Advances in Time-Series Anomaly Detection: Algorithms, Benchmarks, and Evaluation Measures

John Paparrizos
The Ohio State University
Columbus, OH, USA
Aristotle University
Thessaloniki, Greece
paparrizos.1@osu.edu

Paul Boniol
Inria, DI ENS, PSL, CNRS
Paris, France
paul.boniol@inria.fr

Qinghua Liu
The Ohio State University
Columbus, OH, USA
liu.11085@osu.edu

Themis Palpanas
Université Paris Cité; IUF
Paris, France
themis@mi.parisdescartes.fr

Abstract

Recent advances in data collection technology, accompanied by the ever-rising volume and velocity of streaming data, underscore the vital need for time series analytics. In this regard, time-series anomaly detection has been an important activity, entailing various applications in fields such as cyber security, financial markets, law enforcement, and health care. While traditional literature on anomaly detection is centered on statistical measures, the increasing number of machine learning algorithms in recent years call for a structured, general characterization of the research methods for time-series anomaly detection. In this paper, we present a process-centric taxonomy for time-series anomaly detection methods, systematically categorizing traditional statistical approaches and contemporary machine learning techniques. Beyond this taxonomy, we conduct a meta-analysis of the existing literature to identify broad research trends. Given the absence of a one-size-fits-all anomaly detector, we also introduce emerging trends for time-series anomaly detection. Furthermore, we review commonly used evaluation measures and benchmarks, followed by an analysis of benchmark results to provide insights into the impact of different design choices on model performance. Through these contributions, we aim to provide a holistic perspective on time-series anomaly detection and highlight promising avenues for future investigation.

CCS Concepts

• **Computing methodologies** → **Anomaly detection**; • **Mathematics of computing** → **Time series analysis**.

Keywords

Time-Series Analysis; Anomaly Detection; Outlier Detection

ACM Reference Format:
John Paparrizos, Paul Boniol, Qinghua Liu, and Themis Palpanas. 2025. Advances in Time-Series Anomaly Detection: Algorithms, Benchmarks, and Evaluation Measures. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3711896.3736565

ACM ISBN 979-8-4007-1454-2/2025/08
https://doi.org/10.1145/3711896.3736565

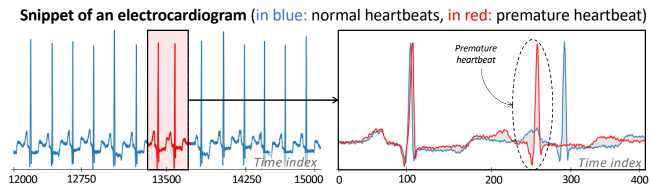


Figure 1: Example of detecting anomalies in time series.

1 Introduction

A wide range of cost-effective sensing, networking, storage, and processing solutions enable the collection of enormous amounts of measurements over time. Recording these measurements results in an ordered sequence of real-valued data points commonly referred to as *time series*. The analysis of time series has become increasingly prevalent for understanding a multitude of natural or human-made processes [108–110]. Unfortunately, inherent complexities in the data generation of these processes, combined with imperfections in the measurement systems as well as interactions with malicious actors, often result in abnormal phenomena. Such abnormal events appear subsequently in the collected data as anomalies. Considering that the volume of the produced time series will continue to rise due to the explosion of Internet-of-Things applications [78, 86], an abundance of anomalies is expected in time-series collections.

The detection of anomalies in time series has received ample academic and industrial attention for over six decades [15, 41, 80, 81, 102, 105, 107]. With the term *anomalies*, we refer to data points or groups of data points that do not conform to some notion of normality or an expected behavior based on previously observed data [47, 56]. Depending on the application, anomalies can constitute (i) noise or erroneous data, which hinders the data analysis; or (ii) data of interest. In the former case, the anomalies are unwanted data that are removed or corrected. In the latter case, the anomalies may identify meaningful events, such as failures or changes in behavior, which require analysis.

The first approaches for detecting time-series anomalies centered around using statistical tests [26]. Since then, a large number of works have appeared in this area, which is still rapidly expanding, and multiple surveys have been written to summarize the state of the art (SOTA) [10, 24]. Unfortunately, the majority of the surveys focus on general-purpose anomaly detection methods and only a portion of them briefly review methods for time-series anomaly detection. Even though traditional anomaly detection methods may treat time series as any other high-dimensional vector and attempt to detect anomalies, our focus is on approaches that are specifically designed to consider characteristics of time series.

To illustrate the importance of this point, in Figure 1, we present an example anomaly where the temporal ordering and the collective consideration of points enable the detection of the anomaly. Depending on the research community, multiple solutions have been proposed to tackle the above-mentioned challenge. Unfortunately, these areas remain mostly disconnected, using different datasets, baselines, and evaluation measures. New algorithms are evaluated only against non-representative approaches, and it is virtually impossible to find a SOTA approach for a concrete use case. Moreover, recent benchmarks have highlighted the absence of a one-size-fits-all anomaly detector [81, 107, 124], as no single method consistently outperforms others across all domains. This underscores the growing importance of automating the anomaly detection process. However, to date, no surveys have systematically reviewed this line of research.

To remedy this issue, this survey presents a novel, comprehensive, process-centric taxonomy for time-series anomaly detection. We collected a comprehensive range of algorithms in the literature and grouped them into families of algorithms with similar approaches (Section 3). Furthermore, to identify research trends, we provide statistics over time on the type and area of proposed approaches (Section 4). Then, we review recent emerging trends in time-series anomaly detection (Section 5). Additionally, we enumerate established and recent evaluation measures used to assess anomaly detection methods (Section 6). Finally, we present experimental evaluation results on benchmark datasets and discuss possible future directions (Section 7).

2 Background

In this section, we provide necessary background, including different time-series anomaly types (Section 2.1), categories of methods concerning supervision (Section 2.2), and the components of anomaly detection pipelines (Section 2.3).

2.1 Types of Anomalies in Time Series

Due to the temporality of the data, anomalies can occur in the form of a single value or collectively in the form of sub-sequences. In the specific context of point, we are interested in finding points that are far from the usual distribution of values that correspond to *healthy* states. In the specific context of sequences, we are interested in identifying anomalous sub-sequences, which are usually not outliers but exhibit rare and, hence, anomalous patterns. In real-world applications, such a distinction between points and sub-sequences becomes crucial because even though individual points might seem normal against their neighboring points, the shape generated by the sequence of these points may be anomalous. Formally, we define three types of time-series anomalies: *point*, *contextual*, and *collective* anomalies. *Point* anomalies refer to data points that deviate remarkably from the rest of the data, as is shown in Figure 2(a). Figure 2(b) illustrates *Contextual* anomalies in which data points lie within the expected range of the distribution (in contrast to point anomalies) but deviate from the expected data distribution given a specific context (e.g., a window). *Collective* anomalies refer to sequences of points that do not repeat a typical (previously observed) pattern. Figure 2(c) depicts a synthetic collective anomaly. The first two categories, namely, point and contextual anomalies, are called

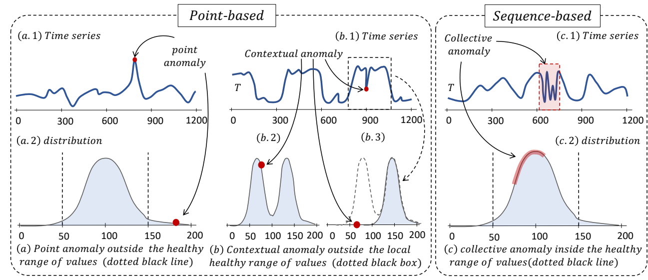


Figure 2: Synthetic illustration of three time series anomaly types: (a) point; (b) contextual; and (c) collective anomalies.

point-based anomalies. whereas, *collective* anomalies are referred to as *sequence-based* anomalies.

2.2 Unsupervised vs. Supervised Methods

An approach can be categorized into three types based on the level of prior knowledge available: (i) experts do not have information on what anomalies to detect; (ii) experts only have information on the expected normal behaviors; (iii) experts have precise examples of which anomalies they have to detect (and have a collection of known anomalies). This gives rise to the distinction between unsupervised (i), semi-supervised (ii), and supervised methods (iii).

2.3 Anomaly Detection Pipelines

An anomaly detection pipeline consists of four parts: *data preprocessing*, *detection method*, *scoring*, and *post-processing*. The data processing step represents how the anomaly detection method processes the time series data at the initial step. We have noticed all the anomaly detection models are somehow based on a windowed approach by converting the time series data into a matrix with rows of sliding window slices of the original time series. Subsequent to this transformation, various detection methods can be applied to the windowed data, producing anomaly scores that quantify the abnormality of each data point. A higher anomaly score implies a greater likelihood of abnormality. Following this, the post-processing step extracts anomalous points or intervals based on the computed anomaly scores. Typically, a threshold is defined, and any data points exceeding this threshold are classified as anomalies.

3 Time-Series Anomaly Detection Taxonomy

In this section, we describe our proposed taxonomy. We divide methods into three core categories: (i) *Distance-based*, (ii) *Density-based*, and (iii) *Prediction-based*.

The *distance-based* family contains methods that focus on analyzing sub-sequences to detect anomalies in time series, mainly by utilizing distance measures to a given model. Instead of measuring nearest-neighbor distances, *density-based* methods focus on detecting globally normal distributions and isolated behaviors. The *prediction-based* methods aim to train a model (on anomaly-free time series) to reconstruct the normal data or predict the future expected normal points. In the following sections, we break down each category into subcategories. Figure 3 illustrates our proposed process-centric taxonomy. Note that the second-level categorization is not mutually exclusive. A model might compress the time series data while adopting a discord-based identification strategy.

