monitoring. In the healthcare industry, they are used to monitor the patients' physical conditions, such as heartbeat and temperature, or to track patients and detect any unusual behavior, such as stroke or a fall. In a similar way, WSNs are used in the farming industry to automate many difficult tasks according to the need of farms, such as providing timely water supply, monitoring the temperature and the weather, or managing the use of fertilizers and pesticides. In the military, wireless sensors are used for surveillance, target tracking/soldier tracking, detection of snipers, and for many other purposes.

WSN consists of numerous interconnected small nodes that can be deployed in a distributed area of interest and can be self-configured. Every node is commonly built with a microcontroller, memory, power supply, radio, and an assortment of sensors. WSN nodes are typically tiny, inexpensive, and have limited amounts of energy, computing power, and storage [1]. In a WSN, nodes communicate wirelessly, collaborating to collect, process, and transmit large amounts of sensed data to selected gateway nodes, which may include base stations, cluster heads, or data mules. These gateway nodes, also known as sink points, allow for extra data processing and analysis.

Depending on the application type and the size of the WSN, data transmission can be accomplished through either a single-hop or a multi-hop approach. In a single-hop network, nodes communicate directly with a central node, limiting scalability and potentially requiring more power. In contrast, multi-hop networks allow communication through multiple intermediate nodes, providing scalability, energy efficiency, increased reliability, and redundancy.

In recent years, the efficiency of data collection in WSN has significantly increased with the integration of data mules [2, 3]. Data mules are mobile devices deployed to improve the effectiveness of data collection. These devices, whether in the form of robots or vehicles, have the ability to function in various environments, including aerial (drones), land-based (ground vehicles), or water-based (underwater vehicles). The primary function of data mules is to go around the sensor network and collect data from individual sensors or sensor clusters and submit it to the base station. Figure 1 shows a generic WSN model using data mules.

Even though WSNs have advanced a lot in recent years, they are still limited in what they can do because of the challenges they face. WSNs transmit their data with wireless links with no infrastructure and are prone to many failures, such as hardware failure or physical damage due to harsh environment [4]. There are many hardware constraints that WSNs face, such as limited power supply, low processing, and below average transmission units. These constraints make it very difficult for WSNs to add any additional functionality or carry out complex tasks at the node level, as WSNs must collect data using imperfect sensor nodes with limited resources and power supply. Thus, the data gathered from these nodes is prone to diverse faults, potentially resulting in abnormal data patterns within the monitoring domain. These data readings that deviate significantly from normal readings and are inconsistent with the rest of the data are called outliers which are also known as anomalous [5].

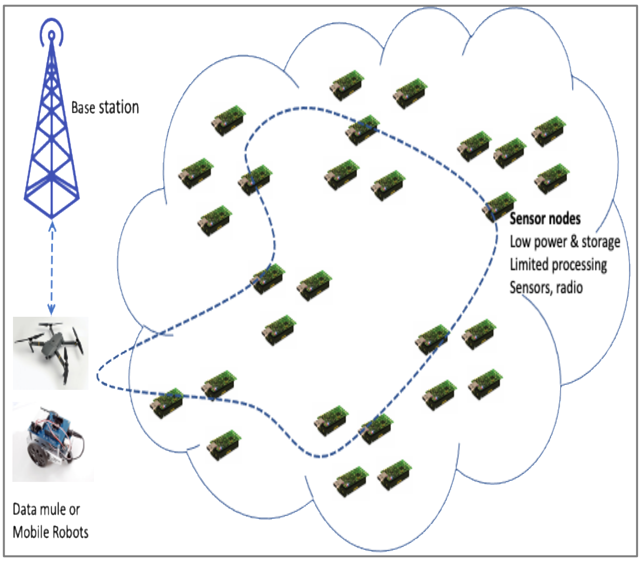

Figure 1. WSN model using data mules

Outlier detection is essential in WSNs as it plays a crucial role in maintaining data quality, system reliability, and facilitating well-informed decision-making. Its primary function is to identify anomalies and errors within sensor data, enabling early warning systems for abnormal conditions, optimizing resource utilization, and assisting in fault diagnosis. The accuracy, resilience, and operating efficiency of WSNs are greatly enhanced in a variety of scenarios and applications by proactively detecting and addressing anomalies.

Outlier detection provides important benefits across various real-world applications such as healthcare, agriculture, military, and environmental monitoring. For instance, in healthcare, WSNs monitor crucial indicators like heart rate and temperature. In this case, the application of outlier detection assists in identifying irregularities in patient data, allowing early interventions for critical conditions. Similarly, in environmental monitoring applications, sensors continuously track parameters such as humidity and temperature in harsh regions, helping in the early detection of abnormal environmental changes. In military domain, outlier detection plays a vital role in identifying unusual activities such as unexpected movements or unauthorized access in sensitive areas.

Outliers in WSNs can be detected at different stages of the data transmission process. Outlier detection techniques can be categorized as centralized, distributed, or hybrid depending on where the outliers are detected. In a centralized approach [6], data is gathered from multiple nodes and sent to a central base station. Finding outliers in the received data is the responsibility of the central base station. By sending data to a central location, this technique improves the detection process but may result in increased communication overhead. Outliers are identified at individual nodes or node clusters using a distributed approach [7]. By finding outliers closer to the data source, this decentralized approach intends to reduce transmission overhead. The hybrid approach utilizes diverse outlier detection algorithms to detect outliers at various locations in the network [8]. By combining the advantages of both distributed and centralized techniques, this approach seeks to improve overall outlier detection performance. Choosing between centralized, distributed, or hybrid approaches often involves deciding on trade-offs between communication overhead, computational efficiency, and network condition adaptability.

In WSN, Outlier detection methods can be categorized based on their underlying methodologies, mainly into statistical and machine methods [9-11]. Statistical methods involve analyzing the statistical properties of data such as mean, median, and z-score to identify outliers based on deviations from the expected distribution. On the other hand, machine methods employ machine learning algorithms to learn patterns from the data and detect outliers by recognizing deviations from these learned patterns. Machine methods include nearest neighbor-based approaches, clustering approaches, Isolation Forests, Support Vector Machines, and others. Some of the mentioned outlier detection methods tend to be more power-efficient than others. Generally, statistical methods involve basic calculations that don't require an excessive amount of power, although the actual impact depends on the complexity of the statistical measures utilized. On the other hand, machine learning-based methods typically involve intensive computations, potentially leading to higher power consumption [11]. Hence, when selecting outlier detection methods for WSNs, it's crucial to prioritize power efficiency to minimize energy consumption and extend the lifespan of sensor nodes without frequent battery replacements. This becomes particularly important when nodes are positioned in remote or challenging-to-access areas, making battery replacement more difficult.

