Science World Journal Vol. 19(No 2) 2024 https://dx.doi.org/10.4314/swj.v19i2.26
www.scienceworldjournal.org
ISSN: 1597-6343 (Online), ISSN: 2756-391X (Print)
Published by Faculty of Science, Kaduna State University

Review Article

A REVIEW OF DATA ANALYTIC ALGORITHMS FOR OUTLIER DETECTION ON THE INTERNET OF THINGS ECOSYSTEM

[1]Iwomi Onyemaechi Joel, *[1]Edje E. Abel, [1]Omede Gracious, [1]Atonuje Ephraim, [1]Ogeh Clement, [1]Akazue I. Maureen, [2]Apanapudor Joshua Sarduana

[1]Department of Computer Science, Delta State University, Abraka
[2]Department of Mathematics, Delta State University, Abraka

*Corresponding Author Email Address: edjeabel@delsu.edu.ng

ABSTRACT
In the last few years, outlier detection has drawn a lot of attention. New technologies, including the Internet of Things (IoT), are recognized as one of the most important sources of data streams, continuously producing enormous amounts of data from several applications. Reducing functional hazards and avoiding hidden problems that result in application downtime can be achieved by looking through this gathered data to identify suspicious events. This paper presents a review of existing algorithms deployed on Internet of things ecosystem that resolved the challenges of data outliers. It further highlights the problems solved, the results and the weaknesses of the existing algorithms. Also, presented a detailed discussion on various programming language and simulation tools adopted to implement and conduct experiment on the prevailing algorithms; as well as metrics used to evaluate their performances. It was discovered that metrics such as accuracy, precision, recall, specificity are mostly adopted as metrics for performance evaluation of the algorithms. Additionally, python programming language and Microsoft Studio IDE simulation tools were mostly used for the development and test-running of the existing algorithms.

Keywords: Outliers, Internet of Things, Simulation Tools, Clustering/Classification process

INTRODUCTION
Due to the rapid development of Internet of Things (IoTs) applications and the advancement of communication technology, IoT applications such as smart cities, smart power grids, and smart homes are now widely employed globally. The number of devices linked to IoTs is expected to increase from 20.35 billion in 2017 to 74.33 billion in 2025, predicts Statista (Sikder and Batarseh, 2023). With the rise in popularity of IoT in recent years, several sensor devices have been widely used in a variety of industries, including the biomedical, financial, and chemical sectors. In addition to altering people's lifestyles, these sensor devices produce a large amount of time series data (Edje and Ekabua, 2015).
In IoTs settings, the sensors send the gathered data to the edge nodes' data receivers after monitoring the surrounding data at a specific frequency. In addition to precisely reflecting changes in external conditions in real time, a substantial amount of IoT data also explains the evolution and change patterns of the status of the network over a specific period of time. As a result, the sensor data gathered from a vast array of sensor devices serves as the foundation and starting point for additional data mining in addition to being a vital data source for real-time data visualization monitoring.
Finding data from the sample set whose behavior features differ unusually from those of other samples is known as Outlier detection. Specific occurrences in the node region (such an increased temperature sensor reading in the event of a fire), sensor failure, and external variables are among the causes of anomalous data (Edje *et al.,* 2021). Abnormal event identification, one of the primary responsibilities of the IoT system's state monitoring, has progressively drawn the attention of academics, and various research accomplishments have been made in this area. Many wireless sensors are placed in locations with constrained bandwidth and energy usage to meet the requirements of the Internet of Things system (Edje and Ekabua, 2015). The Internet of Things' extensive sensor node deployment enables cooperative environmental monitoring (Fahim and Sillitti, 2019).
IoT sensing devices provide vast amounts of dynamic, heterogeneous data. Another common problem is the speed at which unstructured and semi-structured data are created. IoT sensed data has four main characteristics: Sensing data inaccuracy, multisource high heterogeneity, weak semantic data with low-level and massive data dynamicity (Edje *et al.,* 2023). Sensing data inaccuracy refers to the information obtained from Internet of Things (IoT) sensing devices because of a number of restrictions, including erratic readings that cause data anomalies. This adds to the difficulty of directly applying the detected data to its intended use.
The related research survey conducted by previous researches mainly focuses on the components and architecture of the IoT ecosystem, network protocols and other integrated technologies. For example, Fahim and Sillitti (2019) presented a literature regarding anomaly detection methods, with the exception of these prevailing fields of study. It focuses on studies based in application areas such as smart items, industrial systems, transportation systems, health care systems, and intelligent living environments. Also, highlights many research gaps or challenges in the areas of data generation, large-scale unbalanced dataset analysis, statistical method processing issues, and the small number of studies on abnormal behavior prediction in real-world settings. Gaddam et al., (2020), investigated the complexities in identifying abnormalities, outliers, and faulty sensors in the Internet of Things. Also, presented a thorough analysis on set guidelines for selecting appropriate outlier detection model for sensors in an Internet of things setting. Al-amri *et al.,* (2021) investigated various cutting-edge approaches used to address the main issues and central problems with IoT data. Additionally, included are the datasets, assessment criteria, learning mode, window model, anomaly categories, and data nature (Apanapudor *et al.,* 2020). The investigation focuses on research issues associated with data evolving, feature-evolving, windowing, ensemble techniques, parameter selection, data visualizations, heterogeneity of data,

correctness, and large-scale and high-dimensional data. Alghanmi *et al.,* (2021) examine and evaluate the relevant literature on current anomaly detection algorithms that use various machine learning strategies in the Internet of Things. Furthermore, it analyzes various IoT-related anomaly detection datasets, point out the most prevailing problems for various methodologies, and outline a number of potential future research directions (Apanapudor *et al.,* 2023). Chatterjee *et al.,* (2022), highlight and discusses the detection techniques and applications, followed by how IoT anomaly detection systems are categorized. After that, the most recent articles to identify specific application fields and the current issues surrounding IoT ecosystem. Sikder and Batarseh (2023) analyzes the development of Outlier techniques with the support of artificial intelligence. Also presenting the most recent state-of-the-art techniques, their uses, and their results. Next, a concise analysis of the benefits, limitations, and difficulties associated with each technique.