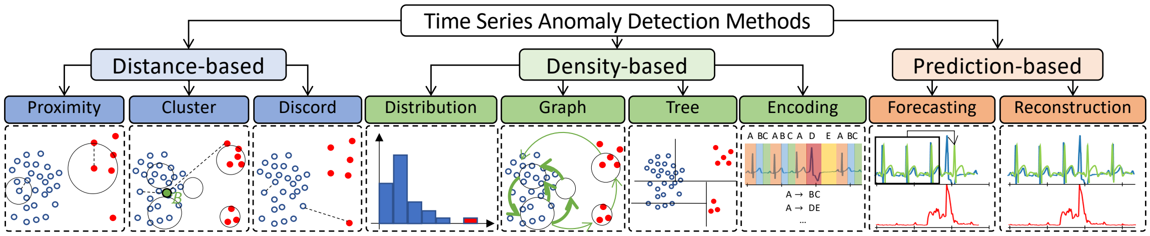


Figure 3: Process-centric Time-Series Anomaly Detection Taxonomy.

Table 1: Summary of the Distance-based anomaly detection methods. I: Univariate, M: Multivariate; S: Supervised, Se: Semi-Supervised and U: Unsupervised.

	Second Level	Prototype	Dim	Method	Stream
KNN [56]	Proximity-based	Nearest Neighbor	M	U	×
KnorrSeq2 [103]	Proximity-based	Nearest Neighbor	M	U	×
LOF [20]	Proximity-based	LOF	M	U	×
COF [138]	Proximity-based	LOF	M	U	×
LOCI [104]	Proximity-based	LOF	M	U	✓
ILOF [115]	Proximity-based	LOF	M	U	✓
DILOF [93]	Proximity-based	LOF	M	U	✓
HSDE [74]	Proximity-based	LOF	I	U	×
k-Means [56]	Clustering-based	k-Means	M	U	×
Hybrid-k-Means [131]	Clustering-based	k-Means	M	U	×
DeepkMeans [?]	Clustering-based	k-Means	M	Se	×
DBSCAN [122]	Clustering-based	DBSCAN	M	U	×
DBStream [53]	Clustering-based	DBSCAN	M	U	✓
MCOD [67]	Clustering-based	-	I	U	×
CBLOF [57]	Clustering-based	LOF	M	U	×
sequenceMiner [22]	Clustering-based	-	I	U	×
NorM (SAD) [13]	Clustering-based	NormA	I	U	×
NormA [14]	Clustering-based	NormA	I	U	×
SAND [18]	Clustering-based	NormA	I	U	✓
TARZAN[64]	Discord-based	-	I	S	×
HOT SAX [63]	Discord-based	-	I	U	×
DAD [159]	Discord-based	-	I	U	×
AMD [157]	Discord-based	-	I	U	×
STAMPI [162]	Discord-based	Matrix Profile	M	U	✓
STOMP [173]	Discord-based	Matrix Profile	M	U	×
MERLIN [?]	Discord-based	Matrix Profile	I	U	×
MERLIN++ [94]	Discord-based	Matrix Profile	I	U	×
SCRIMP [172]	Discord-based	Matrix Profile	I	U	×
SCAMP [174]	Discord-based	Matrix Profile	I	U	×
VALMOD [77]	Discord-based	Matrix Profile	I	U	✓
DAMP [82]	Discord-based	Matrix Profile	I	U	✓
LAMP [175]	Discord-based	Matrix Profile	I	Se	✓

In this case, the model falls within two different sub-categories. In the table of methods, only one of the second-level will be listed to give a clearer representation.

3.1 Distance-based Methods

As its name suggests, the distance-based method detects anomalies purely from the raw time series using distance measures. Given two sequences (or univariate time series), $A \in \mathbb{R}^\ell$ and $B \in \mathbb{R}^\ell$, of the same length, ℓ, we define the distance between A and B as $d(A, B) \in \mathbb{R}$, such as $d(A, B) = 0$ when A and B are the same. Different definitions of d exist in the literature. The widely used classical distance is the Euclidean distance or the z-normalized Euclidean distance (euclidean distance with sequences of mean values equal to 0 and standard deviations equal to 1). Then, Dynamic Time Wrapping (DTW) is commonly used to cope with misalignment issues. Overall, the distance-based algorithms merely treat the numerical value of the time series as it is without further modifications, such as removing seasonality or introducing a new structure built on the data. Within the Distance-based models, there are three second-level categories: proximity-based, clustering-based, and discord-based models. A detailed listing of methods under these subcategories is demonstrated in Table 2.

The **proximity-based** model measures proximity by calculating the distance of a given sub-sequence to its close neighborhood. The isolation of a sub-sequence regarding its closest neighbors is the main criterion to consider if this sub-sequence is an anomaly or not. This notion of isolation about a given neighborhood has been proposed for non-time series data. Thus, the methods contained in this category have been introduced for the general case of multi-dimensional outlier detection. In our specific case, the sub-sequence of a time series can be considered a multi-dimensional point with the number of dimensions equal to the length of the sub-sequence. The most commonly used proximity-based approach is the Local Outlier Factor (LOF) [21], which measures the degree of being an outlier for each instance. Unlike the previous proximity-based models, which directly compute the distance of sub-sequences, LOF depends on how the instance is isolated to the surrounding neighborhood. This method aims to solve the outlier detection task where an outlier is considered as *"an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism"* (Hawkins definition [56]). This definition is coherent with the anomaly detection task in time series where the *different mechanism* can be either an arrhythmia in an electrocardiogram or a failure in the components of an industrial machine. In the past decade, researchers also suggest many variants of the LOF. COF [138], for example, is a connectivity-based variant of LOF. It indicates how far away a point shifts from a pattern, adjusting the notion of isolation to not depend on the density of data clouds. LOCI [104] is another LOF-like algorithm that utilizes different statistics (correlation integral and MDEF) to infer individual points' isolation. Other LOF variants are the ILOF [115] and DILOF [93], which are able to detect anomalies incrementally.

The **clustering-based** model infers anomalies from a cluster partition of the time series sub-sequences. In practice, the anomaly score is calculated by the non-membership of a sub-sequence of each of the clusters learned by the model. Other considerations, such as cluster distance and cluster capacity, can also be considered. The clustering issue is related to the anomaly detection problem in that points may either belong to a cluster or be deemed anomalies. In practice, the fact that a sub-sequence belongs or not to a cluster is assessed by the computation of the distance between this sub-sequence and the cluster centroid or medoid. CBLOF [57] is a LOF-based clustering algorithm, which first clusters the data and then assigns the CBLOF factor to each entry to measure both the size and relative of and among the individual clusters. More recently, NormA [14] is a clustering-based algorithm that summarizes the time series with a weighted set of sub-sequences. The Normal set (weighted collection of sub-sequences to feature the training dataset) results from a clustering algorithm (Hierarchical), and the weights are derived from cluster properties (cardinality, extra-distance clustering, time coverage). An extension of NormA, called SAND [18], has been proposed for streaming time-series anomaly detection. The main difference between NormA and SAND is the mechanism to update the weight in a streaming fashion, but also the clustering step that is performed using the k-shape method [106] instead of a hierarchical clustering method.

The **discord-based** model tries to identify efficiently specific types of sub-sequences in the time series named discord. Formally, a sub-sequence A (or a given length ℓ) is a discord if the distance between its nearest neighbor is the largest among all the nearest neighbors' distances computed between sub-sequences of length ℓ in the time series. Overall, similar to proximity-based methods, the isolation of a sub-sequence regarding its closest neighbors is the main criterion to consider if this sub-sequence is an anomaly or not. However, on contrary to proximity-based methods, discord-based methods have been introduced for the specific case of anomaly detection in time series. Thus, as such methods introduced efficient processes for time series distance computation specifically, we group them into one different sub-category. Matrix Profile [161] represents time series as a matrix of closest neighbor distances. Compared to its predecessor, Matrix Profile proposed a new metadata time series computed effectively, capable of providing various valuable details about the examined time series, such as discords. A family of Matrix Profile anomaly detection methods has also been proposed in the last decade. STAMP [162] proposed an algorithm that can provide an accurate answer at any time during the full computation with time complexity of $O(n^2 log(n))$. STAMPI [162] not only performs the standard all-pairs-similarity-join of sub-sequences for matrix profile methods but also adapts the method incrementally to accommodate streaming purposes. Moreover, MERLIN [**?**] and MERLIN++ [94] have been proposed to identify discords of arbitrary length. Finally, DAMP [82] is able to work on online settings while scaling to fast-arriving streams.

3.2 Density-based Methods

The density-based methods do not treat the time series as simple numerical values but imbue them with more complex architecture. The density-based method processes time series data on top of a representation of the time series that aims to measure the density of the points or sub-sequence space. Such representation varies from graphs, trees, and histograms to a grammar induction rule. There are four second-level categories: distribution-based, graph-based, tree-based, and encoding-based. A comprehensive enumeration of methods within these subcategories is in Table 2.

A **distribution-based** anomaly detection approach is building a distribution from statistical features of the points or sub-sequences of the time series. By examining the distributions of features of the normal sub-sequences, the distribution-based approach tries to recover relevant statistical models. It uses them to infer the abnormality of the data. The One-Class Support Vector Machine (SVM) is a representative distribution-based anomaly detection method that aims to separate instances from the origin while maximizing the margin of separation. This separation can be achieved either through a hyperplane-based approach [126] or a spherical boundary approach [140]. Anomalies are identified as data points that lie far from the decision boundary, indicating deviations from the learned normal data distribution.

Table 2: Summary of the Density-based anomaly detection methods. I: Univariate, M: Multivariate; S: Supervised, Se: Semi-Supervised and U: Unsupervised.

