A Survey on Explainable Anomaly Detection

ZHONG LI, YUXUAN ZHU, and MATTHIJS VAN LEEUWEN, Leiden Institute of Advanced Computer Science (LIACS), Leiden University, The Netherlands

In the past two decades, most research on anomaly detection has focused on improving the accuracy of the detection, while largely ignoring the explainability of the corresponding methods and thus leaving the explanation of outcomes to practitioners. As anomaly detection algorithms are increasingly used in safety-critical domains, providing explanations for the high-stakes decisions made in those domains has become an ethical and regulatory requirement. Therefore, this work provides a comprehensive and structured survey on state-of-the-art explainable anomaly detection techniques. We propose a taxonomy based on the main aspects that characterise each explainable anomaly detection technique, aiming to help practitioners and researchers find the explainable anomaly detection method that best suits their needs.

CCS Concepts: • **Information systems → Decision support systems**; **Data analytics**; **Data mining**;

Additional Key Words and Phrases: Explainable anomaly detection, interpretable anomaly detection, anomaly explanation, anomaly detection, outlier detection, explainable machine learning, explainable artificial intelligence

1 INTRODUCTION

An anomaly is an object that is notably different from the majority of the remaining objects. Depending on the specific application domain, an anomaly can also be called an outlier or a novelty. Moreover, it may also be known as an unusual, irregular, atypical, inconsistent, unexpected, rare, erroneous, faulty, fraudulent, malicious, unnatural, or strange object [181]. Except for a few works such as Reference [181], the term *outlier* is often used as a synonym for *anomaly* in most research. For consistency, we will use the term *anomaly* in this article.

Since the seminal work in [105], anomaly detection has been well studied and there exists a plethora of comprehensive surveys and reviews on it, including but not limited to References [1, 5, 25, 36, 37, 134, 135, 161, 165, 231]. In contrast, we only found a handful of surveys [162, 189, 225] about the *explainability* of anomaly detection methods. As suggested by Langone

23

et al. [111], model explainability represents one of the main issues concerning the adoption of data-driven algorithms in industrial environments. More importantly, for applications in safety critical domains, providing explanations to stakeholders of AI systems has become an ethical and regulatory requirement [50, 217]. However, after a thorough survey of academic publications on explainable anomaly detection, we found that existing surveys are either outdated, have missed some important work, or their proposed taxonomies are relatively coarse and, therefore, unable to characterize the increasingly rich set of explainable anomaly detection techniques available in the literature.

To address this gap in the literature, we conduct a comprehensive and structured survey on state-of-the-art explainable anomaly detection techniques and distil a refined taxonomy that caters to the increasingly rich set of techniques. Overall, this survey intends to provide both practitioners and researchers with an extensive overview of the different types of methods that have been proposed, with their pros and cons, and to help them find the explainable anomaly detection technique most suited to their needs.

Note that some researchers [27, 145, 197] distinguish between the terms 'interpretation' and 'explanation', the terms 'interpretable' and 'explainable', and the terms 'explainability' and 'interpretability'. Specifically, Broniatowski [27] defines explainability as *a model's ability to provide a description of how its outcome came to be* and describes interpretability as *a human's ability to make sense from a given stimulus so that the human can make a decision.* Moreover, Sipple & Youssef [197] argue that explainability is *the algorithmic task of generating the explanation* and interpretability is *the cognitive task of merging the expert's knowledge with the explanation to identify a unique diagnostic condition and to choose the appropriate treatment.* Considering that most researchers in data mining and machine learning treat explainability and interpretability equally, we use those terms interchangeably throughout this article. The next section will clarify what we mean exactly when we say that a technique is explainable.

1.1 Methodology

This survey aims to answer the following research questions and is structured accordingly:

- Q0 What is explainable anomaly detection and why should we care about it?
- Q1 What are the most important aspects that characterise each explainable anomaly detection technique? On this basis, how to classify existing techniques?
- Q2 How do existing techniques interpret anomalies and what are the main differences between them?
- Q3 What are the challenges and associated opportunities in explainable anomaly detection?

In order to answer these research questions, we employ a comparative and iterative surveying procedure that consists of three cycles. In the first cycle, we employ a methodology consisting of two main phases:

- — Database Selection: we select well-known scientific databases for literature collection, i.e., Google Scholar, IEEE Xplore, ACM Digital Library, DBLP, and Web of Science.
- — Literature Selection: we select related research publications that were published between January 1998 to February 2022 using the following keywords: Interpretable/Interpret/Interpreting Anomaly Detection, Explainable/Explain/Explaining Anomaly Detection, Interpretable/Interpret/Interpreting Outlier Detection, Explainable/Explain/Explaining Outlier Detection, Anomaly Interpretation, Anomaly Explanation, Outlier Interpretation, Outlier Explanation. Other useful keywords are: Anomaly/Outlier Description, Anomaly/Outlier Characterization, Outlying Property Detection, Outlying Aspects Mining, Outlying Subspaces Detection.
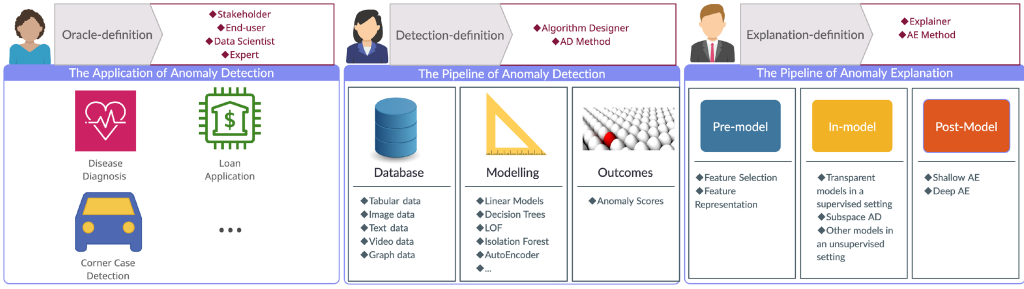
Fig. 1. The procedure of anomaly analysis and the different roles involved in this procedure.

In the second cycle, we inspect research publications that have been referenced by articles collected in the first cycle. In the third cycle, we exclude research publications that are irrelevant, not published in what we consider high-quality venues, or applications of existing methods to certain use cases.

This survey is organised as follows: To answer Q0, Section 2 states the motivations for this work and the terminology used. Section 3 describes the proposed taxonomy for answering Q1. Sections 4–7 survey existing techniques for explainable anomaly detection in a principled manner based on the proposed taxonomy, aiming to answer Q2. Section 8 discusses the open challenges and related opportunities of existing work, and then concludes this survey, answering Q3.

2 THE NEED FOR EXPLAINABLE ANOMALY DETECTION

This section introduces important terminology and concepts, such as anomalies and explainable anomaly detection, and explains why this is an important field of study.

2.1 What is an Anomaly?

First of all, we need to define what an anomaly is. Inspired by Sejr & Schneider-Kamp [189], we assume that there are three roles involved in an anomaly analysis task: (1) a/an *Stakeholder/ End-user/Data Scientist/Expert* that uses the anomaly detection system; (2) an *Algorithm Designer/ Anomaly Detection Method* that does the actual anomaly detection; and (3) an *Algorithm Explainer/ Anomaly Explanation Method* that explains identified anomalies. These three roles are illustrated in Figure 1. The different roles may have different definitions of what an anomaly is, and we distinguish those definitions as follows:

— *Oracle-Definition*: the 'ideal' definition that defines the anomalies that the end-users of the anomaly detection system aim to detect. In other words, this definition defines the *true anomalies* in the real-world application and thus strongly depends on the context and is often hard to formally/precisely formulate.
— *Detection-Definition*: the anomalies that an anomaly detection model can actually capture. This definition is given explicitly or implicitly by the anomaly detection model or technique.
— *Explanation-Definition*: describes why (and when) the anomaly explanation method considers an anomaly as anomalous.

For example, for a credit card fraud detection system, the end-users aim to detect fraudulent behaviour, which is defined as "obtaining services/goods and/or money by unethical means", including bankruptcy fraud, theft fraud, application fraud and behavioral fraud [58]. Therefore, the *Oracle-Definition* is "behaviour that aims to obtain services/goods and/or money by unethical means". However, a given credit card fraud system might only detect anomalous behaviours such as unprecedented high payments and/or payments at a never-before-seen location. Hence, the

Detection-Definition is "unprecedented high payments and/or payments at a never-before-seen location" and this is actually a theft fraud. Moreover, for an identified anomalous payment, the anomaly explanation method could generate the explanation "the payment is flagged as anomalous because it happened at midnight", which follows from the *Explanation-Definition*. Clearly the *Oracle-Definition*, the *Detection-Definition*, and the *Explanation-Definition* can be different from each other.

In general, the *Oracle-Definition* is given based on domain knowledge, which is application-specific. From this point of view, there is no universal definition of an anomaly. A commonly accepted definition by Hawkins [81] is that "an outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism". As this is informal, each specific anomaly detection model has its own definition of an anomaly, either explicitly or implicitly. For example, KNN [172] defines objects with 'far' k-nearest neighbours as anomalies, LOF [26] treats objects with a low local density as anomalies, and Isolation Forest [122] considers 'easily isolated' objects as anomalies. Importantly, this *Detection-Definition* definition can be different from the *Oracle-Definition*, which may lead to problems. For example, an anomaly detector may miss relevant anomalies while detecting 'anomalies' that are uninteresting to end-users. Moreover, depending on the technique used to explain an anomaly, the *Detection-Definition* and *Explanation-Definition* can also be different, especially when the explanation approach does not reflect the decision-making process behind the anomaly detection model.

2.2 What is Explainable Anomaly Detection?

According to Doshi-Velez & Kim [61], interpretability or explainability is defined as the ability to explain or provide meaning to humans in understandable terms. Moreover, Arrieta et al. [13] define **Explainable Artificial Intelligence** (**XAI**) as "Given an audience, an explainable Artificial Intelligence is one that produces details or reasons to make its functioning clear or easy to understand." Furthermore, Murdoch et al. [150] define interpretable or **eXplainable Machine Learning** (**XML**) as "the extraction of relevant knowledge from a machine learning model concerning relationships either contained in data or learned by the model", where the knowledge is considered relevant if it provides insight into the problem faced by the target audience. Accordingly, we define **eXplainable Anomaly Detection** (**XAD**) as *the extraction of relevant knowledge from an anomaly detection model concerning relationships either contained in data or learned by the model*, where the knowledge is considered relevant if it can provide insight into the anomaly detection problem investigated by the end-user. Hereinafter, we utilize XAI and XML interchangeably as they practically mean the same within the scope of this manuscript.

Miller [142] defined XAI as a human-agent interaction problem at the intersection of Artificial Intelligence, **Human-Computer Interaction** (**HCI**), and the Social Sciences (including Philosophy, Cognitive Science, and Social Psychology). Being a subfield of XAI, XAD can also be situated at the intersection of those three domains. Therefore, in addition to considering different XAD tasks and problems together with their algorithmic and computational challenges, it would also be of interest to consider questions such as *how do humans understand an explanation*, *what kind of explanations are human-understandable*, and *how do humans interact with machines to understand explanations?* Thoroughly addressing these questions, however, would require substantial additional coverage and analysis of the literature; to maintain a clear scope and prevent the survey from becoming even longer, we will not address these questions. Instead, we refer to recent articles for perspectives from HCI [202] and social science [142, 143], and leave a broader discussion of these aspects to a future article.

The anomaly analysis process consists of two equally important tasks, namely, *anomaly detection* and *anomaly explanation*. Anomaly explanation refers to the process of finding out why an

anomaly is considered anomalous. Because the terms *anomaly* and *outlier* are used interchangeably, anomaly explanation is also known as outlier explanation, outlier interpretation, outlier description, outlier characterization, outlying property detection, outlying aspects mining, outlying subspaces detection, object explanation, and promotion analysis.

An anomaly can be identified by an anomaly detection algorithm or, otherwise, become known (e.g., from an expert).

— **Case 1 (Model).** If an anomaly is identified by an anomaly detection algorithm, *XAD* aims at explaining the anomaly by making the anomaly detection method interpretable. Specifically, there exist many approaches to make an anomaly detector interpretable. If the anomaly detector is intrinsically interpretable (e.g., logistic regression, shallow decision trees, and rule-based models), it is relatively easy to deduce why the anomaly is flagged as anomalous. In contrast, if the anomaly detector is not intrinsically interpretable (e.g., Isolation Forest [122], RNN [183], and CNN [74]), post-hoc XAI techniques such as SHAP [128], LIME [174], and Anchors [175] can be used to interpret the anomaly detector, namely, to describe why it makes certain decisions. In this case, we aim at making the *Detection-Definition* and *Explanation-Definition* consistent.

— **Case 2 (Data).** If an anomaly is identified by an expert, an anomaly explanation method can only aim at explaining why the given data instance is anomalous, extracting no knowledge from any anomaly detection models. In this case, we attempt to make the *Oracle-Definition* (if any) and *Explanation-Definition* consistent. However, it is also possible that the expert obtains the anomaly by running an existing anomaly detection algorithm, but the design of the algorithm is unavailable to the expert for some reasons (such as confidentiality). Hence, the *Explanation-Definition* may be different from the *Detection-Definition* (which is not known).

In short, the biggest difference between these two cases is about what to explain: the model (and possibly the data) or just the data. **Case 1** is centred around anomaly detection models. If we can understand how the anomaly detection model makes decisions, as a by-product, we can easily explain why an anomaly is flagged as anomalous by the model. In contrast, **Case 2** focuses on anomalies and aims at explaining why they are anomalous where the detection model is not available. The anomaly explanation methods corresponding to this case can be considered as surrogate methods for the unavailable anomaly detection models. For completeness, we will consider both cases in this survey.

2.3 Why Should We Care About XAD?

Due to the widespread application of anomaly detection in many domains, the interpretability of corresponding methods has become increasingly important [162]. For example, anomaly detection algorithms are being used to diagnose diseases in healthcare [212]. In financial services, many banks use anomaly detection methods to detect abnormal behaviour in credit card transactions [6]. In addition, the self-driving car manufacturing industry applies anomaly detection algorithms on camera data to detect corner cases [23]. In other safety-critical areas—such as spacecraft design—anomaly detection algorithms are used to detect sensor faults [70]. As we can see, anomaly detection systems for high-stakes decisions are deeply impacting our daily lives and society. One natural question is, *how can we trust these systems without understanding and validating the underlying rationale of the involved anomaly detection components?* For this reason, XAD aims to not only provide accurate anomaly detection results but also to provide tangible explanations of why a specific object is detected as an anomaly [155].

Providing anomaly detection results with corresponding explanations can help gain the trust of end-users in anomaly detection systems. Moreover, the explanations can also assist end-users

to validate the anomaly detection results in unsupervised settings. Even more, explanations can potentially enable end-users to find the root causes of anomalies and thereby take remedial or preventive actions.

For a long time, however, the anomaly detection community has mainly focused on detection accuracy, largely ignoring the interpretation of corresponding decisions. For instance, Micenková et al. [141] criticise that "almost all existing algorithms stop at the point of providing anomaly ranking and leave the user without any explanation of why some data points deviate and how." Additionally, Dang et al. [53] indicate that "although there is a large number of techniques for discovering global and local anomalous patterns, most attempts focus solely on the aspect of outlier identification, ignoring the equally important problem of outlier interpretation." Aggarwal [1] also points out that "only few outlier detection studies considered providing some qualitative information to explain the form of outlierness." Similarly, Vinh et al. [216] argue that "current outlier detection techniques do not usually offer an explanation as to why the outliers are considered as such, or in other words, pointing out their outlying aspects."

In summary, the anomaly detection community has long been paying more attention to *giving correct answers* rather than *providing explanations* or—even better—*providing correct explanations*. With more and more applications or potential applications of anomaly detection in high-risk decision-making systems, it has become crucial to gain or increase humans' trust in and acceptance of anomaly detection techniques. For this it is important to provide correct answers with correct explanations, i.e., to avoid the *Clever Hans Phenomenon* [112] that—in this context—refers to anomaly detection models utilising spurious correlations and patterns in the data to identify anomalies. Although the identified anomalies are true, these correlations or patterns may be incorrect or undesirable (e.g., violating the laws of physics). Such provably incorrect explanations are unacceptable to end-users and would only harm trust.

2.4 What is a Good XAD Method?

Once explanations are generated by an XAD method, how can one trust them? A natural first step is to evaluate the quality of generated explanations. Studies relevant to this have been conducted in the realm of XAI. For instance, references [20, 76] analyze the XAI literature and propose important properties that should be considered when designing an XAI technique. Next, Barbado et al. [18] defines some criteria to evaluate rule-extraction-based explanation techniques. Moreover, Zhou et al. [229] perform a survey on the quality evaluation of machine learning explanations. Recently, Sipple & Youssef [197] proposes four desiderata for anomaly explanation methods as well as a method for comparing different explanations. However, there is no consensus on what a good XAD technique should be. Based on related work on XAI, we find the following properties to be especially relevant when designing or choosing an XAD technique:

— Accuracy: how accurate is the prediction of unseen anomalous instances as anomalies;
— Fidelity: consistency of *Oracle-Definition*, *Detection-Definition*, and *Explanation-Definition*;
— Comprehensibility: to what extent are the explanations understandable to the end-users;
— Generality: does the technique have special requirements for data type, data size, anomaly detection model type, anomaly detection model size, training regimes, or training restrictions;
— Scalability: does it scale to large input data size and/or a large model;
— Complexity: how many hyper-parameters need to be set by end-users.

The practical implementation and evaluation of XAD techniques are largely dependent on the application domain and end-users and are, therefore, out of the scope of this survey.

3 A TAXONOMY OF EXPLAINABLE ANOMALY DETECTION METHODS

Before we introduce the taxonomy that we propose for the field of XAD, we first briefly review existing surveys and taxonomies.

3.1 Related Work

Compared to the abundance of taxonomies of anomaly detection methods, including but not limited to these surveys throughout the years [3, 36, 37, 78, 86, 161], the categorization of anomaly explanation methods involving XAD techniques has received relatively little attention so far [162, 185, 189, 216, 225].

We discuss the four most notable existing categorizations. Vinh et al. [216] for the first time subdivided anomaly explanation approaches into two categories: *Feature selection-based approaches* that transform the anomaly explanation task into the classical problem of feature selection for classification, and *Score-and-search approaches* that compare the outlyingness degree of an anomaly across all subspaces followed by inspecting the subspace with the highest anomaly score. To the best of our knowledge, Samariya et al. [185] was the first work dedicated to the survey of anomaly explanation methods. They also subdivided related techniques into three categories: *Score-and-Search-based approaches*, *Feature selection-based approaches*, and *Hybrid approaches*. More recently, Panjei et al. [162] introduced a survey on anomaly explanation, wherein they divided relevant techniques into three categories: *Importance Levels of Outliers*, *Causal Interactions Among Outliers*, and *Outlying Attributes*. Meanwhile, Yepmo et al. [225] also presented a review of anomaly explanation methods, categorizing existing techniques into four groups, namely, *Explanations by Feature Importance*, *Explanations by Feature Values*, *Explanations by Data Points Comparison*, and *Explanations by Structure Analysis*. Finally, Reference [189] is also closely related, wherein they have discussed what anomaly explanations are, who needs those explanations, and why there are different types of anomaly explanations.

After a thorough survey of the scientific literature on XAD techniques, we find that existing surveys are less comprehensive than we aim to be in this manuscript. Specifically, each of the above surveys contains no more than 40 relevant works in the field. In contrast, our survey has investigated more than 150 relevant articles. In addition, we find the existing taxonomies to be relatively coarse and sometimes not intuitive. For example, although anomaly score is a very natural ranking of outlying degree, Panjei et al. [162] particularly treat anomaly ranking as a subcategory of anomaly explanation methods. Furthermore, although *Explanations by Feature Importance* and *Explanations by Feature Values* mainly differ in the granularity of provided explanations, Yepmo et al. [225] regard them as two distinct categories. In brief, existing surveys only partially cover existing research, and the proposed taxonomies are insufficient to characterize the increasingly rich field of XAD. For this reason, we perform a comprehensive and structured survey on state-of-the-art XAD techniques. As new articles are published at a rapid pace, we do not claim to have covered all relevant research publications. Furthermore, as we intend to include a wide spectrum of XAD methods, we cannot describe each method in detail. Meanwhile, a refined taxonomy, distilled from existing surveys on XAI techniques, is presented below and used to categorize XAD methods.

3.2 Proposed Taxonomy

Similar to how anomaly detection is an important part of machine learning and data mining, we argue that XAD is also an important constituent of what is nowadays called XAI. XAI has received extensive attention in the past few years due to the emergence and prevalence of black-box models such as deep neural networks. After carefully scrutinizing existing surveys on XAI [13, 20, 30, 34, 62, 71, 120], we found that some criteria are often used to categorize existing XAI techniques. Capitalizing on these findings, we propose six main criteria to taxonomize existing XAD techniques.

First of all, according to the anomaly detection pipeline as shown in Figure 1, we can subdivide XAD techniques into three categories, namely, *Pre-model techniques*, *In-model techniques* and *Post-model techniques*. Specifically, *pre-model techniques*, also known as *ante-hoc techniques*, are constructed and implemented before the anomaly detection process. Techniques such as filter feature selection methods belong to this category. *In-model techniques* use inherently interpretable models and can therefore provide explanations without additional or with little effort when performing anomaly detection. For example, anomaly detection methods based on linear regression, which can simultaneously report the coefficients of the corresponding features, fall into this category. In contrast, *post-model techniques*, also known as *post-hoc techniques*, attempt to explain the decisions made by an anomaly detection model after the construction and implementation of the detection model or when anomalies are obtained from an oracle. For instance, SHAP-based interpretation methods [128] are part of this category.

Second, we distinguish XAD techniques based on whether they provide a *global explanation* or *local explanation*. Specifically, a *global explanation* is based on the understanding of the complete 'model logic' or some important properties of the anomaly detection model, being able to explain how all decisions are made. In contrast, a *local explanation* explains why a specific object is anomalous or how a specific decision is made.

Third, XAI techniques can be further subdivided into *model-agnostic* approaches that can be applied to any anomaly detection model, and *model-specific* approaches that are only applicable to specific anomaly detection models.

Fourth, two aspects of a tabular dataset can be used to generate explanations, i.e., a tabular dataset has features and samples. Therefore, we can subdivide techniques into three subcategories:

— *Feature-based methods* provide explanations based on features. This group of methods generally indicates which features are important and/or the corresponding values of investigated anomalies. Specifically, *subspace* (e.g., a subset or unordered features), *a set of subspaces* (e.g., a set of feature pairs), *feature importance* (e.g., assigning a score or an order to each feature), and *feature values* (e.g., rare combination of feature values) fall into this subcategory. Particularly, some studies attempt to define a set of rules based on a subset of features and their corresponding values, resulting in so-called patterns. Meanwhile, for sequential data such as time series, a pattern consisting of a collection of consecutive observations is usually leveraged to detect and explain anomalies. Each observation can be regarded as a feature or a sample depending on the context. For simplicity, we call them *pattern-based methods*, but they are still essentially *feature-based methods*.

