applied sciences

MDPI

*Review*

# A Review of Machine Learning and Deep Learning Techniques for Anomaly Detection in IoT Data

**Redhwan Al-amri [1,*], Raja Kumar Murugesan [1,*], Mustafa Man [2,*], Alaa Fareed Abdulateef [3], Mohammed A. Al-Sharafi [4,*] and Ammar Ahmed Alkahtani [5]**

1 School of Computer Science and Engineering, Taylor's University, Subang Jaya 47500, Selangor, Malaysia
2 Faculty of Ocean Engineering Technology & Informatics, Universiti Malaysia Terengganu (UMT), Kuala Nerus 21030, Terengganu, Malaysia
3 School of Computing, Universiti Utara Malaysia, Sintok 06010, Kedah, Malaysia; alaa_fareed@ahsgs.uum.edu.my
4 Department of Information Systems, Azman Hashim International Business School, Universiti Teknologi Malaysia, Skudai 81310, Johor, Malaysia
5 Institute of Sustainable Energy (ISE), Universiti Tenaga Nasional (UNITEN), Kajang 43000, Malaysia; ammar@uniten.edu.my
* Correspondence: redhwanmohammedabdullahalamri@sd.taylors.edu.my (R.A.-a.); rajakumar.murugesan@taylors.edu.my (R.K.M.); mustafaman@umt.edu.my (M.M.); alsharafi@ieee.org (M.A.A.-S.)

**Abstract:** Anomaly detection has gained considerable attention in the past couple of years. Emerging technologies, such as the Internet of Things (IoT), are known to be among the most critical sources of data streams that produce massive amounts of data continuously from numerous applications. Examining these collected data to detect suspicious events can reduce functional threats and avoid unseen issues that cause downtime in the applications. Due to the dynamic nature of the data stream characteristics, many unresolved problems persist. In the existing literature, methods have been designed and developed to evaluate certain anomalous behaviors in IoT data stream sources. However, there is a lack of comprehensive studies that discuss all the aspects of IoT data processing. Thus, this paper attempts to fill this gap by providing a complete image of various state-of-the-art techniques on the major problems and core challenges in IoT data. The nature of data, anomaly types, learning mode, window model, datasets, and evaluation criteria are also presented. Research challenges related to data evolving, feature-evolving, windowing, ensemble approaches, nature of input data, data complexity and noise, parameters selection, data visualizations, heterogeneity of data, accuracy, and large-scale and high-dimensional data are investigated. Finally, the challenges that require substantial research efforts and future directions are summarized.

**Keywords:** anomaly detection; data stream; deep learning; Internet of Things; machine learning



**Citation:** Al-amri, R.; Murugesan, R.K.; Man, M.; Abdulateef, A.F.; Al-Sharafi, M.A.; Alkahtani, A.A. A Review of Machine Learning and Deep Learning Techniques for Anomaly Detection in IoT Data. *Appl. Sci.* **2021**, *11*, 5320. https://doi.org/10.3390/app11125320

Academic Editor: Gabriella Tognola

Received: 4 May 2021
Accepted: 4 June 2021
Published: 8 June 2021



## 1. Introduction

The advent of the Internet has revolutionized communication between humans. Similarly, Internet of Things (IoT) devices are reshaping how humans perceive and interact with the physical world. By 2025, IoT systems are expected to cross nearly 75 billion connected devices, tripling the global population [1].

The IoT is a network of heterogeneous objects, such as smartphones, laptops, intelligent devices, and sensors, connected to the Internet through various technologies [2]. The IoT enables various sensors and devices to communicate with each other directly without user interaction [3]. IoT has become one of the biggest data sources in the past few years [4]. Methods such as machine learning algorithms can be used to extract meaningful information from these data.

IoT remains a significant challenge. One technique that effectively analyses the collected data stream is anomaly detection [5,6]. These unexplained phenomena could be

outliers, anomalies, cyber-attack, novelties, exceptions, deviations, surprises, or noise [7,8], where outliers are the data points that are considered out of the ordinary. Detection of these points can be done using outlier detection methods [9]. The anomalies are a special kind of outlier that has actionable pieces of information which could be meaningful. To detect these points, anomaly detection methods are used [10]. Similarly, fault detection is used to detect noise, which is unwanted, and wrong data that has to be removed [11]. Cyber attacks, on the other hand, are more sophisticated; they can be hidden between the data points and hard to detect [12]. Figure 1 illustrates the difference between the above-mentioned terms.

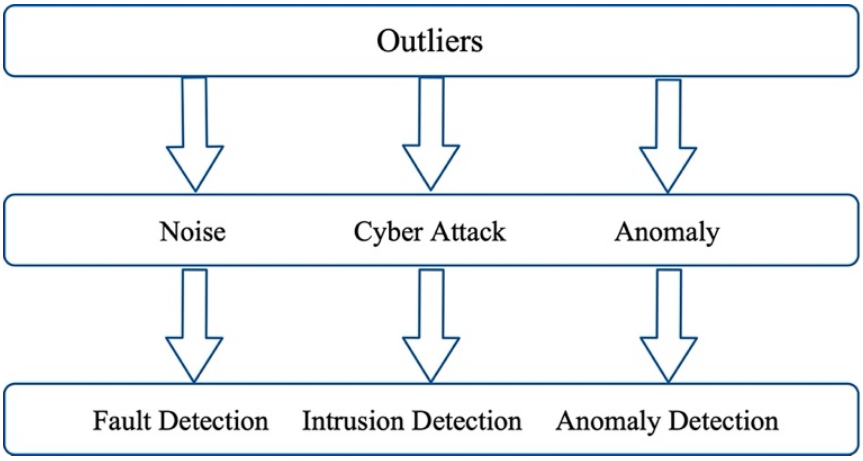



**Figure 1.** Outliers in the data stream.

In many cases, IoT real-time applications generate infinitely massive data streams that pose impose unique limitations and obstacles to machine learning algorithms [13]. These challenges require a careful design of the algorithm to process these data [14]. Most existing data stream algorithms are less efficient and have limited capability requirements [15]. Many studies have investigated the techniques used for anomaly detection, such as [16–19] that address static data and data stream using both statistical and machine learning methods. However, these studies have not focused on evolving data streams.

Some studies have focused on the detection of anomalies in the data stream, such as [16]. However, previous studies have not addressed all the requirements that have to be available in the algorithm to process IoT data streams and the main challenges for choosing an excellent algorithm that suits the IoT data characteristics. For anomaly detection, many algorithms can be used to detect anomalies in the data stream. A good anomaly detection algorithm should consider the following restrictions related to data streams:

- Data points are pushed out continuously, and the speed of arrival of data depends on the data source. Thus, it could be fast or slow.
- The data stream could be potentially infinite, which means there could be no end to the incoming data.
- Features and/or characteristics of the arriving data points may evolve.
- Data points are potentially one-pass, i.e., the data points can be used only once, and discarded after. Thus, fetching important data characteristics is important.

To have good quality anomaly detection, algorithms must have the ability to handle the following challenges before they can be used effectively:

- Ability to handle fast data—the anomaly detection algorithm must be able to handle and process data in real-time when data points from the data source come constantly, as data streams could be huge and should be handled in one pass.
- Ability to process data in given memory—the massive amount of data should not influence the data stream's processing capabilities. Thus, the data stream algorithm should not require unlimited memory for the unlimited data points arriving in the system, and it should be able to process the data within the available memory.

- Ability to handle dimensionality—high-dimensional data have the additional problem of selecting the proper feature vectors or dimensions that could create better clusters. High-dimensional data also require additional calculation and additional processing. Thus, the algorithm should consider such factors in which the clusters' quality will not be affected.
- Ability to handle evolving data streams—data sources are numerous and moving over time; they grow continuously. Therefore, data produced would change, and the algorithm outcomes can change considerably. The activities of streams are known to evolve over time and need a special method to handle them.

Existing research has mostly analyzed anomalies based on machine learning techniques primarily focused only on batch processing. In contrast, this paper focuses on machine learning techniques for anomaly detection in data streams, more specifically on evolving data streams. The major contributions of this research are as follows:

1. Examination of state-of-the-art studies centered on machine learning and deep learning techniques for anomaly detection in data streams;
2. Taxonomy that defines current literature based on the nature of the data, anomaly types, detection learning modes, window model, dataset, and evaluation criteria;
3. Analysis of the existing techniques based on the proposed taxonomy;
4. Highlighting challenges that form the future research direction.

The rest of the paper is organized as follows: Section 2 provides a brief background of the study, which includes the definition of the main terms in the review paper. Section 3 presents a taxonomy of anomaly detection techniques for IoT data stream that includes the machine learning and deep learning techniques used, nature of data, anomaly types, detection learning mode, window models, datasets, and the evaluation criteria. Section 4 discusses the research challenges and the potential future directions. Section 5 presents the research results. Finally, the conclusion is given in Section 6.

## 2. Background

With sensors invading human's daily lives, data streams are growing exponentially. Driven by the expansion of the Internet of Things (IoT) and the connected real-time data sources, many applications generate vast amounts of critical data streams that evolve. Data streams are a continuous, infinite series of data records followed and arranged by embedded or precise timestamps [14]. Analyzing such data streams efficiently offers useful insights for many application domains. Yet, it is a significant challenge. One of the techniques that deal with analyzing the collected data stream effectively is called anomaly detection [5]. Anomaly detection refers to the identification of events or patterns in a dataset that differ significantly from the majority of items or the expected pattern, and those unexpected patterns are referred to as anomalies, novelties, exceptions, noise, surprises, or deviations [7,8,20]. Anomaly detection plays a considerable role in the analysis of anomalies in many applications. In almost every use case, early detection is useful. Looking at a device that monitors a cardiac patient's heart rate continuously, an anomaly may result in a heart attack. Detecting such anomalies minutes in advance is far better than detecting it a few seconds ahead, or detecting it after the event occurs [21]. Detecting anomalies in the data stream has practical and important applications across a wide range of fields [15].

One of the main approaches used for anomaly detection is machine learning techniques. As stated in [22], machine learning enables computers to learn without explicit programming. It is intended to allow a system to learn from the past or the present and to use the knowledge to make future predictions or decisions [18]. Even though "learning" is extremely vital in machine learning, it is not the objective. The primary objective of machine learning is to create a system that can identify relevant patterns in data automatically and correctly [23].