	Second Level	Prototype	Dim	Method	Stream
FAST-MCD [119]	Distribution-based	MCD	M	Se	×
MC-MCD [54]	Distribution-based	MCD	M	Se	×
OCSVM [84]	Distribution-based	SVM	M	Se	×
AOSVM [48]	Distribution-based	SVM	M	U	✓
Eros-SVMs [70]	Distribution-based	SVM	M	Se	×
S-SVM [9]	Distribution-based	SVM	I	Se	×
MS-SVDD [152]	Distribution-based	SVM	M	Se	×
NetworkSVM [167]	Distribution-based	SVM	M	Se	×
HMAD [52]	Distribution-based	SVM	I	Se	×
DeepSVM [149]	Distribution-based	SVM	M	U	×
HBOS [46]	Distribution-based	-	M	U	×
COPOD [75]	Distribution-based	-	M	U	×
ConInd [3]	Distribution-based	-	M	Se	×
MGDD [135]	Distribution-based	-	M	U	✓
OC-KFD [117]	Distribution-based	-	M	U	×
SmartSifter [156]	Distribution-based	-	M	U	✓
MedianMethod [8]	Distribution-based	-	I	U	✓
S-ESD [59]	Distribution-based	ESD	I	U	×
S-H-ESD [59]	Distribution-based	ESD	I	U	×
SH-ESD+ [143]	Distribution-based	ESD	I	U	×
TwoFinger [90]	Graph-based	-	I	Se	×
GeckoFSM [121]	Graph-based	-	M	S	×
Series2Graph [16]	Graph-based	Series2Graph	I	U	×
DADS [125]	Graph-based	Series2Graph	I	U	×
IForest [79]	Tree-based	IForest	M	U	×
IF-LOF [30]	Tree-based	IForest/LOF	M	U	×
Extended IForest [55]	Tree-based	IForest	M	U	×
Hybrid IForest [91]	Tree-based	IForest	M	Se	×
SurpriseEncode [23]	Encoding-based	-	M	U	×
GranmmarViz [127]	Encoding-based	-	I	U	×
Ensemble GI [43]	Encoding-based	-	I	U	×
PST [136]	Encoding-based	Markov Ch.	M	U	×
EM-HMM [111]	Encoding-based	Markov Ch.	M	Se	✓
LaserDBN [100]	Encoding-based	Bayseian Net.	M	Se	×
EDBN [113]	Encoding-based	Bayseian Net.	M	Se	×
KDE-EDBN [114]	Encoding-based	Bayseian Net.	M	Se	×
PCA [128]	Encoding-based	PCA	M	Se	×
RobustPCA [101]	Encoding-based	PCA	M	U	×
DeepPCA [25]	Encoding-based	PCA	M	Se	×
POLY [160]	Encoding-based	-	I	U	×
SSA [160]	Encoding-based	-	I	U	×

A **graph-based** method represents the time series and the corresponding sub-sequences as a graph. The nodes and edges represent the different types of sub-sequences (or representative features) and their evolution in time. For instance, the nodes can be sets of similar sub-sequences (using a predefined distance measure), and the edge weights can be the number of times a sub-sequence of a given node has been followed by a sub-sequence of another node. The detection of anomalies is then achieved using characteristics of the graph, such as the node and edge weights, but also the degree of the nodes. One approach is to convert the time series into a directed graph with nodes representing the usual types of subsequences and edges representing the frequency of the transitions between types of subsequences. Series2Graph [17] is building such kinds of graphs. Moreover, an extension of Series2Graph, named DADS [125], proposes a distributed implementation and, therefore, a much more scalable method for large time series.

A **tree-based** approach aims to divide the point or sub-sequence of a time series using trees. For instance, such trees can be used to split different points or sub-sequences based on their similarity. The detection of anomalies is then based on the statistics and characteristics of the tree, such as its depth. Isolation Forest (IForest) is density-based and the most famous Tree-based approach for anomaly detection. IForest tries to isolate the outlier from the rest [79]. The key idea remains that, in a normal distribution, anomalies are more likely to be isolated (i.e., requiring fewer random partitions to be isolated) than normal instances. Other IForest algorithms have also been proposed recently. Extended IForest [55] is an extension of the traditional method, which removes the branching bias using hyperplanes with random slopes. The random sloping hyperplanes enable an unbiased selection of features free of the branching structure within the dataset. Hybrid IForest [91] is another improvement, enabling a supervised setting and eliminating the dataset's potential confounding due to unbalanced clusters. Finally, IF-LOF [30] combines IForest and LOF by applying IForest and then utilizes LOF to refine the results, which speeds up the process.

A **encoding-based** anomaly detection model compresses or represents the time series into different forms of symbols. It suggests that a time series can be interpreted as a sequence of context-free, discrete symbols or states. For instance, anomalies can be detected by using grammar rules with the symbols extracted from the time series. One approach is to encode and represent the time series with its principal components. PCA [128] investigates the major components of the time series that contribute the most to the covariance structure. The anomaly score is measured by the sub-sequences distance from 0 along the principal components weighted by their eigenvalues. A standard routine is to pick q significant components that can explain 50% variations of the time series and r minor components that explain less than 20% variations. A point is an anomaly if its values of major principles components have a weighted sum exceeding the threshold its minor one has. RobustPCA [101] aims to recover the principal matrix by decomposing the original covariance matrix. Moreover, DeepPCA [25] extends RobustPCA by first using an autoencoder to map the time series into a latent space, then applying PCA to detect anomalies.

3.3 Prediction-based Methods

Prediction-based methods aim to detect anomalies by predicting the expected normal behaviors based on a training set of time series or sub-sequences (containing anomalies or not). For instance, some methods will be trained to predict the next value or sub-sequence based on the previous one. Then, the prediction error is used as an anomaly score. The underlying assumption of prediction-based methods is that normal data are easier to predict, while anomalies are unexpected, leading to higher prediction errors. Such assumptions are valid when the training set contains no or few time series with anomalies. Therefore, prediction-based methods are usually more optimal under semi-supervised settings. An enumeration of methods within these subcategories is in Table 3.

The **forecasting-based** method is a model that, for a given index or timestamp, takes as input points or sub-sequences before this given timestamp and predicts its corresponding value or sub-sequence. In other words, such methods use past values as input to predict the following one. The forecasting error (the difference between the predicted and the real value) is used as an anomaly score. Indeed, such forecasting error is representative of the expectation of the current value based on the previous points or sub-sequences. The larger the error, the more unexpected the value, and thus, potentially abnormal. Forecasting-based approaches assume that the training data (past values or sub-sequences) is almost anomaly-free. Thus, such methods are mostly semi-supervised. Long Short-Term Memory (LSTM) [60] network has been demonstrated to be particularly efficient in learning inner features for sub-sequences classification or time series forecasting. Such a model can also be used for anomaly detection purposes [40, 89]. A stacked LSTM model is trained on *normal* parts of the data. The objective is to predict the future point or the sub-sequence using the historical sub-sequence. Consequently, the model will be trained to forecast a healthy state of the time series, and, therefore, will fail to forecast when it will encounter an anomaly. Telemanom [61] focuses on multivariate time series, where an LSTM network is trained for each channel. The prediction error is further smoothed over time, and low errors are pruned retroactively. RePad [71] considers short-term historical data points to predict future anomalies in streaming data.

In addition to LSTM, the Gated Recurrent Unit (GRU) has also been employed for anomaly detection. MTAD-GAT [168] is the first example of anomaly detection methods based on GRU units. The model utilizes two parallel graph attention layers to preprocess the time series and then implements a GRU network to reconstruct and predict the next values. AD-ITL [151] uses seasonal and raw features as input. The input time series is first used to extract seasonal features and further fed to the GRU network. The GRU then predicts each value of the window, and Local Trend Inconsistency is used as a measure of the error to assess the abnormality between predicted and actual values. Finally, it is important to note that forecasting-based approaches represent a broad concept requiring a model to predict future values based on historical data. Consequently, any regression-based method can be employed as a forecasting approach for anomaly detection.

Table 3: Summary of the Prediction-based anomaly detection methods. I: Univariate, M: Multivariate; S: Supervised, Se: Semi-Supervised and U: Unsupervised.

	Second Level	Prototype	Dim	Method	Stream
ES [129]	Forecasting-based	-	I	Se	×
DES [129]	Forecasting-based	-	I	Se	×
TES [129]	Forecasting-based	-	I	U	×
ARIMA [120]	Forecasting-based	ARIMA	I	U	✓
NoveltySVR [83]	Forecasting-based	SVM	I	U	✓
PCI [163]	Forecasting-based	ARIMA	I	U	✓
OceanWNN [145]	Forecasting-based	-	I	Se	×
MTAD-GAT [168]	Forecasting-based	GRU	M	Se	✓
AD-LTI [151]	Forecasting-based	GRU	M	Se	✓
CoalESN [99]	Forecasting-based	ESN	M	Se	✓
MoteESN [27]	Forecasting-based	ESN	I	Se	✓
HealthESN [29]	Forecasting-based	ESN	I	Se	×
Torsk [58]	Forecasting-based	ESN	M	U	✓
LSTM-AD [88]	Forecasting-based	LSTM	M	Se	×
DeepLSTM [28]	Forecasting-based	LSTM	I	Se	×
DeepAnT [92]	Forecasting-based	LSTM	M	Se	×
Telemanom★ [61]	Forecasting-based	LSTM	M	Se	×
RePAD [71]	Forecasting-based	LSTM	M	U	×
NumentaHTM [2]	Forecasting-based	HTM	I	U	✓
MultiHTM [148]	Forecasting-based	HTM	M	U	✓
RADM [36]	Forecasting-based	HTM	M	Se	✓
MAD-GAN [72]	Reconstruction-based	GAN	M	Se	✓
VAE-GAN [98]	Reconstruction-based	GAN	M	Se	×
TAnoGAN [7]	Reconstruction-based	GAN	M	Se	×
USAD [4]	Reconstruction-based	GAN	M	Se	×
EncDec-AD [87]	Reconstruction-based	AE	M	Se	×
LSTM-VAE [112]	Reconstruction-based	AE	M	Se	✓
DONUT [153]	Reconstruction-based	AE	I	Se	×
BAGEL [73]	Reconstruction-based	AE	I	Se	×
OmniAnomaly [134]	Reconstruction-based	AE	M	Se	×
MSCRED [166]	Reconstruction-based	AE	I	U	×
VELC [165]	Reconstruction-based	AE	I	Se	×
CAE [44]	Reconstruction-based	AE	I	Se	×
DeepNAP [65]	Reconstruction-based	AE	M	Se	✓
STORN [130]	Reconstruction-based	AE	M	Se	✓
Anomaly Transformer [154]	Reconstruction-based	Transformer	M	Se	×
TranAD [141]	Reconstruction-based	Transformer	M	Se	×
DCdetector [158]	Reconstruction-based	Transformer	M	Se	×
MEMTO [132]	Reconstruction-based	Transformer	M	Se	×
SARAD [33]	Reconstruction-based	Transformer	M	Se	×