In WSNs, many researchers have shown that data transmission costs for sensor nodes are higher than computation costs because communication uses more energy than computation [12]. Traditional outlier detection methods have mainly been designed based on centralized approaches, in which all data collected by sensor nodes is transmitted to a base station or cluster head for preprocessing. In fact, most existing anomaly detection techniques assume the availability of extensive computational resources and centralized access to all data within the relevant domain. Although this methodology exhibits a reasonable detection rate, it also leads to higher energy usage and communication overhead, ultimately reducing network lifetime and blocking network traffic. Moreover, processing data only at the base station results in a single point of failure for network operations and delays in anomaly detection and response times, impacting time-sensitive applications such as security monitoring, disaster response, or industrial automation. Consequently, these centralized approaches are unsuitable for WSNs due to the energy constraints characteristic of WSNs. Correspondingly, distributed approaches offer significant benefits, such as reducing communication overhead, providing real-time response, and avoiding single points of failure within the network. However, they also come with certain limitations, including complexity, limited resources like processing power, memory, and energy, as well as potentially lower performance levels. Additionally, this approach faces communication constraints, such as substantial propagation delay, signal absorption, extreme path lengths, rapidly changing time-varying channels, noise, and diffusion. Therefore, communication cost is inherently a fundamental challenge for outlier detection approaches in WSNs. In other words, how to reduce communication costs to extend system lifetime and minimize network traffic in WSNs is a major challenge. Addressing this issue requires innovative approaches that balance the trade-offs between detection accuracy, energy efficiency, and communication overhead.

In this paper, we propose a two-step outlier detection technique for WSNs. At the sensor level, we apply a simple statistical method to address limited energy and computational power constraints. At the base station, we employ unsupervised machine learning to identify outliers that may have been missed at the sensor level. This approach aims to reduce communication overhead, optimize resource usage, and improve network performance and reliability. The objectives of the research can be defined as follows:

1. Minimize the energy consumption and communication costs in WSNs resulting from high data transmission costs between sensors and the centralized location, while maintaining accuracy comparable to centralized detection.

2. Prolong the operational lifespan of sensor nodes and the overall WSN by employing targeted outlier detection techniques aimed at reducing communication costs, reduce network traffic, and improve energy efficiency by efficiently identifying and addressing anomalies within the network.

3. Ensure real-time anomaly detection and response by integrating localized anomaly detection at the node level with centralized detection at the base station. This ensures timely detection and mitigation of anomalies, which is crucial for applications that require immediate response to maintain operational integrity and security.

By achieving these objectives, the study aims to enhance the efficiency, reliability, and sustainability of outlier detection techniques in WSNs, addressing key challenges related to communication costs, resource limitations, and real-time responsiveness.

The rest of the article is organized in various sections including, motivation in Section 2, literature review in Section 3, outlier detection algorithms in Section 4, data sets, analysis, and evaluation metrices in Section 5, results in Section 6 and conclusions in Section 7.

2. MOTIVATION

Wireless Sensor Networks were developed in the 1950s but because of their cost per sensor, limited energy supply, and the computational power needed to run the network, these WSNs were mostly used for military and government applications. Soon the popularity of these networks grew along with their applications and WSNs became a topic of interest in the scientific community. WSNs consist of many different sensors like temperature, humidity, pressure, proximity, and acoustics sensors, which collect data from the environment and send it back to the base station for further processing [13]. For this reason, they can be dispersed in the environment randomly and the data collected from the environment can be used to monitor the environment or to maintain some specific condition in that environment. Even though WSNs have a wide range of applications, the cost it takes to implement WSNs is still not feasible for many day-to-day applications.

To ensure the reliability of data coming from a sensor it is crucial that the data is free from any kind of anomaly and if there is any outlier in the data, it is detected by the system [11]. The detected outlier can be used to make some specific action. For example, when monitoring a forest using temperature sensors, fire in the forest can cause outliers in the data and when the system detects these outliers it can trigger an event such as calling the forest rangers or sounding an alarm. But sometimes these outliers may also be a result of faulty sensor or an error in the data transcription and transmission, or it can be a result of intrusion in the system. These outliers along with noisy data can cause transmission overhead, communication delay and energy drainage [11]. Thus, it is crucial to detect outliers in a WSN to ensure the reliability, efficiency, and security of data, as well as the effective operation of the network, regardless of the type of outlier. A key challenge in the advancement of WSNs is the development of algorithms for outlier detection that achieve accurate and speedy detection while minimizing energy costs.

To remove these outliers from the data many different techniques have been presented by scientists and engineers such as statistical based Zhang et al. [14], clustering based Rajasegarar et al. [15], classification based Feng et al. [16], nearest neighbor based Xie et al. [17], and others. Among these, machine learning-based approaches are preferred for outlier detection due to their high accuracy, adaptability to dynamic data patterns, and real-time capabilities, making them well-suited for outlier detection in WSNs where the environment may vary. Some machine learning models, especially those designed for efficiency, can provide real-time outlier detection. This becomes crucial for applications in WSNs that require timely responses due to their time-sensitive nature. Additionally, machine learning-based techniques have the ability to automatically learn and extract relevant important features from the data, eliminating the need for manual feature engineering and potentially enhancing overall efficiency of complex outliers.