Recent advancement in deep learning techniques has made it also possible to largely improve anomaly detection performance compared to the classical approaches. Deep

learning is defined as a subset of machine learning in artificial intelligence that has networks capable of unsupervised learning from data that are unstructured [5].

## 3. Taxonomy of Anomaly Detection Techniques

The taxonomy defined here reflects the state-of-the-art techniques on anomaly detection using machine learning in IoT data streams. Further, this section also elaborates on the nature of data, types of anomalies and the anomaly detection learning modes, window models used for analyzing the data, the datasets used for evaluating the anomaly detection techniques, and the evaluation criteria used to measure the performance of the anomaly detection technique, as illustrated in Figure 2.

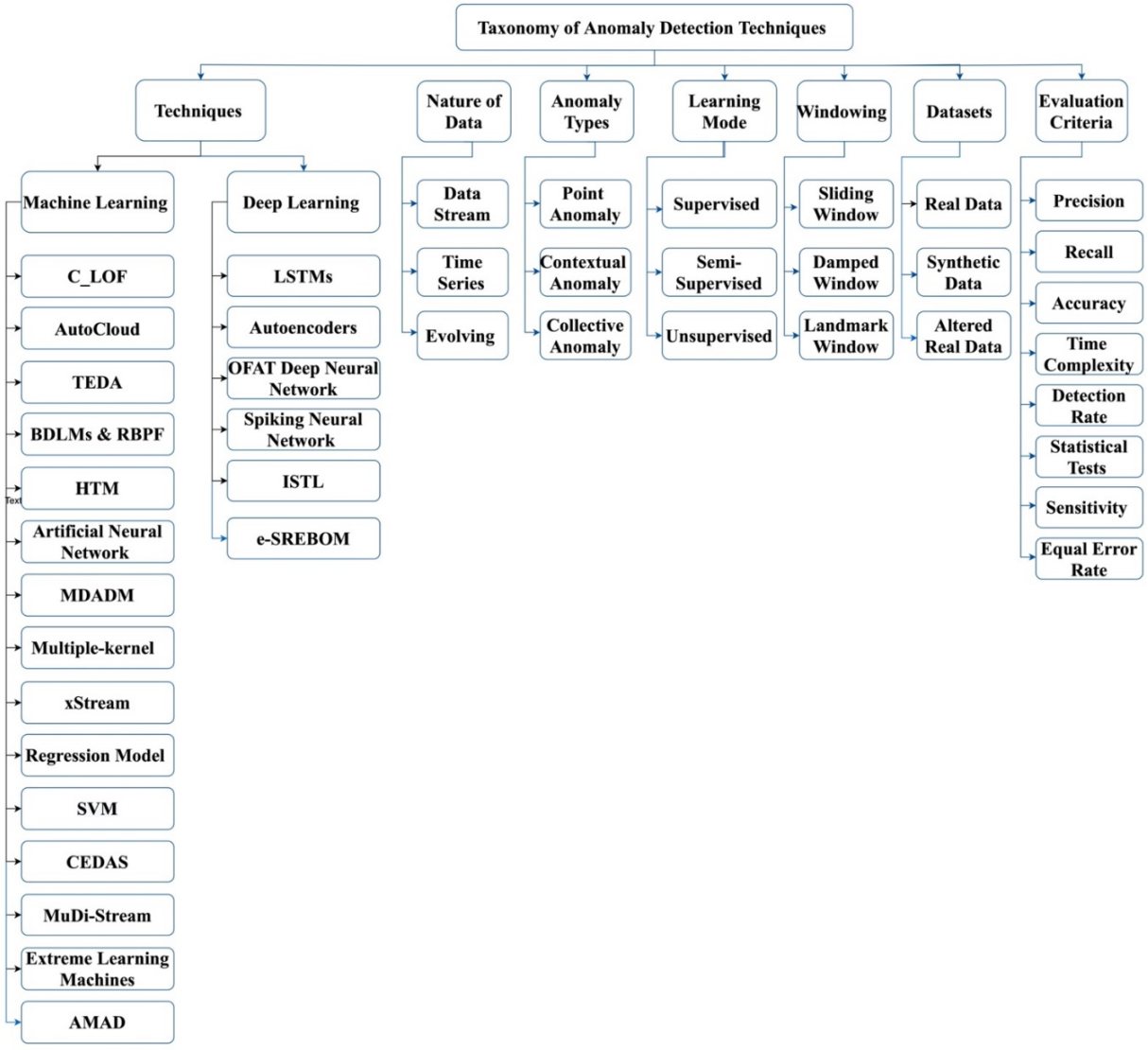



**Figure 2.** Taxonomy of anomaly detection techniques.

### *3.1. Anomaly Detection Techniques*

#### 3.1.1. Machine Learning Techniques

Machine learning anomaly detection techniques for data streams have been proposed in recent years. These techniques can detect data stream anomalies in various core implementations across a broad variety of areas, including manufacturing networks, finance and banking, military, healthcare, insurance, network protection, and the Internet of Things.

Based on a popular and successful anomaly detection algorithm called Local Outlier Factor (LOF), a new data stream algorithm called Cumulative Local Outlier Factor (C-LOF) was developed; however, it only works on static data [14]. Experimental results suggest that even when detecting anomalies in the data stream, C-LOF may overcome masquerading. A serious drawback of this strategy is that the time complexity of C-LOF is high.

An evolving algorithm based on data stream clustering was introduced by Bezerra et al. [24]. The algorithm called AutoCloud is based on the newly introduced Typicality and Eccentricity Data Analytics principle that is used for anomaly detection research. AutoCloud is an evolving electronic and recursive approach that does not require prior knowledge of data before processing. The algorithm can handle concepts, which are inherited problems in data streams. However, the proposed technique overlooks many of the distance problems, dynamically affecting the separation of the clouds.

Another evolving clustering algorithm was proposed based on a mixture of topicalities for data stream mining [25]. It is based on the paradigm of TEDA and separates clustering into two sub-problems: micro and macro-clusters. Experimental results show that the proposed technique yields good results for data clustering and predicts its density even in the face of events that affect data distribution parameters, such as concept drifts. However, the autonomous adaptation process is not entirely considered in this strategy.

Complex linear Bayesian equations have been developed, wherein existing BDLMs and RBPF techniques are combined for the real-time detection of anomalies [26]. This methodology aims to quantify if the variables and parameters are effective in detecting anomalies. However, no effort was made in the proposed technique to resolve the challenge of sensor drift or inadequate conditions for implementation.

The Hierarchical Temporary Memory (HTM) algorithm is used to suggest a data stream method for real-time outlier detection for space imagers [27]. The HTM-based algorithm is consistent with the detection of spatial and temporal outliers and fulfils the specifications without oversight for real-time, continuous data stream detection. The findings suggest that the algorithm would efficiently achieve real-time detection of anomalies. However, the suggested solution was not able to reduce the false-positive rate.

In 2019, an artificial algorithm for the neural network was used by Cauteruccio et al. [28]. A novel method for automated anomalies in heterogeneous sensor networks based on data edge exploration combined with cloud data was constructed. The experimental assessment of the planned solution was carried out on real data gathered in an indoor building environment and then distorted by separate virtual impairments. The research outcomes suggest that the proposed solution would adjust itself to the shifts in the environment and identify the anomalies correctly. However, the study does not consider the evolving features of the data and the definition of drift.

A new multi-source approach for the detection of multi-dimensional data anomalies (MDADM) was developed [29]. The real-time prediction of the probability of abnormal behavior occurring in underground mining is based on the hierarchical edge computing model. Results show the current method has a higher detection performance and less transmission delay than conventional methods. However, the authors overlooked the evolution of data and its value and that its characteristics may change over time, which will affect the overall accuracy of the proposed method for anomaly detection.

A new approach to classifying non-stationary data streams that address problems in detecting anomalies such as infinite time, idea drift, recurring concepts, and concept evolution was suggested based on a multi-kernel approach [30]. These kernels were modified in the stream periodically after acquiring the precise labels of the instances. In the function spaces, newly arrived cases will be graded as to their distance from the boundaries of the groups previously identified. The experimental results indicate the proposed approach is superior in this area to the state-of-the-art approaches. However, the study does not consider the evolving clusters.

For the first time, an outlier detection algorithm called xStream, which addresses feature-evolution in data streams, was suggested by Manzoor et al. [31]. The proposed

algorithm addresses a problem that has not been investigated previously. XStream is an outlier detector based on density and has three main characteristics: (1) it is a constant-space and constant-time algorithm (per incoming update), (2) it measures anomalies on several scales or granularities, and (3) it can handle high-dimensionality by distance-preserving projections, and non-stationary as the stream progresses through (1)-time model updates. For evolving streams with moderate space overhead for which there is no competition, experiments show that xStream is effective and precise. However, the challenge of evolving the cluster, which affects the overall outlier detection, has not been attempted.

A regression-based strategy was developed by Farshichi et al. [32] for detecting contextual anomalies in the control system of air traffic. They also have actionable improvement specifics by using the algorithm for modification detection and time windows on contextual anomalies. The evaluation of the proposed model shows a high accuracy anomaly detection with low delay. However, the researchers did not discuss the challenge of fault detection initiatives.

An SVM algorithm was effectively designed by Bose et al. [33], to provide drivers with real-time warnings and directions on the road anomalies and ensure a safe driving experience. The proposed system uses the common machine learning technique, SVM, to categorize driving events, such as acceleration and braking, and road anomalies, such as bumps and potholes, and it gives drivers real-time alerts and guidance using a local Quick Dynamic Time Warping (FastDTW) algorithm. Nevertheless, the study fails to consider the different categories and features of objects moving on the roads.

In 2018, HTM was used by Rodriguez et al. [34] to detect outliers on real-time network metrics obtained by continuous analysis of the resource utilization of workflow tasks performed. The framework can process data streams online in an unsupervised fashion and successfully adapt to changes in the data based on the underlying statistics. The experiment results illustrate the proposed model's ability to correctly catch output deviations on different resource usage parameters caused by various conflicting workloads implemented into the program. However, a severe weakness with this argument is that detection rate accuracy and latency are still subject to improvement.

Similarly, HTM has been used to create a new method of detecting anomalies based on online sequence memory [35]. Spatial and temporal anomalies within predictable and noisy domains can be detected by the proposed method. The technique achieves, without supervision, real-time, continuous, online detection criteria. The results indicate an improvement in the detection rate of the method in comparison with the state of the art. However, the research did not consider evaluating the technique based on a real dataset with anomalies containing high dimensionality.