— *Sample-based methods* generate explanations based on samples. This type of method typically compares the abnormal object directly to normal objects to demonstrate differences. For instance, *local neighbourhood* (e.g., the nearest objects, which may be normal or abnormal, to an anomaly), *counterexample* (e.g., the nearest normal object to an anomaly), and *contextual anomalies* (e.g., the nearest cluster to an anomaly) belong to this subcategory. Moreover, *exception analysis* in Reference [76] and *representative examples* in References [20, 211] also fall under this category.

— *Feature and Sample-based methods* leverage both aspects.

Fifth, based on the specific techniques used to generate explanations, we can categorize models into the following subcategories, which are not mutually exclusive:

— *Approximation-based methods*, which approximate or mimic complex models with simpler ones that are much easier to interpret. They are also called surrogate models. Examples include LIME [174] and Anchors [175].
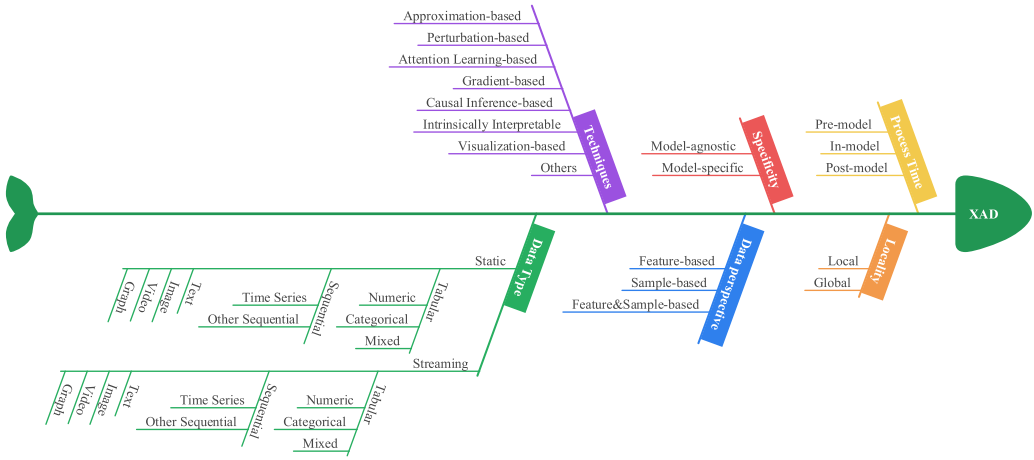
Fig. 2. XAD taxonomy based on six criteria (coloured text boxes). Most existing XAD methods fall into one category for each of the six criteria.

— *Perturbation-based methods*, which examine the influence of output via input changes to generate explanations. Examples include Anchors [175].

— *Reconstruction Error-based methods*, which use reconstruction errors to explain anomalies. Examples include SHAP-based methods [12].

— *Attention Learning-based methods*, which use attention learning to localise anomalies. Examples include Reference [28] and Reference [215].

— *Gradient-based methods*, which measure feature contribution on midput (intermediate outputs) or outputs through back-propagation. Examples include Layer-wise Relevance Propagation [96, 160, 196]. Note that some of these methods may also be Reconstruction Error based.

— *Causal Inference-based methods*, which analyze the causal relations between objects and/or features to explain anomalies. Examples include Reference [124].

— *Visualization-based methods*, which use plots to explain anomalies. Examples include Reference [126], which uses heatmaps that is a kind of saliency masks. Note that many other techniques also leverage visualization to explain anomalies.

— *Intrinsically Explainable methods*. The above-mentioned subcategories are mainly post-model techniques that are used to explain deep learning-based anomaly detection models. Meanwhile, there are in-model techniques that make the anomaly detection model intrinsically explainable. Examples include Rule-based models [83].

— *Miscellaneous other methods*. We assign other techniques to this subcategory by indicating their specific technique. Examples include Pattern Compression [200], and Subspace Anomaly Detection [186].

Sixth and last, we also indicate the types of data to which each XAD technique can be applied. Specifically, the data type can be static or streaming. Furthermore, it can be tabular (numeric, categorical, or mixed), sequential (time series, other sequential), image, text, video, or graph data.

Our overall proposed taxonomy is presented in Figure 2: each of the six criteria can be used to categorize an XAD method. Together these six 'dimensions' can be used to provide a detailed characterization of an existing XAD method, or—the other way around—to find XAD methods satisfying certain requirements.
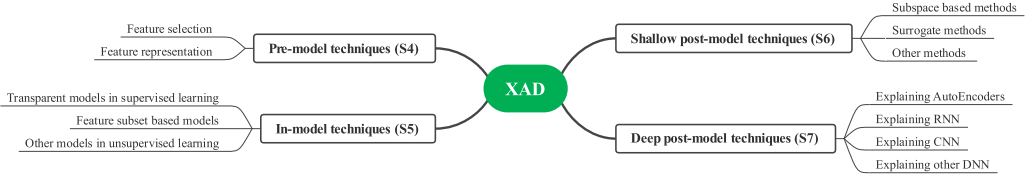
Fig. 3. Structure of the core of this survey, i.e., Sections 4–7 (indicated by S4–7).

3.3 Organisation of the Literature Review

As described in the previous subsection and shown in Figure 2, our taxonomy employs six criteria. To organize our survey by these six criteria; however, we would have to introduce many section levels and some subsections would be much longer than others. We will therefore use another structure (as shown in Figure 3) for the literature review in the following sections, which we will explain next. We will still make ample use of our proposed taxonomy: to partially structure the individual sections, to characterize the methods that we describe, and to provide a full characterization of all methods in a large overview table at the end of each section.

We use the first main criterion to classify *pre-model techniques* (S4), *in-model techniques* (S5), and *post-model techniques* (S6-7) into different sections. As there are so many post-model techniques, we split those into deep learning-based methods (S7) and other, 'shallow' methods (S6). Next, we use the characteristics of each of these categories to define subsections. That is, the *pre-model techniques* section consists of subsections for *feature selection* and *feature representation*. Meanwhile, the *in-model techniques* section includes subsections for *transparent models in supervised learning*, *feature subset-based models*, and *other models in unsupervised learning*. The *shallow post-model techniques* section has subsections for *subspace-based methods*, *surrogate methods*, and *miscellaneous methods*. Finally, the *deep post-model techniques* section contains subsections *explaining AutoEncoders*, *explaining RNNs*, *explaining CNNs*, and *explaining other DNNs*.

4 LITERATURE REVIEW ON PRE-MODEL TECHNIQUES

Opaque models are often criticized for their inexplicability. However, the features used as inputs to models are as critical as, if not more than, the type of models in producing explainable results. In other words, by having more meaningful and informative features whilst retaining fewer irrelevant features, we can build simpler models to learn the relationships exhibited in the data while ensuring comparable anomaly detection accuracy, see Figure 3.

Therefore, this section reviews articles that leverage XAD techniques *before* the anomaly detection process. Specifically, the following pre-model techniques are investigated:

— Feature selection methods that select a subset of original features for anomaly detection;
— Feature representation methods that learn a set of high-level and human-understandable feature representations for anomaly detection.

4.1 Feature Selection for Anomaly Detection

Siddiqui et al. [193] point out that the effort required to investigate an anomaly is usually proportional to the number of features that describe it. Therefore, dimensionality reduction techniques—including feature projection and feature selection methods—can be applied to reduce the number of features that describe an object, thereby, facilitating anomaly explanation. However, feature projection methods such as Principal Component Analysis convert the original features into a new set of features, sacrificing interpretability. In contrast, feature selection methods retain a subset of original features that contain the most important information, greatly improving the

interpretability and effectively alleviating the *curse of dimensionality* problem in high-dimensional data.

There exist very limited unsupervised feature selection methods for anomaly detection. Specifically, Pang et al. [156] and Pang et al. [158] propose two filter-based unsupervised selection methods, namely, CBRW_FS and CBRW, which select a subset of features independently from subsequent anomaly detection methods. These two methods work only on categorical data through modelling the feature-value couplings. By assuming strong similarities between rare instances, He & Carbonell [82] design an optimization framework to jointly select features and instances for anomaly detection on categorical data. However, this assumption is usually not satisfied since anomalies are often isolated and thus distinct from each other.

Meanwhile, Noto et al. [154] and Paulheim & Meusel [167] try to find a relevant feature subset for anomaly detection by exploring the correlations between features. They assume that anomalies are those instances that violate the normal dependencies between features. Therefore, only features that are related to other features are considered relevant for anomaly detection. Unfortunately, this anomaly definition is not applicable to many benchmark anomaly detectors. Moreover, Isolation Forest [122] can also be used to select a subset of features for anomaly detection. The isolation forest-based feature selection method, IBFS [223], simply selects features that contribute the most to the outlyingness of anomalies reported by the Isolation Forest method. To our knowledge, this is the first unsupervised feature selection method specifically designed for *generic* anomaly detection in numeric data. The above three methods are all filter-based, which independently select subsets of features regardless of subsequent anomaly detection methods. Consequently, suboptimal or completely irrelevant features may be selected for anomaly detectors.

A **platform information technology** (**PIT**) system is a system capable of connecting and communicating with other systems, subsystems and devices. To detect attacks in PIT systems, Morris [146] proposes to use **Principal Component Analysis** (**PCA**) or **Independent Component Analysis** (**ICA**) to reduce the number of features considered, thereby promoting interpretability in the subsequent anomaly detection process. Moreover, he suggests using ensemble learning-based methods such as Random Forests to detection anomalies after the dimensionality reduction process. However, every feature obtained using PCA or ICA is a combination of the original features and is, therefore, no longer interpretable.

Some feature selection methods are interleaved with the anomaly detection process, rather than being applied before the anomaly detection process. We call such methods wrapper or embedded feature selection methods depending on their implementations and will introduce them in the next section.

4.2 Feature Representation for Anomaly Detection

Due to the complexity entailed in data such as time series, image, video, and so on, **deep neural network** (**DNN**)-based methods have shown superiority in detecting anomalies in these data. However, DNN-based models are notoriously known for their complexity, which implies uninterpretability. To alleviate this problem, Chen et al. [43] and Wu et al. [219] indicate that using high-level and human-understandable feature representations for anomaly detection can reduce the complexity of subsequent anomaly detection models, thereby improving their interpretability.

Examples can be observed in the domain of time series anomaly analysis. For instance, Ramirez et al. [173] introduce an interpretable anomaly detection and classification framework to analyze human gait. Specifically, they first harness symbolic representations such as Piecewise Aggregate Approximation to represent the collected multivariate time series data. Particularly, they consider the symbolic abstraction of the data as the core of their XAD framework, enhancing interpretability of the results via feature reduction. Second, they apply two discords-based anomaly detection

methods, viz. HOT-SAX [101] and RRA [191], to discovery anomalies, respectively. Third, they determine the final anomalies based on the consensus of these two detection algorithms.

Instead of using symbolic representations, Dissanayake et al. [60] investigate the importance of heart sound segmentation and feature extraction for detecting abnormal heart sound. They suggest that an automated detection method usually consists of three steps: Segmentation, Feature Extraction, and Classification. First, they apply the model proposed by Fernando et al. [69] to perform segmentation. Particularly, the segmentation is based on a feature representation called **Mel-Frequency Cepstral Coefficients** (**MFCCs**). They argue that pre-extracted feature representations such as MFCCs or spectrogram are commonly used in medical domain as they are closely related to the original signal. One can gain important insights into the model prediction results if explaining the feature representations in conjunction with the signal. Second, they utilise a **Convolution Neural Network (CNN)** encoder to extract features. Third, they construct a **Multilayer Perceptron** (**MLP**) Network model to perform anomaly detection. Moreover, to interpret an anomaly, they combine Shapley values and Occlusion maps [227] to investigate how input features impact the prediction.

Schlegl et al. [187] construct a DNN-based model that can learn interpretable feature representations from unlabeled time series, facilitating the evaluation and deployment of subsequent anomaly detection algorithms. First, they set up a so-called *deviation convolution* based model to learn characteristic shapes of normal time series, wherein they impose a separating constraint on the neural network to make it interpretable. Second, they feed these human-interpretable shapes to a convolutional-RNN AutoEncoder, which attempts to reconstruct the input shapes while minimising the reconstruction errors. Therefore, a test instance with a large reconstruction error is considered anomalous.

In the field of video anomaly analysis, Wu et al. [219] propose a **Denoising AutoEncoder** (**DAE**)-based model combined with SHAP to detect and explain anomalies in videos. Since uninterpretable feature representations hide the decision-making process, they first leverage pretrained **Convolutional Neural Network (CNN)** models to extract high-level concept and contextual features. Second, they train a DAE model based on these features to predict the video frame. On this basis, a test instance is considered anomalous if its actual frame is significantly different from its predicted frame. Third, they apply kernel SHAP [128] to find input features which cause the anomaly.

4.3 Summary

As shown in Table 1, all pre-model XAD techniques are model-agnostic except for Reference [173]. In other words, most pre-model XAD techniques can be applied to any subsequent anomaly detection methods. However, fully decoupling the feature selection or feature representation learning from the subsequent anomaly detection methods may lead to sub-optimal detection accuracy.

Furthermore, most reviewed pre-model XAD techniques are feature-based with the exception that He & Carbonell [82] also perform instance selection to improve interpretability. Importantly, all pre-model XAD techniques can provide global explanations in the sense that they render the subsequent anomaly detection models more transparent and interpretable by preserving less irrelevant or redundant features, or providing human-understandable feature representations.

The ultimate goal of using XAD techniques is to ensure that the entire pipeline of anomaly detection is human-understandable. However, we note that high-level and human-understandable feature representations are usually obtained by an opaque model, such as a pre-trained CNN model in Reference [60], which somewhat offsets the benefits of using interpretable feature representations for anomaly detection.

Table 1. Summary of Pre-model XAD Techniques

Ref.	Spec	Pers	Tech	Data	Loc		Pros	Cons
[156]	A	F	Feature selection	Static TC	G		Handles noisy features well	Only applicable to categorical data
[158]	A	F	Feature selection	Static TC	G		Linear time complexity to data size	Only applicable to categorical data
[82]	A	F & S	Feature selection + Instance selection	Static TC	G		Jointly selects features and instances for AD	Assumes strong similarities between rare instances
[154]	A	F	Feature selection	Static TN	G		Robust to noisy and high-dimensional data	Only explores correlations between features
[167]	A	F	Feature selection	Static TN	G		Changes unsupervised AD into supervised AD	Only explores correlations between features
[223]	A	F	Feature selection	Static TN	G		Applicable to generic AD for numeric data	Selects features without considering subsequent AD methods
[146]	A	F	Feature selection	Static TN	G		Applicable to generic AD	Obtained features are not interpretable
[219]	A	F	Pretrained CNN models to extract high-level concept and contextual features; VAE + SHAP	Static video	L	& G	Extracted features are easy to understand	Weak interpretability due to the opacity of CNN
[173]	S	P	Symbolic representation using PAA	Static MTS	G		Enables human-in-the-loop	Only applicable to symbolic based AD such as HOT-SAX and RRA
[60]	A	F	Pre-extracted feature representations (MFCCs/spectrogram); SHAP + Occlusion maps	Heart sound sig-nals/UTS	L	& G	Simple, stable and efficient architecture	Only applicable to DNN
[187]	A	F	Explainable feature representations	Static MTS	G		Easy to visualize	Weak interpretability due to the opacity of RNN-based AD; Fails to learn less frequent shapes

Spec indicates whether a method is model-agnostic (A) or model-specific (S). *Pers* specifies whether a method is feature-based (F), sample-based (S) or pattern-based (P). *Tech* indicates the techniques used in each method. *Data* indicates the data type for which the method is applicable (TN: Tabular Numeric; TC: Tabular Categorical; TM: Tabular Mixed; UTS: Univariate Time Series; MTS: Multivariate Time Series; ES: Event Sequence). *Loc* shows whether a method provides a local explanation (L) or global explanation (G). *Pros* and *Cons* describe advantages and disadvantages of each method, respectively.

Moreover, it can be seen that the reviewed feature selection and feature representation techniques are model-based feature engineering methods, which only leverage machine learning techniques. However, one can employ domain-knowledge-based feature engineering methods to extract features. For instance, Murdoch et al. [150] point out that combining exploratory data

analysis tools with domain knowledge is helpful for extracting meaningful features, thereby improving the interpretability of subsequent anomaly detection.

5 LITERATURE REVIEW ON IN-MODEL TECHNIQUES

This section presents anomaly detection models that are considered to be inherently explainable. These anomaly detection models can provide insights into the relationships they have learned from the data, enabling an end-user to understand the decisions they have made. In general, the following methods are considered intrinsically explainable:

— Commonly seen transparent models in supervised learning, including Linear Models (Linear Regression, Logistic Regression), Decision Trees, Gaussian Process, Rule-based Learners, Generative Additive Models, and Bayesian Models;
— Feature subset based methods, including subspace anomaly detection methods, wrapper or embedded feature selection methods for anomaly detection;
— Miscellaneous other methods (mostly in an unsupervised setting) that reveal the rationale for how anomaly scores are calculated in a comprehensible way.

5.1 Transparent Models in Supervised Learning

According to Lipton [121], a model is transparent if its intrinsic structure satisfies at least one of the following three requirements:

— **Simulatability:** if a model can be simulated by a human, and thus whether it is possible to reason about its entire decision-making process.
— **Decomposability:** if a model can be broken down into multiple parts, and these parts are easy to explain individually.
— **Algorithmic Transparency:** if a human can understand the process by which the model generates output from a given input.

In a supervised setting, commonly seen transparent models include Linear Models (such as Linear Regression and Logistic Regression), Decision Trees, Rule-based Learners in the form of *if-then rules, m-of-n rules, list of rules, falling rule lists* or *decision sets*, Gaussian Process, **Generative Additive Models (GAMs)**, and Bayesian Models. Although anomaly detection is often an unsupervised problem, it can often leverage these methods in some way. However, transparency is not sufficient to guarantee explainability. Specifically, when a transparent model becomes exceedingly complex, it is not human-understandable anymore. Therefore, anomaly detection models that are developed based on these transparent models are considered to be explainable as long as they are not overly complex.

First, rule-based models are often leveraged to learn frequent patterns in the data, enabling interpretable anomaly detection. For instance, He et al. [83] apply frequent pattern mining to identify and explain anomalies in transaction data. Specifically, they leverage the Apriori algorithm [2] to find frequent patterns, and utilise the so-called top-k contradictory frequent patterns to explain each identified anomaly. Similarly, Zhu et al. [230] propose a model to capture frequent motion and background patterns of activities in video data, treating patterns that deviate from learned frequent patterns as anomalies. Likewise, Vaculík & Popelínský [213] put forward the DRGMiner model, which mines frequent patterns in dynamic graphs and considers graphs deviating from these patterns as anomalous. Besides, Mauro et al. [138] propose HyVarRec to detect and explain anomalous traces for context-aware software product lines. Concretely, they apply Satisfiability Modulo Theories [57] to construct a conjunction of constraints that should be satisfied by normal traces when considering their contexts. As a result, a trace that violates the predefined constraints

is considered anomalous. Moreover, Böhmer & Rinderle-Ma [24] develop the ADAR model to detect and explain anomalies in process runtime behaviour. Specifically, ADAR leverages association rule mining to extract a set of ordered rules that normal traces should satisfy. Hence, a test trace with a small support is considered anomalous. Importantly, they also propose a visualization technique called A_Viz to show the rule violation.

Second, decision trees and their variants have also been proposed to be used for the detection of anomalies, resulting in intrinsically explainable detection results. For instance, Kraiem et al. [108] introduce the ***Composition-based Decision Tree*** (**CDT**) to detect and interpret anomalies in time series. Specifically, after preprocessing and labelling of given time series, a CDT is constructed as an extension of a decision tree on this labelled data, extracting rules for describing seen anomalies and detecting unseen anomalies. Also, the authors evaluate the explanation quality in terms of the number of used patterns and the length of rules. Furthermore, Cortes [51] presents an anomaly detection method that performs supervised decision tree splits on features, wherein the one-dimensional confidence intervals of each branch are built for the target feature. As a result, explanations can be obtained from the branching conditions and the general distribution statistics of non-anomalies that fall into the same branch. Besides, Aguilar et al. [4] propose the **Decision Tree-based AutoEncoder (DTAE)** model to detect anomalies. Specifically, they use a decision tree to depict the encoding and decoding portions of AE, determining whether an instance is anomalous by comparing the input with the output. The advantage of using decision trees as encoders and decoders is that each tree contains the rules for categorizing tuples, offering interpretability. Meanwhile, Itani et al. [90] develop the so-called **one-class decision tree** (**OC-Tree**) model, which employs Kernel Density Estimation to divide data subsets into intervals of interest and then encloses the data within hyperrectangles that can be explained by a set of rules. Additionally, Perez & Lavalle [168] devise the alleged User Model to detect potential fraud in bank transactions, where they fit manually selected features into a threshold-based rule model, classifying the model outputs in the form of fraud probability into five categories.

Third, another line of research utilises regression models to perform anomaly detection, providing explanations for identified anomalies. For example, for each data instance, Chen et al. [41] apply LOESS regression [49] by taking each feature in turn as the target variable and the remaining features as predictors based on its neighbours. An instance is considered anomalous in a certain feature if its actual value differs significantly from its predicted value. Particularly, for each identified anomaly, they provide a natural language explanation consisting of its considered neighbours and the associated feature differences. Besides, in Burak Gunay et al. [29], the heating and cooling load patterns of buildings are studied using three inverse models, including a univariate change point model, a regression trees-based model, and a DNN-based model. Particularly, change point models and regression trees are easy to interpret and can generate rules from their output. Moreover, Langone et al. [111] leverage regularized Logistic Regression to identify anomalies in time series. In brief, they first utilise a bucket-based representation to represent the data, and then implement a rolling window procedure to extract features. On this basis, they employ the Kolmogorov-Smirnov distance to select relevant features for anomaly detection, and the resulting features are fed to a Logistic Regression with Elastic Net regularization to detect anomalies.

Fourth, some researchers utilize intrinsically interpretable models such as **Gaussian Processes (GPs)**, **Generalized Additive Models (GAMs)**, and **Dynamic Bayesian Networks (DBNs)** to detect anomalies. For instance, Berns et al. [21] employ GPs to detect anomalies, where a GP is a stochastic process defined over a set of random variables such that every finite subset of these random variables follows a multivariate Gaussian distribution. If the actual value of a test instance deviates significantly from its predicted value, the GP model treats it as an anomaly. Meanwhile, Chang et al. [38] present an XAD model named DIAD based on GAMs. Specifically, a GAM model

is a linear combination of smooth functions, where each function is defined on some variables. Given an anomaly, one can easily infer which features contribute the most to its outlyingness. Moreover, Slavic et al. [199] develop a DBNs-based model to predict the state of a moving object in Autonomous Driving domain, attempting to identify abnormal motion behaviours based on its motion direction and orthogonal direction. A test instance is considered anomalous if its predicted state deviates significantly from its actual state. Due to the good properties of DBN, they can decompose the anomalous motion along its two directions and resort to the corresponding parameters to interpret the anomaly.