However, machine learning -based approaches come with challenges. They are more complex, requiring significant computational power, which may be impractical for nodes in WSNs with limited energy, storage, and computational resources. The majority of existing anomaly detection techniques operate under the assumption of having significant computational resources and centralized access to all data within the relevant domain [18]. Kumar et al. [13] recommended that machine learning algorithms are needed to run centrally (centralized approach). While this centralized strategy overcomes the energy constraints of machine learning -based approaches, it prevents real-time outlier detection, which is crucial, particularly for dynamic outdoor applications. Moreover, not applying any kind of detection technique on the node level (decentralized setting) can cause transmission delay and communication overhead because of too many outliers coming from the sensor which can also decrease the efficiency and efficacy of the network [11, 18]. In a decentralized setting, statistical methods offer practical solutions for the detection of outliers at individual nodes. Typically, statistical methods are computationally less intensive compared to machine learning-bases methods, making them well-suited for effective implementation in such settings. Methods like the z-score or quantile-based measures analysis can be applied without significantly burden nodes with limited processing power. These techniques are easily interpretable and can identify outliers by utilizing information at the local level. However, the challenge is to ensure that statistical methods can effectively detect anomalies without having a global view of the entire network. Additionally, the performance of statistical techniques may face challenges in scenarios involving complex and non-linear patterns, or where the assumptions regarding data distribution are not met. These limitations highlight the potential risks associated with depending exclusively on statistical methods for outlier detection.

Another drawback is that Machine learning models often require a substantial amount of historical labeled training data especially with Supervised Learning methods, which may be challenging to obtain in certain WSN scenarios. One approach to overcome this challenge is the utilization of Unsupervised machine learning algorithms, as they do not require historical labeled inputs for algorithm training. However, the majority of research works focus on using supervised machine learning for outlier detection in WSN. A recent survey on machine learning algorithms for WSNs [13] indicates that 67% of current studies utilize supervised learning algorithms to solve WSN issues, while unsupervised learning methods have been applied in only 18% of WSN problem-solving issues in recent years.

The decision between employing machine learning or statistical methods in centralized and decentralized outlier detection approaches requires careful evaluation of factors including interpretability, adaptability, computational efficiency, data availability, and real-time processing needs. Making informed decisions can lead to the development of outlier detection strategies aligning with the specific characteristics and constraints of WSNs. This, in turn, contributes to the overall enhancement of their reliability and efficiency. This research aims to advance the field of outlier detection in WSNs by investigating and comparing the effectiveness of machine learning and statistical methods within centralized and decentralized frameworks. The primary objective is to find a balance that ensures accurate outlier detection while taking into account the energy constraints of WSN nodes. Ultimately, improving the reliability, efficiency, and real-time responsiveness of WSNs, making valuable contributions to the advancement of outlier detection techniques.

In this research, we introduce a two-phase outlier detection approach designed to identify outliers at both sensor nodes and the base station levels in a WSN. Specifically, at the sensor level, we utilize a statistical technique that is resource-efficient to detect and eliminate outliers while considering energy and computational limitations. On the other hand, we use Unsupervised Machine Learning methods at the base station to find outliers that could have gone undetected at the sensor level, improving the network's overall accuracy and reliability. The primary objective of this approach is to minimize communication overhead and decrease the number of outliers transmitted to the base station. This reduction strategy aims to extend the overall lifespan of the network by minimizing unnecessary data transmission and optimizing resource utilization.

The main contributions of this paper are as follows:

1. Minimize communication overhead in the network by adopting a two-phase outlier detection approach, which effectively addresses outlier detection at both the sensor nodes and the base station levels. By incorporating outlier detection methods at multiple stages, the approach aims to optimize resource utilization, decrease unnecessary data transmission, and ultimately enhance the overall performance and reliability of the network.

2. Compare the effectiveness and performance of the two-step approach (node-level processing + base station) against a scenario where only base station-level is applied. This comparison helps in understanding the additional value of integrating a simple statistical method at the sensor level to remove outliers locally and whether it affects outlier detection compared to base station alone.

3. Assess the effectiveness of diverse unsupervised Machine Learning algorithms for outlier detection at the base station. The aims to gain insights into the practicality and efficacy of these methods in real-world WSN scenarios, particularly when integrated with statistical methods implemented at the node level.

3. LITERATURE REVIEW

In this section, we highlight the most popular and recent studies proposed by researchers for outlier detection in WSNs.

Wang et al. [19] proposed an isolation-based distributed outlier detection framework using nearest-neighbor ensembles (iNNE). This framework consists of a local detector and a global detector. The local detectors are constructed in each node through iNNE algorithm, while the global detector is created combining local detectors from a node and its neighboring nodes. Experimental results showed enhanced detection accuracy and reduced false alarms compared to other techniques. However, limitations include reliance on static parameters and system complexity.

Miao et al. [20] formulated a distributed online one-class SVM algorithm where they use a random approximate function which maps the input data to low-dimension feature space to get a cost function. Then another cost function is created by adding sparse constraint to the previous one, stochastic gradient descent is then applied to this function to get two distributed functions do OCSVM and sparse doOCSVM. The proposed algorithms offer low misdetection rates and high true positive rates with minimal resource usage, ideal for resource-constrained of WSNs. However, challenges like implementation complexity and scalability need addressing for wider real-world deployment.

Poornima and Paramasivan [21] proposed an Online Locally Weighted Projection Regression (OLWPR) for anomaly detection in WSNs, utilizing PCA for dimensionality reduction. LWPR regression predicts sensor values, compared to actual readings, with error determined using a dynamic threshold. Achieving an 86% detection rate and 16% error rate, the method exhibits high detection rates and few false alarms. However, its efficacy may depend on training data quality, facing challenges in computational complexity and scalability.

Gupta et al. [22] utilize the Outlierness Factor-based on Neighbourhood (OFN) technique to analyze and detect the outliers in sensor network. In the proposed approach, the initial step involves determining the neighborhood points, followed by the calculation of the weight assigned to the neighborhood data. The (OFN) technique is applied to categorize outlier data points into events and errors. This classification is based on spatial correlations, representing neighborhood readings, and temporal correlations, representing timestamps of readings. The performance of the proposed approach was evaluated on a real dataset. While the proposed technique showcases notable strengths in efficiency compared to previously considered approaches, it also exhibits potential weaknesses such as sensitivity to parameter settings and the complexity of real-time adaptability to dynamic environmental conditions.

Thangaramya et al. [23] introduced a secured WSN communication model using fuzzy temporal clustering and trust analysis. Their method enhances routing security by identifying malicious nodes and improves communication reliability, packet delivery, delay, and energy consumption. However, implementation complexity and scalability challenges persist, and there's a lack of comparison with other mechanisms regarding outlier detection's impact on packet delivery.