A full online method, called CEDAS, has been introduced for clusters evolving data streams into arbitrary-shaped clusters [36]. It is a two-stage solution that is effective, noise-resistant, and productive in memory and computing use, with low latency as the number of data dimensions. The results revealed the proposed algorithm's ability to deal with changing data sources entirely online. The proposed algorithm compares velocity, purity of the cluster, precision, and memory capacity favorably with similar techniques. However, this technique does not consider the evolution of the feature and the high dimensionality of the data.

On the other hand, a new method called MuDi-Stream [37] consisted of four main components of the online-offline phases. The online phase maintains summary information on the evolving multi-density data stream into core mini clusters. In contrast, to generate final clusters, the offline phase uses an adapted density-based clustering algorithm. An anomaly buffer for handling both noise and multi-density information is the grid-based approach. The algorithm is evaluated on different synthetic and real-world datasets using different quality metrics, and results on scalability are presented. The experimental results show that the approach proposed in this study increases the effectiveness of clustering in multi-density environments. However, the main weakness of this method is that as the

number of empty grids increases, it cannot handle high-dimensional data and thus makes the processing time slower.

Fast anomaly detection algorithms were suggested using Extreme Learning Machine (ELM) [38], to discover operationally significant anomalies. The method is used over large aviation datasets to address elevated computational training time. The authors carried out the experiments on a real benchmark aviation safety dataset in an unsupervised fashion, including 43,000 flight data on radar information, flight trajectories, and distance from nearby aircraft. Nevertheless, the authors did not use the benchmarked data to demonstrate the performance of the proposed method and compare its performance with the state-of-the-art techniques.

Xue et al. [39] proposed a novel dynamic anomaly detection framework on time-evolving attributed networks called (AMAD). AMAD utilized the advantage of the evolving characteristics of the underlying attributed networks and models. Precisely, AMAD progressively updates the detection results by measuring the difference in residuals among the two consecutive timestamps and applying it to the previous residuals. To prevent the negative effects of unwanted and noisy features, the proposed method performs attribute selection while processing the residuals for anomaly detection. Tests performed on both synthetic and real-world datasets show that the proposed method is effective and efficient. Nevertheless, the study fails to handle the high dimensionality of the data and ignore its effect towards the power and memory consumption of the framework.

Table 1 presents a summary of the machine learning techniques used for anomaly detection in data streams. The table highlights the nature of data used for executing the experiments, the types of anomalies found within the data, the windowing model applied, the dataset used, and finally the criteria used for evaluating the proposed techniques.

**Table 1.** Summary of machine learning techniques for data stream anomaly detection.

| Techniques | Nature of the Data | Types of Anomaly | Anomaly Detection Types | Windowing | Dataset | Evaluation Criteria |
|---|---|---|---|---|---|---|
| C_LOF [14] | Data Stream (evolving) | Point anomaly | Unsupervised learning using density | Sliding window | synthetic and real-life datasets. | Precision, Recall, and Accuracy |
| AutoCloud [24] | Data Stream (evolving) | Point anomaly | Unsupervised learning using clustering | Sliding window | Artificial and real dataset | N/A |
| TEDA Clustering [25] | Data Stream (evolving) | Point anomaly | Unsupervised learning using clustering | Sliding window | Own synthetic data sets | Accuracy, Time complexity |
| Combination of (BDLMs) & (RBPF) [26] | Data Stream (evolving) | Point anomaly | Unsupervised learning using density | Sliding window | Artificial dataset | Accuracy, the Detection rate |
| HTM [27] | Data Stream | Point anomaly | Unsupervised learning based on HTM | N/A | Dataset of space imager data stream | Accuracy |
| Artificial Neural Network [28] | Continuous and image data | Point anomaly | Unsupervised learning on patterns of WSN nodes | Sliding window | The experimental tests that have been conducted and cover more than 27 | Accuracy |
| MDADM [29] | Continuous data | Point anomaly | Supervised learning | N/A | Own dataset | Accuracy |

**Table 1.** *Cont.*

| Techniques | Nature of the Data | Types of Anomaly | Anomaly Detection Types | Windowing | Dataset | Evaluation Criteria |
|---|---|---|---|---|---|---|
| Multi-kernel [30] | Data Stream (evolving) | Point anomaly | Unsupervised learning-based multiple kernel learning approach | N/A | KDD99, (SynCN), Cover type | Detection Rate |
| xStream [31] | Data Stream (evolving) | Point anomaly | Unsupervised learning based on density-based ensemble | Sliding window (reference and current) | Spam-SMS, Spam-URL datasets | Detection Rate |
| Regression Model [32] | Continuous data | Contextual anomaly | Supervised learning on historical data | N/A | N/A | Precision, Recall, and Accuracy |
| Super Vector Machine [33] | Continuous data | Contextual anomaly | Supervised learning on historical data | N/A | Own dataset | Accuracy |
| HTM [34] | Continuous data | Point anomaly | Unsupervised learning | N/A | Two real scientific workflows | TP, TN, FP, FN |
| CEDAS [36] | Data Stream (evolving) | Point anomaly | Unsupervised learning based on clustering | N/A | KKDCup99 | Confusion matrix |
| HTM [35] | Data Stream | Point anomaly | Unsupervised learning based on clustering | Sliding thresholds | Numenta Anomaly Benchmark (NAB) | Confusion matrix |
| MuDi-Stream [37] | Data Stream (evolving) | Point anomaly | Unsupervised learning based on the density-based method | Fixed windowing | KDD Cup'99, UCI dataset, DS1, DS2, and DS3 dataset | Confusion matrix |
| Extreme Learning Machines [38] | Continuous data | Collective anomaly | Supervised learning on 43,000 flights data | N/A | Real aviation safety benchmark dataset | The area under the curve |
| AMAD [39] | Data Stream (evolving) | Point anomaly | Unsupervised learning | Window-based | WikiBlogcata Flickr, Congress, Aminer, LargeAmazon | Sensitivity (AUC, ROC curve) |

#### 3.1.2. Deep Learning Techniques

The use of deep learning in anomaly detection is one of the latest advancements in this field. To solve the problem of imbalanced classification for non-stationary time series, time-series anomaly detection for KPIs based on supervised deep learning models with neural convolution and long-short-term memory networks (LSTMs) and a vector auto-encoder (VAE) oversampling algorithm have been developed by researchers Qiu et al. [40]. To verify the performance of KPI-TSAD, Yahoo's benchmark anomaly detection datasets were used, and the detector performed well on the benchmark datasets. The proposed VAEGEN algorithm also produced better results than other common oversampling approaches. The primary drawback of this approach is that it required labelled data to train the model, which in most cases are not usable.

Another new anomaly detection algorithm based on neural network ensembles, called Streaming Autoencoder (SA), was developed by Dong and Japkowicz [41] for evolving data streams. It is a one-class learner that only requires suitable instructional class data, which is reliable even without training. It features an autoencoder threaded ensemble with continuous learning capabilities. The results show anomalies with fewer false alarms are effectively identified by the new approach. One of the limitations of this technique, however, is scalability. The technique loses its capability when it comes to a large volume of data.

In 2020, a neural network showed a promising improvement in the arena of anomaly detection. Wambura et al. [42], used a deep neural network to propose an algorithm named One sketch Fits All Time (OFAT) for addressing the issue of accurate long-range forecast within high-dimensional feature-evolving time series. The proposed algorithm is designed to address the issue generated by the non-stationary nature of feature-evolving time series causing the length of the input's sequence (rows) to change as new data points arrive with their feature values (columns) evolving over time. The experiments performed on real-world datasets and rigorous evaluation evidenced that OFAT has fast processing time, robust performance, and accurate detection. Yet, one of the limitations with this approach is that it does not consider real-time interactive forecasting in data streams.

Similarly, for reliable anomaly detection in data streams, a real-time evolving spiking restricted Boltzmann machine technique named (e-SREBOM) was proposed by Xing et al. [43]. It is a hybrid anti-malware detection technique that is sophisticated, revolutionary, and extremely efficient. It is a combination of the e-SNN and REBOM algorithms that allows the system to adjust to changes automatically. e-SREBOM can respond to high-complexity problems and provides a high degree of generalization. The proposed technique was evaluated on three-dimensional datasets with high complexity. However, the study made no attempt in considering the incremental online learning, with long-/short-term memory abilities, to ensure greater accuracy and efficiency.

In 2020, Nawaratne et al. [44] proposed the Incremental Spatio-Temporal Learner (ISTL) to handle the issues and limitations of anomaly detection and localization for real-time video surveillance. ISTL is an unsupervised deep learning technique that uses active learning and fuzzy aggregation to constantly update and discriminate among new anomalies and normal data with respect to time. Three benchmark datasets are used to illustrate and test ISTL on precision, robustness, and computation complexity. The experiment findings confirm the efficiency of the proposed technique. However, one major drawback of this approach is that it fails to reduce false-negative detection, which effects the overall detection accuracy.

In 2019, a real-time spiking restricted Boltzmann machines (e-SREBOM) approach was proposed [43]. It is a strategy of detecting malware that is hybrid, dynamic, innovative, and incredibly useful. The findings have shown that the proposed algorithm maximizes the efficiency of the classification, thereby reducing the requirements for computational power. One big downside to this strategy is that it lacks gradual online learning for long-/short-term memory capability.

Table 2 presents the summary on the deep learning techniques used for anomaly detection in data streams. The table highlights the nature of data used for executing the experiments, the types of anomalies found within the data, the windowing model applied, the dataset used, and the criteria used for evaluating the proposed techniques.