Finally, an important line of research attempts to introduce interpretable components in a complex anomaly detection model, providing weak interpretability. For instance, Zancato et al. [226] propose the STRIC model to detect anomalies in multivariate time series data. Specifically, STRIC consists of four layers. The first layer attempts to model the trend of time series by using a cascade of linear filters. The second layer implements a linear module to model and remove the seasonality at multiple time scales. Next, the third layer comprises a linear stationary filter bank that is able to approximate any trend or seasonality. Finally, the fourth non-linear layer consists of a randomly initialized Temporal Convolution Network model. Therefore, these four layers constitute a model capable of predicting time series. On this basis, they extend the CUMSUM algorithm [224] to detect anomalies by using the likelihood ratio between two windows of prediction residuals. Particularly, the linear components used in STRIC provide interpretability.

Discussion: To take advantage of these transparent models for anomaly detection, one usually needs to turn the unsupervised anomaly detection problem into a supervised or semi-supervised setting. For instance, References [24, 199, 213] attempt to learn normal behaviours or patterns by training the model on exclusively normal data, and then identify anomalies by comparing a test instance with the expected normal behaviours. Meanwhile, References [41, 108] either directly leverage labelled data or decompose an unsupervised problem into many supervised problems [167].

5.2 Feature Subset-based Models

The methods in this subsection select one or more subsets of features to detect and explain anomalies. Specifically, it contains subspace anomaly detection methods and feature selection methods for anomaly detection. Subspace anomaly detection methods find anomalies that are only detectable in certain subspaces, providing intrinsic explanations based on subspaces. Moreover, wrapper or embedded feature selection methods select a subset of original features that are relevant for anomaly detection, thereby improving the interpretability of detection results. Note that wrapper or embedded feature selection methods select features during the process of performing anomaly detection, not before the anomaly detection process (see Section 4.1). Furthermore, feature selection methods can be considered as a special case of subspace anomaly detection since they actually select a subspace for anomaly detection.

First, subspace anomaly detection usually consists of two steps: finding subspaces and assigning anomaly scores. Subspace anomaly detection has received extensive attention, resulting in a collection of strategies for finding subspaces and assigning anomaly scores. In general, finding subspaces and assigning anomaly scores can be decoupled or intertwined. For instance, Muller et al. [149] propose OUTRES. For each instance, they first use the Kolmogorov–Smirnov goodness of fit test to exclude some subspaces from the powerset of features. Specifically, they exclude subspaces in which the local densities of the given instance and its neighbourhood are uniformly random distributed. Then, for the residual subspaces, they define a dimensionality-unbiased anomaly scoring function to measure the local density deviation of the given instance. Next, they aggregate the anomaly scores of each instance across its non-uniformly random distributed

subspaces. Moreover, Keller et al. [99] present HICS. Specifically, they first look for subspaces with high contrast by measuring the correlation between features in a subspace using statistical tests, viz. the difference between marginal probability density and conditional probability density. Second, they apply an off-the-shelf anomaly ranking method such as LOF [26] on selected subspaces and aggregate the results. It can be seen that both methods [99, 149] decouple subspace search and anomaly scoring. In contrast, Dang et al. [52] leverage spectral graph theory to achieve subspace anomaly detection. Specifically, they first construct an undirected graph that can capture the local geometry of all instances. Second, they attempt to learn an optimal subspace that can separate well an instance from its neighbours, while preserving the intrinsic geometrical structure of data. Correspondingly, a well separated instance is considered anomalous and the corresponding subspace acts as an explanation. The subspace search and anomaly scoring are intertwined in this method.

Some researchers attempt to leverage dimensionality reduction or feature projection techniques to perform subspace anomaly detection. For example, given a data instance with its global nearest neighbours, Kriegel et al. [109] first project these data instances with all d features into subspaces of varying size, where the subspace is spanned by the l largest principle components using robust PCA. Meanwhile, they compute the projection to the subspaces spanned by the remaining $d - l$ principle components as its error vectors. Second, they choose the error vector with the largest L_2-norm value as its anomaly score and explanation. The rationale for using PCA is that the correlation dimensionality is highly related to the intrinsic dimensionality of data. Meanwhile, Bin et al. [22] develop the ASPCA model. Given a dataset with D features, they first compute and order the principal components using sparse PCA [94]. The first k principle components which capture most of the variance are called *normal subspace*, and the remaining $D - k$ components are called *abnormal subspace*. An instance is considered anomalous if it has a large projection length in the *abnormal subspace*. Based on sparse PCA, each feature is a linear combination of a few original features. Therefore, this method can easily obtain the original features that are responsible for an anomaly, resulting in an explanation. Furthermore, Dang et al. [53] introduce the LODI model. For each instance, they first select their neighbours using an information-theoretic tool. Second, they use local dimensionality reduction to select an optimal subspace in which this instance can be maximally separated from its neighbours. An instance that is relatively easy to separate is considered anomalous. More concretely, the local dimensionality reduction problem is solved via matrix eigendecomposition, which can return the corresponding original features that are most important to explain an anomaly. Additionally, Pevný [169] presents Loda, an online anomaly detection model that can also provide explanations. Specifically, Loda first leverages sparse random projections to obtain a collection of one-dimensional subspaces. Second, it constructs a histogram in each subspace, aiming to approximate the probability density. Third, it aggregates these one-dimensional histograms to estimate the joint probability density. Consequently, an instance with low estimated probability density is considered anomalous. For each identified anomaly, Loda can rank features according to their contributions to the anomaly score as an explanation.

As we can see, the above-mentioned methods utilise some well-defined criteria to search subspace and then assign anomaly scores. However, another line of research intends to use random search strategies to search for subspaces. For instance, Keller et al. [100] propose RefOut, which consists of three steps. They first generate an initial subspace pool that is a set of randomly selected subspaces. On this basis, they utilise an off-the-shelf anomaly detection model to perform anomaly detection, resulting in a set of anomaly scores for each instance. Second, for each instance, they generate a refined subspace by maximizing the discrepancy of anomaly scores. Aggregating all these refined subspaces leads to a refined subspace pool. Third, they apply again the anomaly detection model on the refined subspace pool to obtain an anomaly score for each instance. Considering that the cardinality of each refined subspace may be different, they normalize the anomaly

scores to ensure comparability. Accordingly, they return the maximum anomaly score and the corresponding subspace for each instance as an explanation. Similarly, Savkli & Schwartz [186] put forward RSMM. Concretely, they first randomly select m subspaces of dimension k, ensuring that each dimension contributes equally to the final probability model. Second, they construct a mixture model such as Gaussian Mixture Model in each subspace. Third, they compute the geometric averaging of the probability densities in all subspaces as the joint probability density. Therefore, if a test instance is located in a low-density region, it is considered anomalous. Furthermore, to interpret an anomaly, they rank features according to how often they consist in the subspaces where the anomaly is considered an anomaly.

Second, despite the prevalence of subspace anomaly detection, other techniques such as feature selection have also been exploited to facilitate the interpretability of anomaly detection. For instance, Pang et al. [159] introduce a wrapper feature selection framework for anomaly detection. Specifically, they first create an internal evaluation metric for anomaly detection and then select relevant features for detecting anomalies by iteratively maximizing this metric. They have only applied this framework to their proposed anomaly detection model though, which only works on categorical data. Besides, Pang et al. [157] propose an embedded feature selection method for anomaly detection, dubbed CINFO, which is an ensemble of sequential ensemble learners. Specifically, the base learner, namely, the sequential ensemble learner, iteratively and mutually refines the anomaly detection and feature selection processes. In this way, they build many similar base learners, which are then aggregated to produce the final anomaly scores. Hence, the method does not explicitly provide any selected features and lacks interpretability due to the use of an ensemble approach. Meanwhile, Roshan & Zafar [178] develop an **AutoEncoder (AE)**-based model incorporating the SHAP technique to detect and explain anomalies in computer network data. Specifically, they first train an AE model on exclusively normal computer network data with all input features, followed by applying Kernel SHAP to explain the predictions of the trained AE model. Next, they use the trained AE model to detect anomalies in another dataset containing cyberattacks and then apply again Kernel SHAP to explain the predictions, aiming to select a subset of important features to identify anomalies. Finally, these selected features are used to train a refined AE model for anomaly detection.

Discussion: The subspace anomaly detection methods introduced here are considered inherently explainable since they only explain anomalies identified by themselves. In other words, the *Detection-Definition* and *Explanation-Definition* of anomaly are usually consistent. Therefore, the generated explanations are intrinsic regardless of whether the anomaly detection and subspace search processes are interleaved or decoupled. In contrast, the subspace anomaly detection methods that will be presented in the *shallow post-model techniques* section are distinct, as they aim at interpreting anomalies that are identified by other detection models or experts. As a result, the *Detection-Definition* and *Explanation-Definition* of an anomaly are likely to be different since the *Detection-Definition* is generally unknown.

5.3 Other Miscellaneous Models in Unsupervised Learning

In principle, an anomaly detection model that reveals the rationale for how anomaly scores are calculated in a human comprehensible way can be considered intrinsically explainable. Hereinafter, we survey a collection of intrinsically XAD methods that do not belong to the commonly seen transparent methods in supervised learning nor feature subset-based methods. Due to the diversity of these methods, i.e., they share few basic techniques, we organize them according to the type of data they have been designed for.

5.3.1 Models for Tabular Data. A plethora of models have been devised to detect anomalies in tabular data whilst providing intrinsic explanations. For instance, as a typical method, distribution

based anomaly detection models attempt to fit data with probabilistic distributions. Then, data instances that do not conform to the fitted model are considered anomalous. According to Agyemang et al. [5], distribution-based anomaly detection techniques are intrinsically explainable. This is because the identified anomalies can be meaningfully interpreted from a statistical perspective once the probabilistic distribution is known. Dunstan et al. [66] take a different approach and utilise a data cube structure to divide transaction data instances into different regions. They refer to each region as a context and show how each instance can be abnormal in different contexts. More importantly, they create anomaly tables and anomaly lattices to explain anomalies. An anomaly table contains anomalous transactions alongside their contexts. Meanwhile, an anomaly lattice graphically displays the anomalies with their contexts. Rather than using groups to define contexts in which anomalies can be detected, Mejia [139] adapts **Adaptive Resonance Theory** (**ART**) [33] to group instances into clusters such that instances residing in the smallest clusters are considered anomalous. By virtue of the good properties of ART, one can obtain the feature differences between every two clusters, resulting in explanations for the anomalous instances.

Smets & Vreeken [200] take a more global approach to anomaly detection, i.e., they employ the **Minimum Description Length** (**MDL**) principle to determine whether a data instance is anomalous; in brief, the number of bits required to encode it using compression is used as anomaly score. They utilise the Krimp algorithm [195] as the compressor, which is trained on exclusively normal samples to capture normal behaviours. On this basis, they provide explanations by showing which patterns were recognised in the anomalies, as well as by checking whether small changes can turn the anomalies into normal instances. If it can, the anomalies are observation errors rather than real anomalies. Instead of using patterns to represent what is normal, Park & Kim [163] put forward a model that is an ensemble of **Region-Partition** (**RP**) trees. Each RP tree is trained only on normal data and thus represents a partition of the normal data region. Hence, if a test instance can arrive at a leaf node of any individual RP tree, it is considered normal. Otherwise, it is an anomaly. Considering that the size of each RP tree is small, one can easily find the hypercube in which the anomaly is stuck. On this basis, they take the intersection of hypercubes of all RP trees as an explanation for the anomaly.

While the above-mentioned models focus on static tabular data, Dickens et al. [59] propose **Mondrian Pólya Forest** (**MPF**) to detect and explain anomalies on large data streams by combining random trees with non-parametric density estimation approaches. Specifically, the Mondrian Process [179] is a family of hierarchical binary partitions of data and the Pólya Tree [137] is a non-parametric approach that can estimate the density function of binary partitions. They combine the Pólya Tree structure with a truncated Mondrian Process to deal with static data, and combine the Pólya Tree structure with a Mondrian Tree to handle data streams. In this way, they construct a forest, namely, MPF, for density estimation and anomaly detection. As a result, an instance with relatively low estimated density is considered anomalous. Furthermore, with the good properties of MPF, the resulting anomaly scores are probabilistic and, therefore, interpretable.

5.3.2 Models for Sequential Data. One line of research addresses the problem of detecting and providing intrinsic explanations in time series data, which is a type of sequential data. Techniques such as sparse learning and time series decomposition have been leveraged to develop intrinsically interpretable anomaly detection models for time series data. For instance, Li et al. [116] apply **deep Generative Models** (**DGMs**) to detect anomalies in multivariate time series data. Specifically, they first set up a stacking **Variational AutoEncoder** (**VAE**)-based model that constructs a single-channel block-wise reconstruction, followed by stacking it multiple times using a weight sharing technique to handle channel-level similarities. Second, they utilise a graph learning

module to learn a sparse adjacency matrix for every channel, attempting to extract structure information for achieving an explainable reconstruction process. As a result, a test instance with a large reconstruction error is considered anomalous. Meanwhile, Cheng et al. [46] exploit time series decomposition techniques from a multi-scale perspective to identify spatiotemporal abnormalities of human activity. They first employ the **Seasonal-Trend decomposition with the Loess (STL)** method to decompose the time series to look for anomalies. As a result, the periodic as well as trend components of observed data are eliminated during the time series decomposition, and the remaining components capture anomalous activity signatures. By examining the residual elements of the time series for each spatial unit, they are able to identify spatiotemporal anomalies in human activity. Finally, they devise a rule to match anomalies identified at different scales in accordance with their spatiotemporal influence ranges and explain anomalies based on their multi-scale characteristics.

Another line of research focuses on the task of identifying and offering intrinsic explanations in network traffic data, which can be seen as an instance of other sequential data. For example, Grov et al. [75] first group the network traffic data into different sessions, followed by learning two behavioural models, namely, a **Markov Chain (MC)** model and a **Finite State Automata (FSA)** model, on normal sessions. Next, for an incoming session, they compute a similarity measure with respect to these two models, resulting in an anomaly score. More concretely, the MC model returns two probabilities as the anomaly score and the FSA model returns a distance as the anomaly score. Meanwhile, Mulinka et al. [147] present HUMAN, a hierarchical clustering-based method. Specifically, they consider three different clustering methods to group data instances into clusters, assuming that the normal behaviour is represented by the largest cluster. Therefore, data instances residing in the smallest clusters are considered anomalous. Next, they explain the detected anomalies by displaying the clustering results, including the number of clusters, the size of each cluster, and a textual explanation of each cluster. Moreover, Marino et al. [132] propose the **Network Transformer model (NeT)**. First, the network data is represented by a graph where its nodes represent network device IP addresses and the edges describe data packets delivered between different devices. Second, NeT extracts hierarchical features from the graph for anomaly detection. Third, based on these features, NeT employs existing anomaly detection models—such as LOF [26], OCSVM [188], and AutoEncoders—to identify anomalies at various granularity levels. Moreover, NeT provides explanations based on the graph structure, offering a subset of hierarchical features that allow users to pinpoint the devices affected by the anomalies and the connections that caused the anomalies.

5.3.3 Discussion. Due to the lack of a unified definition of anomaly and the diversity of data types, a wide range of in-model XAD techniques have been explored in an unsupervised setting. More concretely, techniques such as Probabilistic Models (e.g., Mondrian Pólya Forest and other distribution or density estimation-based approaches), Data Cube structure, Incremental Clustering, MDL-based Pattern Compression, RP trees are harnessed to detect and explain anomalies in tabular data. Meanwhile, techniques such as MC, FSA, Hierarchical Clustering, Sparse Learning in VAE, Time Series Decomposition, and Hierarchical Features in Graph Representation are adapted to identify and interpret anomalies in sequential data.

5.4 Summary

As shown in Table 2, the in-model techniques presented in this section are model-specific. Moreover, the majority of these methods provide feature-based explanations (including pattern-based explanations), with the exception of References [46, 75, 83], which offer sample-based explanations, and References [41, 53] generate explanations from both perspectives.

Table 2. Summary of In-model XAD Techniques

Ref	Spec	Pers	Tech	Data	Loc	Pros	Cons
[83]	S	P	Frequent pattern mining	Static TC	G	Performs frequent pattern mining and outlier discovery simultaneously	Only applicable to categorical data
[168]	S	S	Rule-based model to produce probability	Static TN	G	Fast	Low accuracy
[230]	S	P	Frequent pattern mining	Static video	G	Able to detect point anomalies, contextual anomalies, and collective anomalies	Needs labeled normal activity data
[213]	S	P	Frequent pattern mining	Streaming dynamic graph	G	Able to handle dynamic graphs	Computationally expensive
[138]	S	P	Satisfiability Modulo Theories based rule models	Static ES	G	Integrates contexts to detect anomalies	Computational expensive; Explanations may be too long to understand
[24]	S	P	Association rule mining + Visualisation	Event logs	L	Able to handle process change and flexible executions; Evaluating the quality of explanations	Assumes the availability of domains, process models, and anomalies in evaluation
[108]	S	F	Composition-based decision tree	Static UTS	G	No manual tuning of hyper-parameters; Evaluating the quality of explanations	Needs labeled data
[90]	S	F	One Class Decision Tree	Static tabular	G	Produces compact and readable results	Parameterization of the KDE is difficult
[4]	S	F	Decision tree based AE	Static TC	G	Intrinsically interpretable AE	Not suitable for dataset with many features; Only applicable to categorical data
[41]	S	F & S	LOESS regression	Static TM	G	Able to handle heterogeneous features; Evaluating the quality of explanations	Sensitivity to important parameters not discussed
[29]	S	F	Change Point Model + Regression Trees	static tabular	G	More insights from multiple models	No explanations for ANN
[111]	S	F	Logistic regression	Streaming MTS	G	Able to predict short-term anomalies	Not able to predict long-term anomalies
[21]	S	F	Gaussian Process	Streaming TN	G	Able to handle unreliable, noisy, or partially missing data	High computation cost; Hard to do model selection; Poor at handling discontinuities

(Continued)

Table 2. Continued

Ref	Spec	Pers	Tech	Data	Loc	Pros	Cons
[199]	S	F	Dynamic Bayesian Networks	Streaming MTS	G	Able to incorporate domain knowledge	Computationally expensive
[226]	S	F	Linear components	Static MTS	G	End-to-end training	Weak interpretability due to deep models
[38]	S	F	Generalized Additive Models	Static TM	G	Able to incorporate a small amount of labeled data	Relies on assumptions about the data generating mechanism
[149]	S	F	Subspace AD	Static TN	L	More scalable than other density-based methods	Weak interpretability due to ensemble
[99]	S	F	Subspace AD	Static TN	L	Adjustable to any AD	Weak interpretability due to ensemble
[109]	S	F	PCA + Subspace AD	Static TN	L	Works with arbitrarily oriented subspaces	Weak interpretability by using error vectors of PCA; Does not work well with high-dimensional data
[22]	S	F	Sparse PCA + Subspace AD	Static TN	G	Fast	Non-trivial parameters setting by end-users
[53]	S	F & S	Subspace AD	Static TN	L	Explores interconnections between neighboring members	High computational cost; Assumes linear separability of anomalies with their neighbours
[100]	S	F	Subspace AD	Static tabular	L	Applicable to any AD model	Computationally expensive
[52]	S	F	Subspace AD	Static TN	L	Ensures quality of explanation via keeping local geometry	Not scalable to high-dimensional data
[186]	S	F	Subspace AD	Static TM	L	Applicable to mixed data; Parallelizable; Applicable to high-dimensional data	Difficult to initialize clusters for GMMs in high-dimensional subspaces
[169]	S	F	Feature projections	Streaming and static TN	L	Fast; Adapted to concept drift; Able to handle missing variables	Only considers one-dimensional projections; Weak interpretability due to ensemble
[159]	S	F	Wrapper feature selection	Static TC	L & G	Works well with noisy features	Only applicable to categorical data
[157]	S	F	Embedded feature selection	Static TC	L	Works with high-dimensional data	Lacks interpretability due to the use of an ensemble approach
[178]	S	F	SHAP based feature selection	Static MTS/ES	G	Does not need labelled data	Only applicable to AE-based model; High time complexity with kernel SHAP

(Continued)

Table 2. Continued

Ref	Spec	Pers	Tech	Data	Loc	Pros	Cons
[5]	S	F	Statistical distribution	Static tabular	L & G	Sound statistical foundation	Assumes probabilistic distribution of data
[66]	S	F	Data cube	Static tabular	L	Considers anomalies in multiple contexts	Computationally expensive
[139]	S	F	Incremental clustering	Static TM	G	Fast	Non-trivial setting of the threshold parameter
[200]	S	P	MDL-based pattern compression	Static TC	L	Provides detailed inspection and characterisation of decisions	Only applicable to categorical data
[75]	S	P	Markov Chain; Finite State Automata	Network traffic data	G	Robust to new unseen anomalies	Trains on normal data
[147]	S	S	Clustering	Network traffic data	G	Able to integrate domain knowledge	Non-trivial setting of parameters
[59]	S	F	Bayesian nonparametric based density estimation	Static and steaming TM	G	Able to handle streaming data	Density estimation does not work well in high-dimensional data
[163]	S	F	RP Tree	Static tabular	G	Automatically determines the threshold on anomaly scores	Weak interpretability due to ensemble
[116]	S	F	Sparse learning + Reconstruction error	Static MTS	G	Considers single-channel anomalies and structural multi-channel anomalies	Only considers linear correlation between channels; Weak interpretability due to VAE
[46]	S	S	Time series decomposition	Spatial-temporal data	G	Considers multi-scales to gain more insights into anomalies	Does not work well with data missing and deficiency
[132]	S	F	Hierarchical features in graph	Static graph	G	Self-supervised training that does not need labeled data	Weak interpretability due to deep models

Spec indicates whether a method is model-agnostic (A) or model-specific (S). Note that all in-model techniques are basically model-specific. *Pers* specifies whether a method is feature-based (F), sample-based (S) or pattern-based (P). *Tech* indicates the techniques used in each method. *Data* indicates the data type for which the method is applicable (TN: Tabular Numeric; TC: Tabular Categorical; TM: Tabular Mixed; UTS: Univariate Time Series; MTS: Multivariate Time Series; ES: Event Sequence). *Loc* shows whether a method provides a local explanation (L) or global explanation (G). Moreover, *Pros* and *Cons* describe the advantages and disadvantages of each method, respectively.