The aforementioned existing researches demonstrated a thorough analysis on outlier detection techniques in IoT and highlighted prevailing challenges that cloud lead to future research directions. However, the simulation and programming language tools as well as performance metrics used for the development of the existing techniques have not been addressed in existing related researches. Therefore, the contributions of this research is highlighted as follows;

- To examine and analyze various machine learning and neural network techniques used for outlier detection in IoT ecosystem. Highlighting the challenges resolved, outcomes, benchmark models and limitations of the algorithms in a tabular form.
- Identify and discusses diverse programming languages and simulation tools adopted to implement and conduct the experimentation of the detection techniques.
- Highlight and discuss different types of metrics adopted to evaluates the performance of the existing detection techniques.

The rest of this article comprises of the research methodology deployed to actualize the current research objectives, followed by result analyses that presents a detailed discussion of the outcomes based on the itemized current research contributions and give an overview of the entire research processes based on the research topic and its benefits to the general public and ends with a concluding remark.

MATERIALS AND METHODS

The research survey understudy was carried out with the approach adopted from the literature review conducted by Edje *et al.,* (2023). Using academic research databases, the exploration of literature contributions spanning the years 2016 to 2023 was determined to be the most pertinent to achieving the goals of the current study. Included in this database are ScienceDirect, IEEE Xplore, Google Scholar, Springer, and Scopus. Articles pertinent to the current study were found using the search terms "internet of things data" OR "mining algorithm" OR "edge" and "storage resource provisioning" OR "IoT data" OR "cloud data center." Nevertheless, a large number of research publications that were unimportant to the study were returned by the search query.

The related articles that were not found in the list of results that were referred were expected to be among them and were included in the subsequent analysis round. Only English-language research articles that were published in journals and conference proceedings were taken into consideration. There were 209 items returned in the first search result. Before being chosen, each piece goes through a series of quality assessment stages. Four sequences make up these phases, and the following are the ones that are highlighted:

- Evaluating the title and removing it if it doesn't conform to the algorithms used in IoT ecosystem (current study).
- Read the abstract, and if it has nothing to do with the current investigation, discard it.
- Examine the opening and conclusion; discard if the contribution is redundant with those of other pertinent papers.
- Analyze the quality of the research contribution analytically and reject publications that fall below a certain minimum standard.

The degree of relevance of the accepted publications to the ongoing research was taken into account. In addition, consideration was given to the articles' writing quality, soundness, clarity, and authenticity of the contributions they actualized. A total of 85 publications that are relevant to the current research objective are evaluated for quality. These 84 articles go through an extraction process in order to extract the information needed to fulfill the research study's objectives. The necessary details are indicated below.

- The algorithms used to predict outlier data in IoT infrastructure.
- Simulation and programming language tools adopted for implementation and experimentation of the algorithms.
- Strengths and weaknesses of each algorithm.
- The metrics each algorithm uses for performance evaluation.

Out of the extraction process, a total of 26 desirable candidate articles were found to be relevant to the current study. A summary of the bibliometric data, comprising 19 journal articles and 1 conference paper. The remaining six papers, which are judged appropriate for usage as related research works in this topic are analyzed in the previous section of this study. Lastly, a qualitative analysis is performed on 20 articles in order to summarize the results.

RESULTS ANALYSIS

This section presents a detailed analysis of the various existing techniques or algorithms deployed for the detection to outliers in IoT ecosystem. It elaborated mostly on the functionalities and processes involved in the prediction of outliers by the algorithms. Furthermore, the challenges solved, the outcomes, weaknesses and the benchmark models used to validate the performance existing algorithms are highlighted in a tabular form.

Analysis of Existing Outlier Technique in IoT

For the segmentation of a time sequence data stream with change in real-time processing, a Dynamic Symbolic Aggregation Approximation (D-SAX) is intended for both adaptive and non-adaptive window sizes (Kolozali et al., 2016). It creates a string representation for each segment after dividing time sequence sensing data into comparable segments. Before being converted to a piecewise aggregation approximation (PAA), the time sequence data first go through a normalization step to obtain standard deviation and mean (average). In order to decrease the quantity of the data, the data is additionally split into the required

number of windows and the mean average of the data dropped in each window is calculated using the PAA. Next, in order to identify the equal-size area for the retrieval of symbolic data representation, a discretization procedure is carried out on the PAA coefficients (each window size) by mapping the PAA coefficients to breakpoints produced by the alphabet size (e.g. c).

An Adaptive K-means Clustering (AKC) algorithm was created by Puschmann et al. (2017) in order to create dynamic clusters and allocate the sensed data to these clusters according to their similarity features. It evaluates the dynamic data and modifies the cluster centroids in accordance with changes in the data stream within a specified time frame. By looking at the distribution of different features that emerge from the streaming data, one can determine how many clusters are obtained during a specific period of time. As a result, the similarity properties of streaming data are used to create clusters. Consequently, alterations in data features or attributes lead to the formation of a new cluster or clusters. For example, the first cluster will receive the Temp features if an incoming streaming data set has the feature (Temp, Temp, Temp, Hum, and Hum…). A new cluster containing the Hum feature data will emerge in response to the introduction of the "Hum".