The **reconstruction-based** method corresponds to a model that compresses the input time series (or sub-sequence) into a latent space (smaller than the input size) and is trained to reconstruct the input time series from the latent space. The difference between the input time series and the reconstructed one (named the reconstruction error) is used as an anomaly score. As for forecasting-based methods, the larger the error, the more unexpected the value, and thus, the more potentially abnormal. Moreover, as the reconstruction error is likely to be small for time series used to train the model, such reconstruction methods are optimal in semi-supervised settings. Autoencoder is a type of artificial neural network used to learn to reconstruct the dataset given as input using a smaller encoding size to avoid identity reconstruction. It will try to learn the best latent representation using a reconstruction loss. Therefore, it will learn to compress the dataset into a shorter code and then uncompress it into a dataset that closely matches the original. The reconstruction error can be used as an anomaly score for the specific anomaly detection task. As the model is trained on the non-anomalous sub-sequence of the time series, it is optimized to reconstruct the normal sub-sequences. Therefore, all the sub-sequences far from the training set will have a bigger reconstruction error. EncDec-AD [87] is the first model that implements an encoder-decoder by using reconstruction error to score anomalies. LSTM-VAE [112] and MSCRED [166] are similar to EncDec-AD but use LSTM and Convolutional LSTM cells in the AutoEncoder architecture. Similarly, OmniAnomaly [134] extends the autoencoder-based methodology by employing GRU and planar normalizing flow for improved anomaly detection.

In addition, Generative Adversarial Network (GAN), which was initially proposed for image generation purposes [49], can also be used for the detection of anomalies. It has two components: (i) one to generate new time series and (ii) one to discriminate the existing time series. For anomaly detection, the generator is trained to produce subsequences labeled as normal, and the discriminator is trained to discriminate the anomalies. Several anomaly detection methods based on GAN have been proposed in the literature, such as MAD-GAN [72], USAD [4], and TAnoGAN [7]. These approaches train GAN on the normal sections of the time series. The anomaly score is based on the combination of discriminator and reconstruction loss. VAE-GAN [98] is another GAN-based model that combines GAN and Variational AutoEncoder. More specifically, the generator is a VAE, which further competes with the discriminator. The anomaly score is computed as the previous two.

Transformers [142] have demonstrated remarkable performance in processing sequential data, spanning natural language tasks [35] and computer-vision applications [38]. For time-series analysis, they leverage the self-attention mechanism to capture long-range temporal dependencies [146]. Unlike RNNs, Transformers process the entire sequence in parallel. Due to the rarity of anomalies, establishing meaningful associations between abnormal points and the overall time series remains highly challenging. AnomalyTransformer [154] addresses this by introducing an "Anomaly-attention" mechanism, which extends self-attention into a two-branch structure to separately capture both prior-association and series-association for each time point. Similarly, TranAD [141] relies on focus score-based self-conditioning to achieve robust multi-modal feature extraction while employing adversarial training for

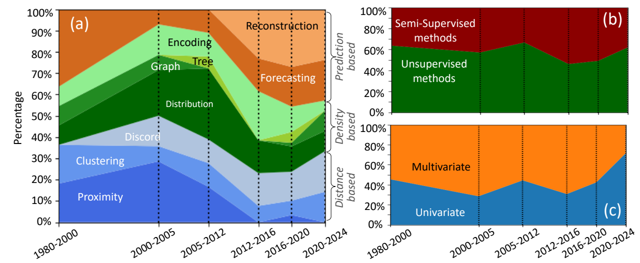


Figure 4: Evolution of anomaly detection methods: (a) Number of methods per second-level category, (b) Distribution by learning paradigm (Unsupervised or Semi-Supervised), (c) Capability to handle Univariate or Multivariate time series.

enhanced stability. DCdetector [158] proposes a dual attention asymmetric design that creates a permuted environment and uses pure contrastive loss to guide the learning process, thus yielding a permutation-invariant representation with superior discrimination capabilities. To address over-generalization, MEMTO [132] integrates a novel memory module that learns how each memory item should be updated in response to incoming data. Finally, extending beyond solely temporal modeling, SARAD [33] incorporates spatial information in addition to autoencoding errors to improve both anomaly detection and diagnosis.

4 Evolution of Anomaly Detection Methods

Until now, we have described the main methods proposed in the literature to detect anomalies in time series. We grouped them into three first-level categories and nine second-level categories. However, these first- or second-level categories do not share the same distribution over time. Figure 4(a) illustrates the number of proposed methods across different time intervals.

We first observe that the number of methods proposed yearly was constant between 1990 and 2016. The number of methods proposed in the literature significantly increased after 2016. This first confirms the growing academic interest in the topic of time-series anomaly detection. We can then dive into the second-level categories, and we observe that the significant increase in methods proposed is caused mainly by the prediction-based approach and, more specifically, by LSTM and AutoEncoders-based approaches. Between 2020 and 2024, such methods represent almost 50% of the newly introduced anomaly detection methods.

We can then measure the evolution of the number of unsupervised and semi-supervised methods over the years. The latter is illustrated in Figure 4(b). We observe that 65% of the anomaly detection methods proposed in the literature were unsupervised between 1980 and 2000, whereas 50% of the methods proposed between 2012 and 2018 were unsupervised. Finally, we can inspect the evolution of the number of methods proposed in the literature that can handle univariate or multivariate time series. Figure 4(c) shows the number of methods for multivariate and univariate time series per interval of years listed on the x-axis.

Surprisingly, we observe that most of the methods proposed between 1990 and 2016 were proposed for multivariate time series, whereas, in the last three years, most of the proposed methods are for univariate time series. However, after a deep inspection, most of the methods proposed before 2016 were designed for point anomaly detection (i.e., well-defined problems for multivariate time series). The recent interest in sub-sequence anomaly detection, joined by the fact that the subsequence anomaly detection problem for multivariate time series is harder to define, leads to a significant increase in methods for univariate time series.

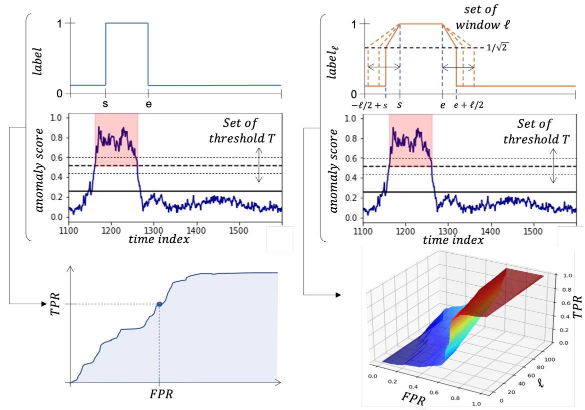
5 Emerging Trends for Anomaly Detection

In recent years, there has been a paradigm shift driven by the emergence of foundation models (FM) [11]. These models exhibit impressive few-shot or even zero-shot generalization capabilities across a broad spectrum of downstream tasks, often surpassing task-specific models. Within this evolving landscape, there are two main categories of works: (i) adapting large language models (LLMs) for time-series anomaly detection and (ii) leveraging foundation models pre-trained on large-scale time-series data to support a variety of time-series applications. Representative of the first direction, OFA [170] fine-tunes the existing GPT backbone using time-series data. Meanwhile, in the second direction, MOMENT [51] offers a family of foundation models designed for general-purpose time-series analysis, trained via a masked time-series modeling strategy. Similarly, UniTS [42] incorporates a modified transformer block to learn universal time-series representations and employs task tokenization to merge predictive and generative objectives.

In addition to this line of research, another emerging approach involves prompting LLMs to directly perform anomaly detection tasks. In this setup, an LLM is given a textual prompt (e.g., "Identify anomalies in this time series") along with the corresponding time-series data, expecting it to pinpoint anomalous segments. However, recent studies [31, 171] reveal that while LLMs detect trivial anomalies, they struggle with complex real-world cases, perform better with visual rather than textual inputs, and lack structured reasoning in anomaly detection. Additionally, LLMs' anomaly recognition does not always align with human intuition, detecting obscure patterns while missing obvious anomalies.

Despite the growing number of anomaly detection algorithms, no single stand-alone detector can consistently outperform others across different domains. A model that achieves good performance on one dataset may perform poorly on another. This raises a critical question: *How can we automate time-series anomaly detection for reliable, adaptable results?*

One approach is to evaluate model effectiveness without relying on labeled anomalies. Unsupervised Evaluation Curves [45] eliminate label dependence by using Mass-Volume and Excess-Mass curves instead of ROC or PR curves. Clustering Quality [96] applies clustering metrics, such as Silhouettes [118], to evaluate anomaly score separation without labeled data. Model Centrality [76] ranks detectors based on their proximity to an assumed ground truth using Kendall's τ distance, though it may cluster poor detectors together. Synthetic Anomaly Injection [50] evaluates models on data with artificial anomalies, with some approaches incorporating STL decomposition [32] for more realistic synthetic datasets, though simulated data may not fully reflect real-world scenarios. Finally, TSADAMS [50] refines rankings by aggregating results from multiple unsupervised metrics using Kemeny rank aggregation [68] and robust variants of the Borda method [19].



(a) AUC (b) Volume Under the Surface

Figure 5: Illustration of evaluation measures for AD.

Another line of research leverages meta-learning [6] to guide model selection using historical datasets with labeled anomalies. These approaches require historical datasets with labeled anomalies to guide the selection of the most suitable model for new data. Given a new dataset, the model selector predicts the best-performing model among the candidates. These methods are categorized based on their optimization functions for training model selectors. Simple meta-learners use direct search strategies: ARGOSMART [97] selects the model that performed best on the most similar past dataset, while ISAC [62] clusters historical datasets and selects the best model from the nearest cluster. Optimization-based meta-learners learn task similarity through advanced performance estimation techniques. MetaOD [169] employs collaborative filtering with matrix factorization to estimate model performance on new datasets. MSAD [137] formulates model selection as a classification problem, training a classifier to recommend the best detector for new time series. Meanwhile, SATzilla [155], UReg [95], and CFact [95] frame it as a regression task, predicting each model's expected performance to make selection. Unlike classification-based methods, regression-based approaches provide both the recommended model and its expected performance, offering greater interpretability.