According to their main characteristics, we subdivide them into three high-level groups, i.e., *transparent models in supervised learning, feature subset-based models*, and *miscellaneous models in unsupervised learning*, for which we make the following observations.

First, *for transparent models in supervised learning*, we find that most methods can provide global explanations as their entire logic can be easily understood by humans due to their transparent nature. However, *decision tree-based models* are hard to explain when the tree is too deep or too wide. To alleviate this problem, feature selection can be leveraged. Meanwhile, an ensemble of decision trees can avoid overfitting of the data, thereby improving the generalization performance. However, an ensemble of trees is not human-understandable. Concerning *rule-based models*, a large set of rules or a long rule is difficult to explain. Therefore, a human-reasonable size is required to maintain interpretability. Furthermore, an inherent problem of using *linear models* for interpretation is that when the model does not fit the training data optimally, it may optimize errors using spurious features that may be difficult to interpret for humans [76]. Overall, for these transparent models to retain their interpretability characteristics, they must be limited in size and the features used should be understandable to the end-users [20].

Second, for *feature subset-based models*, most methods can only provide local interpretations, that is, only a certain output can be interpreted at a time. Besides, some subspace anomaly detection methods do not provide explicit explanations due to the use of ensemble techniques to aggregate anomaly scores in multiple subspaces. However, the contribution of each feature is relatively easy to obtain. Moreover, most subspace anomaly detection methods were originally designed to tackle the issue of *curse of dimensionality* when detecting anomalies in high-dimensional data [231]. Therefore, promoting interpretability is not their main concern. Notably, we observe that there is extremely limited research on the wrapper or embedded feature selection for anomaly detection.

Third, for *miscellaneous models in unsupervised learning*, these methods are quite different from each other, as they are specifically designed for the anomaly detection and not explored in a supervised setting. Given the lack of a unified definition of anomalies and the diversity of data types, it is not surprising that these approaches are very diverse. Importantly, we note that most of these methods can provide global explanations, in the sense that the logic of the whole model is human-understandable, or some important properties of the model can be leveraged to interpret all decisions.

6 LITERATURE REVIEW ON SHALLOW POST-MODEL TECHNIQUES

Post-model methods inspect an anomaly detection model after the detection process is completed, or just inspect a given anomaly without being given an anomaly detection model. In other words, these techniques do not interfere with the anomaly detection process, operating only on the basis of correlating the input of the anomaly detection model (if any) with its output. Due to the proliferation of techniques in this category, in this section, we only introduce techniques designed for non-deep learning processes, and we call them *shallow post-model anomaly explanation* techniques.

Most *shallow post-model anomaly explanation* methods intend to find a subspace or a set of subspaces in which the given anomaly differs the most from other instances, and we call these methods *subspace-based methods. Surrogate methods*, on the other hand, resort to identify another model to explain the anomaly detection model or just the given anomalies. Specifically, a surrogate model can be a transparent model, such as a set of rules or a decision tree, or an opaque model, such as XGBoost or SVM. Importantly, if the surrogate model is an opaque model, it should be easy to interpret by using XAI techniques such as SHAP. Meanwhile, *miscellaneous methods* such as comparing patterns to find differences, leveraging SHAP techniques to measure feature importance, and visualisation also play an important role in shallow anomaly explanation.

6.1 Subspace-based Methods

Given an anomaly or a group of anomalies, *subspace-based methods* aim at finding a subspace or a set of subspaces in which the anomaly deviates the most. Different from the intrinsically explainable *subspace anomaly detection* methods that were introduced in Section 5.2, the subspace based methods investigated hereinafter do not assume the availability of anomaly detection models.

First, different explanation methods usually have different definitions for anomaly, dubbed *Explanation-Definition* in this survey, leading to different measures of abnormality. For instance, Knorr & Ng [106] define strongest and weak outliers, based on which they use so-called intensional knowledge to explain anomalies. For each anomaly identified in the original feature space, they report the minimal subspaces in which it behaves anomalously. Particularly, in their proposed algorithm CELL, for each instance, they utilize the number of neighbours in its local neighbourhood of a given radius as the anomaly score. However, this anomaly measure can be replaced by other anomaly measures such as density, depth, and so on. As far as we know, their work is seminal in anomaly interpretation. Additionally, Zhang et al. [228] propose HOS-Miner to identify outlying subspaces for a given anomaly. Specifically, they define the sum of distances between the anomaly and its k-nearest neighbours as its anomaly score in each subspace, thereby returning the outlying subspace with the lowest dimensionality as an explanation. Similarly, Micenková et al. [141] propose an anomaly explanation technique that works on tabular dataset with numeric features. Specifically, given an anomaly, they look for a subspace in which this instance is well separable from the rest. To achieve this, they first generate a classification dataset consisting of a comparable number of normal and abnormal instances. Second, they apply an existing feature selection method to find a subset of features that are relevant for the classification, namely, separation. Particularly, they define a measure of separability based on the probability density function of a normal distribution as the anomaly measure. Finally, the obtained subspace serves as an explanation for the anomaly.

Second, some researchers attempt to provide explanations from multiple perspectives or contexts. For instance, Angiulli et al. [10] propose a method that is capable of providing explanations from both global and local perspectives. On the one hand, for an anomaly, they measure its abnormality with reference to all data instances in different subspaces, delivering the subspace with the highest abnormality as a global explanation. On the other hand, for an anomaly, they first select a subset of features and the corresponding values to define a reference group, and then compute its abnormality with respect to this reference group in different subspaces. Accordingly, the reference group and the corresponding subspace with the highest abnormality constitute a local explanation. Note that the definitions of global explanation and local explanation used in Reference [10] differ from those we defined in this survey. Particularly, the *abnormality* is defined in terms of the frequency of the anomalous instance and the frequencies of referencing instances. Furthermore, Müller et al. [148] present OutRules, which generates multiple explanations for an anomaly in different contexts. Specifically, OutRules explains an anomaly by generating rules that describe the deviation of this instance in contrast to its context. On the one hand, a subset of features is used to define a context consisting of highly clustered instances. On the other hand, they attempt to find an extended subset of features in which one of these instances is significantly deviating. Concretely, the anomaly measure used in their framework can be instantiated by the underlying anomaly score of any anomaly detection model such as LOF. Similarly, Angiulli et al. [9] devise a method that consists of two steps. Given an anomaly and a dataset, for each feature, they first determine the interval that includes the anomaly and the associated condition, resulting in a set of conditions on all features. Second, they employ an Apriori-like strategy to search for *explanation-property pairs* for the anomaly. More concretely, an *explanation* is a set of conditions used to define a context where the anomaly is located. Meanwhile, a *property* is an additional

condition posed on a feature other than the features that are used to define the context, aiming to distinguish the anomaly from the context. In particular, they define an anomaly measure based on the probability density function of each feature. Consequently, the *explanation-property* pair and the corresponding anomaly score constitute an explanation for the anomaly.

Third, other techniques such as visualisation can be leveraged to further improve the explainability of subspace-based methods. For instance, given a dataset consisting of real-valued features and a list of anomalies, Gupta et al. [79] propose the so-called LOOKOUT approach to explain these anomalies. Specifically, they attempt to find a limited number of 2-dimensional subspaces in which the given anomaly deviates the most from the rest. Particularly, if the anomaly is detected by an anomaly detection model, they utilise the underlying anomaly measure to obtain the anomaly score. Otherwise, they employ any other off-the-shelf model such as LOF to obtain the anomaly score. Moreover, they visualise these subspaces using 2-dimensional scatter plots, known as focus plots in their article, and then present these *focus plots* to the end-users as explanations. Also, the above-mentioned approach OutRules [148] utilises parallel coordinates plots to visualise the anomalies.

Fourth, some methods have been explored to explain anomalies in a group. For instance, Angiulli et al. [11] extend their previous work [10] to explain a group of anomalies. Furthermore, Macha & Akoglu [129] propose x-PACS to explain anomalies in a group in three steps. They first utilise a subspace clustering algorithm to identify clusters that the anomalies form. Second, for each subspace cluster of anomalies, they leverage an axis-aligned hyper-ellipsoid to represent it. Third, they employ the MDL criterion to identify a set of hyper-ellipsoids that are compact, non-redundant, and pure. Particularly, x-PACS does not require a measure of outlyingness since they address the anomaly explanation problem from a subspace clustering perspective.

Fifth, with the emergence of many anomaly explanation methods, some researchers endeavour to formally define the anomaly explanation problem or propose a taxonomy of existing methods. Concretely, Kuo & Davidson [110] formally define the outlier description (namely, anomaly explanation) problem and propose a **Constraint Programming (CP)**-based framework to encode the problem. Particularly, they utilise a neighbourhood density based criterion to measure the outlyingness of an instance in each subspace. On this basis, they introduce a CP framework to learn the optimal subspace to explain an anomaly. Their framework and variants can explain an anomaly in a single subspace, multiple subspaces, or by introducing the human in the loop. Meanwhile, Vinh et al. [216] for the first time divide outlying aspects mining techniques into two categories, viz. feature selection-based approaches and score-and-search approaches, and additionally make two important contributions. First, they formalize the concept of dimensionality-unbiasedness for anomaly scoring functions. They show that some widely used anomaly scoring functions such as distanced-based and density-based scoring measurements violate this important property. Moreover, they put forward two dimensionality-unbiased anomaly scoring functions, namely, Z-score and isolation path score, to measure the outlyingness of an instance in different subspaces. Second, they propose a beam search framework to overcome the limitation of exhaustive search in exponentially large space. Consequently, for instance, they return the subspace with the highest dimensionality-unbiased anomaly score as an explanation. However, Samariya et al. [184] point out an issue of using Z-score normalisation of density to rank subspaces for outlying aspects mining. Particularly, Z-score normalisation has a bias towards data distribution (with high variance subspaces) although it is dimensionality-unbiased. To tackle this issue, they propose another anomaly scoring function called SiNNE for outlying aspects mining. Specifically, SiNNE consists of an ensemble of models where each model is developed based on a subset of data. However, due to the use of ensemble techniques, SiNNE cannot provide explanations for identified anomalies.

Finally, another line of research attempts to explain any given instance that may be anomalous or normal. For example, given a data instance, Duan et al. [64] develop a method to find its minimal outlying subspace, i.e., the subspace with the lowest dimensionality where the query instance is most deviating from others. To achieve this, they first assume that the instances are generated from a probability distribution that is often unknown. Second, they utilise kernel density estimation techniques to approximate the probability density of an instance in each subspace, deriving its anomaly score. Moreover, they employ heuristic techniques to prune the set of possible subspaces that need to be explored.

Discussion: Most subspace-based methods reviewed above suffer from two limitations: high computational costs and poor explanation fidelity. First, most methods intend to find a minimal subspace in which the anomaly deviates the most. To find such a subspace, they usually need to go through the exponentially large search space. Although some pruning techniques such as beam search are leveraged to mitigate this problem, optimality is no longer guaranteed. Second, almost all methods in this category have their own definitions of anomaly (namely, *Explanation-Definition*) when trying to interpret anomalous instances. Importantly, this *Explanation-Definition* is very likely to differ from the *Detection-Definition*, leading to a poor fidelity of the explanation. In other words, if the anomaly detection model is available, the provided explanation may not reflect its actual decision-making process.

6.2 Surrogate Methods

A line of research in shallow post-model XAD techniques is to utilise surrogate models to describe given anomalies or anomaly detection models. In general, the surrogate model can be a transparent model or an opaque model. If a transparent model such as a set of rules or a decision tree is employed to depict the anomaly, the result is directly understandable. However, if an opaque model such as XGboost or SVM is leveraged to approximate the outputs, additional XAI techniques such as SHAP or LIME are required to make the results understandable.

First of all, model-agnostic rule learners are often leveraged to extract a set of rules or patterns as the surrogate model, aiming to explain anomalies. For instance, Ertoz et al. [67] present the MINDS framework for network intrusion detection and explain anomalies by association rules. Specifically, they first utilise an off-the-shelf anomaly detection model such as LOF [26] to detect anomalous network connections. Second, they develop a Discriminating Association Pattern Generator to extract patterns that exclusively characterise normal instances or anomalous instances, respectively. The extracted patterns are human-comprehensible and thus serve as explanations for anomalies. Moreover, they attempt to assign anomalies to different groups based on the extracted patterns. Alternatively, Davidson [55] capitalizes on mixture modelling (a.k.a. model-based clustering) to perform anomaly detection. Specifically, for each data instance, this method calculates the likelihood of this instance belonging to each cluster. If the maximum obtained likelihood is less than a predefined threshold, the instance does not belong to any cluster and is therefore considered anomalous. Moreover, they describe a visualization approach to show normal and abnormal instances based on scatter plots, enabling end-users to quickly understand why an instance is considered anomalous. More importantly, they try to extract rules (in Conjunctive Normal Form) to describe each obtained cluster and those anomalies. By comparing these rules, one can easily understand why an anomaly is anomalous.

Meanwhile, some model-specific rule learners are proposed to extract a set of rules or patterns as the surrogate model, aiming to explain anomalies. For example, Das et al. [54] define a novel formalism, known as *compact description*, to extract rules to describe discovered anomalies. However, this method can only be applied to tree-based ensembles, and the extracted rules are represented using **Disjunctive Normal Form** (**DNF**). Moreover, they also develop an active

anomaly explanation algorithm for generic ensembles, dubbed GLAD. For each detected anomaly, GLAD first identifies the base-learners (namely, ensemble members) that contribute the most to the decision. Second, GLAD applies a model-agnostic explanation method such as LIME on these important base-learners, respectively, to generate explanations for the anomaly. Besides, Barbado et al. [18] apply several rule extraction techniques to OCSVM models [188] for anomaly explanation, and evaluate the quality of generated explanations accordingly. Specifically, these techniques first apply OCSVM to obtain normal and abnormal instances. Second, they use an off-the-shelf clustering method such as K-Prototypes [93] to iteratively divide the non-anomalous instances into different regions until no anomalies are contained in the generated regions. Third, since these regions are in the form of hypercubes, they can directly extract rules from the vertices of these hypercubes to explain why an instance is non-anomalous. More importantly, they define several metrics including comprehensibility, representativeness, stability and diversity to evaluate the quality of explanations. Besides, their methods can provide both local and global explanations. Although it is claimed that the whole process can be adapted to any anomaly detection model, this is not shown.

Second, some rule learners are explored to extract decision trees as the surrogate model, aiming at explaining anomalies. For instance, Xu et al. [222] propose an approach to detect and explain system problems by mining console logs. Concretely, they leverage a **Principle Component Analysis** (**PCA**)-based anomaly detection method [65] to identify anomalies, followed by explaining the results using decision trees to mimic the decision-making process. However, Bin et al. [22] show that using decision trees to explain the PCA model can be misleading, thereby failing to reveal the true decision-making process. Besides, Pevný & Kopp [170] introduce a method called Explainer to explain anomalies using DNF. Specifically, given an anomaly, Explainer first trains a collection of trees known as **Sapling Random Forests** (**SRF**). Each tree in an SRF is a binary decision tree with the aim of separating the anomaly from other normal instances. Second, once a tree is built, they utilise DNF to represent the path from the root node to the node that contains only the anomaly. Third, they aggregate the DNFs from all trees to a compact DNF to interpret the anomaly. Furthermore, Kopp et al. [107] extend Explainer by introducing two k-means-based clustering methods to interpret anomalies when these anomalous instances form natural micro-clusters.

Third, some researchers attempt to utilise well-studied opaque models as surrogate models, and then leverage additional explanation techniques such as SHAP to explain surrogate models. For example, to monitor the average fuel consumption of fleet vehicles, Barbado and Corcho [17] set up an unsupervised anomaly detection process capable of explaining decisions through feature importance. First, they leverage a threshold-based model to detect anomalies. Second, they utilize two types of surrogate models to explain anomalies, including black-box anomaly detection models with a post-hoc local explanation (XGBoost [44] and LightGBM [98] with LIME or SHAP), and transparent anomaly detection models (ElasticNet [232] and EBM [153]). Third, they evaluate these surrogate models in terms of predictive power and explanatory power. Particularly, their explanation method can also integrate domain knowledge given by business rules or counterfactual recommendations. Furthermore, Kiefer & Pesch [102] put forward an ensemble-based anomaly detection model combined with model-agnostic explanation technique to identify and interpret anomalies in financial auditing data. Specifically, they construct an ensemble architecture to incorporate a wide range of unsupervised anomaly detection models, attempting to identify different types of anomalies. To interpret anomalies, they propose a four-step method: synthetic oversampling of anomalies, supervised model approximation (using SVM or XGBoost), LIME-based local explanation, and explanation post-processing (visualisation or natural language description).

Discussion: As can be seen, most methods in this category leverage rule learners to extract a set of rules or patterns to describe anomalies. Importantly, the resulting rules are often represented

using a DNF or Conjunctive Normal Form. Consequently, it might be relatively easy to evaluate the quality of the resulting explanations, but this depends on many factors.

6.3 Miscellaneous Methods

In addition to subspace-based methods and surrogate methods, *miscellaneous methods* such as visualisation and Shapley values are often used to obtain feature importance as explanations. Moreover, pattern comparison is commonly explored in sequential data to interpret anomalies in a post-hoc manner. Due to the diversity of these methods, we again organize them according to the type of data to which they are applicable.

6.3.1 Models for Tabular Data. A wide range of methods has been proposed to explain anomalies in tabular data by showing feature contribution or selecting a subset of features. Importantly, some of these methods are model-agnostic. For instance, Liu et al. [123] introduce the COIN framework that consists of four main steps. For each anomaly, they first find its neighbours based on a distance measure such as Euclidean distance. Second, they leverage existing clustering algorithms to subdivide the anomaly and its neighbours into multiple disjoint clusters. Third, they apply a strategy such as synthetic sampling to expand the size of the anomaly cluster where the anomalous instance is located. Fourth, they train a simple classifier to separate these clusters, deriving an anomaly score and feature contributions from the parameters of the classifier. Moreover, COIN can also incorporate prior knowledge into the explanation process. Importantly, Siddiqui et al. [193] present **Sequential Feature Explanations (SFEs)** to explain detected statistical outliers. Concretely, given an anomaly identified by any density-based detector, SFEs sequentially present a feature to the analyst until the analyst can confidently identify this anomaly. As a result, these features used to identify the anomaly constitute the corresponding explanations. Particularly, Siddiqui et al. [194] apply Isolation Forest [122] to detect cyber attacks and leverage SFEs to generate explanations.

Shapley value-based methods are often leveraged to obtain feature importance as explanations. For example, Park et al. [164] employ SHAP [128] to explain anomalies by showing feature contributions. Concretely, they set up an anomaly detection model by using random forest and employ the SHAP approach to explore the relationship between model results and input variables to generate explanations. Similarly, Kim et al. [103] utilize Isolation Forest on sensor stream data of marine engines to keep track of unusual engine conditions. Moreover, they leverage SHAP to identify which sensor is in charge of each abnormal data event and to quantify its contribution to the observed anomaly. Besides, using reconstruction errors as a measure to detect and explain anomalies is a common practice in unsupervised anomaly detection. However, Takeishi [209] argues that by simply looking at the reconstruction error of each feature, one may fail to find the true cause of the anomaly. This is because a large reconstruction error in one feature may stem from another feature. To mitigate this problem, a method is introduced to compute the Shapley values [205] of reconstruction errors for PCA-based anomaly detection method. The numerical examples show that the Shapley values are superior to reconstruction errors for explaining an anomaly.

Model-specific techniques have also been developed to explain anomalies in tabular data, especially for Isolation Forest [122], a state-of-the-art anomaly detection model. For example, Kartha et al. [95] develop a method to interpret anomalies identified by Isolation Forest by exploring the internal structure of an Isolation Forest to generate a feature importance vector, indicating the contribution of each feature to the anomaly score. Similarly, Carletti et al. [32] propose DIFFI to obtain feature importance scores for explaining Isolation Forest. Specifically, DIFFI provides a global feature importance score for each feature, indicating how that feature affects the overall decisions of Isolation Forest on the training data. Meanwhile, they present a local version of DIFFI,

named local-DIFFI, to provide a local feature importance score for each feature, describing how each feature participates in making individual decisions on the test data. Importantly, they develop a feature selection method for unsupervised anomaly detection problems on this basis. Particularly, Carletti et al. [31] apply DIFFI on real-world semiconductor manufacturing data to demonstrate its effectiveness.

6.3.2 Models for Sequential Data. A common strategy to explain anomalies in sequential data is to contrast the observed pattern with its expected pattern or normal patterns. For instance, Babenko & Pastore [15] leverage LFA [131] to detect anomalies in system logs. On this basis, they present the so-called **Automata Violation Analyzer** (**AVA**) to automatically explain anomalies detected by LFA. Specifically, AVA provides *basic explanations* by comparing the expected event sequence with the observed event sequence, generating relatively simple explanations for the anomalous events. Furthermore, they combine these *basic explanations* to obtain *composite explanations*. Finally, they order these basic and composite explanations by their likelihood of explaining differences between the expected event sequences and the observed event sequences. In addition, Leue & Befrouei [113] design a method to explain counterexamples that are symptoms of deadlocks in concurrent systems. These counterexamples can be considered as anomalies and the authors use sequential pattern mining to produce explanations for these anomalies. Specifically, they extract fixed-length common substrings from anomalous sequences and contrast them with normal sequences to explain the occurrence of anomalies.

Meanwhile, leveraging visualization to explain anomalies in sequential data is also a common practice. For example, Rieck & Laskov [176] propose a technique for explaining intrusion detection results. Specifically, they present two methods for anomaly detection, viz. global anomaly detection and local anomaly detection. For each payload, the global anomaly detection method computes its distance to the centre of all payloads as its anomaly score; In contrast, the local anomaly detection method computes the average distance to its k-nearest neighbours as its anomaly score. To explain an anomaly, they present a visualization tool to show the feature differences between the anomalous payload and the normal payloads. A large difference in a feature means that the corresponding feature value is anomalous. Furthermore, they also highlight the network content corresponding to the feature value in the original payload. Besides, Alizadeh et al. [7] implement an **AutoRegressive Integrated Moving Average** (**ARIMA**)-based model together with a **Virtual Reality** (**VR**) tool to detect and interpret abnormal vehicle operating states. Specifically, modern vehicles are often equipped with multiple sensors to collect data used to monitor their operating status. To detect anomalies in such multi-channel time series data, they develop an ARIMA model for each channel (i.e., for each individual univariate time series). Hence, a large difference between the actual value and the predicted value indicates an anomaly. Importantly, they build a VR tool to visualize residuals from ARIMA models, aiming to better understand anomalies. Moreover, Markou et al. [133] create a tool for exploiting internet data to find abnormalities in transportation networks and connecting them to unique events. First, the baseline normality corresponding to GPS data for taxi journeys in New York City is trained based on historical mobility data. Next, they scan various days to look for days where demand deviates greatly from normality in order to identify abnormalities. To investigate the severity of daily traffic abnormalities, they consider the Z-score formula of kernel density values. The current traffic situation is considered abnormal if the Z-score value exceeds a given threshold. To explain the anomaly, they diagram the time and place of the anomaly and utilize that information to look for nearby unusual events using Google Searches.