A Weighted Component Human Activity Recognition (WHAR) classifier algorithm was suggested by Santamaria et al. (2018) to identify the normal and abnormal activities of the patients under observation. It starts a few constants that are used to indicate how many clusters there are. Selected are the initial membership matrix with specified threshold values and the weighted component (fuzzier). As each cluster member receives a data point, the weighted component controls class overlaps. In order to get outliers and normal data, the threshold value is also used to assess the convergence of the classification process iterations. According to Verma et al. (2018), a Bayesian Belief Network (BBN) algorithm is introduced for the classification of sensory data. It divides the obtained data into two categories: abnormal and normal. The abnormal event class's data suggests that the patient's health is in a serious or critical condition. The Naïve Bayes classification approach is used to actualize the classification process. The chance of all sample data falling into the range of the predefined normal value is classed as a normal event class when a predetermined value is set as the normal value. Conversely, the abnormal event class is identified when the sampled data value's probability surpasses that of the normal event class.

In order to identify anomalies in the multimodal distribution of sensing data retrieved from end nodes, an Ellipsoidal Neighborhood Outlier Factor (ENOF) is introduced (Lyu et al., 2017). Initially, a set of hyperellipsoidal clusters is obtained by using ENOF to identify the ellipsoids drifting relatively from their proximity neighborhood to densities the neighborhood. Therefore, the outlier score level is determined by dividing the average neighborhood range density of neighbors by the neighborhood range of the ellipsoids. To identify the anomalous clusters, a threshold is therefore computed using a parameter and the standard deviation of the ENOF score. As a result, clusters that have an ENOF score above the cutoff are considered anomalous.

Based on sensed data received from sensor devices, a Linear Prediction Spectrum (LPS) method is presented for the detection of voice disorder (Ali et al., 2017). It divides the vocal track into distinct tubes from the glottis to the lips in order to discern between disordered and normal voices by analyzing the energy variation of the spectrum. After that, it uses inverse filtering to do an estimated analysis on the source signal, which forces the computation of the spectrum by utilizing the estimated source signal to ascertain the energy distribution in vowels and running speech to identify voice disorders (outliers).

In order to accurately classify desired aspects of sensed data, Raafat et al. (2017) suggested a Feed-forward Neural Networks (FFNN) algorithm based on Homoscedasticity Measurement Leven's Test (HMLT). It is used to extract different features by looking for abrupt shifts in the de-noised signal. To categorize the sensed data into abnormal and normal data, the collected features are thus fed into the FFNN. Sending the data from its input layer to the concealed layer makes this a reality. The activation function over the sum of the input characteristics multiplied by a set of weights parameters is computed by the neurons in the hidden layers. so producing the findings as felt data that is either normal or pathological.

A Dynamic Probabilistic Clustering (GDPC) approach based on Gaussian distribution was able to solve challenge such as inconsistency is data clustering (Diaz-Rozo et al., 2018). It makes an estimate of the drifts in the sensor data as well as the model parameters. With the aid of the Brier Score method, it additionally offers the membership likelihood of each instance (data) to each cluster. The irregularity of subsequent probabilities from those cases (sensed data) that are expected is identified using the Brier score. When the sensed data value parameter is higher than the Brier score's predetermined threshold value, drifts or changes are identified. These drifts are recognized as anomalies. The Brier score modifies its behavior and stability for incoming sensor data once drifts have been identified.

Nesa et al. (2018) offer the Non-Parametric Sequence-Based Learning (N-PSL) algorithm. Consequently, for the prediction of outliers based on abrupt changes (event) in sensor readings and error sensed data recovered from malfunctioning sensing devices. It takes into account the use of perception data, which is crucial in an IoT platform for self-check outlier detection brought about by errors and events from malfunctioning sensor nodes. Grey relational analysis serves as the foundation for the non-parametric sequence learning technique. Initially, the average image of each sensed data point is computed in order to normalize the detected data. The relative mass function in each class is obtained by computing the difference between each instance of the sequence image sensed data and then calculating the IRG coefficients for each sequence (class) sensed data. As a result, classes with lower values for the outlier and greater values for the inlier are expected to comprise the outlier.

Box plot Adjustment K-Nearest-Neighbor technique is proposed for the detection of outliers in multidimensional Dataset (Rehman and Belhaouri, 2021). It converts the dataset into a unidimensional isolated space in other to identify the outliers in new space. The distance vector is calculated, considering all dimension while measuring the distance between data records. Thus, minimizes the presence of noisy data points and computational cost based on time efficiency and memory space usage. Consequently, Yang *et al.* (2021) developed a Mean-shift technique for the prediction of outliers. It substitutes every data points using its k-nearest neighbors which discards the presence of outliers before clustering without having knowledge of the outliers. Hence, outliers are identified based on the distance shifted. Ijaz *et al*., (2020) proposed an outlier detection model for the prediction of outliers in IoT infrastructure. It utilizes the combination of Density-based Support Clustering and Isolation forest algorithms for the removal of noisy data records and retrieval of relevant datasets. Thus, the Random

Forest classifier is deployed to train relevant datasets in other to identify actual outliers.

A technique based on Compressed Sensing and Online Extreme Learning Machine Auto-encoder (COELMAE) is developed for the identification and removal of outliers from datasets generated from IoT sensors (Yu *et al.,* 2020). At the initial stage, historical dataset is trained using the extreme learning machine auto-encoder to obtain the initial output. The datasets generated continuously from Online are compressed to reduce the amount of datasets. Noisy data are also removed during the compression process and retains the relevant data features using the sparse transformation method. Then, the reconstruction algorithm SP is applied to reconstruct the compressed dataset and inserted the original dataset into initialized online extreme learning machine auto-encoder to obtain the output dataset. The reconstruction error for the output dataset or values is computed and if the reconstructed value is greater than it is classified as outlier else it is regarded as normal data.