Zidi et al. [24] propose a two-phase detection technique using Support Vector machine. In the first phase a decision function along with support vectors are established in anticipated time, which is then fed to cluster head to classify data. In phase two an observation vector is created using the last three data measurements on which the decision function is applied. If the result is positive the data is correct otherwise it is considered an anomaly. The algorithm provides the most detection accuracy as compared to other algorithms used in the experiment with an accuracy of more than 99%. Despite its resource-efficient nature, SVM-based fault detection may encounter complexities in implementation and parameter tuning, potentially affecting its performance in dynamic and nonlinear scenarios. Nevertheless, its adaptability and validation in real-world applications highlight its suitability for practical deployment in wireless sensor networks.

Yu et al. [25] proposed an unsupervised contextual outlier detection method in WSNs. The method identifies outlier correlations based on contextual spatial and temporal neighbor values using DBSCAN and Grid partitioning. The experimental results demonstrate that this method can accurately and efficiently detect not only individual anomalies but also anomalous events. The proposed method for anomaly detection in WSNs demonstrates adaptability and comprehensive detection capabilities. However, its effectiveness relies on the selection of clustering method, which has a certain influence on the detection results.

Chander and Kumaravelan [12] proposed a two-step approach for secure routing and outlier detection in WSNs. First, the fuzzy rule and cluster-based secured routing with outlier detection (FRCSROD) algorithm secures communication. Second, the fuzzy rule and distance-based outlier detection algorithm (FRDOA) identifies malicious nodes. This approach enhances security, reliability, and packet delivery rates while reducing communication delays. However, the complexity may increase computational overhead and impact scalability and resource use.

4. OUTLIER DETECTION ALGORITHMS

We implemented and evaluated five machine learning algorithms for outlier detection in WSNs. In the next subsections we briefly discuss overview of the utilized unsupervised machine learning algorithms.

4.1 OCSVM

OCSVM [26, 27] is an unsupervised machine learning algorithm that works by creating a hypersphere around the data in a feature space and creating a hyperplane which has a maximum distance between the data points and the origin. The second class in OCSVM is considered to be the origin. OCSVM uses a parameter v (Nu) which lies between (0, 1) and it is the upper bound on the fractions of outliers we want to allow. OCSVM is commonly used in various fields such as cybersecurity, fraud detection, and industrial monitoring where detecting anomalies is crucial for maintaining system integrity and security. OCSVM can be computationally efficient in analyzing high-dimensional data with a reasonable sample size, however, its utility diminishes when dealing with extensive datasets or extremely high-dimensional feature spaces.

4.2 HBOS

The Histogram Based Outlier Score [28] is an unsupervised machine learning algorithm used to detect outliers. It works by creating a univariate histogram for each feature with dynamic or static bandwidth. Afterwards each datapoint is given a score based on the histogram created and the greater the score, the more likely the datapoint is an outlier. HBOS is effective for real-time anomaly detection, making it ideal for quick and scalable solutions. It identifies unusual network traffic, detects payment fraud, monitors business transactions, and tracks patients' vital signs. HBOS processes data quickly, suitable for large datasets, but its storage needs can be high due to histograms and outlier scores scaling with dataset size.

4.3 IForest

Isolation Forest [29] make use of isolation trees and work in a similar way as random forest. The principle behind Isolation Forest is that in each dataset outliers are much easier to isolate than a normal datapoint. IForest randomly picks up a feature and builds isolation tree on that feature until the datapoint is completely isolated. The lesser the number of partitions it took to isolate a datapoint, the higher the chances of that datapoint to be an outlier. IForest is used in various domains: detecting intrusions in network security, identifying fraudulent transactions in finance, and spotting unusual patient data in healthcare. Its time and space complexity can vary from moderate to high, depending on dataset size, number of features, and tree depth.

4.4 KNN

K-Nearest Neighbour [30] is a non-parametric lazy algorithm that uses proximity to classify datapoints into different classes. KNN uses Euclidean distance to calculate the distance between the datapoint and its K neighbours and then sorts the distance in order of smallest to largest. The datapoints that have the largest distance from their neighbours in considered to be an outlier. KNN is a versatile and effective algorithm used in recommendation systems, targeted marketing, cybersecurity, and environmental monitoring. It is efficient for small to medium-sized datasets, but its scalability decreases with larger datasets due to extensive distance calculations.

4.5 CBLOF

Cluster Based Local Outlier Factor [31] arranges a given dataset into several clusters and then calculates the size of those clusters and identifies them into small and large clusters.

For each datapoint outlier score is calculated using the CBLOF which is measured using the size of the clusters. CBLOF is used in cybersecurity to detect network anomalies, in healthcare for monitoring patient data, and in finance to identify fraudulent transactions. Its time complexity varies with the clustering algorithm and dataset size, and it requires storage space that scales with the dataset's size and features.

5. EXPERIMENTAL SETUP

5.1 Data set

The dataset used in this research was published by Suthaharan et al. [32], who belongs to the department of Computer Science, at the University of North Carolina at Greensboro. The data was collected from multi-hop and single-hop network scenarios using real sensor networks and protocols. Sensors were deployed in both indoor and outdoor environments and were used to collect the reading of temperature and humidity. In the indoor settings, it's reasonable to presume that conditions were more stable and controlled compared to outdoor environments. Factors like consistent room temperature, controlled humidity levels, and limited exposure to external elements could characterize these indoor setups. On the other hand, outdoor deployments are subject to natural environmental fluctuations including changes in temperature and humidity, exposure to sunlight, wind speed, and other outdoor elements.

The data collection spanned 6 hours. At the midpoint of this duration, anomalies were generated in one sensor node in each setup (indoor and outdoor) by using a hot water kettle, which raised both the temperature and humidity simultaneously. Introducing anomalies through the use of a hot water kettle likely caused abrupt increase in temperature and humidity. This simulates sudden environmental changes that sensors might experience in real-world situations, such as equipment failures, unexpected weather fluctuations, or localized environmental disturbances. These anomalies serve as crucial data points for training anomaly detection algorithms within WSNs. Table 1 shows the number of instances in each data sample of the dataset.

Table 1. Number of instances in each data sample of the dataset

Setting	Normal	Anomaly
Single hop-indoor	4300	117
Single hop -outdoor	5009	32
Multi hop -indoor	4633	57
Multi hop -outdoor	4590	100

5.2 Analysis

In this research we propose a two-phase detection technique to detect outliers in WSNs. Phase one is performed on the sensor nodes in which outliers above a certain threshold are removed from the data set. Phase two will be performed at the base station, here we will be using a machine learning algorithm to further detect outliers that were not detected in phase one. We used five different unsupervised machine learning algorithms to detect outliers to check the performance of algorithms when combined with standard deviation.