Discussion: This subsection reviewed a wide range of shallow XAD techniques that are explored to interpret anomalies in a post-hoc manner. In contrast to *subspace-based methods* and *surrogate methods*, the techniques investigated here vary by data type. For instance, Pattern Comparison and Visualisation are commonly utilized to explain anomaly in sequential data. Methods in this group

usually do not have an explicit *Explanation-Definition* of anomalies since they directly illustrate the anomalies by comparing patterns or using visualisation tools. Meanwhile, feature importance that is obtained by using SHAP techniques, separation or isolation-based measure, plays an important role in explaining anomalies in tabular data. Using Shapley values techniques such as SHAP to obtain feature importance usually does not require a definition of anomaly. Moreover, model-specific techniques such as References [7, 32, 95] generally have consistent *Explanation-Definition* and *Detection-Definition* of anomaly as they explore the internal structure of an anomaly detection model to generate explanations. On the contrary, model-agnostic techniques such as Reference [123] usually have an *Explanation-Definition* that may differ from the *Detection-Definition*.

6.4 Summary

While Table 3 provides the full characterization for all methods discussed in this section based on the six criteria of our taxonomy, we here make some general observations on shallow post-model XAD techniques.

First, nearly all *subspace-based methods* and *surrogate methods* are model-agnostic in the sense that they are applicable to any anomaly detection model or given anomalies. In other words, these methods do not explore the internal structure of an anomaly detection model and, therefore, cannot have a full grasp of the underlying decision-making mechanism, rendering the provided explanations less useful and potentially resulting in weak interpretability. In contrast, *miscellaneous methods* are mainly model-specific, as they explore the internal structure of anomaly detection models to generate feature importance as explanations. As a result, the obtained explanations are more reliable and actionable.

Second, all these methods provide feature-based explanations (including pattern-based explanations) except that Davidson [55] provides sample-based explanations. We consider the lack of sample-based explanation methods to be a gap in the literature that might be of interest for future research.

Third, most shallow post-model XAD techniques, especially *subspace-based methods* and *miscellaneous methods*, can only provide local explanations. In other words, they can merely interpret an individual anomaly at a time. As a result, the explanation may be highly sensitive to noise or biased since the employed XAD methods are short of a holistic perspective on the decision-making process and logic.

7 LITERATURE REVIEW ON DEEP POST-MODEL TECHNIQUES

Deep learning, based on artificial neural networks, has become prevalent in anomaly detection due to its capability to learn expressive feature representations and/or anomaly scores for complex data such as text, audio, images, videos and graph [161]. A wealth of deep anomaly detection methods, including those based on AE, **Long Short-Term Memory** (**LSTM**), CNN, **Generative Adversarial Network** (**GAN**) and other neural networks, have been proposed and have been shown to be more accurate than traditional methods when it comes to detecting anomalies in complex data. However, although deep anomaly detection methods tend to have high detection accuracy, they are often criticized for their poor interpretability. For this reason, some studies have attempted to leverage post-hoc XAI techniques to improve the interpretability of corresponding neural networks. Importantly, which XAI techniques are available may vary depending on the specific neural network used. For instance, AE-based models typically employ reconstruction errors to explain anomalies, while LSTM-based models generally leverage SHAP techniques to interpret anomalies. Therefore, we will present the review results according to the type of neural network used to perform anomaly detection, which is correlated with the data type that can be used (e.g., CNNs for images and RNNs for sequential data).

Table 3. Summary of Surveyed Shallow Post-model Techniques

Ref	Spec	Pers	Tech	Data	Loc	Pros	Cons
[106]	A	F	Others (Subspace)	Static tabular	L	Applicable to any AD; Good explanation fidelity (exception in post-hoc)	High computational cost
[228]	A	F	Others (Subspace)	Static tabular	L	Applicable to any AD	Poor explanation fidelity; Compares scores in subspaces with different dimensionalities
[10]	A	F	Others (Subspace)	Static TC	L & G	Applicable to any AD	Poor explanation fidelity; Only applicable to categorical data
[148]	A	F	Others (Subspace + Context)	Static TN	L	Applicable to any AD; Considers multiple contexts	Poor explanation fidelity
[141]	A	F	Others (Subspace)	Static TN	L	Applicable to any AD	Poor explanation fidelity
[64]	A	F	Others (Subspace)	Static TN	L	Applicable to any AD	Poor explanation fidelity; KDE does not work with high-dimensional data
[110]	A	F	Others (Subspace)	Static TN/TC	L	Applicable to any AD; Enables human in the loop	Poor explanation fidelity; Poor scalability
[216]	A	F	Others (Subspace)	Static TN	L	Applicable to any AD; Fast search; Dimensionality unbiased	Poor scalability in high-dimensional data; Poor explanation fidelity
[79]	A	F	Visualisation	Static TN	L	Applicable to any AD; Easy to understand by non-experts	Only considers 2D subspace
[9]	A	F	Others (Subspace + Contextual Contrast)	Static TM	L	Applicable to any AD; Considers both numeric and categorical features	KDE does not work well in high-dimensional subspace
[129]	A	F	Others (Subspace + Rule Extraction)	Static TN	L	Applicable to any AD; Explains anomalies in groups	Poor explanation fidelity
[67]	A	P	Approximate (Rule Extraction/Association Rule Mining)	Static ES	L	Applicable to any AD	Poor explanation fidelity
[55]	S	S	Approximate (Rule Extraction) + Visualisation	Static TM	G	Provides visual explanations	Only applicable to clustering based AD
[222]	A	F	Approximate (Rule Extraction/Decision Trees) + Visualisation	Static logs	G	Good scalability	Post-hoc explanations can be misleading
[170]	A	F	Approximate (Rule Extraction/Decision Trees)	Static TN	L	Applicable to any AD; Good scalability	Poor explanation fidelity
[54]	S	F	Approximate (Rule Extraction)	Static and streaming TM	L	Able to handle streaming data	Only applicable to tree-based ensembles
[18]	A	F	Approximate (Rule Extraction)	Static TM	L & G	Easy to evaluate the explanation quality	Only considers OCSVM
[17]	A	F	Approximate (XGBoost/LightGBM + LIME/SHAP; ElasticNet/EBM)	Static MTS	G	Able to integrate domain knowledge	Not suitable for explaining only one anomalous point

(Continued)

Table 3. Continued

Ref	Spec	Pers	Tech	Data	Loc	Pros	Cons
[102]	A	F	Approximate (SVM/XGBoost + LIME) + Visualisation	Static UTS	G	Applicable to any AD; End-user dependent explanations	Poor explanation fidelity
[15]	S	P	Others (Pattern comparison)	Streaming ES	L	Able to handle streaming data	Only applicable to LFA-like AD
[113]	A	P	Others (Pattern comparison)	Static ES	L	Generalisable	Needs many anomalies
[176]	S	F	Visualization	Static ES	L	Provides visual explanations	Only applicable to distance based AD methods
[123]	A	F	Others (Neighbours + Data Augmentation + Classification + Feature Contribution)	Static TN	L	Incorporates prior knowledge; Applicable to various anomaly detectors	Poor explanation fidelity; Important parameters to set by user; Only considers individual anomalies
[193]	S	F	Others (Separation-Based)	Static TN	L	Provides quantitative evaluation of explanation quality	Only applicable to density-based AD
[95]	S	F	Others (Isolation based feature importance)	Static TM	L	High explanation fidelity	Only applicable to Isolation Forest
[32]	S	F	Others (Isolation based feature importance)	Static TM	L & G	Provides local and global explanations; High explanation fidelity	Only applicable to Isolation Forest
[7]	S	P	Visualisation	Static MTS	L	Interpretable statistical model and visual interpretation	Only applicable to ARIMA
[133]	S	F	Visualization	Static TS	L	Provides visual explanations	Only applicable to Spatiotemporal model
[209]	S	F	Shapley values of reconstruction errors	Static tabular	L	More reliable compared to reconstruction error based explanation	Only applicable to PCA
[164]	A	F	SHAP	Static tabular	L	Applicable to any AD	Poor explanation fidelity
[103]	A	F	SHAP	Streaming MTS	L	Applicable to any AD	Poor explanation fidelity

Spec indicates whether a method is model-agnostic (A) or model-specific (S). *Pers* specifies whether a method is feature-based (F), sample-based (S) or pattern-based (P). *Tech* indicates the techniques used in each method. *Data* represents the type of data to which each method can be applied. *Data* indicates the data type for which the method is applicable (TN: Tabular Numeric; TC: Tabular Categorical; TM: Tabular Mixed; UTS: Univariate Time Series; MTS: Multivariate Time Series; ES: Event Sequence). *Loc* shows whether a method provides a local explanation (L) or global explanation (G). *Pros* and *Cons* describe the advantages and disadvantages of each method, respectively.

7.1 Explaining AutoEncoders

An AE is a type of neural network that first encodes the given data instances into some low-dimensional feature representation space and then decodes them back under the constraint of minimizing the reconstruction error. Several types of AEs have been introduced, including vanilla AE such as replicator neural network, **Sparse AutoEncoders (SAE)**, DAE, **Contractive AutoEncoders (CAE)**, VAE, and other variants [16]. AEs are widely used for anomaly detection, based on the assumption that anomalies are more difficult to reconstruct from the compressed feature representation space than normal instances.

First of all, Shapley values-based techniques such as SHAP are typically used to obtain feature contributions for explaining AEs. For instance, Giurgiu & Schumann [72] extend SHAP to explain anomalies identified via a GRU-based AutoEncoder in multivariate time series data. Specifically, they modify kernel SHAP [128] to output the windows that contribute the most to the anomaly and also the windows that counteract the most to the anomaly as explanations. Besides, to detect and explain anomalies in mobile **Radio Access Network (RAN)** data, Chawla et al. [40] set up a Sparse SAE-based anomaly detection algorithm and then applies kernel SHAP to explain the results. Furthermore, Jakubowski et al. [91] propose a VAE model combined with Shapley values to detect and interpret anomalies in an asset degradation process. Concretely, they compute Shapley values to generate both local and global explanations for anomalies. Additionally, Serradilla et al. [192] utilise different machine learning approaches to detect, predict and explain anomalies in press machines to achieve predictable maintenance. To interpret an anomaly detected by AE, they first leverage t-SNE [214] to visualise the learned latent feature spaces. Next, they employ the GradientExplainer tool [127], which combines SHAP, Integrated Gradients [206], and SmoothGrad [201], to analyze which input features are associated with the anomaly.

Second, many methods attempt to track reconstruction errors to obtain feature contribution by exploring the internal structure of AEs. Therefore, these methods are generally model-specific. For instance, Ikeda et al. [88] design a **Multimodal AutoEncoder (MAE)** model to detect anomalies emerging in ICT systems. More importantly, by using sparse optimization, they also propose an algorithm to estimate the contributing dimensions in an AE to anomalies as explanations. Besides, Nguyen et al. [151] introduce a framework called GEE to detect and explain anomalies in network traffic. Specifically, they train a VAE model on a normal dataset to learn the normal behaviour of a network, and then employ gradient-based fingerprinting technique to identify the main features causing the anomaly. Similarly, Memarzadeh et al. [140] propose a DGM-based on VAE. Particularly, they achieve model interpretability by evaluating feature importance through the random-permutation method. Additionally, Chen et al. [45] put forward DAEMON, which trains an **Adversarial AutoEncoder (AAE)** to learn the typical pattern of multivariate time series, and then use the reconstruction error to identify and explain anomalies. Meanwhile, to monitor wireless spectrum and identify unexpected behaviour, Rajendran et al. [171] present an AAE-based anomaly detection method named SAIFE. Since the AAE is trained in three phases, viz. *reconstruction*, *regularization*, and *semi-supervised* [130], SAIFE attempts to localize the anomalous regions based on the reconstruction errors coupled with the semi-supervised features, providing explanations for the anomalies. Furthermore, Ikeda et al. [89] set up an anomaly detection model based on VAE, and then estimate the features that contribute the most to the identified anomalies as explanations. Concretely, they present an approximative probabilistic model based on the trained VAE to estimate contributing features via exploring the so-called *true latent distribution*. The *true latent distribution* defines how an anomalous instance would be if it were normal. Importantly, they argue that directly estimating feature contribution based on the deviating latent distribution or reconstruction errors will lead to high false positives and/or negatives.

Third, some researchers attempt to utilise surrogate models such as LIME and rule learners to explain AEs. For example, Wu & Wang [220] propose a neural network-based model incorporating LIME techniques to detect and interpret fraudulent credit card transactions. Specifically, the anomaly detection model contains an AE and an MLP classifier, which are trained in an adversarial manner. To interpret an anomaly, they apply three independent LIME-based models to explain the AE, MLP, and AE & MLP models, respectively. Besides, Song et al. [203] develop the EXAD system to identify and interpret anomalies from Apache Spark traces. First, the EXAD system adapts AE and LSTM to perform anomaly detection. Second, they propose three ways to explain anomalies. The first one is to build a conjunction of the atomic predicates, which can be solved by a greedy algorithm but cannot guarantee the performance. To overcome this limitation, the second one attempts to use an entropy-based reward function to build atomic predicates. Furthermore, they present these constructed predicates in a Conjunctive Normal Form. The third one is to approximate the anomaly detection neural networks using a decision tree. From the decision tree, they generate explanations in a DNF. Additionally, De Moura et al. [56] present the **Lane Change Detector (LCD)** model to detect and explain when the surrounding vehicles of an ego vehicle change their lanes. Specifically, the LCD model consists of three independent AE models trained on three different datasets. On this basis, they set up a decision rule set based model by extracting rules from the reconstruction errors produced by these three separate models, to determine when an anomaly happens. Besides, Gnoss et al. [73] first annotate journal entries with previously trained AutoEncoders and then train three XAI models using these annotations. First, they utilise Decision Tree and Linear Regression, two intrinsically interpretable models, to simulate AE. The feature importance values of Decision Tree and the odd ratio values are calculated to show which feature is relevant to the anomalies. Additionally, they also leverage SHAP to explain the AE model.

Fourth, visualisation techniques such as Heatmaps and Saliency Maps are often constructed to help explain AEs. For instance, Kitamura & Nonaka [104] set up an encoder-decoder-based model to detect anomalies in images. To generate explanations for an anomaly, they first develop a feature extractor that is trained on a dataset consisting of normal images and their corresponding reconstructed images. Second, using this feature extractor to extract latent features, their method attempts to find the difference in the feature-level between the input image and the reconstructed image. On this basis, their method localizes and visualizes abnormal regions as explanations for the anomaly. Besides, Feng et al. [68] develop a Two-Stream AE-based model to detect abnormal events in videos and then utilise a Feature Map Visualization method to interpret the anomalies. Moreover, Guo et al. [77] set up a Sequence-to-Sequence VAE-based model to detect anomalies in event sequences. To reveal anomalous events, they investigate the differences between the anomalous sequence together with its reconstructed sequence and a set of normal sequences close to the anomalous sequence in the latent space. Importantly, they build a visualization tool to facilitate the comparisons. In addition, Szymanowicz et al. [208] develop a method for detecting and automatically explaining anomalous events in video. They first design an encoder-decoder architecture based on U-Net [177] to detect anomalies, thereby generating saliency maps by computing per-pixel differences between actual and predicted frames. Second, based on the per-pixel squared errors in the saliency maps, they introduce an explanation module that can provide spatial location and human-understandable representation of the identified anomalous event.

Finally, a wide range of methods such as feature selection, Markov Chain Monte Carlo, and providing similar historic anomalies, are also explored to facilitate the interpretability of AE-based anomaly detection. For example, Chakraborttii & Litz [35] develop an AE-based model to detect **Solid-State Drive (SSD)** failures. To produce explanations, they investigate the reconstruction error per feature, wherein a feature with a reconstruction error greater than the average error is considered a significant cause. Particularly, they apply three types of feature selection techniques,

viz. Filter, Wrapper and Embedded, to select important features to train the AE model, facilitating the interpretability of resulted anomaly detection model. Besides, Li et al. [115] develop a VAE and **genetic algorithm (GA)**-based framework, called VAGA, to detect anomalies in high-dimensional data and search corresponding abnormal subspaces. Concretely, for each identified anomaly, they utilize a GA to search the subspace where the anomaly deviates most. Additionally, Li et al. [117] introduce *InterFusion*, a model based on **hierarchical Variational AutoEncoder (HVAE)** and **Markov Chain Monte Carlo (MCMC)** for detecting and explaining anomalies in multivariate time series data. Specifically, given an anomaly, they set up a MCMC-based method to find a set of the most anomalous metrics as explanations. Furthermore, Assaf et al. [14] develop a **Convolutional AutoEncoders (ConvAE)**-based anomaly detection method and an explainability framework to detect and explain anomalies in data storage systems, respectively. Particularly, for each anomaly, they attempt to use cosine similarity over the embedding space to find similar historical anomalies, thereby explaining the anomaly through association.

Discussion: AEs are the most widely used deep learning method to detect anomalies in tabular data, sequence data, image data, video data and graph data. As a result, a plethora of methods are also proposed to explain AEs. Concretely, XAD techniques such as reconstruction error-based feature contribution, Kernel SHAP, GradientExplainer, LIME, rule extraction and feature map visualisation are often leveraged to obtain explanations. Importantly, most of these explanation methods only provide weak interpretability, as they only explain a single anomaly at a time by exploring some important properties of AE-based detection models.

7.2 Explaining Recurrent Neural Networks

A **Recurrent Neural Network (RNN)** is a specific type of neural network that is capable of learning features and long-term dependencies in sequential data [183]. Specifically, sequential data refers to any data that is ordered into sequences, including time series, text streams, DNA sequences, audio clips, video clips, and so on. To address the different challenges of modelling sequential data, various RNN architectures have been proposed. More concretely, frequently used RNNs include deep RNNs with MLP, **Bidirectional RNN (BiRNN)**, **Recurrent Convolutional Neural Networks (RCNN)**, **Multi-Dimensional Recurrent Neural Networks (MDRNN)**, LSTM, **Gated Recurrent Unit (GRU)**, Memory Networks, **Structurally Constrained Recurrent Neural Network (SCRNN)**, **Unitary Recurrent Neural Networks (Unitary RNN)**, and so on. Particularly, by assuming normal instances are temporally more predictable than anomalous instances, RNNs are extensively used to identify anomalies in sequential data because of their ability to model temporal dependencies.

First of all, Shapley values-based techniques such as SHAP are the most typical method used to obtain feature contributions, aiming to explain anomalies identified by RNNs. For instance, Zou and Petrosian [210] utilise Decision Trees [42] and DeepLog [63] to detect anomalies in system logs, and then explain the results using the Shapley value approach. To explain an anomaly, they treat each event in the logs as a player without examining the model structure to generate Shapley values. Moreover, Hwang & Lee [87] propose a bidirectional stackable LSTM-based anomaly detection model for industrial control system anomaly detection. For each identified anomaly, they employ SHAP values to obtain a contribution score of each feature as an explanation. Similarly, Jakubowski et al. [92] examine the issue of anomaly detection when hot rolling slabs into coils. They utilise LSTM to construct a modified AutoEncoder architecture in order to find anomalies. Importantly, they are able to pinpoint the origin of the majority of the abnormalities identified by the deep learning model through analysis of the SHAP interpretation. Furthermore, Nor et al. [152] present a probabilistic LSTM-based model combined with SHAP to detect and interpret anomalies in gas turbines. More importantly, they evaluate the quality of post-hoc explanations from two

aspects, viz. *local accuracy* and *consistency*. Specifically, *local accuracy* describes the relationship between feature contributions and predictions, while *consistency* checks whether the interpretation is consistent with changes in the input features.

Second, some researchers attempt to utilise surrogate models such as LIME to explain RNN. For example, Herskind Sejr et al. [84] create a predictive neural network-based unsupervised system by training an LSTM model and using reconstruction errors to assess data abnormalities. Importantly, the system offers two layers of anomaly interpretation: deviations from model predictions, and interpretations of model predictions, in order to make the process transparent to developers and users. They employ Mean Absolute Error to illustrate how observations diverge from assumptions at the first level. For the second level, they simulate a black-box model to provide an explanation using LIME. Additionally, Mathonsi & van Zyl [136] present **Multivariate Exponential Smoothing Long Short-Term Memory** (**MES-LSTM**) that combines statistics and deep learning. Particularly, they integrate SHAP and LIME and introduce a metric—called Mean Discovery Score—that aims at showing which predictors are most strongly associated with the anomalies.

Third, other methods such as **Layer-wise Relevance Propagation** (**LRP**), Integrated Gradients, and Attention Mechanism, are also leveraged to explain RNN-based anomaly detection. For instance, due to the complexity of log systems and the unstructured nature of the resulting logs, Patil et al. [166] use LSTM to detect anomalies in such systems. To generate explanations for each identified anomaly, they utilise LRP to generate relevance scores for every feature at every timestep. Moreover, Han et al. [80] present *InterpretableSAD*, a Negative Sampling based method for detecting and interpreting anomalies in sequential log data. First, due to the scarcity of anomalous instances, they adapt a data augmentation strategy via negative sampling to generate a dataset that contains sufficient anomalous samples. Second, they train an LSTM model based on this augmented labelled dataset. Third, they apply Integrated Gradients to identify anomalous events that lead to the outlyingness. Furthermore, to detect anomalies in system logs, Brown et al. [28] implement four attention mechanisms in LSTM and prove that compared to Bidirectional LSTM, the attention mechanism augmented LSTM not only retains high performance but also provides information about feature importance and relationship mapping between features, which provides explainability.

Discussion: RNNs are primarily employed to detect anomalies in sequence data. Typical XAD techniques for interpreting anomalies identified by RNN-based models include Shapley-value-based methods, surrogate models, and other versatile techniques such as LRP, Integrated Gradients, and Attention Mechanism. These post-hoc explanation methods are usually computationally expensive, making it difficult to provide real-time explanations.

7.3 Explaining Convolutional Neural Networks

A CNN is a specific type of neural network inspired by the visual cortex of animals. CNNs are widely used in the computer vision field because of their strong ability to extract features from image data with convolution structures. Moreover, CNNs have also been shown to be useful for extracting complex hidden features in sequential data [74]. Accordingly, a variety of CNN architectures have been proposed, including LeNet, AlexNet, GoogleNet, VGGNet, Inception V4, ResNet, and so on. Some studies have attempted to utilize CNNs for anomaly detection, especially in the fields of intrusion detection, image anomaly detection, and so on.