Conversely, Zhu et al., (2020) proposed a Grid-based Approximate Average Outlier Detection (GAAOD) technique for the identification of outliers in IoT sensed datasets. At the initial stage, it applied a grid-based index to remove noisy data records from streaming dataset as it self-adapt to incoming data records. Then, a Min-Heap-based method is applied to calculate distance upper/lower-bound between records and their K^{th} nearest neighbors respectively. Thereafter, the K-skyband algorithm is used to identify candidate outliers. Furthermore, Xu et al., (2023) proposed Synthetic Minority Oversampling (SMOTE) technique for the detection of outliers on sensed dataset. It initially preprocessed the dataset to remove excessive noisy and redundant records, retaining relevant datasets which are trained with the support of Auto-tuned hyper-parameters algorithm to retrieve outliers.

A Compressed-enabled Isolation Forest technique is proposed for the prediction of outliers in sensed data (Liu et al., 2021). Firstly, the sensory data are compressed by calculating the reconstruction error threshold on each data object in the dataset. When the average value of the data object is greater than the error threshold, it is regarded as irrelevant data record and discarded. But if the average value of the data object is less than the error threshold it is regarded as relevant data records. The relevant data records are regrouped together as the compressed dataset which are trained with Isolation forest algorithm to identify the expected outliers.

A Hierarchical Clustering-enabled Long Short-Term Memory (LSTM) Neural Network technique is proposed for the detection of outliers in sensed data (Shukla *et al.*, 2020). Initially, each data point in the Hierarchical clustering approach is treated as a separate cluster. It then goes through the two processes again. It first locates the two clusters that are adjacent to one another. It then combines the clusters to create a single, large cluster. Until the complete data set is gathered in a single cluster, the procedure is repeated. Therefore, the number of clusters is based on the fluctuation in distance between two clusters that are merged. Then, the formulated clusters containing data records are trained with the LSTM Neural network approach for the prediction of outliers.

The development of Isolation Forest (IF)-enabled Local Outlier Factor(LOF) technique is used for the prediction of outliers in datasets generated by IoT sensors (Alsini *et al.,* 2020). The IF approach estimates the isolation score for all data records and processes the data points into recurrent random splits that are dependent on feature selections. The outlier score for each data record in the tree or cluster is then calculated by setting a range between the maximum and minimum values based on the selected value, to determine the path length needed to isolate the outlier. Then the LOF approach is applied to predict the actual outliers. When evaluating data points, LOF takes into account the outlier factor in relation to the density of the nearby neighbors, or a degree of measurement.

An abnormal and identification techniques were combined together for the detection of outliers in IoT infrastructure (Shao and Chen, 2022). The sensed dataset was normalized using Z-score technique after which it was de-noised using the Gaussian Smoothing Filtering algorithm. Then, distance matrix between the data points is calculated using the composite temporal series similarity measurement criteria, and any potential abnormality data is then detected using the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm. The geometry of the spatial cross-correlation coefficient of the terminal nodes collected as the input of the cascaded fuzzy logic system is then used to identify the actual abnormal data.

A Deep-Variational Auto-Encoder (VAE) technique is developed for the detection of outliers in the IoT infrastructure (Gouda *et al.,* 2022). The generated sensed dataset was initially preprocessed by discarding the mean and scaling to unit standard. Then, from the low-dimensional representation of the latent variables in the input data, the VAE is used to identify a reconstructed output representation. After which, an outlier score is derived from the reconstruction error between the original and reconstructed observations. Table 1 depicts a summary overview of the existing algorithms, highlighting the challenges solved, outcomes, weaknesses and the benchmark models deployed for the performance validation of the existing algorithms

Table 1: Summary overview of the Existing Algorithms or Techniques

Article Title	Technique	Problem Solved	Achievement	Simulation Tools	Metrics	Benchmark	Weakness
Unsupervised outlier detection in multidimensional data (Rehman and Belhaouari, 2021)	Boxplot Adjustment K-Nearest Neighbor	Noisy data points and Computational complexity	Improved detection rates with minimum computational cost	Python	Area Under Curve (AUC), Detection Rates	Local Outlier Factor (LOF), K-nearest Neighbor	Computationally intensive
Mean-shift outlier detection and filtering (Yang et al., 2021)	Mean-Shift	Data records that deviates significantly	Improved outlier detection rate with minimum computationa	Python 3.7	Area Under Curve, Detection Rates,	Isolation Forest, Local Outlier Factor, One	Caught in local optima search space entrapment

			l cost			Class Support Vector and Angular-based Outlier Detection (ABOD)	
Data-Driven Cervical Cancer Prediction Model with Outlier Detection and Over-Sampling Methods (Ijaz et al., 2020)	Isolation Forest, Density-based Support Clustering and Random Forest.	Prevalent noisy data objects and disparity between records.	Improved accurate outlier detection Rates	Python	Accuracy, Precision, F1-score, Recall and Specificity	Support Vector, K-nearest Neighbor, Naïve Bayes	Ineffective on over sampling of dataset.
On the Effect of Adaptive and Non-adaptive Analysis of Time-Series Sensory Data (Kolozali, 2016)	Dynamic Symbolic Aggregation Approximation (DSAX)	The complexity of aggregating massive sensory data retrieved from various sources	Achieved optimal data aggregation quality for prediction of error data.	MATLAB	Sensitivity, Reconstruction Rate,	Not Specified	Unable to give insight knowledge about the sensing data retrieved regarding drifts and consistent data.
Adaptive Clustering for Dynamic IoT Data Streams (Puschmann et al., 2017)	Adaptive K-Means Clustering algorithm	Deficiency in clustering streaming sensed data.	Improved accuracy based on Silhouette coefficient.	Python	Accuracy and Execution Time	Not Specified	Inability to overcome greater drifts of objects within dataset.