Additionally, ensemble learning improves anomaly detection by integrating multiple detectors, enhancing robustness while mitigating individual model weaknesses [37]. Methods fall into two categories: (i) aggregating scores from all models and (ii) selecting and combining a subset. Outlier Ensemble [1] introduces AVG (mean of scores), MAX (highest score per point), and AOM (average of maximums) to balance variance and bias. SELECT [116] strategically selects ensemble components using Vertical and Horizontal selection. IOE [85] iteratively refines anomaly scores by averaging the closest matches to a pseudo-ground truth. HITS [85] ranks detectors based on hubness scores [66].

6 Evaluating Anomaly Detection Methods

The measures used to evaluate anomaly detection vary significantly in terms of their characteristics. Consequently, evaluating and selecting the most appropriate method for a given scenario has emerged as a major challenge in this field. In this section, we present an overview of evaluation measures used to assess the performance of anomaly detectors and categorize them based on the requirement of threshold setting.

Threshold-based evaluation requires setting a threshold to classify each point as an anomaly or not based on the anomaly score. Generally, a higher anomaly score value indicates a more abnormal point. After setting the threshold, we can classify the time points as either normal or abnormal based on whether they exceed the threshold. By comparing prediction to the true-labeled anomalies, the points can fall into one of the following four categories: True Positive, True Negative, False Positive, and False Negative. Given these four categories, several point-wise evaluation measures can be used to assess the performance, such as Precision, Recall and F-Score. These metrics ignore the sequential nature of time series. A range-based measure [139] was recently proposed to address the shortcomings of point-based measures. The point-based Precision and Recall can be extended to calculating range-based F-score.

Two threshold-independent metrics commonly employed in classification can be adapted for anomaly detection: AUC-PR [39] and AUC-ROC [34] as depicted in Figure 5(a). These two metrics are primarily designed for point-based anomalies, treating each point independently and assigning equal weight to the detection of each point in calculating the overall AUC. However, these metrics may not be ideal for assessing subsequence anomalies. To address these limitations, an extension of the ROC and PR curves called Range-AUC [105] has been introduced specifically for subsequences. By adding a buffer region at the outliers' boundaries as shown in Figure 5(b), it accounts for the false tolerance of labeling in the ground truth and assigns higher anomaly scores near the outlier boundaries. However, the buffer length in Range-AUC, denoted as l, needs to be predefined. If not properly set, it can strongly influence range-AUC measures. To eliminate this influence, VUS [12, 105] computes Range-AUC for different buffer lengths from 0 to the l, which leads to the creation of a three-dimensional surface in the ROC-PR space as shown in Figure 5(b). Different evaluation measures have different properties. The selection of evaluation metrics should be approached with caution, considering the specific requirements of the task (refer to [81, 105, 133] for additional details).

7 Benchmarks & Discussion

In previous sections, we observed that a substantial number of time-series anomaly detection methods have been developed over the past several decades. However, evaluating the performance of these methods across diverse application backgrounds, domains, and anomaly types has become a significant challenge. Multiple surveys and experimental analyses have been proposed in recent years [5, 10, 124]. These evaluations have been based on several collections of labeled time series and benchmarks. The benchmarks are presented in chronological order in Table 4. However, the quality of time-series anomaly detection datasets poses critical challenges, with common issues such as mislabeling, bias, and limited feasibility hindering progress in evaluation and benchmarking practices [81, 150]. Addressing these limitations, TSB-AD [81] offers a heterogeneous and curated collection of time-series anomaly detection datasets, along with a rigorous and comprehensive benchmark to systematically assess various anomaly detection methods.

Moreover, we summarize the research insights drawn upon the experimental results of previous benchmark studies. First, there

Table 4: Summary of existing benchmarks.

Benchmark	Dataset				Algorithm			Evaluation	
	# Datasets	# Curated TS	Uni	Multi	Stat	NN	FM	HP	# Measures
Wu & Keogh [150]	1	250	✓	×	-	-	-	-	-
TODS [69]	5	0	✓	✓	7	2	0	×	3
TimeEval [124]	15	0	✓	✓	49	22	0	×	3
TSB-UAD [107]	18	0	✓	×	9	3	0	×	9
TimeSeAD [144]	2	21	×	✓	0	28	0	✓	3
Zhang *et al.* [164]	15	0	✓	✓	11	6	0	✓	4
TSB-AD [81]	40	1070	✓	✓	25	10	5	✓	10

is no one-size-fits-all anomaly detector across all settings. Certain methods excel in specific contexts yet underperform in others [81, 144, 147, 164]. Second, statistical approaches generally demonstrate robust performance, while neural network-based methods often fall short of their presumed advantages [81, 124]. While in TSB-AD [81], researchers also observe that neural networks and foundation models still strive to excel in detecting point anomalies and in handling multivariate cases. Third, simpler architectures such as CNNs and LSTMs generally outperform more complex designs, such as advanced transformer architectures [81, 123]. Finally, foundation models excel at detecting point-based anomalies but face difficulties with extended sequence anomalies. Their predictive mechanism often relies on limited look-back windows, constraining the available temporal context. Consequently, performance diminishes, and noise increases when dealing with long sequence anomalies. In addition, flawed point-adjustment techniques can artificially inflate their results, creating an illusion of progress [81].

Even though a large number of unsupervised methods have been proposed for univariate time-series anomaly detection, not much attention has been paid to multivariate time series, streaming time series, series with missing values, series with non-continuous timestamps, heterogeneous time series, or a combination of the above. Such times series are often encountered in practice, thus we need robust and accurate methods for these cases, as well.

8 Conclusions

We presented an extensive and process-centric taxonomy for time-series anomaly detection. The existing literature is classified into three primary categories and nine subcategories. Additionally, we conduct a meta-analysis on the evolution of time series anomaly detection algorithms, providing a holistic overview of the field's progression. Furthermore, we highlight recent advancements in automated anomaly detection, emphasizing the need for approaches that leverage model selection, ensembling, and generation to enhance detection performance. By looking into the evaluation measures and the performance of diverse anomaly detectors, we advocate the necessity for methodologies that can effectively handle the complex characteristics of real-world time series.

Acknowledgments

Supported by EU Horizon projects AI4Europe (101070000), TwinODIS (101160009), ARMADA (101168951), DataGEMS (101188416), RECITALS (101168490), by $Y\Pi AI\Theta A$ & NextGenerationEU project HARSH ($Y\Pi 3TA-0560901$), Cisco Systems, and Meta.

References

[1] Charu C Aggarwal and Saket Sathe. 2015. Theoretical foundations and algorithms for outlier ensembles. *Acm sigkdd explorations newsletter* 17, 1 (2015), 24–47.
[2] Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. 2017. Unsupervised Real-Time Anomaly Detection for Streaming Data. 262 (2017), 134–147.