First, one line of research attempts to utilise surrogate models such as LIME and rule learners to explain CNN. For example, Cheong et al. [47] set up a **SpatioTemporal Convolutional Neural Network–based Relational Network** (**STCNN-RN**) to detect anomalous events in financial markets. For each anomaly, they apply LIME to provide a local explanation by indicating the contribution of each feature. Besides, Levy et al. [114] propose an end-to-end anomaly detection model

named AnoMili, which can also provide real-time explanations. Specifically, AnoMili consists of four stages. First, they introduce a physical intrusion detection mechanism by using AE. Second, if no anomalous device is discovered, they train a CNN-based classifier on voltage signals of each device, aiming to detect spoofing attacks. Third, they utilise LSTM to build a context-based anomaly detection mechanism, which detects anomalous messages based on their context. Finally, to interpret an anomalous message, they leverage decision tree to locally approximate the detection result and also apply SHAP TreeExplainer [128] to identify the most important features in real-time.

Second, visualisation techniques are often combined with other techniques such as Gradient Backpropagation and LRP to explain CNN-based anomaly detection. For instance, Saeki et al. [182] present a CNN-based method to detect and explain machinery faults based on vibration data. For each detected anomaly, they utilize grad-CAM [190], which is a gradient-based localization approach, to obtain an importance map in the feature space. Fourth, they combine the results of grad-CAM with a visualization approach called Guided Backpropagation [204]. Concretely, this method can visualize the predictions via backpropagation from the output space to the input space, generating explanations for the anomaly. Moreover, Chong et al. [48] introduce a CNN-based Teacher−Student Network-based model combined with LRP technique to detect and explain anomalies. To interpret an anomaly, they provide an example-based explanation by showing its top prototypes (namely, top nearest neighbours). Importantly, they apply LRP to show a pixel-level similarity between the anomaly and each of its top prototypes. Additionally, Szymanowicz et al. [207] introduce a model to detect and explain anomalies in videos. Specifically, they implement R-CNN to detect objects in video, and then employ Dual Relation Graph for human-object interaction recognition. The video is encoded with a collection of **human-object interaction vectors (HOI vectors)** for each frame. When the likelihood of the HOI vector in a scenario is less than a threshold, an anomaly is proclaimed. After using PCA to reduce the dimension of non-anomalies, they train a **Gaussian Mixture Model (GMM)**. A video frame is deemed abnormal if any of its HOI vectors are lower than the threshold probability under the GMM. The distance between the anomalous HOI vector and the usual HOI vector is then weighted and visualized as a 2D heatmap to help understand abnormalities.

Third, some researchers attempt to directly utilise the semantic anomaly scores as explanations. For instance, Hinami et al. [85] utilise a general CNN model and context-sensitive anomaly detectors to identify and explain abnormal events in films. Specifically, they set up a Fast R-CNN-based model to learn multiple concepts in videos and then extract semantic features. On this basis, they apply a context-sensitive anomaly detector to obtain semantic anomaly scores, which can be seen as explanations for anomalies.

Discussion: CNN-based anomaly detection models are mainly leveraged to detect anomalies in image data. To explain anomalies identified by CNNs, XAD techniques such as surrogate models (LIME and rule learners), Gradient Backpropagation, LRP, and visualisations are commonly used. However, some post-hoc explanation methods, especially surrogate models, may suffer from poor explanation fidelity. In other words, the generated explanations may not reflect the actual anomaly detection process of CNNs.

7.4 Explaining Other Deep Neural Networks

In addition to AEs, RNNs and CNNs, other DNNs—such as GANs, Deep OCSVM, and **Deviation Network (DevNet)**—can also be used for anomaly detection. Therefore, the interpretation of these types of networks is also relevant.

First, some studies propose explanation methods for general DNNs. For instance, Amarasinghe et al. [8] propose a framework for explainable DNN-based anomaly detection. Specifically, they assume the anomaly detection is performed in a supervised setting and leverage LRP

to obtain the input feature relevance for making a decision. Besides, Sipple [196] trains an anomaly detector using Neural Network with negative sampling to detect device failures in the Internet of Things. For each identified anomaly, they leverage Integrated Gradients techniques to attribute the anomaly score to each feature and provide a contrastive nearest normal instance as explanations.

Second, some researchers utilise techniques such as self-attention learning-based feature selection or gradient back propagation-based feature contribution to explain a DevNet. For instance, Xu et al. [221] propose **Attention-guided Triplet deviation network for Outlier interpretatioN (ATON)** to explain anomalies in a post-hoc fashion. Specifically, ATON is composed of two main modules, viz. the feature embedding module and the customized self-attention learning module. The feature embedding module transforms the original feature space into an embedding space with extended high-level information. Meanwhile, given an anomaly, the customized self-attention learning module can obtain the contribution of each learned feature to its separability. Based on the embedding module and the corresponding attention coefficients, they distil a subset of the original features that lead to the separability of the anomalous instance. Meanwhile, Pang et al. [160] put forward FASD, a weakly-supervised framework to detect anomalies when a few labeled anomalies of interest are available. Specifically, they instantiate this framework as a DevNet model, which assumes that the anomaly scores of normal instances are drawn from a Gaussian prior distribution and the anomaly scores of anomalies come from the upper tail of the prior. To interpret an anomaly, they compute the contribution of each input feature to the final anomaly score through gradient-based back propagation.

Third, deep Taylor decomposition [144] is leveraged to explain models such as OCSVM, KDE, and so on. For example, Kauffmann et al. [96] first convert the OCSVM models to neural networks, and then they modify the deep Taylor decomposition method to be applicable to these neural networks. In addition, they show its superiority to other explanation methods such as Distance Decomposition, Gradient-Based Method, SHAP Values, and Edge Detection, which are commonly used in deep learning to produce pixel-wise explanations of decisions. However, this method itself has many parameters to tune when applied to different methods or datasets, sometimes rendering the explanation method itself not explainable. Moreover, it also makes many strong assumptions and approximations. Similarly, Kauffmann et al. [97] reveal the widespread occurrence of Clever Hans phenomena in unsupervised anomaly detection models. Concretely, they propose an XAI procedure based on Deep Taylor Decomposition to highlight relevant features for detecting anomalies and apply it on models including AutoEncoder reconstruction-based detectors, Deep One-Class and KDE-based detectors, generating pixel-wise explanations of outlyingness.

Finally, visualisation techniques can be leveraged to help explain anomalies. For instance, Liu et al. [125] create the deep temporal clustering framework seq2cluster, which can cluster and detect anomalies in time series with varying lengths. The Temporal Segmentation, Temporal Compression network, and GMM Estimation modules make up seq2cluster. In particular, each sequence is divided into non-overlapping temporal segments via the Temporal Segmentation module. A low-dimensional representation of each time segment is what the Temporal Compression network aims at achieving. Moreover, the Estimation Network for GMMs utilises the latent space representation to perform density estimation. Therefore, data instances can be clustered in latent space to find anomalies based on the likelihood of each segment sample. The results of anomaly detection can also be more easily interpreted when anomalies found in the latent space are adequately visualized.

Discussion: In addition to the above-mentioned DNNs, namely, AEs, RNNs, CNNs, GANs, Deep OCSVM, and DevNet, other DNNs such as Graph Neural Networks [39] and Transformers [119]

have become prevalent in anomaly detection. Therefore, the interpretation methods of these DNNs are also relevant.

7.5 Summary

To wrap up our review of post-model XAD techniques for DNNs, Table 4 gives an overview of all techniques discussed and we have several high-level observations.

First, most deep post-model XAD techniques are model-specific in the sense that they are only applicable to a family of specific neural networks or all neural networks. This is in stark contrast with most shallow post-model XAD techniques, which are typically model-agnostic. This is because these deep post-model XAD techniques provide explanations by exploring the internal structure of the neural network. By doing so, although these explanation methods cannot be generalized to other anomaly detection models, the resulting explanations are usually faithful as the *Explanation-Definition* is in compliance with the *Detection-Definition*. However, techniques such as SHAP, LIME, and some rule learners are model-agnostic and are, therefore, more likely to suffer from poor fidelity.

Second, nearly all deep post-model XAD techniques provide only feature-based explanations; the only exceptions are References [14, 48, 72], which also produce sample-based explanations. Regarding the techniques used, the Shapley values-based approach is the most popular one. More importantly, one can see that most deep post-model XAD techniques are proposed to explain anomalies detected in sequential data such as time series and system logs.

Third, nearly all deep post-model XAD techniques can only provide local explanations. In other words, they can only explain a single anomaly at a time. Due to the complexity of neural networks, it is extremely challenging, if possible, to understand the entire decision-making process. To help end-users understand why an instance is reported as anomalous, deep post-model XAD techniques often inspect some important properties of the neural networks, such as feature contribution to reconstruction errors, thereby providing weak interpretability.

8 CONCLUSION AND FUTURE OPPORTUNITIES

We reviewed more than 150 articles that harness XAD techniques to explain anomalies. Specifically, we first introduced three different definitions of anomaly and then clarified what XAD is and why it is needed. On this basis, and inspired by existing surveys on XAI, we proposed a taxonomy consisting of six main criteria, enabling the categorization of the increasingly rich field of XAD. For purposes of brevity and organisation, we structured the review into four high-level categories (corresponding to Sections S4–7) and twelve fine-grained categories (corresponding to the subsections of S4–7). Throughout the survey, we identified a number of research challenges that may offer opportunities for future research, which we will summarize next.

8.1 Definition of Anomaly and XAD

A long-standing problem in anomaly analysis is the lack of a uniform definition of an anomaly, leading to a wide range of anomaly detection methods. The diversity of anomaly definitions and anomaly detection methods leads to the need for a large variety of anomaly explanation methods. Although is not necessarily problematic in itself (and maybe unavoidable), the lack of uniform definitions for anomaly detection and XAD hampers communication of researchers between different (sub)fields, such as computer vision, natural language processing, data mining, and social science. This makes it hard to find related work and leads to the re-invention of methods, causing unnecessary delays in scientific progress. More importantly, the evaluation and comparison of XAD methods becomes difficult and subjective, due to the lack of a uniform, objective, and precise definition of XAD.

Table 4. Summary of Surveyed Deep Post-model Techniques

Ref.	Spec	Pers	Tech	Data	Loc	Pros	Cons
[88]	S (MAE)	F	Reconstruction error-based feature contribution using sparse optimization	Static network-flow data	L	Able to handle cross-domain data	Only applicable to AEs
[72]	A (GRU-based AE)	S & P	Kernel SHAP based feature importance	Static MTS	L	Applicable to any AD	Assumes feature independence in Kernel SHAP
[40]	A (SAE)	F	Kernel SHAP based feature importance	Streaming ES	L	Applicable to any AD	Assumes feature independence in Kernel SHAP
[104]	S (AEs)	F	Feature-level reconstruction error + Visualisation	Static image	L	Provides visual explanations	Only applicable to AEs
[35]	S (AEs)	F	Reconstruction error based feature contribution + Feature selection	Static telemetry logs	L	Able to handle evolving data	Only applicable to AEs
[192]	S (AEs)	P	GradientExplainer + Visualization	Static MTS	L	Provides visual explanations	Only applicable to AEs
[68]	S (CNN-based AE)	F	Feature map visualization	Static video	L	Provides visual explanations	Post-hoc explanations may be misleading
[220]	A (AE+MLP)	F	Approximate (LIME)	Static TN	L	Applicable to any AD	Poor explanation fidelity
[203]	A (AE+LSTM)	F	Approximate (Rule extraction)	Streaming ES	L	Handles streaming data; Applicable to any AD	Post-hoc explanations may not be reliable
[151]	S (VAE)	F	Gradient based feature contribution	Streaming ES	L	Handles streaming data	Only applicable to reconstruction based DNN
[77]	S (VAE)	P	Reconstruction error based pattern comparison + Visualization	Static ES	L	Provides interactive visual explanations	Only applicable to reconstruction based AD
[115]	A (VAE)	F	GA based subspace search	Static TM	L	Applicable to any AD	High computational cost
[91]	A (VAE)	F	Shapley values based feature contribution	Static MTS	L & G	Applicable to any AD	High computational cost
[89]	S (VAE)	F	Others (True latent distribution based feature contribution)	Static TN	L	More reliable than reconstruction error-based explanations	Only applicable to VAE
[117]	S (VAE)	F	Others (MCMC-based method)	Streaming MTS	L	Provides real-time explanations	Unable to handle evolving data
[140]	A (VAE)	F	Perturbation based feature importance	Static MTS	L	Applicable to any AD	Needs a few labelled data
[14]	S (ConvAE)	F & S	Similar historic anomaly + Reconstruction error based feature contribution	Static MTS	L	End-to-end framework	Only applicable to Convolutional AEs
[56]	S (AEs)	F	Approximate (Rule extraction from reconstruction error)	Streaming MTS	G	Handles streaming data	Needs many anomalous samples; Only applicable to reconstruction-based AD
[45]	S (AAE)	F	Reconstruction error based feature contribution	Streaming MTS	L	Provides quantitative evaluation of explanation quality	Only applicable to reconstruction-based AD

(Continued)

Table 4. Continued

Ref.	Spec	Pers	Tech	Data	Loc	Pros	Cons
[171]	S (AAE)	F	Reconstruction errors coupled with the semi-supervised features in AAE	Static MTS	L	Works in both unsupervised and semi-supervised settings	Only applicable to AAE
[208]	S (U-Net based AEs)	F	Saliency maps + Human-understandable representation	Static video	L	Provides visual explanations	Only applicable to AEs
[73]	A (AEs)	F	SHAP + Approximate (Decision tree + Linear regression)	Static TM	L	Applicable to any AD	Poor explanation fidelity; Manual evaluation of explanation quality
[8]	S (DNN)	F	LRP based feature relevance	Any data type	L & G	Not easy to understand for non-experts	Needs labelled data; Only applicable to certain DNNs
[166]	S (LSTM)	F	LRP based feature relevancy	Static ES	L	Not easy to understand for non-experts	Needs labelled data; Only applicable to certain DNNs
[210]	A (DT + LSTM)	F	Shapley values based feature importance	Static ES	L	Applicable to any AD	Poor explanation fidelity
[87, 92, 152]	A (LSTM)	F	SHAP	MTS	L	Applicable to any AD	Poor explanation fidelity
[84]	A (LSTM)	F	Approximate (LIME)	Streaming UTS	L	Applicable to any AD; Handles streaming data	Not easy to understand for non-experts
[80]	S (LSTM)	F	Integrated Gradients	Static ES	L	Not needs labelled data	Only applicable to DNN; Unable to handle evolving data
[28]	S (LSTM)	F	Attention mechanism	Streaming ES	L	Handles streaming data	Only applicable to RNNs
[136]	A (MES-LSTM)	F	Approximate (LIME) + SHAP	Static MTS	L	Applicable to any AD	Poor explanation fidelity
[182]	S (CNN)	F	grad-CAM + Visualization	Static UTS	L	Provides visual explanations	Only applicable to CNNs
[47]	A (CNN)	F	Approximate (LIME)	Static UTS	L	Applicable to any AD	Poor explanation fidelity
[48]	S (CNN)	F & S	LRP based feature importance + Prototypes explanation + Visualisation	Static image	L	Provides visual explanations	Hard to set the number of prototypes
[114]	A (AE + CNN + LSTM)	P	Approximate (Decision Tree) + SHAP TreeExplainer	Streaming MTS	L	Applicable to any AD; Provides real-time detection and explanations	Poor explanation fidelity
[207]	S (R-CNN)	F	Others (PCA + GMM) + Visualisation	Static video	L	Provides visual explanations	Unable to handle unseen data in training phrase
[85]	S (R-CNN)	F	Others (Semantic Anomaly Score)	Static video	L	Joint abnormal event detection and recounting	Weak interpretability using only semantic anomaly scores
[196]	S (DNN)	F & S	Integrated Gradients	Streaming TS	L	Handles streaming data	Only applicable to DNN
[221]	A (DevNet)	F	Others (Self-attention learning based feature selection)	Static TM	L	Applicable to any AD	Poor explanation fidelity

(Continued)

Table 4. Continued

Ref.	Spec	Pers	Tech	Data	Loc	Pros	Cons
[160]	S (DevNet)	F	Gradient back propagation based feature contribution	Static image	L	End-to-end training	Only applicable to specific DNN
[96, 97]	S (OCSVM)	F	Approximate (NN) + LRP-type Back-propagation based feature importance at pixel-level	Static image	L	Better performance compared to others	Only applicable to OCSVM and potentially some distance-based methods
[125]	S (DNN)	F	Attention mechanism + GMM + Visualisation	Static UTS	L	Handles time series with varying lengths; Provides visual explanations	Only applicable to specific DNN

Spec indicates whether a method is model-agnostic (A) or model-specific (S). If it is model-specific, we also indicate the models to which it applies. However, for completeness, we also indicate the involved DNN framework for model-agnostic techniques. *Pers* specifies whether a method is feature-based (F), sample-based (S) or pattern-based (P). *Data* indicates the data type for which the method is applicable (TN: Tabular Numeric; TC: Tabular Categorical; TM: Tabular Mixed; UTS: Univariate Time Series; MTS: Multivariate Time Series; ES: Event Sequence). *Loc* shows whether a method provides a local explanation (L) or global explanation (G).

8.2 Evaluation of XAD

Despite the clearly stated needs for XAD in various domains, and especially those domains involving high-stakes decisions, the question of how XAD techniques should be evaluated remains unanswered. Over the past few years, the long-standing problem of measuring and assessing machine learning explainability has received certain attention [34]. However, most of these methods are specifically designed for classification or clustering problems, and extending these methods to anomaly detection problems is non-trivial.

Particularly, the fidelity of post-hoc explanations merits close scrutiny when evaluating XAD techniques. Inconsistency between the *Oracle-Definition* and *Detection-Definition* of an anomaly may lead to the identification of anomalies that are not of interest to the end-users. Moreover, inconsistency between the *Detection-Definition* (if available) and *Explanation-Definition* of an anomaly may lead to poor explanation fidelity. In other words, the explanations do not reflect the actual decision-making process of an anomaly detection model. In general, pre-model and in-model XAD techniques do not suffer from this problem, while most post-hoc XAD techniques—that only correlate inputs with outputs, without exploring the internal structure of detection models—are afflicted with this problem. For this reason, some researchers [180] appeal not to use opaque models and then explain them in a post-hoc manner for high-stakes decisions. Instead, an intrinsically explainable model should be used. We emphasize that the same argument also applies to anomaly detection and explanation.

8.3 XAD with Prior Knowledge

Most XAD techniques attempt to provide explanations solely based on information contained in existing data instances (anomalous or not) and/or anomaly detection models. However, sometimes additional knowledge about data instances or anomaly detection models may be acquired, in the form of algebraic equations, simulation results, logic rules, knowledge graphs, human feedback, and so on. Importantly, this prior knowledge can be integrated to augment training data, choose a network architecture, initialize parameters or validate model outputs. This paradigm of integrating prior knowledge into machine learning is called *Informed Machine Learning* [218], which has received increasing attention over the past few years. Particularly, Beckh et al. [19] have performed a survey on methods that integrate prior knowledge into machine learning for improving explainability, wherein they subdivide these methods into three categories, including the integration of knowledge into machine learning problems such as classification, regression, clustering, or anomaly detection, the integration of knowledge into explanation method, and deriving knowledge from the explanation results and then integrating it into the machine learning pipeline. Therefore, repurposing these methods for anomaly detection is undoubtedly beneficial to improve interpretability, and thus is a promising future direction.

8.4 Adversarial Attacks in XAD

Belle & Papantonis [20] point out that some widely used XAI techniques are vulnerable to adversarial attacks. In particular, post-hoc XAI techniques such as LIME and SHAP are easily manipulated [198]. From the reviewed results, we can see that most of these methods in XAI have been repurposed for XAD. As a result, anomaly explanations obtained by using these techniques have the possibility of being manipulated or attacked. To circumvent this problem, attention should be paid when selecting an XAD method. In particular, more efforts should be made to develop interpretation methods that take into account adversarial attacks in the future.

8.5 Scalability of XAD

Last but not least, the scalability of XAD plays an important role in some applications. For example, large internet companies often develop an anomaly detection system to monitor a large number of

key performance indicators, aiming to ensure the reliability of their service platform [118]. However, after identifying anomalies, they have to find the root causes and then take remedial actions as soon as possible. To achieve this in an automated manner, an anomaly interpretation method that can process large amounts of data and provide near real-time interpretation is required. However, most existing XAD techniques—such as subspace anomaly detection and Shapley value-based methods—have a high computational cost. Therefore, the development of scalable XAD techniques (with high fidelity) is an important direction for future research.

REFERENCES

[1] Charu C. Aggarwal. 2015. Outlier analysis. In *Proceedings of the Data Mining*. Springer, 237–263.

[2] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB*, Vol. 1215. Santiago, Chile, 487–499.

[3] Shikha Agrawal and Jitendra Agrawal. 2015. Survey on anomaly detection using data mining techniques. *Procedia Computer Science* 60 (2015), 708–713.

[4] Diana Laura Aguilar, Miguel Angel Medina Perez, Octavio Loyola-Gonzalez, Kim-Kwang Raymond Choo, and Edoardo Bucheli-Susarrey. 2023. Towards an interpretable autoencoder: A decision tree-based autoencoder and its application in anomaly detection. *IEEE Transactions on Dependable and Secure Computing* 20, 02 (2023), 1048–1059.

[5] Malik Agyemang, Ken Barker, and Rada Alhajj. 2006. A comprehensive survey of numeric and symbolic outlier mining techniques. *Intelligent Data Analysis* 10, 6 (2006), 521–538.

[6] Mohiuddin Ahmed, Abdun Naser Mahmood, and Md Rafiqul Islam. 2016. A survey of anomaly detection techniques in financial domain. *Future Generation Computer Systems* 55 (2016), 278–288.

[7] Morteza Alizadeh, Michael Hamilton, Parker Jones, Junfeng Ma, and Raed Jaradat. 2021. Vehicle operating state anomaly detection and results virtual reality interpretation. *Expert Systems with Applications* 177 (2021), 114928.

[8] Kasun Amarasinghe, Kevin Kenney, and Milos Manic. 2018. Toward explainable deep neural network based anomaly detection. In *Proceedings of the 2018 11th International Conference on Human System Interaction (HSI)*. IEEE, 311–317.

[9] Fabrizio Angiulli, Fabio Fassetti, Giuseppe Manco, and Luigi Palopoli. 2017. Outlying property detection with numerical attributes. *Data Mining and Knowledge Discovery* 31, 1 (2017), 134–163.

[10] Fabrizio Angiulli, Fabio Fassetti, and Luigi Palopoli. 2009. Detecting outlying properties of exceptional objects. *ACM Transactions on Database Systems* 34, 1 (2009), 1–62.

[11] Fabrizio Angiulli, Fabio Fassetti, and Luigi Palopoli. 2012. Discovering characterizations of the behavior of anomalous subpopulations. *IEEE Transactions on Knowledge and Data Engineering* 25, 6 (2012), 1280–1292.