Table 1: Continue

Article Title	**Technique**	**Problem Solved**	**Achievement**	**Simulation Tools**	**Metrics**	**Benchmark**	**Weakness**
A real IoT device deployment for e-Health applications under lightweight communication protocols, activity classifier and edge data filtering (Santamaria et al., 2018)	Fuzzy-based Human Activity Recognition classifier algorithm	The complexity of data overlapping in massive sensed data.	Improved accuracy detection of outliers with minimum computation resources.	MATLAB	Accuracy and Average Serving Time	Not Specified	Loss of relevant features during removal of noisy objects
Anomaly Detection for Internet of Things Based on Compressed Sensing and Online Extreme Learning Machine Autoencoder (Yu et al., 2020)	Compressed Sensing and Online Extreme Learning Machine Auto-encoder techniques	Unlabeled datasets and high computational cost	Improved detection rate with minimum computational cost	Python	Accuracy, Precision, Recall and F1-score	Isolation Forest	Not Specified
KNN-Based Approximate Outlier Detection Algorithm Over IoT Streaming Data (Zhu et al., 2020)	Grid-based Approximate Average Outlier Detection technique	High Computational Complexity	Improved detection rates with minimum execution time.	Python	Memory usage amount and Execution time	K-nearest Neighbor	Computationally intensive

A data-driven approach for intrusion and anomaly detection using automated machine learning for the Internet of Things (Xu et al., 2023)	Synthetic Minority Oversampling technique	Inaccurate multi-classification of outliers	Improved classification accuracy	MATLAB	Accuracy, Recall and F1-score	Isolation Forest and Linear Principal Component	Computational Intensive.
Sensors Anomaly Detection of Industrial Internet of Things Based on Isolated Forest Algorithm and Data Compression (Liu et al., 2021)	Compressed-based Isolation Forest technique	Inaccurate classification of outliers	Improved accurate classification of outliers with minimum time execution	Pyhton	Accuracy, Precision, Recall and Execution Time	K-means	Time consuming

Table 1: Continue

Article Title	**Technique**	**Problem Solved**	**Achievement**	**Simulation Tools**	**Metrics**	**Benchmark**	**Weakness**
Fog Assisted-IoT Enabled Patient Health Monitoring in Smart Homes (Verma, 2018)	Bayesian Belief Network algorithm	Delay in the classification of sensed data acquisition	Improved accuracy of classifying dataset with less time.	Python	Accuracy, Execution time, Recall, Precision F-measure and ROC curve	Not Specified	Not considering the spatio-temporal correlations among sensed data set
Fog-Empowered Anomaly Detection in IoT Using Hyperellipsoidal Clustering (Lyu et al., 2017)	Hyperellipsoidal clustering algorithm	The Issue of high latency and energy consumption	Reduction in energy consumption and latency while improving anomaly prediction accuracy.	MATLAB	Accuracy	Not Specified	There is need for further improvement on latency due to increase usage of computation resource
An Automatic Health Monitoring System for Patients Suffering from Voice Complications in Smart Cities (Ali et al., 2017)	Linear Prediction Spectrum algorithm	Ineffective detection of voice disorder of patients	Efficient detection of voice disorder with sustained vowel and running speech based on enhanced accuracy.	MATLAB	Accuracy	Not Specified	Not Specified
Scalable and Robust Outlier Detector using Hierarchical Clustering and Long Short-Term Memory (LSTM) Neural Network for the Internet of Things (Shukla et al., 2020)	Hierarchical Clustering and Long Short-Term Memory (LSTM) Neural Network Approaches	Fluctuation distance within data records	Improved outlier detection with a better accuracy.	Python	Precision, Recall, F-measure	Not Specified	Limited in detecting outliers within data records
Improving the outlier detection method in concrete mix design by combining the isolation forest and	Isolation Forest-Local Outlier Factor Technique	Inability to ascertain actual outliers in dataset	Improved detection rates with minimal execution time	Java	Accuracy, Execution time and AUC	Local Outlier Factor	It is flawed regarding to the flow of sequences of an outlier.

local outlier factor (Alsini et al., 2020)							

Table 1: Continue

Article Title	Technique	Problem Solved	Achievement	Simulation Tools	Metrics	Benchmark	Weakness
Fog Intelligence for Real-Time IoT Sensor Data Analytics (Raafat et al., 2017)	Homoscedasticity measurement Leven's Test Feed-forward Neural Networks algorithm	Improper selection of threshold values that leads to partial classification.	Improved classification accuracy.	Java	Accuracy, F1-score, Precision, Specificity, Sensitivity and Sensitivity	KNN, Random Forest, Decision Tree	Duplicate sensed data and highly computation intensive.
Clustering of Data Streams with Dynamic Gaussian Mixture Models. An IoT Application in Industrial Processes (Diaz-Rozo et al, 2018)	Gaussian-based Dynamic Probabilistic algorithm	Inefficiency in clustering dynamic sensing data to detect drifts sensed data.	Improves drifts detection accuracy to the tune of 98.7% and sensitivity of 96% indicating that almost all detection are true positives.	MATLAB	Recall, Specificity, F-score and Accuracy	KNN	There are about twice the amount of Instances detected as turning points for concept drift
Non-parametric sequence-based Learning approach for outlier detection in IoT (Nesa et al., 2018)	Non-parametric sequence learning algorithm	Problem of self-check identification using perception for error/ event outliers detection	Enhances classification accuracy with optimal detection of error/event outlier	Python	Accuracy, Specificity, Precision, Sensitivity and Recall	Support Vector Machine, Linear Discriminant Analysis, Classification and Regression Trees	Difficult to detect outliers in global space as the dataset increases is size
Abnormal Data Detection and Identification of Distribution IoT to Monitor Terminal on Spatiotemporal Correlation (Shao and Chen, 2022)	DBSCAN algorithm and Cascaded Fuzzy Logic	Self-check and monitoring the low-voltage terminal unit.	Improved abnormal data detection rate.	Python	FI-score, Recall and Precision	Local Area Factor (LOF), One-Class SVM	Not Specified
Unsupervised Outlier Detection in IOT Using Deep VAE (Gouda et al., 2022)	deep Variational Auto-Encoder (VAE) technique	Inaccurate detection of outlier in large volume of dataset	Improved detection rate of outliers in large dataset volume	Python Jupyter Notebook on Ubuntu	Precision, Recall, and (ROC)	Not Applicable	Under-performance and accuracy of unlabeled dataset.