[3] Jérôme Antoni and Pietro Borghesani. 2019. A Statistical Methodology for the Design of Condition Indicators. 114 (2019), 290–327.
[4] Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A Zuluaga. 2020. Usad: Unsupervised anomaly detection on multivariate time series. In *SIGKDD*. 3395–3404.
[5] Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A. Zuluaga. 2022. Do Deep Neural Networks Contribute to Multivariate Time Series Anomaly Detection? *Pattern Recogn.* 132, C (2022), 9 pages.
[6] Maroua Bahri, Flavia Salutari, Andrian Putina, and Mauro Sozio. 2022. AutoML: state of the art with a focus on anomaly detection, challenges, and research directions. *International Journal of Data Science and Analytics* 14, 2 (2022), 113–126.
[7] Md Abul Bashar and Richi Nayak. 2020. TAnoGAN: Time series anomaly detection with generative adversarial networks. In *SSCI*. 1778–1785.
[8] Sabyasachi Basu and Martin Meckesheimer. 2007. Automatic Outlier Detection for Time Series: An Application to Sensor Data. 11, 2 (2007), 137–154.
[9] Arpita Bhargava and AS Raghuvanshi. 2013. Anomaly detection in wireless sensor networks using S-Transform in combination with SVM. In *2013 5th International Conference and Computational Intelligence and Communication Networks*. 111–116.
[10] Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A Lozano. 2021. A review on outlier/anomaly detection in time series data. *ACM computing surveys (CSUR)* 54, 3 (2021), 1–33.
[11] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv* (2021).
[12] Paul Boniol, Ashwin K Krishna, Marine Bruel, Qinghua Liu, Mingyi Huang, Themis Palpanas, Ruey S Tsay, Aaron Elmore, Michael J Franklin, and John Paparrizos. 2025. VUS: effective and efficient accuracy measures for time-series anomaly detection. *The VLDB Journal* 34, 3 (2025), 32.
[13] Paul Boniol, Michele Linardi, Federico Roncallo, and Themis Palpanas. 2020. Automated anomaly detection in large sequences. In *ICDE*. 1834–1837.
[14] Paul Boniol, Michele Linardi, Federico Roncallo, Themis Palpanas, Mohammed Meftah, and Emmanuel Remy. 2021. Unsupervised and scalable subsequence anomaly detection in large data series. *The VLDB Journal* (2021).
[15] Paul Boniol, Qinghua Liu, Mingyi Huang, Themis Palpanas, and John Paparrizos. 2024. Dive into time-series anomaly detection: A decade review. *arXiv* (2024).
[16] Paul Boniol and Themis Palpanas. 2020. Series2Graph: Graph-Based Subsequence Anomaly Detection for Time Series. 13, 11 (2020), 14.
[17] Paul Boniol and Themis Palpanas. 2020. Series2Graph: graph-based subsequence anomaly detection for time series. *PVLDB* 13, 12 (2020), 1821–1834.
[18] Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J Franklin. 2021. SAND: streaming subsequence anomaly detection. *PVLDB* 14, 10 (2021), 1717–1729.
[19] J-C de Borda. 1781. Mémoire sur les élections au scrutin: Histoire de l'Académie Royale des Sciences. *Paris, France* 12 (1781).
[20] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. 2000. LOF: identifying density-based local outliers. In *SIGMOD*. 93–104.
[21] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. LOF: Identifying Density-based Local Outliers. In *SIGMOD*.
[22] Suratna Budalakoti, Ashok N Srivastava, and Matthew Eric Otey. 2008. Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 39, 1 (2008), 101–113.
[23] Soumen Chakrabarti, Sunita Sarawagi, and Byron Dom. 1998. Mining Surprising Patterns Using Temporal Description Length. In *VLDB*, Vol. 24. 606–617.
[24] Raghavendra Chalapathy and Sanjay Chawla. 2019. Deep learning for anomaly detection: A survey. *arXiv* (2019).
[25] Raghavendra Chalapathy, Aditya Krishna Menon, and Sanjay Chawla. 2017. Robust, Deep and Inductive Anomaly Detection. In *Machine Learning and Knowledge Discovery in Databases*. Springer International Publishing, Cham, 36–51.
[26] Ih Chang, George C Tiao, and Chung Chen. 1988. Estimation of time series parameters in the presence of outliers. *Technometrics* 30, 2 (1988), 193–204.
[27] Marcus Chang, Andreas Terzis, and Philippe Bonnet. 2009. Mote-Based Online Anomaly Detection Using Echo State Networks. In *DCOOS (Lecture Notes in Computer Science, Vol. 5516)*. 72–86.
[28] S. Chauhan and L. Vig. 2015. Anomaly Detection in ECG Time Signals via Deep Long Short-Term Memory Networks. In *DSAA*. 1–7.
[29] Qing Chen, Anguo Zhang, Tingwen Huang, Qianping He, and Yongduan Song. 2020. Imbalanced Dataset-Based Echo State Networks for Anomaly Detection. 32, 8 (2020), 3685–3694.
[30] Zhangyu Cheng, Chengming Zou, and Jianwei Dong. 2019. Outlier detection using isolation forest and local outlier factor. In *Proceedings of the conference on research in adaptive and convergent systems*. 161–168.
[31] Winnie Chow, Lauren E Gardiner, Haraldur T Hallgrimsson, Maxwell A Xu, and Shirley You Ren. 2024. Towards Time-Series Reasoning with LLMs. In *NeurIPS Workshop on Time Series in the Age of Large Models*.
[32] Robert B Cleveland, William S Cleveland, Jean E McRae, and Irma Terpenning. 1990. STL: A seasonal-trend decomposition. *J. Off. Stat* 6, 1 (1990), 3–73.
[33] Zhihao Dai, Ligang He, Shuanghua Yang, and Matthew Leeke. 2024. SARAD: Spatial Association-Aware Anomaly Detection and Diagnosis for Multivariate Time Series. In *NeurIPS*.
[34] Jesse Davis and Mark Goadrich. 2006. The relationship between Precision-Recall and ROC curves. In *ICML*. 233–240.
[35] Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* (2018).
[36] Nan Ding, Huanbo Gao, Hongyu Bu, Haoxuan Ma, and Huaiwei Si. 2018. Multivariate-Time-Series-Driven Real-Time Anomaly Detection Based on Bayesian Network. 18, 10 (2018), 3367.
[37] Xibin Dong, Zhiwen Yu, Wenming Cao, Yifan Shi, and Qianli Ma. 2020. A survey on ensemble learning. *Frontiers of Computer Science* 14 (2020), 241–258.
[38] Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* (2020).
[39] Tom Fawcett. 2006. An introduction to ROC analysis. *Pattern Recognition Letters* 27, 8 (2006), 861–874. ROC Analysis in Pattern Recognition.
[40] Pavel Filonov, Andrey Lavrentyev, and Artem Vorontsov. 2016. Multivariate industrial time series with cyber-attack simulation: Fault detection using an lstm-based predictive data model. *arXiv* (2016).
[41] Anthony J Fox. 1972. Outliers in time series. *Journal of the Royal Statistical Society: Series B (Methodological)* 34, 3 (1972), 350–363.
[42] Shanghua Gao, Teddy Koker, Owen Queen, Thomas Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. 2024. UniTS: A unified multi-task time series model. In *NeurIPS*.
[43] Yifeng Gao, Jessica Lin, and Constantin Brif. 2020. Ensemble Grammar Induction For Detecting Anomalies in Time Series. In *EDBT*.
[44] Gabriel Garcia, Gabriel Michau, Mélanie Ducoffe, Jayant Sen Gupta, and Olga Fink. 2020. Time Series to Images: Monitoring the Condition of Industrial Assets with Deep Learning Image Processing Algorithms. (2020).
[45] Nicolas Goix. 2016. How to evaluate the quality of unsupervised anomaly detection algorithms? *arXiv* (2016).
[46] Markus Goldstein and Andreas Dengel. 2013. Histogram-based Outlier Score (HBOS): A fast Unsupervised Anomaly Detection Algorithm.
[47] Markus Goldstein and Seiichi Uchida. 2016. A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PloS one* 11, 4 (2016), e0152173.
[48] Vanessa Gómez-Verdejo, Jerónimo Arenas-García, Miguel Lazaro-Gredilla, and Ángel Navia-Vazquez. 2011. Adaptive one-class support vector machine. *IEEE Transactions on Signal Processing* 59, 6 (2011), 2975–2981.
[49] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *NeurIPS* 27 (2014).
[50] Mononito Goswami, Cristian Challu, Laurent Callot, Lenon Minorics, and Andrey Kan. 2023. Unsupervised Model Selection for Time-series Anomaly Detection. *ICLR*. (2023).
[51] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. 2024. MOMENT: A Family of Open Time-series Foundation Models. In *ICML*.
[52] Nico Görnitz, Mikio Braun, and Marius Kloft. 2015. Hidden Markov Anomaly Detection. In *Proceedings of the International Conference on Machine Learning (ICML) (ICML'15)*. 1833–1842.
[53] Michael Hahsler and Matthew Bolaos. 2016. Clustering Data Streams Based on Shared Density between Micro-Clusters. *TKDE* 28, 6 (2016), 1449–1461.
[54] Johanna Hardin and David M Rocke. 2004. Outlier detection in the multiple cluster setting using the minimum covariance determinant estimator. *Computational Statistics & Data Analysis* 44, 4 (2004), 625 – 638.
[55] Sahand Hariri, Matias Carrasco Kind, and Robert J Brunner. 2019. Extended isolation forest. *TKDE* 33, 4 (2019), 1479–1489.
[56] D. M Hawkins. 1980. *Identification of Outliers*. OCLC: 945065134.
[57] Zengyou He, Xiaofei Xu, and Shengchun Deng. 2003. Discovering cluster-based local outliers. *Pattern recognition letters* 24, 9-10 (2003), 1641–1650.
[58] Niklas Heim and James E. Avery. 2019. *Adaptive Anomaly Detection in Chaotic Time Series with a Spatially Aware Echo State Network*. arXiv:1909.01709
[59] Jordan Hochenbaum, Owen S. Vallis, and Arun Kejariwal. 2017. *Automatic Anomaly Detection in the Cloud Via Statistical Learning*. arXiv:1704.07706
[60] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (1997), 1735–1780.
[61] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. 2018. Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding. In *SIGKDD*. 387–395.
[62] Serdar Kadioglu, Yuri Malitsky, Meinolf Sellmann, and Kevin Tierney. 2010. ISAC–instance-specific algorithm configuration. In *ECAI 2010*. 751–756.
[63] Eamonn Keogh, Jessica Lin, and Ada Fu. 2005. Hot sax: Efficiently finding the most unusual time series subsequence. In *ICDM*. 8–pp.

[64] Eamonn Keogh, Stefano Lonardi, and Bill'Yuan-chi' Chiu. 2002. Finding surprising patterns in a time series database in linear time and space. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. 550–556.

[65] Chunggyeom Kim, Jinhyuk Lee, Raehyun Kim, Youngbin Park, and Jaewoo Kang. 2018. DeepNAP: Deep Neural Anomaly Pre-Detection in a Semiconductor Fab. 457-458 (2018), 1–11.

[66] Jon M Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)* 46, 5 (1999), 604–632.

[67] Maria Kontaki, Anastasios Gounaris, Apostolos N Papadopoulos, Kostas Tsichlas, and Yannis Manolopoulos. 2011. Continuous monitoring of distance-based outliers over data streams. In *ICDE*. 135–146.

[68] Anna Korba, Stephan Clémençon, and Eric Sibony. 2017. A learning theory of ranking aggregation. In *Artificial Intelligence and Statistics*. PMLR, 1001–1010.

[69] Kwei-Herng Lai, Daochen Zha, Junjie Xu, Yue Zhao, Guanchu Wang, and Xia Hu. 2021. Revisiting Time Series Outlier Detection: Definitions and Benchmarks. In *NeurIPS*.

[70] Bouchra Lamrini, Augustin Gjini, Simon Daudin, Pascal Pratmarty, François Armando, and Louise Travé-Massuyès. 2018. Anomaly Detection using Similarity-based One-Class SVM for Network Traffic Characterization.. In *DX*.

[71] Ming-Chang Lee, Jia-Chun Lin, and Ernst Gunnar Gran. 2020. RePAD: Real-Time Proactive Anomaly Detection for Time Series. In *AINA*. 1291–1302.

[72] Dan Li, Dacheng Chen, Baihong Jin, Lei Shi, Jonathan Goh, and See-Kiong Ng. 2019. MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. In *International conference on artificial neural networks*. Springer, 703–716.

[73] Zeyan Li, Wenxiao Chen, and Dan Pei. 2018. Robust and Unsupervised KPI Anomaly Detection Based on Conditional Variational Autoencoder. In *IPCCC*. 1–9.

[74] Zhihua Li, Ziyuan Li, Ning Yu, Steven Wen, et al. 2017. Locality-based visual outlier detection algorithm for time series. *Security and Communication Networks* 2017 (2017).