**Table 2.** Summary of deep learning techniques for data stream anomaly detection.

| Techniques | Nature of the Data | Types of Anomaly | Anomaly Detection Types | Windowing | Dataset | Evaluation Criteria |
|---|---|---|---|---|---|---|
| LSTMs [40] | Time-Series | Point anomaly | Supervised learning using deep learning | Sliding window | Yahoo Webscope | Confusion matrix. |
| Autoencoder [41] | Data Stream (evolving) | Point anomaly | Unsupervised learning based on Ensembles neural networks | Sliding window | HTTP, SMTP, SMTP+HTTP, COVERTYPE, SHUTTLE, Weather | AUC |
| (OFAT) Deep neural network [42] | Time series | Point anomaly | Supervised learning | Window-based | Web traffic dataset, Avocado dataset, Temperature dataset | Statistical tests (average Rank), Mean Average Score (MAS) |
| Evolving spiking neural network [43] | Data Stream (evolving) | Point anomaly | Unsupervised learning | Sliding window | 3 Benchmark dataset | Accuracy |
| ISTL [44] | Data Stream (evolving) | Point anomaly | Unsupervised learning based on deep learning | Sliding Window | UCSD Pedestrian datasets, Ped 1 and Ped 2) and CUHK Avenue dataset | Accuracy (ACU), Equal Error Rate (EER), |
| (e-SREBOM) [43] | Data Stream (evolving) | Point anomaly | Unsupervised learning using Spiking Neural Networks (eSNN) | Window-based | Water_tower_dataset, gas_dataset, electric_dataset | Accuracy, Speed, Time to learn |

### *3.2. Nature of the Data*

The selection of the anomaly detection technique depends on the nature of the data being analyzed. The Internet of Things (IoT) is a major source of data streams. A data stream is a continuous series of data records followed and arranged by implicit or explicit timestamps [14]. This data sequence has three key features. First, it is a stream of continuous data flow. Hence, the algorithm should process the data in a limited time. Second, the data stream is unlimited. In other words, the number of data points entering is infinite. Thus, storing such a massive volume of data is another major obstacle. Lastly, data streams change over time, i.e., evolve [16]. Detecting anomalies in the data stream has practical and important applications across a wide range of fields [15]. In realistic implementations, the significance of detecting anomalies in the data stream accelerates its growth by ensuring precision and immediacy [15]. Several methods have been used to propose learning algorithms for anomaly detection in data streams such as C-LOF [14], AutoCloud [24], TEDA Clustering [25], Evolving spiking neural network [43], Combination of (BDLMs) & (RBPF) [26], KPI-TSAD [40], HTM [27], ensembles neural networks [41], Multiple kernel learning [30], xStream [31], CEDAS [36], MuDi-Stream, [37], Long Short-Term Memory [15,45], Density-based Clustering [46], and others.

### *3.3. Anomaly Types*

Anomalies are observations that deviate significantly from other observations as to arouse suspicion that it was generated by a different mechanism [47]. Anomalies can be categorized as follows.

#### 3.3.1. Point Anomaly

Point anomaly occurs when anomalies differ significantly from the expected patterns. The detection of this type of anomaly involves the observation of any point that can be detected as different from other data flows. It is also referred to as an outlier [19]. Figure 3 demonstrates a point anomaly.

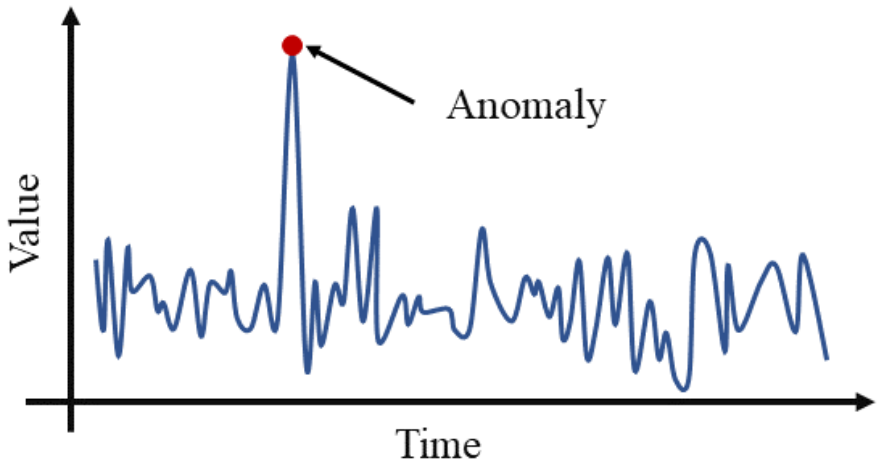



**Figure 3.** Point anomaly detection.

#### 3.3.2. Contextual Anomaly

Another type of anomaly is observed for any data pattern that occurs as usual and anomalous in one scenario. The detection of spatial anomalies includes comprehension of the meaning [19], which typically occurs inside time-series data sources. A common example is heavy traffic jams during rush hour, which may be a contextually anomalous traffic activity after midnight because of a crash, poor visibility, or other causes related to foggy conditions. An example of contextual anomaly is shown in Figure 4.

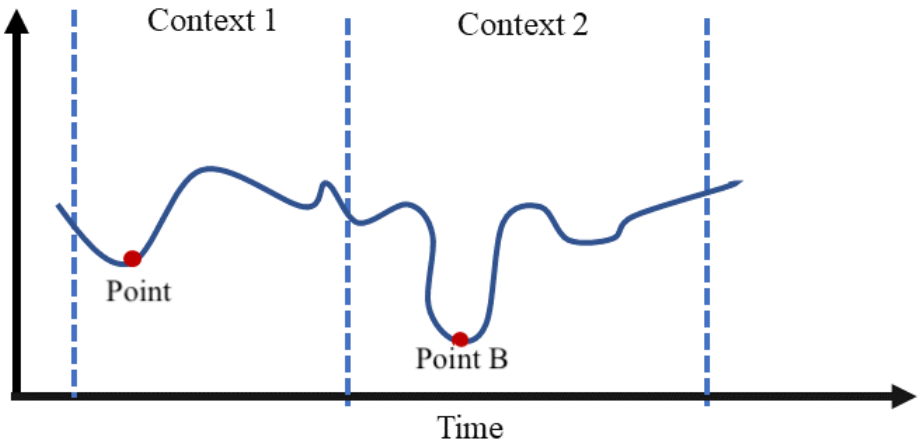



**Figure 4.** Contextual anomaly detection.

#### 3.3.3. Collective Anomaly

A series of observations is evaluated to recognize collective anomaly behavior. Any variance from the usual pattern may result in collective anomalies over consecutive time intervals concerning complete data patterns [19]. For example, a single time interval calculation is not enough to evaluate the operation of the heart, whereas cumulative signals can determine usual or anomalous behavior, as illustrated in Figure 5.

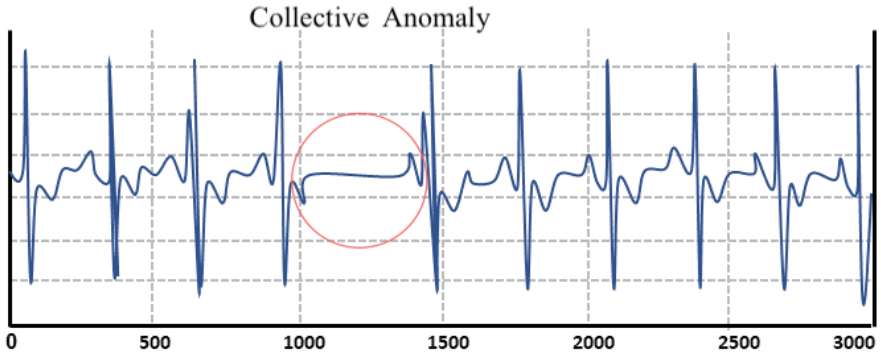



**Figure 5.** Collective anomaly for heart rate monitoring signal.

### *3.4. Anomaly Detection Learning Modes*

Anomaly detection learning modes can be classified into three types based on the availability of labels in the datasets used to create baselines [48], including supervised, semi-supervised, and unsupervised anomaly detection, as shown in Figure 6.

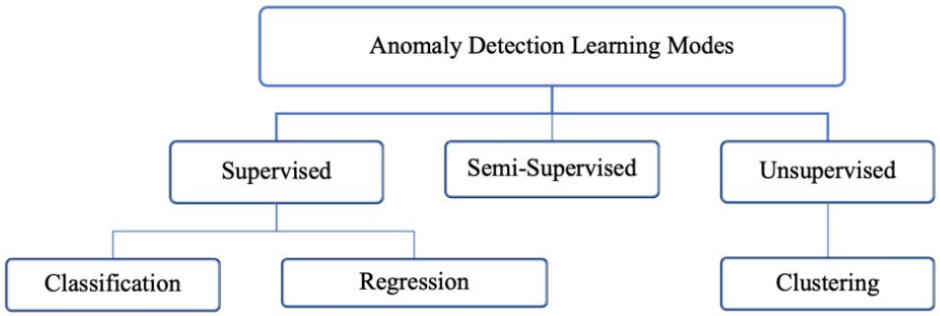



**Figure 6.** IoT anomaly detection learning modes.

#### 3.4.1. Supervised Anomaly Detection

The supervised anomaly detection approach detects anomalies by creating a set of grouping rules that help to predict future data. One type of supervised anomaly technique is the classification-based detection of anomalies [18]. Supervised anomaly detection can be divided into two categories, namely, classification and regression.

##### Classification

The SVM is a separate hyperplane formally defined as a discriminative classifier. With smart transportation [33], SVM is used to provide drivers with real-time alerts and instructions on the streets' anomalies to ensure a safe driving experience. Similarly, in smart cities [49], SVM is utilized for predicting anomalies and attacks that target the IoT systems.

The Naive Bayes classification method is based on the Bayesian Theorem and is primarily compatible when the dimensionality of the input is high. Bayesian techniques have been used in smart homes [34] to propose a two-tier intrusion detection system using a machine learning approach to classify records and monitor suspicious patterns across the network in the service provider's data center. Additionally, Gunupudi et al. [50] used the naïve base classifiers to propose a self-constructing feature clustering method for anomalydetection.

The K-Nearest Neighbor (k-NN) algorithm is an example of supervised machine learning methods utilized for solving classification and regression issues by assuming similarities in devices deployed in a proximate location. The k-NN has been implemented in the industrial system to detect cyber-physical attacks for cyber manufacturing systems by Wu et al. [12]. Similarly, k-NN was used to propose a self-constructing feature clustering method for detecting anomalies by [50].

Regression

Regression algorithms use the input features to predict the data's output values faded into the system. For example, in intelligent transportation, a regression-based model was developed by Farshichi et al. [32] to detect contextual anomalies in the air traffic control framework by identifying the correlation between accidents and resource measurements in the log reports.

The Decision Tree approach constructs regression or classification techniques in a tree structure. This approach separates the dataset into small groups, while at the same time an associated decision tree is increasingly constructed in the industrial system. A decision tree is used to predict anomalies and attacks in smart cities' IoT systems [49].