Simulation Environment and Programming Language Adopted

This section discusses various simulation and programming language tools that were adopted for the implementation and experimentation of the existing algorithm to aid performance evaluation and validation processes.

It was also discovered that different programming languages and simulation tools are used in the implementation and simulation of the existing techniques. The majority of the time, simulation technologies like Weka, OmNet++, Contiki Cooja, Visual Basic.Net, and CloudSim were used to obtain verified performance evaluations of the models that were already in place. On the other hand, the present models were implemented mostly using programming languages including Python, Java, MATLAB, C, and R-Studio. Weka comes with a variety of graphical user interfaces, algorithms, and visualization tools for use in data analysis and predictive modeling. Numerous typical data mining tasks, such as feature selection, data preparation, clustering, regression, and classification, can be managed by it. Weka requires that the input be formatted in the Attribute-Relational File Format and have a

name ending in "arff.".
The Objective Modular Network Testbed in C++, or OmNet++, is primarily used to build network simulators. It has also been used recently to simulate data mining procedures. OMNeT++ is freely available for use in non-commercial simulations, such as those run by academic institutions and in educational contexts.
In COOJA, a simulated Contiki is a real, compiled, and operational Contiki system. COOJA oversees and manages the system. Different Contiki libraries can be built and loaded to simulate different kinds of sensor nodes (heterogeneous networks) within the same COOJA simulation. COOJA uses a few functions to operate and evaluate a Contiki system. For instance, the simulator accesses all of the Contiki system's memory for analysis or provides instructions to the system on how to react to an event.
The.NET framework is required for Visual Basic.NET to function, and the language produces highly scalable and reliable programs. With VB.NET, you may create fully object-oriented programs that are equivalent to those created in other languages like C++, Java, or C#. Programs made with VB.NET can also be used with applications developed in Visual C++, Visual C#, and Visual J#. In VB.NET, everything is handled as an object.
The infrastructure and services of cloud computing are modeled using an open-source framework called CloudSim. It was made by the CLOUDS Lab team and is entirely written in Java. It is used to model and simulate a cloud computing system in order to test a hypothesis prior to developing software and reproduce tests and outcomes.
Python is a programming language that may be used to develop software, generate websites, automate tasks, and analyze and visualize data. The models that are now in use can also be implemented using the Java programming language. Without having to write in numerical codes, programmers can construct computer instructions with Java by employing commands that are based in English. A programming language called R-studio is used for statistical analysis and data visualization. It has been embraced by the fields of data mining, bioinformatics, and data analysis. The R language comes with a ton of extension packages that include reusable code, example data, and documentation. The abbreviation MATLAB stands for "Matrix Laboratory." It is a fourth-generation programming language. MATLAB is multi-paradigm. As such, it is compatible with several programming paradigms, such as object-oriented, functional, and visual.

Performance Metric Adopted

There are different types of performance metrics used to evaluate the performance of the existing algorithms. The most prevailing adopted metrics include Accuracy, Recall, Sensitivity, Precision, Area Under Curve, Specificity, FI-score, Root Mean Squared Error and Execution time.

Accuracy is the degree to which the measured value resembles the true or standard value. This is done by dividing the entire number of correct predictions by the total number of datasets. It's not always the most trustworthy, though, which is why data scientists create confusion matrices and employ metrics like recall and precision in its place. Therefore, **Recall** is computed by dividing the total number of false negatives and true positives by the number of true positives. The number of accurate positive predictions divided by the total number of positive predictions is how **Precision (PREC)** is computed. Positive predictive value (PPV) is another name for it. The number of accurate positive predictions divided by the total number of positives is how **Sensitivity (SN)** is computed. It is also known as true positive rate (TPR) or recall (REC). Consequently, the number of accurate negative predictions divided by the total number of negatives is how **Specificity (SP)** is computed. Another name for it is true negative rate (TNR). The harmonic mean of recall and precision is regarded as **FI-score,** while **Area Under Curve (AUC)** is limited to evaluating classifiers that provide a probability or confidence score for the prediction. **Root Mean Square Error** calculates the mean difference between the values that a model predicts and the actual values. It offers an estimate of the accuracy or how effectively the model can anticipate the desired result. A model is considered better if its Root Mean Squared Error value is smaller.
Execution time is the stage at which the instructions in computer programs or code are carried out is referred to as execution time. Libraries that are used during run-time are used. Reading program instructions to carry out tasks or complete activities is one of the fundamental operations that take place throughout execution time.