To assess the effectiveness of the proposed approach in terms of their ability to detect outliers in a WSN, two types of analyses were performed.

EXPERIMENT 1: The aim of the initial experiment was to evaluate the effectiveness of various machine learning algorithms in terms of their ability to detect outliers at the base station. Here we performed five different unsupervised machine learning algorithms OCSVM, HBOS, IForest, KNN, and CBLOF. Unsupervised machine learning algorithms are used as they require no labeled inputs to train the algorithm and it can detect hidden patterns in a dataset on its own. On other hand, supervised machine learning needs labelled inputs and outputs to train the algorithm and learn overtime. Therefore, unsupervised machine learning is more suitable for WSNs as WSNs are randomly deployed into nature where the availability of labelled historical data is sometimes not possible.

This experiment involved two main phases: the training phase and the testing phase. In the training phase, classifiers were trained using all observations including both normal and anomalous observations, without the use of labels. During the testing phase, all observations are categorized into either the normal data class or the anomaly class based on the previously trained model. In this process, the classifier assigns labels to each observation based on its predictions, distinguishing between normal and anomalous instances.

Throughout the testing process, the classifiers recorded the number of True Positives (TPs), True Negatives (TNs), False Positives (FPs), and False Negatives (FNs). These recorded values are then used to calculate performance metrics mentioned in the next subsection, offering a comprehensive evaluation of the classifiers' performance.

EXPERIMENT 2: The aim of the experiment was to evaluate the effectiveness and the performance of the proposed two-phase detection approach to compare it with the centralized approach at detecting anomalies in WSNs. This experiment has two phases. In the first phase on sensor nodes, outliers above a threshold are removed. The second phase at the base station uses a machine learning algorithm to identify undetected outliers from phase one.

Phase 1: At the node level we used simple standard deviation to mark any reading that is two standard deviations away from the mean to be labelled as an outlier and we removed those reading form the datasets (the value of standard deviation can be set according to the environment the network will be configured in). The remining data that was not removed will then be used in step two to detect further anomalies that were not able to be detected in phase one. The aim of this phase is to minimize the number of outliers by setting a range depending on the environment the sensor is in, which will decrease the number of readings to be sent to the base station.

Phase 2: At the base station, where more power resources are available, we performed the same machine learning algorithms we used in experiment 1 to detect outliers in the dataset that were not detected in phase one. This phase involved training classifiers on all observations (normal and anomalous) without labels. In the testing phase, the classifier categorized observations into normal or anomaly classes. In the testing phase, we recorded True Positives (TPs), True

Negatives (TNs), False Positives (FPs), and False Negatives (FNs) to compute performance metrics.

The proposed algorithm works through the following steps:

1. Read the dataset.

2. Calculate upper and lower limits for the dataset based on the mean and standard deviation.

3. Find data points in the dataset that fall outside the calculated upper and lower limits.

4. Create a subset containing only the data points that are inside the calculated upper and lower limits.

5. Initialize Features and Labels on the cleaned dataset.

6. Initialize Model

7. Fit the Model: Train the model on the feature data.

8. Use the trained model to predict outliers in the feature data.

9. Calculate various performance metrics such as accuracy, true positive rate (TPR), false positive rate (FPR), and Matthew's correlation coefficient (MCC).

Machine learning algorithms for outlier detection were implemented using the PyOD package [33], a specialized Python package designed for anomaly detection. Default parameter values are employed in each machine learning approach.

5.3 Evaluation metrices

In the context of outlier detection, a confusion matrix is used to assess the effectiveness of the outlier detection model. Unlike traditional binary classification settings, outlier detection focuses on finding instances that substantially vary from the norm. The confusion matrix for outlier detection typically includes the following components:

- True Positive (TP): Instances that are actual outliers and are correctly identified as outliers by the model.
- False Positive (FP): Instances that are not outliers but are incorrectly identified as outliers by the model.
- Ture Negative (TN): Instances that are not outliers and are correctly identified as non-outliers by the model.
- False Negative (FN): Instances that are actual outliers but are incorrectly identified as non-outliers by the model.

These values are then used to calculate performance metrics like accuracy, false positive rate (FPR), and true positive rate (TPR), which offers a concise overview of a model's performance. To evaluate the performance of the algorithms that we used in this research, we have used four evaluations metrices.

a) Detection Accuracy (DA): which is calculated as the number of predictions that were correctly classified by the algorithm. It is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

b) True Positive Rate, which is the measure of how many positive cases were positive in actual. It is defined as:

$$TPR = \frac{TP}{TP + FN} \tag{2}$$

c) False Positive Rate. which is the measure of positive outcomes that were considered incorrect. It is defined as:

$$FPR = \frac{FP}{FP + TN} \tag{3}$$

d) Matthew's correlation coefficient (MCC), and it measures the difference between actual value and predicted value. It is defined as:

$$\mathrm{M}CC = \frac{TP * TN - FP * FN}{\sqrt{(TP+FP)\ (TP+FN)}} \tag{4}$$

6. RESULTS

This section reports the results derived from the conducted experiments. The main purpose of this research is to evaluate and compare the performance of the proposed two-phase detection approach with the centralized approach at detecting anomalies in WSNs, and to find which algorithms perform better in both settings.

6.1 Phase one results

In this subsection, we present and discuss the results obtained from phase one of the study. Table 2 provides a comprehensive record, documenting the initial count of outliers in each of the four datasets and presenting the outcomes after performing standard deviation on the datasets. As observed, the utilization of the standard deviation method on the datasets provides notable success in efficiently eliminating outliers across different scenarios. Specifically, in the single-hop indoor dataset, we successfully removed 72% of outliers. Similarly, for the single-hop outdoor dataset, 90% of outliers was effectively removed. In the case of the multi-hop indoor dataset, 62% of outliers was successfully removed, and in the multi-hop outdoor dataset, 85% of outliers was eliminated during phase one. Collectively, these outcomes represent a significant 77.59% reduction in outliers during the initial phase of the experiment.