[75] Zheng Li, Yue Zhao, Nicola Botta, Cezar Ionescu, and Xiyang Hu. 2020. COPOD: Copula-Based Outlier Detection. In *ICDM*.

[76] Zinan Lin, Kiran Thekumparampil, Giulia Fanti, and Sewoong Oh. 2020. Infogan-cr and modelcentrality: Self-supervised model training and selection for disentangling gans. In *international conference on machine learning*. PMLR, 6127–6139.

[77] Michele Linardi, Yan Zhu, Themis Palpanas, and Eamonn Keogh. 2020. Matrix profile goes MAD: variable-length motif and discord discovery in data series. *Data Mining and Knowledge Discovery* 34 (2020), 1022–1071.

[78] Chunwei Liu, John Paparrizos, and Aaron J Elmore. 2024. Adaedge: A dynamic compression selection framework for resource constrained devices. In *ICDE*. 1506–1519.

[79] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2008. Isolation forest. In *ICDM*. 413–422.

[80] Qinghua Liu, Paul Boniol, Themis Palpanas, and John Paparrizos. 2024. Time-Series Anomaly Detection: Overview and New Trends. *PVLDB* 17, 12 (2024), 4229–4232.

[81] Qinghua Liu and John Paparrizos. 2024. The Elephant in the Room: Towards A Reliable Time-Series Anomaly Detection Benchmark. In *NeurIPS 2024*.

[82] Yue Lu, Renjie Wu, Abdullah Mueen, Maria A Zuluaga, and Eamonn Keogh. 2022. Matrix profile XXIV: scaling time series anomaly detection to trillions of datapoints and ultra-fast arriving data streams. In *SIGKDD*. 1173–1182.

[83] Junshui Ma and Simon Perkins. 2003. Online Novelty Detection on Temporal Sequences. In *SIGKDD*. 613.

[84] Junshui Ma and Simon Perkins. 2003. Time-series novelty detection using one-class support vector machines. In *IJCNN*, Vol. 3. 1741–1745.

[85] Martin Q Ma, Yue Zhao, Xiaorong Zhang, and Leman Akoglu. 2023. The need for unsupervised outlier model selection: A review and evaluation of internal evaluation strategies. *ACM SIGKDD Explorations Newsletter* 25, 1 (2023).

[86] Mohammad Saeid Mahdavinejad, Mohammadreza Rezvan, Mohammadamin Barekatain, Peyman Adibi, Payam Barnaghi, and Amit P Sheth. 2017. Machine learning for Internet of Things data analysis: A survey. *Digital Communications and Networks* (2017).

[87] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. 2016. *LSTM-Based Encoder-Decoder for Multi-Sensor Anomaly Detection.* arXiv:1607.00148

[88] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, and Puneet Agarwal. 2015. Long Short Term Memory Networks for Anomaly Detection in Time Series. In *ESANN*, Vol. 23.

[89] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, Puneet Agarwal, et al. 2015. Long Short Term Memory Networks for Anomaly Detection in Time Series.. In *Esann*, Vol. 2015. 89.

[90] Carla Marceau. 2000. Characterizing the Behavior of a Program Using Multiple-Length N-Grams. In *NSPW*. 101–110.

[91] Pierre-François Marteau, Saeid Soheily-Khah, and Nicolas Béchet. 2017. Hybrid isolation forest-application to intrusion detection. *arXiv* (2017).

[92] Mohsin Munir, Shoaib Ahmed Siddiqui, Andreas Dengel, and Sheraz Ahmed. 2019. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series. 7 (2019), 1991–2005.

[93] Gyoung S Na, Donghyun Kim, and Hwanjo Yu. 2018. Dilof: Effective and memory efficient local outlier detection in data streams. In *SIGKDD*. 1993–2002.

[94] Takaaki Nakamura, Ryan Mercer, Makoto Imamura, and Eamonn Keogh. 2023. MERLIN++: parameter-free discovery of time series anomalies. *Data Mining and Knowledge Discovery* 37, 2 (2023), 670–709.

[95] Jose Manuel Navarro, Alexis Huet, and Dario Rossi. 2023. Meta-Learning for Fast Model Recommendation in Unsupervised Multivariate Time Series Anomaly Detection. In *AutoML Conference 2023*.

[96] Thanh Trung Nguyen, Uy Quang Nguyen, et al. 2016. An evaluation method for unsupervised anomaly detection algorithms. *Journal of Computer Science and Cybernetics* 32, 3 (2016), 259–272.

[97] Mladen Nikolić, Filip Marić, and Predrag Janičić. 2013. Simple algorithm portfolio for SAT. *Artificial Intelligence Review* 40, 4 (2013), 457–465.

[98] Zijian Niu, Ke Yu, and Xiaofei Wu. 2020. LSTM-based VAE-GAN for time-series anomaly detection. *Sensors* 20, 13 (2020), 3738.

[99] Oliver Obst, X. Rosalind Wang, and Mikhail Prokopenko. 2008. Using Echo State Networks for Anomaly Detection in Underground Coal Mines. In *IPSN*. 219–229.

[100] Alberto Ogbechie, Javier Díaz-Rozo, Pedro Larrañaga, and Concha Bielza. 2017. Dynamic Bayesian Network-Based Anomaly Detection for In-Process Visual Inspection of Laser Surface Heat Treatment. In *ML4CPS*. 17–24.

[101] Randy Paffenroth, Kathleen Kay, and Les Servi. 2018. *Robust PCA for Anomaly Detection in Cyber Networks.* arXiv:1801.01571

[102] ES Page. 1957. On problems in which a change in a parameter occurs at an unknown point. *Biometrika* 44, 1/2 (1957), 248–252.

[103] Girish Keshav Palshikar. 2005. Distance-based outliers in sequences. In *Distributed Computing and Internet Technology: ICDCIT 2005*. 547–552.

[104] Spiros Papadimitriou, Hiroyuki Kitagawa, Phillip B Gibbons, and Christos Faloutsos. 2003. Loci: Fast outlier detection using the local correlation integral. In *ICDE*. 315–326.

[105] John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron Elmore, and Michael J Franklin. 2022. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *PVLDB* 15, 11 (2022), 2774–2787.

[106] John Paparrizos and Luis Gravano. 2016. k-Shape: Efficient and Accurate Clustering of Time Series. *SIGMOD* 45, 1 (2016), 69–76.

[107] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S Tsay, Themis Palpanas, and Michael J Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. *PVLDB* 15, 8 (2022), 1697–1711.

[108] John Paparrizos, Haojun Li, Fan Yang, Kaize Wu, Jens E d'Hondt, and Odysseas Papapetrou. 2024. A survey on time-series distance measures. *arXiv* (2024).

[109] John Paparrizos, Chunwei Liu, Aaron J Elmore, and Michael J Franklin. 2020. Debunking four long-standing misconceptions of time-series distance measures. In *SIGMOD*. 1887–1905.

[110] John Paparrizos, Fan Yang, and Haojun Li. 2024. Bridging the gap: A decade review of time-series clustering methods. *arXiv* (2024).

[111] Daehyung Park, Zackory Erickson, Tapomayukh Bhattacharjee, and Charles C. Kemp. 2016. Multimodal Execution Monitoring for Anomaly Detection during Robot Manipulation. In *ICRA*. 407–414.

[112] Daehyung Park, Yuuna Hoshi, and Charles C. Kemp. 2018. A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder. 3, 3 (2018), 1544–1551.

[113] Stephen Pauwels and Toon Calders. 2019. An Anomaly Detection Technique for Business Processes Based on Extended Dynamic Bayesian Networks. In *SAC*. 494–501.

[114] Stephen Pauwels and Toon Calders. 2019. Detecting Anomalies in Hybrid Business Process Logs. 19, 2 (2019), 18–30.

[115] Dragoljub Pokrajac, Aleksandar Lazarevic, and Longin Jan Latecki. 2007. Incremental local outlier detection for data streams. In *CIDM*. 504–515.

[116] Shebuti Rayana and Leman Akoglu. 2016. Less is more: Building selective anomaly ensembles. *Acm transactions on knowledge discovery from data (tkdd)* 10, 4 (2016), 1–33.

[117] Volker Roth. 2006. Kernel Fisher Discriminants for Outlier Detection. 18, 4 (2006), 942–960.

[118] Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics* 20 (1987), 53–65.

[119] Peter J. Rousseeuw and Katrien Van Driessen. 1999. A Fast Algorithm for the Minimum Covariance Determinant Estimator. *Technometrics* 41, 3 (1999), 212–223.

[120] Peter J. Rousseeuw and Annick M. Leroy. 1987. *Robust regression and outlier detection.*

[121] Stan Salvador and Philip Chan. 2005. Learning States and Rules for Detecting Anomalies in Time Series. 23, 3 (2005), 241–255.

[122] Jörg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. 1998. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Mining and Knowledge Discovery* 2, 2 (01 Jun 1998), 169–194.

[123] M Saquib Sarfraz, Mei-Yen Chen, Lukas Layer, Kunyu Peng, and Marios Koulakis. 2024. Position Paper: Quo Vadis, Unsupervised Time Series Anomaly Detection?. In *ICML*.

[124] Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. 2022. Anomaly Detection in Time Series: A Comprehensive Evaluation. *PVLDB* 15, 9 (2022), 1779–1797.

[125] Johannes Schneider, Phillip Wenig, and Thorsten Papenbrock. 2021. Distributed Detection of Sequential Anomalies in Univariate Time Series. *The VLDB Journal* 30, 4 (2021), 579–602.

[126] Bernhard Schölkopf, Robert C Williamson, Alex Smola, John Shawe-Taylor, and John Platt. 1999. Support vector method for novelty detection. *NeurIPS* 12 (1999).

[127] Pavel Senin, Jessica Lin, Xing Wang, Tim Oates, Sunil Gandhi, Arnold P. Boedihardjo, Crystal Chen, and Susan Frankenstein. 2015. Time Series Anomaly Discovery with Grammar-Based Compression. 481–492 pages.

[128] S-C. Chen K. Sarinnapakorn Shyu, M-L. and LW. Chang. 2003. A Novel Anomaly Detection Scheme Based on Principal Component Classifier. (2003).