#### 3.4.2. Semi-Supervised Anomaly Detection

Semi-supervised anomaly detection is an approach where only standard data models and other data are classified as measuring anomalies [18].

In the healthcare system, Bayesian network-based methods have been used in [27] to propose a long-term, semi-supervised monitoring system based on IoT to assess the quality of maternal sleep. Similarly, a reinforcement-learning technique was suggested in intelligent transportation by Lu et al. [51] to detect the motor outlier over the temperature data stream of the unmanned aerial vehicle. Temporal and spatial–temporal techniques have also been suggested in the smart object domain by Chen et al. [52] to detect anomalous behavior within the environmental datasets.

#### 3.4.3. Unsupervised Anomaly Detection

Unsupervised anomaly detection is emphasized by unlabeled information, which does not require separate training and testing stages. Clustering-based detection of anomalies is a general example [18]. The unsupervised anomaly detection approach is grouped into one category, namely clustering, as shown below.

Clustering

The objective of using clustering algorithms is to identify the normal data in the input. The input space has a structure such that certain patterns occur more frequently than others, and we want to see what usually happens and what does not happen. In statistics, this step is called estimating the density. Clustering is one method for estimating the density. For example, utilizing coupling-edge data analysis and cloud data analysis automatically detects anomalies in heterogeneous sensor networks.

In an intelligent transportation system, Luo and Zhong [53] built a stacked denoising autoencoder model to detect anomalies in a gas turbine engine. Furthermore, PNN was evaluated in the industrial system over a real-time thermal power system dataset to explore the anomalies [54]. In smart cities, neural network algorithms have been utilized [55] to detect attacks on IoT architecture. Similarly, neural networks were used by Legrand et al. [56] to detect outliers in the smart home large-scale dataset.

One of the most frequently used unsupervised machine-learning algorithms is K-means clustering. Janakiraman and Nielsen [38] used K-means clustering to design a detection and firewall method for anomalies for the IoT site microservices in smart cities.

A gaussian mixture model (GMM) is a probability method that has the assumption that the entire data points are produced from a mixture of a finite number of unknown parameter Gaussian distributions. The K-means clustering companied with GMM has been used in road transport by Riveiro et al. [57] to propose a visual analytics framework for road traffic anomaly detection.

The healthcare system [58] used HMM to design a non-intrusive sleep analyzer for detecting real-time sleep anomalies. In the industrial system, Zang et al. [59] proposed a special feature-extraction method for detecting anomalies within time series based on the transfer probability of a Markov chain.

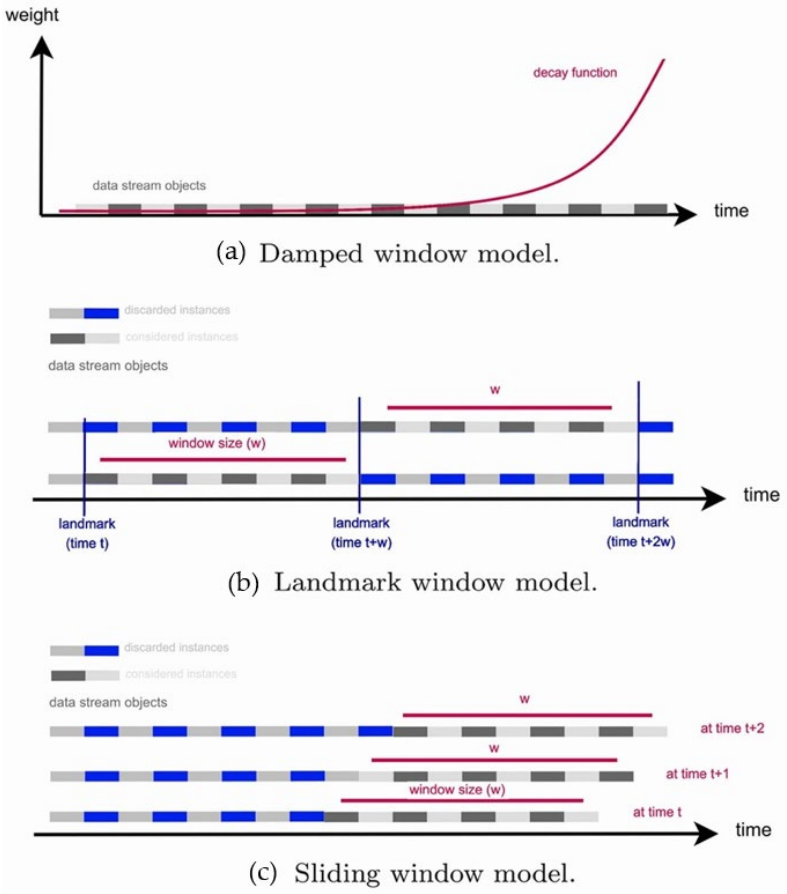
Furthermore, Kumar et al. [60] used clustering for monitoring the energy consumption of indoor office devices. In intelligent transportation, a structured sparse subspace clustering was used by He et al. [61] and proposed to detect anomalies. Similarly, Han et al. [62] designed an ANOVA-based technique to detect a vehicle's anomalous behavior.

Clustering has also been used in database management system as predictive modelling and anomaly detection. In such domain application, an incremental equivalence class transformation (i-Eclat) algorithm has been proposed to serves as the association rule mining database engine in testing frequent itemset mining (FIMI) datasets from online repository [10].

### *3.5. Window Models*

In time window modeling, the data are divided into some basic windows, and these basic windows are utilized as update units. There are three types of window models as listed below [63]:

- Fading (Damped) window model: a weight is allocated to every data point based on a fading concept, and more weights are allocated to latest data compared to old data. The usage of a damped window models reduces the impact of the old data on the mining performance (Figure 7a).
- Landmark window model: the window is defined by a particular time point called a landmark and a current time point. It is utilized for mining throughout the history of data streams (Figure 7b).
- Sliding window model: data can be counted from a certain range in the past to the present. The concept behind the "sliding window" is to do a thorough review of both the recent data points together with the summarization of the old data points (Figure 7c).




(a) Damped window model.

(b) Landmark window model.

(c) Sliding window model.

**Figure 7.** Window models.

A summary of the window models that includes their pros and cons is given in Table 3. The selection of a window model varies based on the requirements of the domain applications [63].

**Table 3.** Window models in clustering data streams.

| Window Model | Definition | Pros | Cons |
| --- | --- | --- | --- |
| Fading (damped) window model | Assigning weights to data points | Compatible for applications in which old data has a significant effect on the mining results, the effect decreases (faded) with respect to time | Infinite time window (the window collects the entire history of the data, the size of the window continues to expand as time passes) |
| Landmark window model | Analyzing the complete data stream history | Compatible for one-pass clustering algorithms | The entire data are equally relevant and the volume of data within the window will rapidly increase to un-processable sizes. |
| Sliding window model | Analyzing the very latest data points | Compatible for applications in which the interest exists only in the very latest data such as stock-marketing | Disregards part of streams |

### 3.6. Datasets

To evaluate the anomaly detection techniques, researchers use either real, synthetic, or altered real datasets. In this review we found most techniques were evaluated using real datasets, which are publicly available. One of the most commonly used datasets in the field of anomaly detection is KDD99, which has been used in many studies such as [30,36,37]. Similarly, the Numenta Anomaly Benchmark (NAB) dataset has gained popularity recently as it consists of seven categories of datasets injected with anomalies, and it has been used in [35]. Yahoo is another real-world data set, which is also publicly available and been used in [40]. Other real world benchmark datasets are also available and have been used by many researchers. However, when the data for some application domains are available, researchers tend to simulate the real data and generate data that will reflect the real-world environment, as in the case of [14,25,29]. With synthetic data, researchers have the advantage of having the data labeled, and they can inject outliers/anomalies inside the data to evaluate the detection performed with the techniques used.

### 3.7. Evaluation Criteria

To measure the performance of anomaly detection techniques, the "accuracy" metric was used in [27–29]. Nevertheless, in the event of imbalanced datasets, the reported accuracy will not offer an accurate representation of the technique's efficiency. To measure the performance more accurately, metrics such as True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN), precision, recall, and F1 scores are used [34–37]. TP offers the information regarding how many positive cases are accurately detected. TN provides the information regarding how many negative cases are accurately labeled as negative cases. FP gives the information regarding the false labeled cases as a positive case. Similarly, FN offers the information regarding the cases that are positive but falsely labeled as negative. Precision is known as the number of class members classified accurately over the total number of cases classified as class members [14]. Recall is known as the number of class members classified correctly over the total number of class members [32]. In anomaly detection, high precision and high recall are needed to develop a high-quality technique. In such scenarios, F-measure is applied to provide an equal importance to precision and recall. Recently, receiving operating characteristics, such as Area Under the Curve (AUC), have also gained popularity in measuring the anomaly detection technique's performance [39]. AUC is the two-dimensional area underneath the curve and interpreted as a probability, which the model ranks a random positive example more highly than a random negative

example. It offers robust evaluation in comparison with accuracy in cases of imbalanced datasets, usually utilized with machine learning and deep learning techniques.

## 4. Results

Based on our review, there are still many open research challenges and issues to be resolved despite the development seen within the anomaly detection research. Furthermore, future directions are needed the most on anomaly detection approaches in the data stream. Therefore, the focus of this research will be limited to highlighting open challenges related to anomaly detection in the data stream, which is presented in Section 5.

Since a massive volume of data comes in the form of data streams characterized by some problems, it is therefore necessary to address the challenging issue of detecting anomalies within evolving data streams efficiently. Most existing data stream algorithms for anomaly detection are losing their effectiveness in the existence of high-dimensional data. Therefore, it is necessary to redesign the current models to detect anomalies accurately and efficiently. More precisely, where there are many characteristics, there may be a set of anomalies that appear at a given time in only a subset of dimensions. This set of anomalies seems natural regarding a different sub-set to dimensions and/or time frame.

Addressing the anomaly detection problem in a feature-evolving data stream is another primary concern. This challenge occurs when the data evolve over time, and the features of the data change. Furthermore, new/old data dimensions appear/disappear over time. It is a promising field with several possible use cases, such as detecting anomalies in IoT devices. Moreover, another challenge is handling and processing data in real-time when data points from the data source come constantly. Historically, data streams could be massive and should be handled in one pass. In addition, the algorithm should be able to process data in given memory, where enormous amounts of data should not influence the data stream's processing capabilities. Thus, the data stream algorithm should not require unlimited memory for the unlimited data points arriving in the system. Instead, it should be able to process the data within the available memory. Furthermore, the necessity for large-scale IoT implementation is increasing rapidly, with a significant security issue. Yet, a concern arises regarding anomaly detection scalability and how machine learning algorithms can handle large-scale sets. Scalability is a significant issue faced by most of the existing anomaly detection techniques, where some of these techniques lose their efficiency when it comes to large-scale deployment.