In the upcoming subsections, we present and analyze the results of the two-step approach (node-level processing + base station) in comparison to a situation where only base station-level outlier detection methods are employed. The primary objective is to determine whether the utilization of standard deviation method at the sensor level for outlier removal has an impact on the performance compared to an approach only relying on the base station.

6.2 Comparison accuracy

We observe that HBOS performs the best in terms of outlier detection without applying phase one and with higher number of outliers in the dataset with an accuracy of more than 97.7% most of the time but at the same time it is also affected the most when it comes to detect outliers with lesser outliers in the dataset. The accuracy of HBOS drops from 99.59 to 90.31% with a difference of 9.28% when the number of outliers drops from 100 to 15 after performing phase one in Table 3. KNN on other hand is least affected by applying phase 1 and gives an almost constant accuracy of more than 94% in all four datasets with and without performing phase 1. In single-hop outdoor-data, KNN gave 94.5% accuracy with phase 1 and 94.1% without performing phase 1. OSCVM, IForest, and CBLOF have little effect on the accuracy while we perform standard deviation in phase 1, with an average fluctuation of around 2%.

6.3 Comparison TPR

The results of TPR form Table 4 show that KNN gives the highest TPR of 1.0 most of the time and standard deviation does not affect its TPR. OCSVM also gives a TPR of 1.0 in indoor situations on both multi-hop and single-hop data readings and applying phase 1 on the dataset does not affect the true positive but in outdoor sensors it does drop by 0.3. This is also true for the other algorithms as most of the time they give a reading of 1.0 in indoor situations but in outdoor data readings a lesser number of outliers give lesser TPR.

6.4 Comparison FPR

According to the results of FPR in Table 5, HBOS gives the least FPR values that range between 0.00 and 0.02 without applying phase one. After that KNN gives a slightly higher values of FPR than HBOS, which ranged between 0.04 and 0.05 for both scenarios and is least affected by phase one but HBOS on other hand is affected the most and the FPR values increase to 0.07 – 0.09. CBLOF, IForest and OCSVM give almost identical FPR reading and are not that affected by the addition of phase one to the experiment.

6.5 Comparison MCC

Looking at Figures 2 and 3 below, we can observe that all classifiers achieved MCC values above 0.0, which indicates that there is a strong relationship between reality and prediction. Moreover, according to the results of the figures below, we can see that HBOS performs in indoor environments without applying phase one and gives a reading 0.84 and 0.91 but these readings drop to 0.28 and 0.17 respectively. After that KNN can be ranked in second position and it also performs better than other algorithms with the addition of phase one. OCSVM, CBLOF and IForest perform almost the same in terms of performance according to MCC.

Table 2. Number of outliers before and after phase 1

	Outliers Before Node-Level Removal	Outliers Count After Node-Level Removal	Reduction Percentage
Dataset			
Single hop-indoor	117	32	%72
Single hop -outdoor	32	3	%90
Multi hop -indoor	58	22	%62
Multi hop -outdoor	100	15	%85

Table 3. Accuracy comparsion

					Accuracy Comparison					
	Base Station Only					Node-Level Processing + Base Station				
Dataset	OCSVM	HBOS	IFORST	KNN	CBOLF	OCSVM	HBOS	IFORST	KNN	CBOLF
Single hop-indoor	92.64	98.98	92.66	95.72	92.69	90.72	92.67	90.72	95.38	90.74
Single hop-outdoor	90.60	97.72	90.58	94.10	90.56	90.02	91.93	89.98	94.50	90.00
Multi hop outdoor	90.64	97.87	90.43	95.33	90.43	89.94	90.89	89.86	95.16	86.60
Multi hop-indoor	92.13	99.59	92.15	95.33	92.15	90.31	90.71	90.31	94.70	90.31

Table 4. TPR comparison

					TPR Comparison					
	Base Station Only					Node-Level Processing + Base Station				
Dataset	OCSVM	HBOS	IFORST	KNN	CBOLF	OCSVM	HBOS	IFORST	KNN	CBOLF
Single hop-indoor	1.0	1.0	1.0	1.0	1.0	1.0	0.97	1.0	1.0	1.0
Single hop-outdoor	0.97	0.72	0.94	1.0	0.94	0.67	0.45	0.33	1.0	0.67
Multi hop outdoor	0.76	0.26	0.67	0.93	0.67	0.45	0.14	0.36	0.86	0.09
Multi hop-indoor	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Table 5. FPR comparison

					FPR Comparison					
	Base Station Only					Node-Level Processing + Base Station				
Dataset	OCSVM	HBOS	IFORST	KNN	CBOLF	OCSVM	HBOS	IFORST	KNN	CBOLF
Single hop-indoor	0.07	0.01	0.07	0.04	0.07	0.09	0.07	0.09	0.04	0.09
Single hop-outdoor	0.09	0.02	0.09	0.05	0.09	0.01	0.08	0.09	0.05	0.09
Multi hop outdoor	0.09	0.01	0.09	0.04	0.09	0.09	0.08	0.09	0.04	0.09
Multi hop-indoor	0.08	0.00	0.08	0.04	0.08	0.09	0.09	0.09	0.05	0.09

6.6 Results discussion

The primary objective of this study is to assess and compare the effectiveness of the proposed two-phase detection approach with the centralized method in identifying anomalies within WSNs. Specifically, the research endeavors to determine whether applying standard deviation method at the node level has a negligible impact on the performance machine learning methods, particularly assessing whether it does not significantly influence the outcomes of anomaly detection. Looking at the results, it is evident that during phase one, we successfully eliminated 62% to 90.62% of the outliers from the datasets. These numbers greatly reduce the amount of data being transmitted over the network and thus can result in energy saving, less communication overhead, and less transmission delay. Furthermore, the results indicate that the addition of standard deviation does not significantly impact the accuracy of most algorithms. Specifically, KNN displays minimal influence, with an average fluctuation of only 0.63% when standard deviation is applied. The accuracy of the other three algorithms (OCSVM, IFORST, CBOLF) is also affected only by an average of around 2%, and HBOS shows a fluctuation around 8%, on average. Therefore, we can conclude that it is safe to apply phase one with the most unsupervised machine learning algorithms and that applying standard deviation at node level does not affect the performance of machine learning algorithms by a significant amount. Other metrics results also confirm this finding. In both scenarios, where standard deviation was not applied at the node level and when it was applied, consistently low FPR values were obtained. The FPR ranged from 0.0% to 0.09% without applying standard deviation at the node level, and from 0.01% to 0.09% when it was applied at the node level. Furthermore, the results indicate that KNN consistently achieves the TPR of 1.0, irrespective of whether standard deviation is applied at the node level or not. The remaining algorithms also obtained high TPR values, mostly reaching or closely 1.0 in indoor data for both scenarios. However, an average reduction of 0.3 in TPR is observed for outdoor data after the implementation of standard deviation. Finally, MCC values exceeding 0.0 in both scenarios affirm a strong association between reality and prediction.