[129] Ralph D. Snyder and Stephen J. Withers. 1983. *Exponential smoothing with finite sample correction.* Number 1983,1.

[130] Maximilian Soelch, Justin Bayer, Marvin Ludersdorfer, and Patrick van der Smagt. 2016. *Variational Inference for On-Line Anomaly Detection in High-Dimensional Time Series.* arXiv:1602.07109

[131] Hongchao Song, Zhuqing Jiang, Aidong Men, and Bo Yang. 2017. A Hybrid Semi-Supervised Anomaly Detection Model for High-Dimensional Data. *Computational Intelligence and Neuroscience* 2017 (15 Nov 2017), 8501683.

[132] Junho Song, Keonwoo Kim, Jeonglyul Oh, and Sungzoon Cho. 2023. Memto: Memory-guided transformer for multivariate time series anomaly detection. *NeurIPS* 36 (2023), 57947–57963.

[133] Sondre Sørbø and Massimiliano Ruocco. 2023. Navigating the Metric Maze: A Taxonomy of Evaluation Metrics for Anomaly Detection in Time Series. *arXiv* (2023).

[134] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. 2019. Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network. In *SIGKDD*. 2828–2837.

[135] S. Subramaniam, T. Palpanas, D. Papadopoulos, V. Kalogeraki, and D. Gunopulos. 2006. Online Outlier Detection in Sensor Data Using Non-Parametric Models. In *VLDB*. 187–198.

[136] Pei Sun, Sanjay Chawla, and Bavani Arunasalam. 2006. Mining for Outliers in Sequential Databases. In *ICDM*. 94–105.

[137] Emmanouil Sylligardos, Paul Boniol, John Paparrizos, Panos E. Trahanias, and Themis Palpanas. 2023. Choose Wisely: An Extensive Evaluation of Model Selection for Anomaly Detection in Time Series. *PVLDB* 16, 11 (2023), 3418–3432.

[138] Jian Tang, Zhixiang Chen, Ada Wai-Chee Fu, and David W Cheung. 2002. Enhancing effectiveness of outlier detections for low density patterns. In *PAKDD*. 535–548.

[139] Nesime Tatbul, Tae Jun Lee, Stan Zdonik, Mejbah Alam, and Justin Gottschlich. 2018. Precision and recall for time series. In *NeurIPS*. 1924–1934.

[140] David MJ Tax and Robert PW Duin. 2004. Support vector data description. *Machine learning* 54, 1 (2004), 45–66.

[141] Shreshth Tuli, Giuliano Casale, and Nicholas R Jennings. 2022. TranAD: deep transformer networks for anomaly detection in multivariate time series data. *VLDB* 15, 6 (2022), 1201–1214.

[142] A Vaswani. 2017. Attention is all you need. *NeurIPS* (2017).

[143] Rafael G. Vieira, Marcos A. Leone Filho, and Robinson Semolini. 2018. An Enhanced Seasonal-Hybrid ESD Technique for Robust Anomaly Detection on Time Series. In *Simpósio Brasileiro de Redes de Computadores (SBRC)*, Vol. 36.

[144] Dennis Wagner, Tobias Michels, Florian CF Schulz, Arjun Nair, Maja Rudolph, and Marius Kloft. 2023. Timesead: Benchmarking deep multivariate time-series anomaly detection. *Transactions on Machine Learning Research* (2023).

[145] Yi Wang, Linsheng Han, Wei Liu, Shujia Yang, and Yanbo Gao. 2019. Study on Wavelet Neural Network Based Anomaly Detection in Ocean Observing Data Series. 186 (2019), 106129.

[146] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. 2023. Transformers in time series: a survey. In *IJCAI*. 6778–6786.

[147] Phillip Wenig, Sebastian Schmidl, and Thorsten Papenbrock. 2022. TimeEval: a benchmarking toolkit for time series anomaly detection algorithms. *PVLDB* 15, 12 (2022), 3678–3681.

[148] Jia Wu, Weiru Zeng, and Fei Yan. 2018. Hierarchical Temporal Memory Method for Time-Series-Based Anomaly Detection. 273 (2018), 535–546.

[149] P. Wu, J. Liu, and F. Shen. 2020. A Deep One-Class Neural Network for Anomalous Event Detection in Complex Scenes. *TNNLS* 31, 7 (2020), 2609–2622.

[150] Renjie Wu and Eamonn Keogh. 2021. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. *TKDE* (2021).

[151] Wentai Wu, Ligang He, Weiwei Lin, Yi Su, Yuhua Cui, Carsten Maple, and Stephen Jarvis. 2020. *Developing an Unsupervised Real-Time Anomaly Detection Scheme for Time Series with Multi-Seasonality.* arXiv:1908.01146

[152] Yanshan Xiao, Bo Liu, Longbing Cao, Xindong Wu, Chengqi Zhang, Zhifeng Hao, Fengzhao Yang, and Jie Cao. 2009. Multi-sphere support vector data description for outliers detection on multi-distribution data. In *ICDM workshops*. 82–87.

[153] Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, et al. 2018. Unsupervised Anomaly Detection via Variational Auto-Encoder for Seasonal KPIs in Web Applications. In *WWW*. 187–196.

[154] Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. 2022. Anomaly Transformer: Time Series Anomaly Detection with Association Discrepancy. In *ICLR*. https://openreview.net/forum?id=LzQQ89U1qm_

[155] Lin Xu, Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. 2008. SATzilla: portfolio-based algorithm selection for SAT. *Journal of artificial intelligence research* 32 (2008), 565–606.

[156] Kenji Yamanishi, Jun-ichi Takeuchi, Graham Williams, and Peter Milne. 2004. On-Line Unsupervised Outlier Detection Using Finite Mixtures with Discounting Learning Algorithms. 8, 3 (2004), 275–300.

[157] Chao-Lung Yang and Wei-Ju Liao. 2017. Adjacent Mean Difference (AMD) method for dynamic segmentation in time series anomaly detection. In *2017 IEEE/SICE International Symposium on System Integration (SII)*. 241–246.

[158] Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. 2023. Dcdetector: Dual attention contrastive representation learning for time series anomaly detection. In *SIGKDD*. 3033–3045.

[159] Dragomir Yankov, Eamonn Keogh, and Umaa Rebbapragada. 2008. Disk aware discord discovery: Finding unusual time series in terabyte sized datasets. *Knowledge and Information Systems* 17 (2008), 241–262.

[160] Yuan Yao, Abhishek Sharma, Leana Golubchik, and Ramesh Govindan. 2010. Online Anomaly Detection for Sensor Systems: A Simple and Efficient Approach. *Perform. Eval.* 67, 11 (2010), 1059–1075.

[161] C.-C.M. Yeh, Y. Zhu, L. Ulanova, N. Begum, Y. Ding, H.A. Dau, D.F. Silva, A. Mueen, and E.J. Keogh. 2016. Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets. In *ICDM*.

[162] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. 2016. Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In *ICDM*. 1317–1322.

[163] Yufeng Yu, Yuelong Zhu, Shijin Li, and Dingsheng Wan. 2014. Time Series Outlier Detection Based on Sliding Window Prediction. 2014 (2014), 1–14.

[164] Aoqian Zhang, Shuqing Deng, Dongping Cui, Ye Yuan, and Guoren Wang. 2023. An Experimental Evaluation of Anomaly Detection in Time Series. *PVLDB* 17, 3 (2023), 483–496.

[165] Chunkai Zhang, Shaocong Li, Hongye Zhang, and Yingyang Chen. 2020. VELC: A New Variational AutoEncoder Based Model for Time Series Anomaly Detection. arXiv:1907.01702

[166] Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V. Chawla. 2019. A Deep Neural Network for Unsupervised Anomaly Detection and Diagnosis in Multivariate Time Series Data. In *AAAI*, Vol. 33. 1409–1416.

[167] Rui Zhang, Shaoyan Zhang, Sethuraman Muthuraman, and Jianmin Jiang. 2007. One Class Support Vector Machine for Anomaly Detection in the Communication Network Performance Data. In *ELECTROSCIENCE*. 31–37.

[168] Hang Zhao, Yujing Wang, Juanyong Duan, Congrui Huang, Defu Cao, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. 2020. Multivariate time-series anomaly detection via graph attention network. In *ICDM*. 841–850.

[169] Yue Zhao, Ryan Rossi, and Leman Akoglu. 2021. Automatic unsupervised outlier model selection. *NeurIPS* 34 (2021), 4489–4502.

[170] Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. 2023. One fits all: Power general time series analysis by pretrained lm. *NeurIPS* 36 (2023), 43322–43355.

[171] Zihao Zhou and Rose Yu. 2024. Can LLMs Understand Time Series Anomalies? *arXiv* (2024).

[172] Yan Zhu, Chin-Chia Michael Yeh, Zachary Zimmerman, Kaveh Kamgar, and Eamonn Keogh. 2018. Matrix profile XI: SCRIMP++: time series motif discovery at interactive speeds. In *ICDM*. 837–846.

[173] Yan Zhu, Zachary Zimmerman, Nader Shakibay Senobari, Chin-Chia Michael Yeh, Gareth Funning, Abdullah Mueen, Philip Brisk, and Eamonn Keogh. 2016. Matrix profile ii: Exploiting a novel algorithm and gpus to break the one hundred million barrier for time series motifs and joins. In *ICDM*. 739–748.

[174] Zachary Zimmerman, Kaveh Kamgar, Nader Shakibay Senobari, Brian Crites, Gareth Funning, Philip Brisk, and Eamonn Keogh. 2019. Matrix profile XIV: scaling time series motif discovery with GPUs to break a quintillion pairwise comparisons a day and beyond. In *Proceedings of the ACM Symposium on Cloud Computing*. 74–86.

[175] Zachary Zimmerman, Nader Shakibay Senobari, Gareth Funning, Evangelos Papalexakis, Samet Oymak, Philip Brisk, and Eamonn Keogh. 2019. Matrix profile XVIII: time series mining in the face of fast moving streams using a learned approximate matrix profile. In *ICDM*. 936–945.