Table 4 illustrates the summary of the reviewed anomaly detection techniques and their capability in meeting the criteria that describe a good quality anomaly detection technique. The reviewed techniques have been evaluated in their ability to perform data projection, handling noisy data, working within a limited memory and limited time. In addition, their ability to address evolving data, high-dimensional data, evolving features, and finally in scalability are addressed.

**Table 4.** Anomaly detection techniques summary.

| Techniques/Methods | Projection | Handling Noisy Data | Limited Time | Limited Memory | Handling Evolving Data | Handling High Dimensional Data | Evolving Features | Scalability |
|---|---|---|---|---|---|---|---|---|
| C_LOF [14] | | | | √ | | √ | | √ |
| AutoCloud [24] | | | | √ | √ | √ | | √ |
| TEDA Clustering [25] | | | | √ | √ | √ | | √ |
| Combination of (BDLMs) & (RBPF) [26] | | √ | | | | | | √ |
| HTM [27] | | √ | | √ | | | | |
| Artificial Neural Network [28] | | | | √ | | | | √ |
| MDADM [29] | | √ | | √ | √ | | | |
| Multi-kernel [30] | √ | √ | | | √ | √ | √ | √ |
| xStream [31] | √ | | √ | | | | | |

**Table 4.** *Cont.*

| Techniques/Methods | Projection | Handling Noisy Data | Limited Time | Limited Memory | Handling Evolving Data | Handling High Dimensional Data | Evolving Features | Scalability |
|---|---|---|---|---|---|---|---|---|
| Regression Model [32] | | | √ | | | | | |
| Super Vector Machine [33] | | √ | | √ | | | | √ |
| HTM [34] | √ | √ | | √ | √ | √ | | |
| CEDAS [36] | | √ | | √ | | | | |
| HTM [35] | | √ | √ | √ | √ | | | |
| MuDi-Stream [37] | | | | √ | | √ | | |
| Extreme Learning Machines [38] | √ | √ | | √ | √ | √ | | √ |
| AMAD [39] | √ | √ | √ | √ | √ | √ | | √ |
| LSTMs [40] | | √ | | √ | | | | |
| Autoencoder [41] | | √ | | | √ | | | √ |
| (OFAT) Deep neural network [42] | | √ | | | √ | √ | √ | √ |
| Evolving spiking neural network [43] | | √ | | | √ | | | |
| ISTL [44] | | | √ | √ | √ | | | |
| (e-SREBOM) [43] | | √ | √ | √ | √ | | | √ |

## 5. Research Challenges and Future Directions

Even though various anomaly detection techniques have been proposed in the literature, there are still several issues to be solved for anomaly detection. Currently, there is no single best technique for the problem; rather, there are several techniques that may be more applicable to certain data types and certain application domains. Below we present a summary of the major challenges found within the state-of-the-art techniques reviewed:

1. Evolving Data Stream

As a vast volume of knowledge falls in the form of data streams labeled by such anomalies, efficiently overcoming the challenging task of detecting anomalies in evolving data streams is necessary [25].

Data streams pose external detection problems, such as detecting in restricted memory and limited time, updating the data once they enter, and managing data in a changing fashion to capture the fundamental changes when detecting them [29]. Data evolution includes algorithms to adjust their configuration and parameters over time and as new knowledge arises. Detection algorithms fail to adjust to complex conditions, such as the ever-changing IoT domain, unlike static records [64].

Further, most existing are less efficient in detecting anomalies in data stream and have poor capability requirements [15]. Detecting anomalies in the IoT data stream environment, known for its evolving characteristics, results in low detection accuracy with a high false-positive rate [43]. The evolving data stream is a challenge that must be addressed in the environment of IoT anomaly detection [24,65].

2. Feature-Evolving

In a feature-evolving data source, another difficulty would be solving the issue of anomaly detection. The concern is that the data change, and the properties of the data also shift. In comparison, over time, new/old dimensions of data appear/disappear. This area is fascinating, with many potential applications, such as outlier detection in IoT systems in which the sensors periodically go off/on (representing the number of dimensions) [31].

3. Windowing

The precision is limited (windowing) because of the short data processing used based on fixed interval timing [66]. Another major challenge is determining which frequency is

ideal for retraining the models, because most of the current approaches use predefined interval timing [66,67].

4. Ensemble Approaches

Another area of growth is ensemble approaches. Ensemble methods are well established for increasing the efficacy of detection anomalies by detecting and running the accuracy of time [41]. Another worthwhile potential course of research would then be the ensemble detection of deviations, which shows great promise in boosting the detection accuracy of the algorithms. More specific models can be recommended for resolving unexplored regions. To detect anomalies within the environment of the data stream, initial attempts to examine the ensemble are recommended. However, this field of study is still unexplored and requires more comprehensive models.

5. Nature of Input Data

Many existing challenges need to be solved within IoT anomaly detection. As highlighted by Azimi et al. [68], labelled data availability is a major issue in IoT anomaly detection because the occurrence of anomalies may not be regular. In addition, obtaining the real system data is complicated and requires a lengthy process to reach the operating system data [19]. A wide gap exists in formalizing obtaining knowledge logs and sensory data flow, developing a model, and validating it in real-life environments.

During the study, many experiments have been reported, linked mainly to the usual behavior of the system [19]. The most developed methods are based on normal behavior training, and anything that differs from normal labelled data is considered anomalous. More precise and reliable techniques are required to deal with complicated datasets of real scenarios.

In addition, the availability of a suitable dataset for public anomaly detection is generally a key issue for training and validating techniques for real-time anomaly detection [69]. Such datasets must have a wide variety of new normal and abnormal behaviors, and they should be labelled clearly and constantly updated to prevent any new types of abnormal behavioral threats. Most existing datasets for anomaly detection often lack from wrong labelling, poor diversity of attacks, and compatibility with real-time detection [70]. New data sets for anomaly detection demand realistic environments with a variety of normal and abnormal scenarios. Additionally, the fundamental truth that includes anomalies must be produced to increase the credibility of the dataset when testing a new system of anomaly detection.

6. Data Complexity and Noise

Data complexity, such as imbalanced datasets, unexpected noises, and redundancy within the data, is one of the main challenges in the development of a model for anomaly detection [40]. Well-developed approaches for curating the datasets are required to collect useful information and knowledge.

7. Parameters Selection

IoT data streams are often generated from non-stationary environments with no advanced information on the data distribution, which affects the choice of a proper set of model parameters for detecting anomalies [25].

8. Data visualizations

The visualization of the anomaly analysis has highlighted the existence of a gap. New techniques and solutions are needed for the analysis of visual systems to be implemented. Therefore, these gaps related to the fields of the anomaly detection process should be explored [8].

9. Heterogeneity of data

The heterogeneous IoT sensors and devices are sources of an unlimited volume of data streams that demonstrate all types of environmental characteristics, such as light,

temperature, humidity, noise, electric current, voltage, power, etc. [28]. Such a data stream requires instant processing for handling timely and critical scenarios, such as healthcare monitoring for a patient and environmental safety monitoring [71]. Every device can transmit data many times per second, and with a large number of connected devices, a typical data processing platform might be necessary for dealing with billions of such incoming events every day [72].

10. Time Complexity

The data stream's main characteristic is its massive volume of data arriving continuously, which requires the algorithm to work in real-time. However, a major challenge would be the time complexity of detecting the anomalies [14,73,74] because there is always a trade-off between accuracy and time complexity.

11. Accuracy

Despite having the capabilities of learning algorithms to detect and identify anomalous behavior in real-time, these algorithms are still subject to optimization to improve accuracy, including the reduction in the false positive detection rate, especially in large-scale sensor networks [15,27,69,75].

12. Scalability

Scalability is another major requirement for anomaly detection algorithms because many algorithms lose their efficiency when handling a large volume of data [41].

13. High-Dimensional Data

Most current data stream algorithms for anomaly detection lose their effectiveness in the presence of high-dimensional data [25]. Therefore, accurately and efficiently redesigning existing models to detect outliers is necessary. More precisely, when many characteristics are observed, a set of outliers may appear at a given time in only a subset of dimensions. This set of outliers appears natural regarding the different subset dimensions and/or time frame.

The many features are another challenge faced by anomaly detection algorithms in selecting the most important data features [37]. Therefore, the reduction in features is critical in choosing the most important ones that display the whole data.

## 6. Conclusions

Anomaly detection has attracted significant attention among researchers in recent years, due to the advancement of sensing technologies categorized with low cost and high impact in diverse application domains. Detection of anomalies greatly eliminates functional threats, removes unseen complications, and prevents downtime of the processes. Machine learning and deep learning anomaly detection techniques play important roles in detecting data stream anomalies in various core implementations across a broad variety of IoT application domains. Yet, there are still several challenges to be addressed to solve the problem of anomaly detection. These challenges include evolving data streams, feature-evolving, windowing, ensemble approaches, nature of data, data complexity and noise, parameters selection, data visualizations, heterogeneity of data, time complexity, accuracy, scalability, and high dimensionality. This paper presents the results of a study on machine learning and deep learning techniques used for anomaly detection in IoT data streams. The review offers a complete overview of the developed techniques, nature of data, types of anomalies, detection learning modes, window models, and dataset and evaluation metrics used to measure the performance of the proposed techniques. Consequently, this can help the research community to gain detailed information about the latest developed techniques related to anomaly detection in IoT data. Further, the paper also suggests some future directions of research that can contribute to the development of new techniques, which could help in improving the anomaly detection in IoT data.

Apart from that, the paper has some limitations that are to be considered as future work as well. These limitations include investigating anomaly detection techniques other

that machine learning and deep learning, such as statistical techniques. Furthermore, preprocessing of the data used in anomaly detection techniques was also out of the scope of the paper. Finally, analyzing the anomaly detection techniques in use case manner/application domains was again out of the scope of this paper.