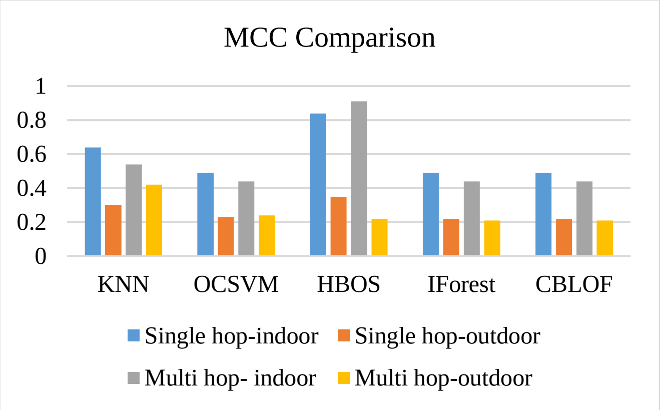

Figure 2. Base station only

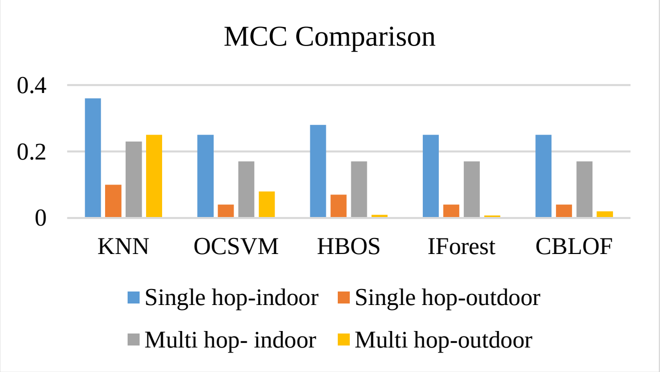

Figure 3. Node-level processing + base station

The observed performance variations in outlier detection algorithms like HBOS, when applied to datasets with fewer outliers, can be ascribed to algorithmic characteristics such as sensitivity to data distribution, threshold adjustments, adaptability to data variability, and changes in data density and statistical assumptions. In contrast, robust algorithms like KNN are less impacted because they depend on local neighborhood information. Machine learning algorithms such as SVM and IForest show intermediate sensitivity, adjusting their models based on training data but still influenced by distribution changes. Recommendations include careful algorithm selection, parameter tuning, ensemble approaches, and data preprocessing to improve outlier detection across varying outlier densities and dataset characteristics.

7. CONCLUSION

The proposed two-phase outlier detection approach in wireless sensor networks (WSNs) provides several significant benefits positively impacting network efficiency, energy consumption, and communication overhead. By minimizing the number of outliers transmitted through the network, the proposed method optimizes data transmission and processing, thereby improving overall network performance and resource utilization. This reduction in outliers also leads to lower energy consumption at both sensor nodes and the base station, contributing to extended network lifetime and decreases maintenance requirements. Additionally, the method lowers communication overhead by reducing outlier data transmission, which enhances bandwidth utilization, decreases network traffic, and improves data delivery reliability. The quantitative results of the method, with an average reduction of 77.59% in outliers during phase one and nearly 90% accuracy in phase two, illustrate its effectiveness in improving data quality, processing efficiency, and anomaly detection capabilities in WSNs. These results are particularly groundbreaking as they showcase a practical and scalable solution for real-world WSN deployments, paving the way for more reliable, energy-efficient, and robust sensor networks. Ultimately, these advancements contribute to the broader goal of creating more dependable and sustainable IoT ecosystems, where outlier detection plays a vital role in ensuring data integrity, system reliability, and operational efficiency.

REFERENCES

[1] Kocakulak, M., Butun, I. (2017). An overview of Wireless Sensor Networks towards internet of things. In 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1-6. https://doi.org/10.1109/CCWC.2017.7868374

[2] Bhadauria, D., Tekdas, O., Isler, V. (2011). Robotic data mules for collecting data over sparse sensor fields. Journal of Field Robotics, 28(3): 388-404. https://doi.org/10.1002/rob.20384

[3] Saravanan, G., Rangachar, M.J.S. (2018). Data mules-oriented particle swarm optimisation-based mobile sink data gathering techniques with analytical data analysis using linear regression. International Journal of Business Information Systems, 27(2): 193-204. https://doi.org/10.1504/IJBIS.2018.089111

[4] Matin, M.A., Islam, M.M. (2012). Overview of wireless sensor network. Wireless Sensor Networks-Technology and Protocols, 1(3). https://doi.org/10.5772/49376

[5] Hawkins, D.M. (1980). Identification of Outliers (Vol. 11). London: Chapman and Hall. https://doi.org/10.1007/978-94-015-3994-4

[6] Chatzigiannakis, V., Papavassiliou, S., Grammatikou, M., Maglaris, B. (2006). Hierarchical anomaly detection in distributed large-scale sensor networks. In 11th IEEE Symposium on Computers and Communications

(ISCC'06), pp. 761-767. https://doi.org/10.1109/ISCC.2006.1691116

[7] De Paola, A., Gaglio, S., Re, G.L., Milazzo, F., Ortolani, M. (2014). Adaptive distributed outlier detection for WSNs. IEEE Transactions on Cybernetics, 45(5): 902-913. https://doi.org/10.1109/TCYB.2014.2338611

[8] Saeedi Emadi, H., Mazinani, S.M. (2018). A novel anomaly detection algorithm using DBSCAN and SVM in wireless sensor networks. Wireless Personal Communications, 98: 2025-2035. https://doi.org/10.1007/s11277-017-4961-1

[9] Chandola, V., Banerjee, A., Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys (CSUR), 41(3): 1-58. https://doi.org/10.1145/1541880.1541882