[12] Liat Antwarg, Ronnie Mindlin Miller, Bracha Shapira, and Lior Rokach. 2021. Explaining anomalies detected by autoencoders using Shapley Additive Explanations. *Expert Systems with Applications* 186 (2021), 115736.

[13] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, Chatila Raja, and Herrera Francisco. 2020. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion* 58 (2020), 82–115.

[14] Roy Assaf, Ioana Giurgiu, Jonas Pfefferle, Serge Monney, Haris Pozidis, and Anika Schumann. 2021. An anomaly detection and explainability framework using convolutional autoencoders for data storage systems. In *Proceedings of the 29th International Conference on International Joint Conferences on Artificial Intelligence*. 5228–5230.

[15] Anton Babenko, Leonardo Mariani, and Fabrizio Pastore. 2009. Ava: Automated interpretation of dynamically detected anomalies. In *Proceedings of the 18th International Symposium on Software Testing and Analysis*. 237–248.

[16] Dor Bank, Noam Koenigstein, and Raja Giryes. 2020. Autoencoders. *CoRR* abs/2003.05991 (2020). arXiv:2003.05991

[17] Alberto Barbado and Óscar Corcho. 2022. Interpretable machine learning models for predicting and explaining vehicle fuel consumption anomalies. *Engineering Applications of Artificial Intelligence* 115 (2022), 105222.

[18] Alberto Barbado, Óscar Corcho, and Richard Benjamins. 2022. Rule extraction in unsupervised anomaly detection for model explainability: Application to OneClass SVM. *Expert Systems with Applications* 189 (2022), 116100.

[19] Katharina Beckh, Sebastian Müller, Matthias Jakobs, Vanessa Toborek, Hanxiao Tan, Raphael Fischer, Pascal Welke, Sebastian Houben, and Laura von Rueden. 2023. Harnessing Prior Knowledge for Explainable Machine Learning: An Overview. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. IEEE, 450–463.

[20] Vaishak Belle and Ioannis Papantonis. 2021. Principles and practice of explainable machine learning. *Frontiers in Big Data* 4 (2021), 39.

[21] Fabian Berns, Markus Lange-Hegermann, and Christian Beecks. 2020. Towards gaussian processes for automatic and interpretable anomaly detection in industry 4.0. In *Proceedings of the IN4PL*. 87–92.

[22] Xingyan Bin, Ying Zhao, and Bilong Shen. 2016. Abnormal subspace sparse PCA for anomaly detection and interpretation. In *ODDx3'15:ACM SIGKDD 2015 Workshop*. Association for Computing Machinery.

[23] Daniel Bogdoll, Maximilian Nitsche, and J. Marius Zöllner. 2022. Anomaly detection in autonomous driving: A survey. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4488–4499.

[24] Kristof Böhmer and Stefanie Rinderle-Ma. 2020. Mining association rules for anomaly detection in dynamic process runtime behavior and explaining the root cause to users. *Information Systems* 90 (2020), 101438.

[25] Azzedine Boukerche, Lining Zheng, and Omar Alfandi. 2020. Outlier detection: Methods, models, and classification. *ACM Computing Surveys* 53, 3 (2020), 1–37.

[26] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. LOF: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*. 93–104.

[27] David Broniatowski. 2021. Psychological foundations of explainability and interpretability in artificial intelligence. (2021).

[28] Andy Brown, Aaron Tuor, Brian Hutchinson, and Nicole Nichols. 2018. Recurrent neural network attention mechanisms for interpretable system log anomaly detection. In *Proceedings of the 1st Workshop on Machine Learning for Computing Systems*. 1–8.

[29] H. Burak Gunay, Weiming Shen, Guy Newsham, and Araz Ashouri. 2019. Detection and interpretation of anomalies in building energy use through inverse modeling. *Science and Technology for the Built Environment* 25, 4 (2019), 488–503.

[30] Nadia Burkart and Marco F. Huber. 2021. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research* 70 (2021), 245–317.

[31] Mattia Carletti, Marco Maggipinto, Alessandro Beghi, Gian Antonio Susto, Natalie Gentner, Yao Yang, and Andreas Kyek. 2020. Interpretable anomaly detection for knowledge discovery in semiconductor manufacturing. In *Proceedings of the 2020 Winter Simulation Conference (WSC'20)*. IEEE, 1875–1885.

[32] Mattia Carletti, Matteo Terzi, and Gian Antonio Susto. 2023. Interpretable anomaly detection with diffi: Depth-based feature importance of isolation forest. *Engineering Applications of Artificial Intelligence* 119 (2023), 105730.

[33] Gail A. Carpenter and Stephen Grossberg. 1987. Neural dynamics of category learning and recognition: Attention, memory consolidation, and amnesia. In *Proceedings of the Advances in Psychology*. Elsevier, 239–286.

[34] Diogo V. Carvalho, Eduardo M. Pereira, and Jaime S. Cardoso. 2019. Machine learning interpretability: A survey on methods and metrics. *Electronics* 8, 8 (2019), 832.

[35] Chandraniil Chakraborttii and Heiner Litz. 2020. Explaining SSD failures using anomaly detection. In *Proceedings of the Non-Volatile Memory Workshop*. 1.

[36] Raghavendra Chalapathy and Sanjay Chawla. 2019. Deep learning for anomaly detection: A survey. *CoRR* abs/1901.03407 (2019). arXiv:1901.03407

[37] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *ACM Computing Surveys* 41, 3 (2009), 1–58.

[38] Chun-Hao Chang, Jinsung Yoon, Sercan Arik, Madeleine Udell, and Tomas Pfister. 2022. Data-efficient and interpretable tabular anomaly detection. *CoRR* abs/2203.02034 (2022). arXiv:2203.02034

[39] Anshika Chaudhary, Himangi Mittal, and Anuja Arora. 2019. Anomaly detection using graph neural networks. In *Proceedings of the 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMIT-Con'19)*. IEEE, 346–350.

[40] Ashima Chawla, Paul Jacob, Saman Feghhi, Devashish Rughwani, Sven van der Meer, and Sheila Fallon. 2020. Interpretable unsupervised anomaly detection for RAN cell trace analysis. In *Proceedings of the 2020 16th International Conference on Network and Service Management (CNSM)*. IEEE, 1–5.

[41] Liang-Chieh Chen, Tsung-Ting Kuo, Wei-Chi Lai, Shou-De Lin, and Chi-Hung Tsai. 2012. Prediction-based outlier detection with explanations. In *Proceedings of the 2012 IEEE International Conference on Granular Computing*. IEEE, 44–49.

[42] Mike Chen, Alice X. Zheng, Jim Lloyd, Michael I. Jordan, and Eric Brewer. 2004. Failure diagnosis using decision trees. In *Proceedings of the International Conference on Autonomic Computing, 2004*. IEEE, 36–43.

[43] N. F. Chen, Zhiyuan Du, and Khin Hua Ng. 2018. Scene graphs for interpretable video anomaly classification. In *Proceedings of the Conference on Neural Information Processing Systems Workshop on Visually Grounded Interaction and Language*.

[44] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining*. 785–794.

[45] Xuanhao Chen, Liwei Deng, Feiteng Huang, Chengwei Zhang, Zongquan Zhang, Yan Zhao, and Kai Zheng. 2021. Daemon: Unsupervised anomaly detection and interpretation for multivariate time series. In *Proceedings of the 2021 IEEE 37th International Conference on Data Engineering (ICDE'21)*. IEEE, 2225–2230.

[46] Ximeng Cheng, Zhiqian Wang, Xuexi Yang, Liyan Xu, and Yu Liu. 2021. Multi-scale detection and interpretation of spatio-temporal anomalies of human activities represented by time-series. *Computers, Environment and Urban Systems* 88 (2021), 101627.

[47] Mei-See Cheong, Mei-Chen Wu, and Szu-Hao Huang. 2021. Interpretable stock anomaly detection based on spatio-temporal relation networks with genetic algorithm. *IEEE Access* 9 (2021), 68302–68319.

[48] Penny Chong, Ngai-Man Cheung, Yuval Elovici, and Alexander Binder. 2021. Toward scalable and unified example-based explanation and outlier detection. *IEEE Transactions on Image Processing* 31 (2021), 525–540.

[49] William S. Cleveland. 1979. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association* 74, 368 (1979), 829–836.

[50] European Commission. 2020. On Artificial Intelligence–A European Approach to Excellence and Trust. Available online at: https://commission.europa.eu/publications/white-paper-artificial-intelligence-european-approach-excellence-and-trust_en (accessed on 19 May 2022).

[51] David Cortes. 2020. Explainable outlier detection through decision tree conditioning. *CoRR* abs/2001.00636 (2020). arXiv:2001.00636

[52] Xuan Hong Dang, Ira Assent, Raymond T. Ng, Arthur Zimek, and Erich Schubert. 2014. Discriminative features for identifying and interpreting outliers. In *Proceedings of the 2014 IEEE 30th International Conference on Data Engineering*. IEEE, 88–99.

[53] Xuan Hong Dang, Barbora Micenková, Ira Assent, and Raymond T. Ng. 2013. Local outlier detection with interpretation. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 304–320.

[54] Shubhomoy Das, Md Rakibul Islam, Nitthilan Kannappan Jayakodi, and Janardhan Rao Doppa. 2019. Active anomaly detection via ensembles: Insights, algorithms, and interpretability. *CoRR* abs/1901.08930 (2019). arXiv:1901.08930

[55] Ian Davidson. 2007. Anomaly detection, explanation and visualization. *SGI, Tokyo, Japan, Tech. Rep* (2007).

[56] Oliver De Candido, Maximilian Binder, and Wolfgang Utschick. 2021. An interpretable lane change detector algorithm based on deep autoencoder anomaly detection. In *Proceedings of the 2021 IEEE Intelligent Vehicles Symposium (IV’21)*. IEEE, 516–523.

[57] Leonardo De Moura and Nikolaj Bjørner. 2011. Satisfiability modulo theories: Introduction and applications. *Communications of the ACM* 54, 9 (2011), 69–77.

[58] Linda Delamaire, Hussein Abdou, and John Pointon. 2009. Credit card fraud and detection techniques: A review. *Banks and Bank Systems* 4, 2 (2009), 57–68.

[59] Charlie Dickens, Eric Meissner, Pablo G. Moreno, and Tom Diethe. 2020. Interpretable anomaly detection with mondrian polya forests on data streams. *CoRR* abs/2008.01505 (2020). arXiv:2008.01505

[60] Theekshana Dissanayake, Tharindu Fernando, Simon Denman, Sridha Sridharan, Houman Ghaemmaghami, and Clinton Fookes. 2020. A robust interpretable deep learning classifier for heart anomaly detection without segmentation. *IEEE Journal of Biomedical and Health Informatics* 25, 6 (2020), 2162–2171.

[61] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *CoRR* abs/1702.08608 (2017). arXiv:1702.08608

[62] Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. 2018. Explainable artificial intelligence: A survey. In *Proceedings of the 2018 41st International Convention on Information and Communication Technology, Electronics, and Microelectronics (MIPRO’18)*. IEEE, 0210–0215.

[63] Min Du, Feifei Li, Guineng Zheng, and Vivek Srikumar. 2017. Deeplog: Anomaly detection and diagnosis from system logs through deep learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 1285–1298.

[64] Lei Duan, Guanting Tang, Jian Pei, James Bailey, Akiko Campbell, and Changjie Tang. 2015. Mining outlying aspects on numeric data. *Data Mining and Knowledge Discovery* 29, 5 (2015), 1116–1151.

[65] Ricardo Dunia and S. Joe Qin. 1997. Multi-dimensional fault diagnosis using a subspace approach. In *Proceedings of the American Control Conference*.

[66] N. Dunstan, I. Despi, and C. Watson. 2009. Anomalies in multidimensional contexts. *WIT Transactions on Information and Communication Technologies* 42 (2009), 173.

[67] Levent Ertoz, Eric Eilertson, Aleksandar Lazarevic, Pang-Ning Tan, Vipin Kumar, Jaideep Srivastava, and Paul Dokas. 2004. Minds-minnesota intrusion detection system. *Next Generation Data Mining* (2004), 199–218.

[68] Jiangfan Feng, Yukun Liang, and Lin Li. 2021. Anomaly detection in videos using two-stream autoencoder with post hoc interpretability. *Computational Intelligence and Neuroscience* 2021 (2021).

[69] Tharindu Fernando, Houman Ghaemmaghami, Simon Denman, Sridha Sridharan, Nayyar Hussain, and Clinton Fookes. 2019. Heart sound segmentation using bidirectional LSTMs with attention. *IEEE Journal of Biomedical and Health Informatics* 24, 6 (2019), 1601–1609.

[70] Sylvain Fuertes, Gilles Picart, Jean-Yves Tourneret, Lotfi Chaari, André Ferrari, and Cédric Richard. 2016. Improving spacecraft health monitoring with automatic anomaly detection techniques. In *Proceedings of the 14th International Conference on Space Operations*. 2430.

[71] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 2018. Explaining explanations: An overview of interpretability of machine learning. In *Proceedings of the 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA'18)*. IEEE, 80–89.

[72] Ioana Giurgiu and Anika Schumann. 2019. Additive explanations for anomalies detected from multivariate temporal data. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2245–2248.

[73] Nico Gnoss, Martin Schultz, and Marina Tropmann-Frick. 2022. XAI in the audit domain-explaining an autoencoder model for anomaly detection. In *17th International Conference on Wirtschaftsinformatik*. AISeL.

[74] Oleg Gorokhov, Mikhail Petrovskiy, and Igor Mashechkin. 2017. Convolutional neural networks for unsupervised anomaly detection in text data. In *Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 500–507.

[75] Gudmund Grov, Marc Sabate, Wei Chen, and David Aspinall. 2019. Towards intelligible robust anomaly detection by learning interpretable behavioural models. *NISK J* 32 (2019), 1–16.

[76] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *ACM Computing Surveys (CSUR)* 51, 5 (2018), 1–42.

[77] Shunan Guo, Zhuochen Jin, Qing Chen, David Gotz, Hongyuan Zha, and Nan Cao. 2022. Interpretable anomaly detection in event sequences via sequence matching and visual comparison. *IEEE Transactions on Visualization and Computer Graphics* 28 (2022), 4531–4545.

[78] Manish Gupta, Jing Gao, Charu C. Aggarwal, and Jiawei Han. 2013. Outlier detection for temporal data: A survey. *IEEE Transactions on Knowledge and Data Engineering* 26, 9 (2013), 2250–2267.

[79] Nikhil Gupta, Dhivya Eswaran, Neil Shah, Leman Akoglu, and Christos Faloutsos. 2018. Beyond outlier detection: Lookout for pictorial explanation. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 122–138.

[80] Xiao Han, He Cheng, Depeng Xu, and Shuhan Yuan. 2021. InterpretableSAD: Interpretable anomaly detection in sequential log data. In *Proceedings of the 2021 IEEE International Conference on Big Data (Big Data'21)*. IEEE, 1183–1192.

[81] Douglas M. Hawkins. 1980. *Identification of Outliers*. Springer.

[82] Jingrui He and Jaime Carbonell. 2010. Co-selection of features and instances for unsupervised rare category analysis. In *Proceedings of the 2010 SIAM International Conference on Data Mining*. SIAM, 525–536.

[83] Zengyou He, Xiaofei Xu, Zhexue Joshua Huang, and Shengchun Deng. 2005. FP-outlier: Frequent pattern based outlier detection. *Computer Science and Information Systems* 2, 1 (2005), 103–118.

[84] Jonas Herskind Sejr, Thorbjørn Christiansen, Nicolai Dvinge, Dan Hougesen, Peter Schneider-Kamp, and Arthur Zimek. 2021. Outlier detection with explanations on music streaming data: A case study with danmark music group Ltd. *Applied Sciences* 11, 5 (2021), 2270.

[85] Ryota Hinami, Tao Mei, and Shin'ichi Satoh. 2017. Joint detection and recounting of abnormal events by learning deep generic knowledge. In *Proceedings of the IEEE International Conference on Computer Vision*. 3619–3627.

[86] Victoria Hodge and Jim Austin. 2004. A survey of outlier detection methodologies. *Artificial Intelligence Review* 22, 2 (2004), 85–126.

[87] Chanwoong Hwang and Taejin Lee. 2021. E-sfd: Explainable sensor fault detection in the ics anomaly detection system. *IEEE Access* 9 (2021), 140470–140486.

[88] Yasuhiro Ikeda, Keisuke Ishibashi, Yuusuke Nakano, Keishiro Watanabe, and Ryoichi Kawahara. 2018. Anomaly detection and interpretation using multimodal autoencoder and sparse optimization. *CoRR* abs/1812.07136 (2018). arXiv:1812.07136

[89] Yasuhiro Ikeda, Kengo Tajiri, Yuusuke Nakano, Keishiro Watanabe, and Keisuke Ishibashi. 2018. Estimation of dimensions contributing to detected anomalies with variational autoencoders. *CoRR* abs/1811.04576 (2018). arXiv:1811.04576

[90] Sarah Itani, Fabian Lecron, and Philippe Fortemps. 2020. A one-class classification decision tree based on kernel density estimation. *Applied Soft Computing* 91 (2020), 106250.

[91] Jakub Jakubowski, Przemysław Stanisz, Szymon Bobek, and Grzegorz J. Nalepa. 2021. Anomaly detection in asset degradation process using variational autoencoder and explanations. *Sensors* 22, 1 (2021), 291.

[92] Jakub Jakubowski, Przemysław Stanisz, Szymon Bobek, and Grzegorz J. Nalepa. 2021. Explainable anomaly detection for Hot-rolling industrial process. In *Proceedings of the 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA'21)*. IEEE, 1–10.

[93] Jinchao Ji, Tian Bai, Chunguang Zhou, Chao Ma, and Zhe Wang. 2013. An improved k-prototypes clustering algorithm for mixed numeric and categorical data. *Neurocomputing* 120 (2013), 590–596.

[94] Ian T. Jolliffe, Nickolay T. Trendafilov, and Mudassir Uddin. 2003. A modified principal component technique based on the LASSO. *Journal of Computational and Graphical Statistics* 12, 3 (2003), 531–547.

[95] Nirmal Sobha Kartha, Clément Gautrais, and Vincent Vercruyssen. 2021. Why are you weird? Infusing interpretability in isolation forest for anomaly detection. In *Proceedings of the Explainable Agency in AI Workshop (AAAI'21).* 51–57.

[96] Jacob Kauffmann, Klaus-Robert Müller, and Grégoire Montavon. 2020. Towards explaining anomalies: A deep Taylor decomposition of one-class models. *Pattern Recognition* 101 (2020), 107198.

[97] Jacob Kauffmann, Lukas Ruff, Grégoire Montavon, and Klaus-Robert Müller. 2020. The clever Hans effect in anomaly detection. *CoRR* abs/2006.10609 (2020). arXiv:2006.10609

[98] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (Long Beach, California, USA) (NIPS'17).* Curran Associates Inc., 3149–3157.

[99] Fabian Keller, Emmanuel Muller, and Klemens Bohm. 2012. HiCS: High contrast subspaces for density-based outlier ranking. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering.* IEEE, 1037–1048.

[100] Fabian Keller, Emmanuel Müller, Andreas Wixler, and Klemens Böhm. 2013. Flexible and adaptive subspace search for outlier analysis. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management.* 1381–1390.

[101] Eamonn Keogh, Jessica Lin, and Ada Fu. 2005. Hot sax: Efficiently finding the most unusual time series subsequence. In *Proceedings of the 5th IEEE International Conference on Data Mining (ICDM'05).* IEEE, 8–pp.

[102] Sebastian Kiefer and Günter Pesch. 2021. Unsupervised anomaly detection for financial auditing with model-agnostic explanations. In *Proceedings of the German Conference on Artificial Intelligence (Künstliche Intelligenz).* Springer, 291–308.

[103] Donghyun Kim, Gian Antariksa, Melia Putri Handayani, Sangbong Lee, and Jihwan Lee. 2021. Explainable anomaly detection framework for maritime main engine sensor data. *Sensors* 21, 15 (2021), 5200.

[104] Shogo Kitamura and Yuichi Nonaka. 2019. Explainable anomaly detection via feature-based localization. In *Proceedings of the International Conference on Artificial Neural Networks.* Springer, 408–419.

[105] Edwin M. Knorr and Raymond T. Ng. 1998. Algorithms for mining distance-based outliers in large datasets. In *Proceedings of the VLDB.* Citeseer, 392–403.

[106] Edwin M. Knorr and Raymond T. Ng. 1999. Finding intensional knowledge of distance-based outliers. In *Proceedings of the VLDB.* Citeseer, 211–222.

[107] Martin Kopp, Tomáš Pevnỳ, and Martin Holena. 2014. Interpreting and clustering outliers with sapling random forests. In *Proceedings of the European Conference on Information Technologies–Applications and Theory.* 61–67.

[108] Ines Ben Kraiem, Faiza Ghozzi, André Péninou, Geoffrey Roman-Jimenez, and Olivier Teste. 2021. Human-interpretable rules for anomaly detection in time-series. In *Proceedings of the International Conference on Extending Database Technology.* 457–462.

[109] Hans-Peter Kriegel, Peer Kröger, Erich Schubert, and Arthur Zimek. 2012. Outlier detection in arbitrarily oriented subspaces. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining.* IEEE, 379–388.

[110] Chia-Tung Kuo and Ian Davidson. 2016. A framework for outlier description using constraint programming. In *Proceedings of the AAAI Conference on Artificial Intelligence.*

[111] Rocco Langone, Alfredo Cuzzocrea, and Nikolaos Skantzos. 2020. Interpretable anomaly prediction: Predicting anomalous behavior in industry 4.0 settings via regularized logistic regression tools. *Data and Knowledge Engineering* 130 (2020), 101850.

[112] Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2019. Unmasking clever hans predictors and assessing what machines really learn. *Nature Communications* 10, 1 (2019), 1–8.

[113] Stefan Leue and Mitra Tabaei Befrouei. 2012. Counterexample explanation by anomaly detection. In *Proceedings of the International SPIN Workshop on Model Checking of Software.* Springer, 24–42.

[114] Efrat Levy, Nadav Maman, Asaf Shabtai, and Yuval Elovici. 2022. AnoMili: Spoofing prevention and explainable anomaly detection for the 1553 military avionic bus. *CoRR* abs/2202.06870 (2022). arXiv:2202.06870

[115] Jiamu Li, Ji Zhang, Jian Wang, Youwen Zhu, Mohamed Jaward Bah, Gaoming Yang, and Yuquan Gan. 2021. VAGA: Towards accurate and interpretable outlier detection based on variational auto-encoder and genetic algorithm for high-dimensional data. In *Proceedings of the 2021 IEEE International Conference on Big Data (Big Data'21).* IEEE, 5956–5958.

[116] Wenkai Li, Wenbo Hu, Ning Chen, and Cheng Feng. 2022. Stacking VAE with graph neural networks for effective and interpretable time series anomaly detection. *AI Open* 3 (2022), 101–110.