**Author Contributions:** Conceptualization, R.A.-a. and A.F.A.; methodology, R.A.-a.; formal analysis, R.A.-a. and R.K.M.; investigation, R.A.-a. and A.F.A.; resources, M.M., A.A.A. and M.A.A.-S. data curation, R.A.-a.; writing—original draft preparation, R.A.-a. and A.F.A.; writing—review and editing, R.A.-a., R.K.M. and M.A.A.-S.; visualization, R.A.-a.; supervision, R.K.M.; funding acquisition, M.M., M.A.A.-S. and A.A.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by Taylor's University, Malaysia, through its Taylor's Ph.D. Scholarship programs, and in part by the Universiti Malaysia Terengganu, Malaysia, under Grants number: UMT GRANT with Vot: 53290, Vot: 55191 (GGRG) and FRGS/1/2018/ICT04/UMT/02/.

**Institutional Review Board Statement:** N/A.

**Informed Consent Statement:** N/A.

**Data Availability Statement:** N/A.

**Acknowledgments:** The authors would like to thank the support received in part by Taylor's University, Malaysia, through its Ph.D. Scholarship programs, and in part by the Universiti Malaysia Terengganu, Malaysia, under Grants number: UMT GRANT with Vot: 53290, Vot: 55191 (GGRG) and FRGS/1/2018/ICT04/UMT/02/.

**Conflicts of Interest:** The authors have no conflict of interest.

## References

1. ReferencesRatasich, D.; Khalid, F.; Geissler, F.; Grosu, R.; Shafique, M.; Bartocci, E. A Roadmap Toward the Resilient Internet of Things for Cyber-Physical Systems. *IEEE Access* **2019**, *7*, 13260–13283. [CrossRef]
2. Deng, X.; Jiang, P.; Peng, X.; Mi, C. An Intelligent Outlier Detection Method with One Class Support Tucker Machine and Genetic Algorithm Toward Big Sensor Data in Internet of Things. *IEEE Trans. Ind. Electron.* **2018**, *66*, 4672–4683. [CrossRef]
3. Fadele, A.A.; Othman, M.; Hashem, I.A.T.; Yaqoob, I.; Imran, M.; Shoaib, M. A novel countermeasure technique for reactive jamming attack in internet of things. *Multimed. Tools Appl.* **2018**, *78*, 29899–29920. [CrossRef]
4. Misra, N.N.; Dixit, Y.; Al-Mallahi, A.; Bhullar, M.S.; Upadhyay, R.; Martynenko, A. IoT, big data and artificial intelligence in agriculture and food industry. *IEEE Internet Things J.* **2020**, *4662*, 1. [CrossRef]
5. Munir, M.; Siddiqui, S.A.; Dengel, A.; Ahmed, S. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series. *IEEE Access* **2018**, *7*, 1991–2005. [CrossRef]
6. Man, M.; Jusoh, J.A.; Saany, S.I.A.; Abu Bakar, W.A.W.; Ibrahim, M.H. Analysis study on R-Eclat algorithm in infrequent itemsets mining. *Int. J. Electr. Comput. Eng.* **2019**, *9*, 5446–5453. [CrossRef]
7. Mahdavinejad, M.S.; Rezvan, M.; Barekatain, M.; Adibi, P.; Barnaghi, P.; Sheth, A.P. Machine learning for internet of things data analysis: A survey. *Digit. Commun. Netw.* **2018**, *4*, 161–175. [CrossRef]
8. Vilenski, E.; Bak, P.; Rosenblatt, J.D. Multivariate anomaly detection for ensuring data quality of dendrometer sensor networks. *Comput. Electron. Agric.* **2019**, *162*, 412–421. [CrossRef]
9. Singh, M.; Pamula, R. An outlier detection approach in large-scale data stream using rough set. *Neural Comput. Appl.* **2019**, *32*, 9113–9127. [CrossRef]
10. Bakar, W.A.W.A.; Man, M.; Man, M.; Abdullah, Z. I-Eclat: Performance enhancement of Eclat via incremental approach in frequent itemset mining. *Telecomnika* **2020**, *18*, 562–570. [CrossRef]
11. Chakraborty, T.; Nambi, A.U.; Chandra, R.; Sharma, R.; Swaminathan, M.; Kapetanovic, Z.; Appavoo, J. Fall-curve: A novel primitive for IoT Fault Detection and Isolation. In Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems, Shenzhen, China, 4–7 November 2018; pp. 95–107.
12. Wu, M.; Song, Z.; Moon, Y.B. Detecting cyber-physical attacks in CyberManufacturing systems with machine learning methods. *J. Intell. Manuf.* **2019**, *30*, 1111–1123. [CrossRef]
13. Kozitsin, V.; Katser, I.; Lakontsev, D. Online Forecasting and Anomaly Detection Based on the ARIMA Model. *Appl. Sci.* **2021**, *11*, 3194. [CrossRef]
14. Yu, K.; Shi, W.; Santoro, N. Designing a Streaming Algorithm for Outlier Detection in Data Mining—An Incrementa Approach. *Sensors* **2020**, *20*, 1261. [CrossRef]
15. Ding, N.; Ma, H.; Gao, H.; Ma, Y.; Tan, G. Real-time anomaly detection based on long short-Term memory and Gaussian Mixture Model. *Comput. Electr. Eng.* **2019**, *79*, 106458. [CrossRef]

16. Salehi, M.; Rashidi, L. A Survey on Anomaly detection in Evolving Data [with Application to Forest Fire Risk Prediction]. *SIGKDD Explor. Newsl.* **2018**, *20*, 13–23. [CrossRef]
17. Donevski, M.; Zia, T. A Survey of Anomaly and Automation from a Cybersecurity Perspective. In Proceedings of the 2018 IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, United Arab Emirates, 9–13 December 2018; pp. 1–6. [CrossRef]
18. Habeeb, R.A.A.; Nasaruddin, F.H.; Gani, A.; Hashem, I.A.T.; Ahmed, E.; Imran, M. Real-time big data processing for anomaly detection: A Survey. *Int. J. Inf. Manag.* **2019**, *45*, 289–307. [CrossRef]
19. Fahim, M.; Sillitti, A. Anomaly Detection, Analysis and Prediction Techniques in IoT Environment: A Systematic Literature Review. *IEEE Access* **2019**, *7*, 81664–81681. [CrossRef]
20. Baydargil, H.; Park, J.-S.; Kang, D.-Y. Anomaly Analysis of Alzheimer's Disease in PET Images Using an Unsupervised Adversarial Deep Learning Model. *Appl. Sci.* **2021**, *11*, 2187. [CrossRef]
21. Chauhan, S.; Vig, L.; Ahmad, S. ECG anomaly class identification using LSTM and error profile modeling. *Comput. Biol. Med.* **2019**, *109*, 14–21. [CrossRef]
22. Shanthamallu, U.S.; Spanias, A.; Tepedelenlioglu, C.; Stanley, M. A Brief Survey of Machine Learning Methods and their Sensor and IoT Applications. In Proceedings of the 2017 8th International Conference on Information, Intelligence, Systems & Applications (IISA), Larnaca, Cyprus, 27–30 August 2017; pp. 1–8. [CrossRef]
23. Praveen Kumar, D.; Amgoth, T.; Annavarapu, C.S.R. Machine learning algorithms for wireless sensor networks: A survey. *Inf. Fusion* **2018**, *49*, 1–25. [CrossRef]
24. Bezerra, C.G.; Costa, B.S.J.; Guedes, L.A.; Angelov, P.P. An evolving approach to data streams clustering based on typicality and eccentricity data analytics. *Inf. Sci.* **2020**, *518*, 13–28. [CrossRef]
25. Maia, J.; Severiano, C.A.; Guimarães, F.G.; de Castro, C.L.; Lemos, A.P.; Galindo, J.C.F.; Cohen, M.W. Evolving clustering algorithm based on mixture of typicalities for stream data mining. *Future Gener. Comput. Syst.* **2020**, *106*, 672–684. [CrossRef]
26. Nguyen, L.H.; Goulet, J. Real-time anomaly detection with Bayesian dynamic linear models. *Struct. Control. Health Monit.* **2019**, *26*, 1–17. [CrossRef]
27. Song, L.; Liang, H.; Zheng, T. Real-Time Anomaly Detection Method for Space Imager Streaming Data Based on HTM Algorithm. In Proceedings of the 2019 IEEE 19th International Symposium on High Assurance Systems Engineering (HASE), Hangzhou, China, 3–5 January 2019; pp. 33–38. [CrossRef]
28. Cauteruccio, F.; Fortino, G.; Guerrieri, A.; Liotta, A.; Mocanu, D.C.; Perra, C.; Terracina, G.; Vega, M.T. Short-long term anomaly detection in wireless sensor networks based on machine learning and multi-parameterized edit distance. *Inf. Fusion* **2019**, *52*, 13–30. [CrossRef]
29. Peng, Y.; Tan, A.; Wu, J.; Bi, Y. Hierarchical Edge Computing: A Novel Multi-Source Multi-Dimensional Data Anomaly Detection Scheme for Industrial Internet of Things. *IEEE Access* **2019**, *7*, 111257–111270. [CrossRef]
30. Siahroudi, S.K.; Moodi, P.Z.; Beigy, H. Detection of evolving concepts in non-stationary data streams: A multiple kernel learning approach. *Expert Syst. Appl.* **2018**, *91*, 187–197. [CrossRef]
31. Manzoor, E.; Lamba, H.; Akoglu, L. xStream: Outlier Detection in Feature-Evolving Data Streams. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 1963–1972. [CrossRef]
32. Farshchi, M.; Weber, I.; Della Corte, R.; Pecchia, A.; Cinque, M.; Schneider, J.-G.; Grundy, J. Contextual anomaly detection for a critical industrial system based on logs and metrics. In Proceedings of the 2018 14th European Dependable Computing Conference (EDCC), Iasi, Romania, 10–14 September 2018; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2018. [CrossRef]
33. Bose, B.; Dutta, J.; Ghosh, S.; Pramanick, P.; Roy, S. D&RSense: Detection of Driving Patterns and Road Anomalies. In Proceedings of the 2018 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Bhimtal, India, 23–24 February 2018; pp. 1–7. [CrossRef]
34. Rodriguez, M.A.; Kotagiri, R.; Buyya, R. Detecting performance anomalies in scientific workflows using hierarchical temporal memory. *Future Gener. Comput. Syst.* **2018**, *88*, 624–635. [CrossRef]
35. Ahmad, S.; Lavin, A.; Purdy, S.; Agha, Z. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing* **2017**, *262*, 134–147. [CrossRef]
36. Hyde, R.; Angelov, P.; MacKenzie, A. Fully online clustering of evolving data streams into arbitrarily shaped clusters. *Inf. Sci.* **2017**, *382-383*, 96–114. [CrossRef]
37. Amini, A.; Saboohi, H.; Herawan, T.; Wah, T.Y. MuDi-Stream: A multi density clustering algorithm for evolving data stream. *J. Netw. Comput. Appl.* **2016**, *59*, 370–385. [CrossRef]
38. Janakiraman, V.M.; Nielsen, D. Anomaly detection in aviation data using extreme learning machines. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; IEEE: New York, NY, USA, 2016.
39. Xue, L.; Chen, Y.; Luo, M.; Peng, Z.; Liu, J. An anomaly detection framework for time-evolving attributed networks. *Neurocomputing* **2020**, *407*, 39–49. [CrossRef]
40. Qiu, J.; Du, Q.; Qian, C. KPI-TSAD: A Time-Series Anomaly Detector for KPI Monitoring in Cloud Applications. *Symmetry* **2019**, *11*, 1350. [CrossRef]
41. Dong, Y.; Japkowicz, N. Threaded ensembles of autoencoders for stream learning. *Comput. Intell.* **2017**, *34*, 261–281. [CrossRef]