[10] Zhang, Y., Meratnia, N., Havinga, P. (2010). Outlier detection techniques for wireless sensor networks: A survey. IEEE Communications Surveys & Tutorials, 12(2): 159-170. https://doi.org/10.1109/SURV.2010.021510.00088

[11] Ayadi, A., Ghorbel, O., Obeid, A.M., Abid, M. (2017). Outlier detection approaches for wireless sensor networks: A survey. Computer Networks, 129: 319-333. https://doi.org/10.1016/j.comnet.2017.10.007

[12] Chander, B., Kumaravelan, G. (2022). Outlier detection strategies for WSNs: A survey. Journal of King Saud University-Computer and Information Sciences, 34(8): 5684-5707. https://doi.org/10.1016/j.jksuci.2021.02.012

[13] Kumar, D.P., Amgoth, T., Annavarapu, C.S.R. (2019). Machine learning algorithms for wireless sensor networks: A survey. Information Fusion, 49: 1-25. https://doi.org/10.1016/j.inffus.2018.09.013

[14] Zhang, Y., Hamm, N.A., Meratnia, N., Stein, A., Van de Voort, M., Havinga, P.J. (2012). Statistics-based outlier detection for wireless sensor networks. International Journal of Geographical Information Science, 26(8): 1373-1392. https://doi.org/10.1080/13658816.2012.654493

[15] Rajasegarar, S., Leckie, C., Palaniswami, M., Bezdek, J.C. (2006). Distributed anomaly detection in wireless sensor networks. In 2006 10th IEEE Singapore International Conference on Communication Systems, pp. 1-5. https://doi.org/10.1109/ICCS.2006.301508

[16] Feng, Z., Fu, J., Du, D., Li, F., Sun, S. (2017). A new approach of anomaly detection in wireless sensor networks using support vector data description. International Journal of Distributed Sensor Networks, 13(1): 1550147716686161. https://doi.org/10.1177/1550147716686161

[17] Xie, M., Hu, J., Han, S., Chen, H.H. (2012). Scalable hypergrid k-NN-based online anomaly detection in wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems, 24(8): 1661-1670. https://doi.org/10.1109/TPDS.2012.261

[18] Da Silva, A.P.R., Martins, M.H., Rocha, B.P., Loureiro, A.A., Ruiz, L.B., Wong, H.C. (2005). Decentralized intrusion detection in wireless sensor networks. In Proceedings of the 1st ACM International Workshop on Quality of Service & Security in Wireless and Mobile Networks, pp. 16-23. https://doi.org/10.1145/1089761.1089765

[19] Wang, Z.M., Song, G.H., Gao, C. (2019). An isolation-based distributed outlier detection framework using nearest neighbor ensembles for wireless sensor networks. IEEE Access, 7: 96319-96333. https://doi.org/10.1109/ACCESS.2019.2929581

[20] Miao, X., Liu, Y., Zhao, H., Li, C. (2018). Distributed online one-class support vector machine for anomaly detection over networks. IEEE Transactions on Cybernetics, 49(4): 1475-1488. https://doi.org/10.1109/TCYB.2018.2804940

[21] Poornima, I.G.A., Paramasivan, B. (2020). Anomaly detection in wireless sensor network using machine learning algorithm. Computer Communications, 151: 331-337. https://doi.org/10.1016/j.comcom.2020.01.005

[22] Gupta, U., Bhattacharjee, V., Bishnu, P.S. (2021). Outlier detection in wireless sensor networks based on neighbourhood. Wireless Personal Communications, 116: 443-454. https://doi.org/10.1007/s11277-020-07722-3

[23] Thangaramya, K., Kulothungan, K., Indira Gandhi, S., Selvi, M., Santhosh Kumar, S.V.N., Arputharaj, K. (2020). Intelligent fuzzy rule-based approach with outlier detection for secured routing in WSN. Soft Computing, 24: 16483-16497. https://doi.org/10.1007/s00500-020-04955-z

[24] Zidi, S., Moulahi, T., Alaya, B. (2017). Fault detection in wireless sensor networks through SVM classifier. IEEE Sensors Journal, 18(1): 340-347. https://doi.org/10.1109/JSEN.2017.2771226

[25] Yu, X., Lu, H., Yang, X., Chen, Y., Song, H., Li, J., Shi, W. (2020). An adaptive method based on contextual anomaly detection in Internet of Things through wireless sensor networks. International Journal of Distributed Sensor Networks, 16(5): 1550147720920478. https://doi.org/10.1177/1550147720920478

[26] Schölkopf, B., Williamson, R.C., Smola, A., Shawe-Taylor, J., Platt, J. (1999). Support vector method for novelty detection. Advances in Neural Information Processing Systems, 12.

[27] Shin, H.J., Eom, D.H., Kim, S.S. (2005). One-class support vector machines—An application in machine fault detection and classification. Computers & Industrial Engineering, 48(2): 395-408. https://doi.org/10.1016/j.cie.2005.01.009

[28] Goldstein, M., Dengel, A. (2012). Histogram-based outlier score (HBOS): A fast unsupervised anomaly detection algorithm. KI-2012: Poster and Demo Track, 1: 59-63.

[29] Liu, F.T., Ting, K.M., Zhou, Z.H. (2008). Isolation forest. In 2008 Eighth IEEE International Conference on Data Mining, pp. 413-422. https://doi.org/10.1109/ICDM.2008.17

[30] Angiulli, F., Pizzuti, C. (2002). Fast outlier detection in high dimensional spaces. In European Conference on Principles of Data Mining and Knowledge Discovery, pp. 15-27. https://doi.org/10.1007/3-540-45681-3_2

[31] He, Z., Xu, X., Deng, S. (2003). Discovering cluster-based local outliers. Pattern Recognition Letters, 24(9-10): 1641-1650. https://doi.org/10.1016/S0167-8655(03)00003-5

[32] Suthaharan, S., Alzahrani, M., Rajasegarar, S., Leckie, C., Palaniswami, M. (2010). Labelled data collection for anomaly detection in wireless sensor networks. In 2010 Sixth International Conference on Intelligent Sensors, Sensor Networks and Information Processing, pp. 269-274. https://doi.org/10.1109/ISSNIP.2010.5706782

[33] Zhao, Y., Nasrullah, Z., Li, Z. (2019). Pyod: A python toolbox for scalable outlier detection. arXiv preprint arXiv:1901.01588