DISCUSSION

According to the study, machine learning algorithms are mostly used to identify abnormalities in the IoT ecosystem. In terms of IoT, anomalies or outliers are the predictions of irrelevant dataset generated from IoT devices such as sensors and wireless sensor network. With a short execution time, the majority of the algorithms used for this purpose produced notable performance outcomes based on accuracy, precision, recall, and specificity e.t c, as discussed extensively in previous section of this article. This suggests that regardless of the size (big data), dynamicity and velocity of the datasets, machine learning algorithms are dependable and effective for preprocessing and processing dataset generated by IoT devices.
To select or extract relevant features for classification to anticipate actual anomalies or outliers, the majority of these algorithms uses the clustering process. Programming languages are used mostly in simulation environments to conduct the studies. This may allow for more study in the future by carrying out the experiment in a real-time setting. Therefore, pave the way to deploy algorithmic functionalities to predict abnormalities in the health status of patients. It will assist medical professionals in better diagnosing illnesses and symptoms, which will enable patients to receive treatments more quickly. Additionally, based on the anticipated outcome, prospective patients will receive the necessary therapies based on their current state of health. Unauthorized users of company's web application and network system can easily be detected with the support of the existing algorithms. Furthermore, possible events such as gas leakages from industrial gas pipes transporting gas to various locations across the globe can be detected and providing solution to stop the leakage real time. Also the prediction of two or more vehicles tending to collide while in motion, and averting such disastrous accident from occurring. Thereby preventing sudden death and fatal accident as a result of the collision.
Based on the benefits of the current study as discussed above, its contribution to knowledge as compared to other literature review work, comprise of a detailed discussion of various techniques adopted for the prediction of outliers on dataset generated in IoT ecosystem. Also, a brief discussion on the types of simulation and implementation tools used to conduct experimentation on various techniques deployed for the prediction of outliers, followed by the presentation of diverse performance metrics adopted to validate

and evaluate the performance of the existing techniques deployed for outlier detection in IoT.

Conclusion

Over the last decades, researchers have focused on outlier detection because of the development of low-cost, highly impactful IoT sensor technologies across a wide range of application domains. Identifying outliers significantly reduces functional risks, gets rid of hidden issues, and prevents operations from stopping. In a wide range of core implementations and IoT application fields, machine learning and deep learning detection algorithms are crucial for identifying outliers in data streams. However, there are still a number of issues that need to be resolved in order to resolve the outlier detection problem. These difficulties include heterogeneous data, time complexity, accuracy, scalability, and high dimensionality. This study reviews existing algorithms used in the IoT ecosystem to address data outlier problems. It also emphasizes the issues resolved, the outcomes, and the weaknesses of the algorithms. Additionally, a thorough explanation of several simulation tools and programming languages required to implement and experiment the prevalent algorithms ware also addressed with metrics for evaluating how well they performed. Future research would be embarking on developing ensemble unsupervised learning algorithm for the prediction of outliers in IoT-enabled Health Systems.

REFERENCES

Ali, Z., Muhammad, G., & Alhamid, M. F. (2017). An Automatic Health Monitoring System for Patients Suffering from Voice Complications in Smart Cities. *IEEE Access*, 5, pp. 3900–3908. https://doi.org/10.1109/ACCESS.2017.2680467

Alsini, R., Almakrab, A., Ibrahim, A., & Ma, X. (2021). Improving the outlier detection method in concrete mix design by combining the isolation forest and local outlier factor. *Construction and Building Materials,* 270, pp. 31-96. https://doi.org/10.1016/j.conbuildmat.2020.121396

Anuroop Gaddem, Tim Wilkin, Maia Angelova and Jyotheesh Gaddam (2020). Detecting Sensor Faults, Anomalies and Outliers in the Internet of Things: A Survey on the Challenges and Solutions, *Electronics (MDPI*), 9(3), pp. 1-15

Apanapudor JS. Umukoro J., Kwonu FZ., & Okposo N. (2023). Optimal Solution Techniques for Control Problem of Evaluation Equations, *Science World Journal,* 18(3), pp. 503-508

Apanapudor JS., Aderibigbe FM. & Okwonu FZ. (2020). An Optimal Penalty Constant for Discrete Optimal Control Regulator Problems, *Journal of Physics: Conference Series*, 1529(4), pp. 042-73.

Ayan Chatterjee Bestoun S. Ahmed (2022). IoT anomaly detection methods and applications: A survey, *Internet of Things (Elsevier)*, 19, pp. 1-17.

Chatterjee, A., & Ahmed, B. S. (2022). IoT anomaly detection methods and applications: A survey. Internet of Things, 19, pp. 100-568. https://doi.org/10.1016/J.IOT.2022.100568

Diaz-Rozo, J., Bielza, C., & Larranaga, P. (2018). Clustering of Data Streams with Dynamic Gaussian Mixture Models: An IoT Application in Industrial Processes. IEEE Internet of Things Journal, 5(5), pp. 3533–3547. https://doi.org/10.1109/JIOT.2018.2840129

Edje E. Abel & Ekabua Obeten (2015). Funding E-Health in Nigeria by NGOS/Multinational Organization: Overview and Perspectives, *International Journal of Computer Applications*, 111(11), pp. 37-41.

Edje E. Abel & Ekabua Obeten (2015). Restaurant Customer Self-Ordering System: A Solution to Reduce Customer/Guest Waiting Time at the Point of Sale, *International Journal of Computer Applications*, 111(11), pp. 19-22.