42. Wambura, S.; Huang, J.; Li, H. Long-range forecasting in feature-evolving data streams. *Knowl. Based Syst.* **2020**, *206*, 106405. [CrossRef]
43. Xing, L.; Demertzis, K.; Yang, J. Identifying data streams anomalies by evolving spiking restricted Boltzmann machines. *Neural Comput. Appl.* **2020**, *32*, 6699–6713. [CrossRef]
44. Nawaratne, R.; Alahakoon, D.; De Silva, D.; Yu, X. Spatiotemporal Anomaly Detection Using Deep Learning for Real-Time Video Surveillance. *IEEE Trans. Ind. Informatics* **2020**, *16*, 393–402. [CrossRef]
45. Hundman, K.; Constantinou, V.; Laporte, C.; Colwell, I.; Soderstrom, T. Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining 2018, London, UK, 19–23 August 2018; pp. 387–395. [CrossRef]
46. Garg, S.; Kaur, K.; Batra, S.; Kaddoum, G.; Kumar, N.; Boukerche, A. A multi-stage anomaly detection scheme for augmenting the security in IoT-enabled applications. *Future Gener. Comput. Syst.* **2020**, *104*, 105–118. [CrossRef]
47. Cook, A.A.; Misirli, G.; Fan, Z. Anomaly Detection for IoT Time-Series Data: A Survey. *IEEE Internet Things J.* **2020**, *7*, 6481–6494. [CrossRef]
48. Goldstein, M.; Uchida, S. A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data. *PLoS ONE* **2016**, *11*, e0152173. [CrossRef]
49. Hasan, M.; Islam, M.; Zarif, I.I.; Hashem, M. Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. *Internet Things* **2019**, *7*, 100059. [CrossRef]
50. Gunupudi, R.K.; Nimmala, M.; Gugulothu, N.; Gali, S.R. CLAPP: A self constructing feature clustering approach for anomaly detection. *Future Gener. Comput. Syst.* **2017**, *74*, 417–429. [CrossRef]
51. Lu, H.; Li, Y.; Mu, S.; Wang, D.; Kim, H.; Serikawa, S. Motor Anomaly Detection for Unmanned Aerial Vehicles Using Reinforcement Learning. *IEEE Internet Things J.* **2018**, *5*, 2315–2322. [CrossRef]
52. Chen, L.-J.; Ho, Y.-H.; Hsieh, H.-H.; Huang, S.-T.; Lee, H.-C.; Mahajan, S. ADF: An Anomaly Detection Framework for Large-Scale PM2.5 Sensing Systems. *IEEE Internet Things J.* **2018**, *5*, 559–570. [CrossRef]
53. Luo, H.; Zhong, S. Gas turbine engine gas path anomaly detection using deep learning with Gaussian distribution. In Proceedings of the 2017 Prognostics and System Health Management Conference (PHM-Harbin) 2017, Harbin, China, 9–12 July 2017; IEEE: New York, NY, USA, 2017. [CrossRef]
54. Hajdarevic, A.; Dzananovic, I.; Banjanovic-Mehmedovic, L.; Mehmedovic, F. Anomaly detection in thermal power plant using probabilistic neural network. In Proceedings of the 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 25–29 May 2015; pp. 1118–1123. [CrossRef]
55. Diro, A.A.; Chilamkurti, N. Distributed attack detection scheme using deep learning approach for Internet of Things. *Future Gener. Comput. Syst.* **2018**, *82*, 761–768. [CrossRef]
56. Legrand, A.; Niepceron, B.; Cournier, A.; Trannois, H. Study of Autoencoder Neural Networks for Anomaly Detection in Connected Buildings. In Proceedings of the 2018 IEEE Global Conference on Internet of Things (GCIoT), Alexandria, Egypt, 5–7 December 2018; IEEE: New York, NY, USA, 2018; pp. 1–5. [CrossRef]
57. Riveiro, M.; Lebram, M.; Elmer, M. Anomaly Detection for Road Traffic: A Visual Analytics Framework. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2260–2270. [CrossRef]
58. Tonchev, K.; Koleva, P.; Manolova, A.; Tsenov, G.; Poulkov, V. Non-intrusive sleep analyzer for real time detection of sleep anomalies. In Proceedings of the 2016 39th International Conference on Telecommunications and Signal Processing (TSP), Vienna, Austria, 27–29 June 2016; pp. 400–404. [CrossRef]
59. Zang, D.; Liu, J.; Wang, H. Markov chain-based feature extraction for anomaly detection in time series and its industrial application. In Proceedings of the 2018 Chinese Control and Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; IEEE: New York, NY, USA, 2018; pp. 1059–1063. [CrossRef]
60. Kumar, D.; Bezdek, J.C.; Rajasegarar, S.; Palaniswami, M.; Leckie, C.; Chan, J.; Gubbi, J. Adaptive Cluster Tendency Visualization and Anomaly Detection for Streaming Data. *ACM Trans. Knowl. Discov. Data* **2016**, *11*, 1–40. [CrossRef]
61. He, Y.; Peng, Y.; Wang, S.; Liu, D.; Leong, P.H.W. A Structured Sparse Subspace Learning Algorithm for Anomaly Detection in UAV Flight Data. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 90–100. [CrossRef]
62. Han, M.L.; Lee, J.; Kang, A.R.; Kang, S.; Park, J.K. A Statistical-Based Anomaly Detection Method for Connected Cars in Internet. In *Internet of Vehicles—Safe and Intelligent Mobility*; Springer: Cham, Switzerland, 2015; pp. 89–97. [CrossRef]
63. Sayed, D.; Rady, S.; Aref, M. Enhancing CluStream Algorithm for Clustering Big Data Streaming over Sliding Window. In Proceedings of the 2020 12th International Conference on Electrical Engineering (ICEENG), Cairo, Egypt, 7–9 July 2020; IEEE: New York, NY, USA, 2020; pp. 108–114. [CrossRef]
64. Gottwalt, F.; Chang, E.; Dillon, T. CorrCorr: A feature selection method for multivariate correlation network anomaly detection techniques. *Comput. Secur.* **2019**, *83*, 234–245. [CrossRef]
65. Maciąg, P.S.; Kryszkiewicz, M.; Bembenik, R.; Lobo, J.L.; Del Ser, J. Unsupervised Anomaly Detection in Stream Data with Online Evolving Spiking Neural Networks. *Neural Netw.* **2021**, *139*, 118–139. [CrossRef]
66. Vergeles, A.; Khaya, A.; Prokopenko, D.; Manakova, N. Unsupervised Real-Time Stream-Based Novelty Detection Technique an Approach in a Corporate Cloud. In Proceedings of the 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 21–25 August 2018; IEEE: New York, NY, USA, 2018; pp. 166–170. [CrossRef]

67. Stiawan, D.; Idris, M.Y.; Malik, R.F.; Nurmaini, S.; Budiarto, R. Anomaly detection and monitoring in Internet of Things communication. In Proceedings of the 2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE), Yogyakarta, Indonesia, 5–6 October 2016; Institute of Electrical and Electronics Engineers: New York, NY, USA, 2016; pp. 1–4. [CrossRef]
68. Azimi, I.; Oti, O.; Labbaf, S.; Niela-Vilen, H.; Axelin, A.; Dutt, N.; Liljeberg, P.; Rahmani, A.M. Personalized Maternal Sleep Quality Assessment: An Objective IoT-based Longitudinal Study. *IEEE Access* **2019**, *7*, 93433–93447. [CrossRef]
69. Moustafa, N.; Hu, J.; Slay, J. A holistic review of Network Anomaly Detection Systems: A comprehensive survey. *J. Netw. Comput. Appl.* **2019**, *128*, 33–55. [CrossRef]
70. Wang, H.; Bah, M.J.; Hammad, M. Progress in Outlier Detection Techniques: A Survey. *IEEE Access* **2019**, *7*, 107964–108000. [CrossRef]
71. Lee, I. Big data: Dimensions, evolution, impacts, and challenges. *Bus. Horiz.* **2017**, *60*, 293–303. [CrossRef]
72. Alaba, F.A.; Othman, M.; Hashem, I.A.T.; Alotaibi, F. Internet of Things security: A survey. *J. Netw. Comput. Appl.* **2017**, *88*, 10–28. [CrossRef]
73. Gibert, X.; Patel, V.M.; Chellappa, R. Deep Multitask Learning for Railway Track Inspection. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 153–164. [CrossRef]
74. Santos, J.; Leroux, P.; Wauters, T.; Volckaert, B.; De Turck, F. Anomaly detection for Smart City applications over 5G low power wide area networks. In *Proceeding of the NOMS 2018—2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23–27 April 2018*; IEEE: Piscataway, NJ, USA, 2018; pp. 1–9. [CrossRef]
75. Da Costa, K.A.; Papa, J.P.; Lisboa, C.O.; Munoz, R.; de Albuquerque, V.H.C. Internet of Things: A survey on machine learning-based intrusion detection approaches. *Comput. Netw.* **2019**, *151*, 147–157. [CrossRef]