[117] Zhihan Li, Youjian Zhao, Jiaqi Han, Ya Su, Rui Jiao, Xidao Wen, and Dan Pei. 2021. Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.* 3220–3230.

[118] Zhihan Li, Youjian Zhao, Rong Liu, and Dan Pei. 2018. Robust and rapid clustering of kpis for large-scale anomaly detection. In *Proceedings of the 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS'18)*. IEEE, 1–10.

[119] Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2022. A survey of transformers. *AI Open* 3 (2022), 111–132.

[120] Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. 2020. Explainable AI: A review of machine learning interpretability methods. *Entropy* 23, 1 (2020), 18.

[121] Zachary C. Lipton. 2018. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue* 16, 3 (2018), 31–57.

[122] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2008. Isolation forest. In *Proceedings of the 2008 8th IEEE International Conference on Data Mining*. IEEE, 413–422.

[123] Ninghao Liu, Donghwa Shin, and Xia Hu. 2018. Contextual outlier interpretation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (Stockholm, Sweden) (IJCAI'18)*. AAAI Press, 2461–2467.

[124] Wei Liu, Yu Zheng, Sanjay Chawla, Jing Yuan, and Xie Xing. 2011. Discovering spatio-temporal causal interactions in traffic data streams. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1010–1018.

[125] Yinxi Liu, Kai Yang, Shaoyu Dou, and Pan Luo. 2022. Interpretable anomaly detection in variable-length co-evolving rhythmic sequences. In *1st Workshop on Artificial Intelligence for Anomalies and Novelties (AI4AN)*.

[126] Philipp Liznerski, Lukas Ruff, Robert A. Vandermeulen, Billy Joe Franks, Marius Kloft, and Klaus-Robert Müller. 2020. Explainable deep one-class classification. In *International Conference on Learning Representations*.

[127] S. Lundberg and S. I. Lee. 2021. A game theoretic approach to explain the output of any machine learning model. Available online at: https://github.com/shap/shap (accessed on 19 May 2022).

[128] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (Long Beach, California, USA) (NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 4768–4777.

[129] Meghanath Macha and Leman Akoglu. 2018. Explaining anomalies in groups with characterizing subspace rules. *Data Mining and Knowledge Discovery* 32, 5 (2018), 1444–1480.

[130] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. 2016. Adversarial autoencoders. *ICLR Workshop track*.

[131] Leonardo Mariani and Fabrizio Pastore. 2008. Automated identification of failure causes in system logs. In *Proceedings of the 2008 19th International Symposium on Software Reliability Engineering (ISSRE'08)*. IEEE, 117–126.

[132] Daniel L. Marino, Chathurika S. Wickramasinghe, Craig Rieger, and Milos Manic. 2022. Self-supervised and interpretable anomaly detection using network transformers. *CoRR* abs/2202.12997 (2022). arXiv:2202.12997

[133] Ioulia Markou, Filipe Rodrigues, and Francisco C. Pereira. 2017. Use of taxi-trip data in analysis of demand patterns for detection and explanation of anomalies. *Transportation Research Record* 2643, 1 (2017), 129–138.

[134] Markos Markou and Sameer Singh. 2003. Novelty detection: A review–part 1: statistical approaches. *Signal Processing* 83, 12 (2003), 2481–2497.

[135] Markos Markou and Sameer Singh. 2003. Novelty detection: A review–part 2: neural network based approaches. *Signal Processing* 83, 12 (2003), 2499–2521.

[136] Thabang Mathonsi and Terence L. van Zyl. 2021. A statistics and deep learning hybrid method for multivariate time series forecasting and mortality modeling. *Forecasting* 4, 1 (2021), 1–25.

[137] R. Daniel Mauldin, William D. Sudderth, and Stanley C. Williams. 1992. Polya trees and random distributions. *The Annals of Statistics* (1992), 1203–1221.

[138] Jacopo Mauro, Michael Nieke, Christoph Seidl, and Ingrid Chieh Yu. 2017. Anomaly detection and explanation in context-aware software product lines. In *Proceedings of the 21st International Systems and Software Product Line Conference-Volume B*. 18–21.

[139] Manuel Mejia-Lavalle. 2010. Outlier detection with innovative explanation facility over a very large financial database. In *Proceedings of the 2010 IEEE Electronics, Robotics and Automotive Mechanics Conference*. IEEE, 23–27.

[140] Milad Memarzadeh, Bryan Matthews, and Thomas Templin. 2022. Multiclass Anomaly Detection in Flight Data Using Semi-Supervised Explainable Deep Learning Model. *Journal of Aerospace Information Systems* 19, 2 (2022), 83–97.

[141] Barbora Micenková, Raymond T. Ng, Xuan-Hong Dang, and Ira Assent. 2013. Explaining outliers by subspace separability. In *Proceedings of the 2013 IEEE 13th International Conference on Data Mining*. IEEE, 518–527.

[142] Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* 267 (2019), 1–38.

[143] Tim Miller. 2021. Contrastive explanation: A structural-model approach. *The Knowledge Engineering Review* 36 (2021), e14.

[144] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. 2017. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition* 65 (2017), 211–222.

[145] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2018. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing* 73 (2018), 1–15.

[146] Brian Morris. 2019. Explainable anomaly and intrusion detection intelligence for platform information technology using dimensionality reduction and ensemble learning. In *Proceedings of the 2019 IEEE AUTOTESTCON*. IEEE, 1–5.

[147] Pavol Mulinka, Pedro Casas, Kensuke Fukuda, and Lukas Kencl. 2020. HUMAN-hierarchical clustering for unsupervised anomaly detection and interpretation. In *Proceedings of the 2020 11th International Conference on Network of the Future (NoF'20)*. IEEE, 132–140.

[148] Emmanuel Müller, Fabian Keller, Sebastian Blanc, and Klemens Böhm. 2012. OutRules: A framework for outlier descriptions in multiple context spaces. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 828–832.

[149] Emmanuel Müller, Matthias Schiffer, and Thomas Seidl. 2011. Statistical selection of relevant subspace projections for outlier ranking. In *Proceedings of the 2011 IEEE 27th International Conference on Data Engineering*. IEEE, 434–445.

[150] W. James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. 2019. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences* 116, 44 (2019), 22071–22080.

[151] Quoc Phong Nguyen, Kar Wai Lim, Dinil Mon Divakaran, Kian Hsiang Low, and Mun Choon Chan. 2019. Gee: A gradient-based explainable variational autoencoder for network anomaly detection. In *Proceedings of the 2019 IEEE Conference on Communications and Network Security (CNS'19)*. IEEE, 91–99.

[152] Ahmad Kamal Mohd Nor, Srinivasa Rao Pedapati, and Masdi Muhammad. 2021. Application of explainable AI (XAI) for anomaly detection and prognostic of gas turbines with uncertainty quantifica-tion. Preprints (September 2021). https://doi.org/10.20944/preprints202109.0034.v1

[153] Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. 2019. Interpretml: A unified framework for machine learning interpretability. *CoRR* abs/1909.09223 (2019). arXiv:1909.09223

[154] Keith Noto, Carla Brodley, and Donna Slonim. 2012. FRaC: A feature-modeling approach for semi-supervised and unsupervised anomaly detection. *Data Mining and Knowledge Discovery* 25, 1 (2012), 109–133.

[155] Guansong Pang and Charu Aggarwal. 2021. Toward explainable deep anomaly detection. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4056–4057.

[156] Guansong Pang, Longbing Cao, and Ling Chen. 2016. Outlier detection in complex categorical data by modelling the feature value couplings. In *Proceedings of the IJCAI International Joint Conference on Artificial Intelligence*.

[157] Guansong Pang, Longbing Cao, Ling Chen, Defu Lian, and Huan Liu. 2018. Sparse modeling-based sequential ensemble learning for effective outlier detection in high-dimensional numeric data. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*.

[158] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. 2016. Unsupervised feature selection for outlier detection by modelling hierarchical value-feature couplings. In *Proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM'16)*. IEEE, 410–419.

[159] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. 2017. Learning homophily couplings from Non-IID data for joint feature selection and noise-resilient outlier detection. In *Proceedings of the IJCAI*. 2585–2591.

[160] Guansong Pang, Choubo Ding, Chunhua Shen, and Anton van den Hengel. 2021. Explainable deep few-shot anomaly detection with deviation networks. *CoRR* abs/2108.00462 (2021). arXiv:2108.00462

[161] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. 2021. Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–38.

[162] Egawati Panjei, Le Gruenwald, Eleazar Leal, Christopher Nguyen, and Shejuti Silvia. 2022. A survey on outlier explanations. *The VLDB Journal* 31 (2022), 977–1008.

[163] Cheong Hee Park and Jiil Kim. 2021. An explainable outlier detection method using region-partition trees. *The Journal of Supercomputing* 77, 3 (2021), 3062–3076.

[164] Sungwoo Park, Jihoon Moon, and Eenjun Hwang. 2020. Explainable anomaly detection for district heating based on shapley additive explanations. In *Proceedings of the 2020 International Conference on Data Mining Workshops (ICDMW'20)*. IEEE, 762–765.

[165] Animesh Patcha and Jung-Min Park. 2007. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks* 51, 12 (2007), 3448–3470.

[166] Aum Patil, Amey Wadekar, Tanishq Gupta, Rohit Vijan, and Faruk Kazi. 2019. Explainable LSTM model for anomaly detection in HDFS log file using layerwise relevance propagation. In *Proceedings of the 2019 IEEE Bombay Section Signature Conference (IBSSC'19)*. IEEE, 1–6.

[167] Heiko Paulheim and Robert Meusel. 2015. A decomposition of the outlier detection problem into a set of supervised learning problems. *Machine Learning* 100, 2 (2015), 509–531.

[168] Deysy Galeana Perez and Manuel Mejia Lavalle. 2011. Outlier detection applying an innovative user transaction modeling with automatic explanation. In *Proceedings of the 2011 IEEE Electronics, Robotics, and Automotive Mechanics Conference*. IEEE, 41–46.

[169] Tomáš Pevnỳ. 2016. Loda: Lightweight on-line detector of anomalies. *Machine Learning* 102, 2 (2016), 275–304.

[170] Tomáš Pevnỳ and Martin Kopp. 2014. Explaining anomalies with sapling random forests. In *Proceedings of the Information Technologies-Applications and Theory Workshops, Posters, and Tutorials (ITAT'14)*.

[171] Sreeraj Rajendran, Wannes Meert, Vincent Lenders, and Sofie Pollin. 2018. SAIFE: Unsupervised wireless spectrum anomaly detection with interpretable features. In *Proceedings of the 2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN'18)*. IEEE, 1–9.

[172] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. 2000. Efficient algorithms for mining outliers from large data sets. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*. 427–438.

[173] Erica Ramirez, Markus Wimmer, and Martin Atzmueller. 2019. A computational framework for interpretable anomaly detection and classification of multivariate time series with application to human gait data analysis. In *Proceedings of the Artificial Intelligence in Medicine: Knowledge Representation and Transparent and Explainable Systems*. Springer, 132–147.

[174] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1135–1144.

[175] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[176] Konrad Rieck and Pavel Laskov. 2009. Visualization and explanation of payload-based anomaly detection. In *Proceedings of the 2009 European Conference on Computer Network Defense*. IEEE, 29–36.

[177] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 234–241.

[178] Khushnaseeb Roshan and Aasim Zafar. 2021. Utilizing XAI technique to improve autoencoder based model for computer network anomaly detection with shapley additive explanation (SHAP). *International Journal of Computer Networks & Communications* 13 (2021), 1–20.

[179] Daniel M. Roy and Yee Whye Teh. 2008. The mondrian process. In *Proceedings of the NIPS*, Vol. 21.

[180] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (2019), 206–215.

[181] Lukas Ruff, Jacob R. Kauffmann, Robert A. Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G. Dietterich, and Klaus-Robert Müller. 2021. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE* (2021).

[182] Mao Saeki, Jun Ogata, Masahiro Murakawa, and Tetsuji Ogawa. 2019. Visual explanation of neural network based rotation machinery anomaly detection system. In *Proceedings of the 2019 IEEE International Conference on Prognostics and Health Management (ICPHM'19)*. IEEE, 1–4.

[183] Hojjat Salehinejad, Sharan Sankar, Joseph Barfett, Errol Colak, and Shahrokh Valaee. 2017. Recent advances in recurrent neural networks. *CoRR* abs/1801.01078 (2018). arXiv:1801.01078

[184] Durgesh Samariya, Sunil Aryal, Kai Ming Ting, and Jiangang Ma. 2020. A new effective and efficient measure for outlying aspect mining. In *Proceedings of the International Conference on Web Information Systems Engineering*. Springer, 463–474.

[185] Durgesh Samariya, Jiangang Ma, Sunil Aryal, and Kai Ming Ting. 2020. A comprehensive survey on outlying aspect mining methods. *CoRR* abs/2005.02637 (2020). arXiv:2005.02637

[186] Cetin Savkli and Catherine Schwartz. 2021. Random subspace mixture models for interpretable anomaly detection. *CoRR* abs/2108.06283 (2021). arXiv:2108.06283

[187] Thomas Schlegl, Stefan Schlegl, Nikolai West, and Jochen Deuse. 2021. Scalable anomaly detection in manufacturing systems using an interpretable deep learning approach. *Procedia CIRP* 104 (2021), 1547–1552.

[188] Bernhard Schölkopf, Robert C. Williamson, Alex Smola, John Shawe-Taylor, and John Platt. 1999. Support vector method for novelty detection. In *Proceedings of the 12th International Conference on Neural Information Processing Systems (Denver, CO) (NIPS'99, Vol. 12)*. MIT Press, Cambridge, MA, USA, 582–588.

[189] Jonas Herskind Sejr and Anna Schneider-Kamp. 2021. Explainable outlier detection: What, for whom and why? *Machine Learning with Applications* 6 (2021), 100172.

[190] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*. 618–626.

[191] 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. In *Proceedings of the EDBT*. 481–492.

[192] Oscar Serradilla, Ekhi Zugasti, Julian Ramirez de Okariz, Jon Rodriguez, and Urko Zurutuza. 2021. Adaptable and explainable predictive maintenance: Semi-supervised deep learning for anomaly detection and diagnosis in press machine data. *Applied Sciences* 11, 16 (2021), 7376.

[193] Md Amran Siddiqui, Alan Fern, Thomas G. Dietterich, and Weng-Keen Wong. 2019. Sequential feature explanations for anomaly detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 13, 1 (2019), 1–22.

[194] Md Amran Siddiqui, Jack W. Stokes, Christian Seifert, Evan Argyle, Robert McCann, Joshua Neil, and Justin Carroll. 2019. Detecting cyber attacks using anomaly detection with explanations and expert feedback. In *Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'19)*. IEEE, 2872–2876.

[195] Arno Siebes, Jilles Vreeken, and Matthijs van Leeuwen. 2006. Item sets that compress. In *Proceedings of the 2006 SIAM International Conference on Data Mining*. SIAM, 395–406.

[196] John Sipple. 2020. Interpretable, multidimensional, multimodal anomaly detection with negative sampling for detection of device failure. In *Proceedings of the International Conference on Machine Learning*. PMLR, 9016–9025.

[197] John Sipple and Abdou Youssef. 2022. A general-purpose method for applying explainable AI for anomaly detection. In *Proceedings of the Foundations of Intelligent Systems: 26th International Symposium, ISMIS 2022*. Springer, 162–174.

[198] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. 2020. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 180–186.

[199] Giulia Slavic, Pablo Marin, David Martin, Lucio Marcenaro, and Carlo Regazzoni. 2021. Interpretable anomaly detection using a generalized Markov Jump particle filter. In *Proceedings of the 2021 IEEE International Conference on Autonomous Systems (ICAS'21)*. IEEE, 1–5.

[200] Koen Smets and Jilles Vreeken. 2011. The odd one out: Identifying and characterising anomalies. In *Proceedings of the 2011 SIAM International Conference on Data Mining*. SIAM, 804–815.

[201] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. 2017. Smoothgrad: Removing noise by adding noise. In *ICML Workshop on Visualization for Deep Learning ICML*.

[202] Grégory Smits, Marie-Jeanne Lesot, Véronne Yepmo Tchaghe, and Olivier Pivert. 2022. PANDA: Human-in-the-loop anomaly detection and explanation. In *Proceedings of the Information Processing and Management of Uncertainty in Knowledge-Based Systems: 19th International Conference, IPMU 2022*. Springer, 720–732.

[203] Fei Song, Yanlei Diao, Jesse Read, Arnaud Stiegler, and Albert Bifet. 2018. EXAD: A system for explainable anomaly detection on big data traces. In *Proceedings of the 2018 IEEE International Conference on Data Mining Workshops (ICDMW'18)*. 1435–1440. DOI : https://doi.org/10.1109/ICDMW.2018.00204

[204] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2014. Striving for simplicity: The all convolutional net. In *ICLR workshop track*.

[205] Mukund Sundararajan and Amir Najmi. 2020. The many Shapley values for model explanation. In *Proceedings of the International Conference on Machine Learning*. PMLR, 9269–9278.

[206] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the International Conference on Machine Learning*. PMLR, 3319–3328.

[207] Stanislaw Szymanowicz, James Charles, and Roberto Cipolla. 2021. X-MAN: Explaining multiple sources of anomalies in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3224–3232.

[208] Stanislaw Szymanowicz, James Charles, and Roberto Cipolla. 2022. Discrete neural representations for explainable anomaly detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 148–156.

[209] Naoya Takeishi. 2019. Shapley values of reconstruction errors of pca for explaining anomaly detection. In *Proceedings of the 2019 International Conference on Data Mining Workshops (ICDMW'19)*. IEEE, 793–798.

[210] Jinying Zou and Ovanes L. Petrosian. 2020. Explainable AI: Using Shapley Value to Explain Complex Anomaly Detection ML-Based Systems. In *Machine Learning and Artificial Intelligence - Proceedings of MLIS 2020, Virtual Event, October 25-28, 2020 (Frontiers in Artificial Intelligence and Applications)*, Vol. 332. IOS Press, 152–164.

[211] Sarah Tan, Matvey Soloviev, Giles Hooker, and Martin T. Wells. 2020. Tree space prototypes: Another look at making tree ensembles interpretable. In *Proceedings of the 2020 ACM-IMS on Foundations of Data Science Conference*. 23–34.

[212] Arijit Ukil, Soma Bandyoapdhyay, Chetanya Puri, and Arpan Pal. 2016. IoT healthcare analytics: The importance of anomaly detection. In *Proceedings of the 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA'16)*. IEEE, 994–997.

[213] Karel Vaculík and Luboš Popelínský. 2016. DGRMiner: Anomaly detection and explanation in dynamic graphs. In *Proceedings of the International Symposium on Intelligent Data Analysis*. Springer, 308–319.

[214] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, 11 (2008).

[215] Shashanka Venkataramanan, Kuan-Chuan Peng, Rajat Vikram Singh, and Abhijit Mahalanobis. 2020. Attention guided anomaly localization in images. In *Proceedings of the European Conference on Computer Vision*. Springer, 485–503.

[216] Nguyen Xuan Vinh, Jeffrey Chan, Simone Romano, James Bailey, Christopher Leckie, Kotagiri Ramamohanarao, and Jian Pei. 2016. Discovering outlying aspects in large datasets. *Data Mining and Knowledge Discovery* 30, 6 (2016), 1520–1555.

[217] Paul Voigt and Axel Von dem Bussche. 2017. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing* 10, 3152676 (2017), 10–5555.

[218] Laura von Rueden, Sebastian Mayer, Katharina Beckh, Bogdan Georgiev, Sven Giesselbach, Raoul Heese, Birgit Kirsch, Julius Pfrommer, Annika Pick, Rajkumar Ramamurthy, Michal Walczak, Jochen Garcke, Christian Bauckhage, and Jannis Schuecker. 2023. Informed machine learning – A taxonomy and survey of integrating prior knowledge into learning systems. *IEEE Transactions on Knowledge and Data Engineering* 35, 1 (2023), 614–633.

[219] Chongke Wu, Sicong Shao, Cihan Tunc, Pratik Satam, and Salim Hariri. 2022. An explainable and efficient deep learning framework for video anomaly detection. *Cluster Computing* 25 (2022), 2715–2737.

[220] Tung-Yu Wu and You-Ting Wang. 2021. Locally interpretable one-class anomaly detection for credit card fraud detection. In *Proceedings of the 2021 International Conference on Technologies and Applications of Artificial Intelligence (TAAI'21)*. IEEE, 25–30.

[221] Hongzuo Xu, Yijie Wang, Songlei Jian, Zhenyu Huang, Yongjun Wang, Ning Liu, and Fei Li. 2021. Beyond outlier detection: Outlier interpretation by attention-guided triplet deviation network. In *Proceedings of the Web Conference 2021*. 1328–1339.

[222] Wei Xu, Ling Huang, Armando Fox, David Patterson, and Michael I. Jordan. 2009. Detecting large-scale system problems by mining console logs. In *Proceedings of the ACM SIGOPS 22nd Symposium on Operating Systems Principles*. 117–132.

[223] Qibo Yang, Jaskaran Singh, and Jay Lee. 2019. Isolation-based feature selection for unsupervised outlier detection. In *Proceedings of the Annual Conference of the PHM Society*.

[224] Emmanuel Yashchin. 1993. Performance of CUSUM control schemes for serially correlated observations. *Technometrics* 35, 1 (1993), 37–52.

[225] Véronne Yepmo, Grégory Smits, and Olivier Pivert. 2022. Anomaly explanation: A review. *Data and Knowledge Engineering* 137 (2022), 101946.

[226] Luca Zancato, Alessandro Achille, Giovanni Paolini, Alessandro Chiuso, and Stefano Soatto. 2021. STRIC: Stacked residuals of interpretable components for time series anomaly detection. *CoRR* abs/2202.12457 (2021). arXiv:2202.12457

[227] Matthew D. Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision*. Springer, 818–833.

[228] Ji Zhang, Meng Lou, Tok Wang Ling, and Hai Wang. 2004. HOS-miner: A system for detecting outlying subspaces of high-dimensional data. In *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB'04)*. Morgan Kaufmann Publishers Inc., 1265–1268.

[229] Jianlong Zhou, Amir H. Gandomi, Fang Chen, and Andreas Holzinger. 2021. Evaluating the quality of machine learning explanations: A survey on methods and metrics. *Electronics* 10, 5 (2021), 593.

[230] Yingying Zhu, Nandita M. Nayak, and Amit K. Roy-Chowdhury. 2012. Context-aware activity recognition and anomaly detection in video. *IEEE Journal of Selected Topics in Signal Processing* 7, 1 (2012), 91–101.

[231] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. 2012. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 5, 5 (2012), 363–387.

[232] Hui Zou and Trevor Hastie. 2005. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67, 2 (2005), 301–320.