Edje, A. E., Abd Latiff, M. S., & Chan, W. H. (2023). IoT data analytic algorithms on edge-cloud infrastructure: A review. Digital Communications and Networks, 9(6), pp. 1486–1515. https://doi.org/10.1016/J.DCAN.2023.10.002

Edje, A. E., Abd Latiff, S. M., & Chan, H. W. (2021). Enhanced Non-Parametric Sequence-based Learning Algorithm for Outlier Detection in the Internet of Things. Neural Processing Letters, 53(3), pp. 1889–1919. https://doi.org/10.1007/s11063-021-10473-2

Ijaz, M. F., Attique, M., & Son, Y. (2020). Data-Driven Cervical Cancer Prediction Model with Outlier Detection and Over-Sampling Methods. Sensors, 20(10), 2809. https://doi.org/10.3390/s20102809

Kolozali, S., Puschmann, D., Bermudez-Edo, M., & Barnaghi, P. (2016). On the Effect of Adaptive and Nonadaptive Analysis of Time-Series Sensory Data. IEEE Internet of Things Journal, 3(6), pp. 1084–1098. https://doi.org/10.1109/JIOT.2016.2553080

Liu, D., Zhen, H., Kong, D., Chen, X., Zhang, L., Yuan, M., & Wang, H. (2021). Sensors Anomaly Detection of Industrial Internet of Things Based on Isolated Forest Algorithm and Data Compression. Scientific Programming, 2021, pp. 1–9. https://doi.org/10.1155/2021/6699313

Lyu, L., Jin, J., Rajasegarar, S., He, X., & Palaniswami, M. (2017). Fog-Empowered Anomaly Detection in IoT Using Hyperellipsoidal Clustering. IEEE Internet of Things Journal, 4(5), pp. 1174–1184. https://doi.org/10.1109/JIOT.2017.2709942

Muhammad Fahim and Alberto Sillitti (2019). Anomaly Detection, Analysis and Prediction Techniques in IoT Environment: A Systematic Literature Review, *IEEE Access, 7*, pp. 81664-81681.

Nazmul Kabir Sikder and Feras A. Batarseh (2023). Outlier Detection Unsing AI: A Survey, *AI Assurance*, pp. 231-291

Nan Shao and Yu Chen (2022). Abnormal Data Detection and Identification Method of Distribution Internet of Things Monitoring Terminal Based on Spatiotemporal Correlation, Energies (MDPI), 15(6), pp. 1-19.

Nesa, N., Ghosh, T., & Banerjee, I. (2018). Non-parametric sequence-based learning approach for outlier detection in IoT. Future Generation Computer Systems, 82, pp. 412–421. https://doi.org/10.1016/j.future.2017.11.021

Nusaybah Alghanmi, Reem Alotaibi and Seyed M. Buhari (2021). Machine Learning Approaches for Anomaly Detection in IoT: An Overview and Future Research Directions, *Wireless Personal Communications (Springer)*, 122, pp. 2309-2324.

Puschmann, D., Barnaghi, P., & Tafazolli, R. (2017). Adaptive Clustering for Dynamic IoT Data Streams. IEEE Internet of Things Journal, 4(1), pp. 64–74. https://doi.org/10.1109/JIOT.2016.2618909

Raafat, H. M., Hossain, M. S., Essa, E., Elmougy, S., Tolba, A. S., Muhammad, G., & Ghoneim, A. (2017). Fog Intelligence for Real-Time IoT Sensor Data Analytics. IEEE Access, 5, pp. 24062–24069. https://doi.org/10.1109/ACCESS.2017.2754538

Redhwan Al-amri, Raja Kumar Murugesan, Mustafa Man, Alaa Fareed Abdulateef, Mohammed A. Al-Sharafi and Ammar Ahmed Alkahtan (2021). A Review of Machine Learning and Deep Learning Techniques for Anomaly Detection in IoT Data, *Applied Sciences (MDPI)*, 11(12), pp. 2-23.

Rehman, A., & Belhaouari, S. B. (2021). Unsupervised outlier detection in multidimensional data. Journal of Big Data, 8(1), pp. 67-80. https://doi.org/10.1186/s40537-021-00469-z

Santamaria, A. F., de Rango, F., Serianni, A., & Raimondo, P. (2018). A real IoT device deployment for e-Health applications under lightweight communication protocols, activity classifier and edge data filtering. Computer Communications, 128, pp. 60–73. https://doi.org/10.1016/j.comcom.2018.06.010

Shukla, R. M., & Sengupta, S. (2020). Scalable and Robust Outlier Detector using Hierarchical Clustering and Long Short-Term Memory (LSTM) Neural Network for the Internet of Things. Internet of Things, 9, pp. 100-167. https://doi.org/10.1016/j.iot.2020.100167

Verma, P., & Sood, S. K. (2018). Fog Assisted-IoT Enabled Patient Health Monitoring in Smart Homes. IEEE Internet of Things Journal, 5(3), pp. 1789–1796. https://doi.org/10.1109/JIOT.2018.2803201

Walaa Gouda , Sidra Tahir, Saad Alanazi, Maram Almufareh & Ghadah Alwakid (2022). Unsupervised Outlier Detection in IOT Using Deep VAE, Sensors (MDPI), 22(17), pp.1-14

Xu, H., Sun, Z., Cao, Y., & Bilal, H. (2023). A data-driven approach for intrusion and anomaly detection using automated machine learning for the Internet of Things. Soft Computing, 27(19), 14469–14481. https://doi.org/10.1007/s00500-023-09037-4

Yang, J., Rahardja, S., & Fränti, P. (2021). Mean-shift outlier detection and filtering. *Pattern Recognition*, 115, pp. 107-874. https://doi.org/10.1016/j.patcog.2021.107874

Yu, Y., Wu, X., & Yuan, S. (2020). Anomaly Detection for Internet of Things Based on Compressed Sensing and Online Extreme Learning Machine Autoencoder. *Journal of Physics:* Conference Series, 1544(1), pp. 012-027. https://doi.org/10.1088/1742-6596/1544/1/012027

Zhu, R., Ji, X., Yu, D., Tan, Z., Zhao, L., Li, J., & Xia, X. (2020). KNN-Based Approximate Outlier Detection Algorithm Over IoT Streaming Data. *IEEE Access*, 8, pp. 42749–42759.