3-2022

Deep learning for anomaly detection: A review

Guansong PANG
Singapore Management University, gspang@smu.edu.sg

Chunhua SHEN
University of Adelaide

Longbing CAO
University of Technology Sydney

Anton Van Den HENGEL
University of Adelaide

Deep Learning for Anomaly Detection: A Review

GUANSONG PANG and CHUNHUA SHEN, University of Adelaide
LONGBING CAO, University of Technology Sydney
ANTON VAN DEN HENGEL, University of Adelaide

Anomaly detection, a.k.a. outlier detection or novelty detection, has been a lasting yet active research area in various research communities for several decades. There are still some unique problem complexities and challenges that require advanced approaches. In recent years, deep learning enabled anomaly detection, i.e., *deep anomaly detection*, has emerged as a critical direction. This article surveys the research of deep anomaly detection with a comprehensive taxonomy, covering advancements in 3 high-level categories and 11 fine-grained categories of the methods. We review their key intuitions, objective functions, underlying assumptions, advantages, and disadvantages and discuss how they address the aforementioned challenges. We further discuss a set of possible future opportunities and new perspectives on addressing the challenges.

CCS Concepts: • **Computing methodologies** → **Anomaly detection**; **Machine learning**; **Scene anomaly detection**; **Neural networks**; • **Security and privacy** → **Intrusion/anomaly detection and malware mitigation**;

Additional Key Words and Phrases: Anomaly detection, deep learning, outlier detection, novelty detection, one-class classification

ACM Reference format:
Guansong Pang, Chunhua Shen, Longbing Cao, and Anton van den Hengel. 2021. Deep Learning for Anomaly Detection: A Review. *ACM Comput. Surv.* 54, 2, Article 38 (March 2021), 38 pages.
https://doi.org/10.1145/3439950

1 INTRODUCTION

Anomaly detection, a.k.a. outlier detection or novelty detection, is referred to as the process of detecting data instances that significantly deviate from the majority of data instances. Anomaly detection has been an active research area for several decades, with early exploration dating back as far as the 1960s [52]. Due to the increasing demand and applications in broad domains, such as risk management, compliance, security, financial surveillance, health and medical risk, and AI safety, anomaly detection plays increasingly important roles, highlighted in various communities including data mining, machine learning, computer vision, and statistics. In recent years, deep learning has shown tremendous capabilities in learning expressive representations of complex data such as high-dimensional data, temporal data, spatial data, and graph data, pushing the

Authors' addresses: G. Pang (corresponding author), C. Shen, and A. van den Hengel, University of Adelaide, Adelaide, South Australia, 5005; emails: pangguansong@gmail.com, {chunhua.shen, anton.vandenhengel}@adelaide.edu.au; L. Cao, University of Technology Sydney, Sydney, New South Wales, 2007; email: longbing.cao@uts.edu.au.

0360-0300/2021/03-ART38 $15.00
https://doi.org/10.1145/3439950

boundaries of different learning tasks. Deep learning for anomaly detection, *deep anomaly detection* for short, aim at learning feature representations or anomaly scores via neural networks for the sake of anomaly detection. A large number of deep anomaly detection methods have been introduced, demonstrating significantly better performance than conventional anomaly detection on addressing challenging detection problems in a variety of real-world applications. This work aims to provide a comprehensive review of this area. We first discuss the problem nature of anomaly detection and major largely unsolved challenges, then systematically review the current deep methods and their capabilities in addressing these challenges, and finally presents a number of future opportunities.

As a popular area, a number of studies [2, 4, 16, 28, 53, 62, 178] have been dedicated to the categorization and review of anomaly detection techniques. However, they all focus on conventional anomaly detection methods only. One work closely related to ours is Reference [26]. It presents a good summary of a number of real-world applications of deep anomaly detection, but only provides some very high-level outlines of selective categories of the techniques, from which it is difficult, if not impossible, to gain the sense of the approaches taken by the current methods and their underlying intuitions. By contrast, this review delineates the formulation of current deep detection methods to gain key insights about their intuitions, inherent capabilities and weakness on addressing some largely unsolved challenges in anomaly detection. This forms a deep understanding of the problem nature and the state of the art, and brings about genuine open opportunities. It also helps explain why we need deep anomaly detection.

In summary, this work makes the following five major contributions:

- *Problem nature and challenges.* We discuss some unique problem complexities underlying anomaly detection and the resulting largely unsolved challenges.
- *Categorization and formulation.* We formulate the current deep anomaly detection methods into three principled frameworks: deep learning for generic feature extraction, learning representations of normality, and end-to-end anomaly score learning. A hierarchical taxonomy is presented to categorize the methods based on 11 different modeling perspectives.
- *Comprehensive literature review.* We review a large number of relevant studies in leading conferences and journals of several relevant communities, including machine learning, data mining, computer vision and artificial intelligence, to present a comprehensive literature review of the research progress. To provide an in-depth introduction, we delineate the basic assumptions, objective functions, key intuitions and their capabilities in addressing some of the aforementioned challenges by all categories of the methods.
- *Future opportunities.* We further discuss a set of possible future opportunities and their implication to addressing relevant challenges.
- *Source codes and datasets.* We solicit a collection of publicly accessible source codes of nearly all categories of methods and a large number of real-world datasets with *real anomalies* to offer some empirical comparison benchmarks.

2 ANOMALY DETECTION: PROBLEM COMPLEXITIES AND CHALLENGES

Owing to the unique nature, anomaly detection presents distinct problem complexities from the majority of analytical and learning problems and tasks. This section summarizes such intrinsic complexities and unsolved detection challenges in complex anomaly data.

2.1 Major Problem Complexities

Unlike those problems and tasks on majority, regular or evident patterns, anomaly detection addresses minority, unpredictable/uncertain and rare events, leading to some unique problem complexities to all (both deep and shallow) detection methods:

- **Unknownness.** Anomalies are associated with many unknowns, e.g., instances with unknown abrupt behaviors, data structures, and distributions. They remain unknown until actually occur, such as novel terrorist attacks, fraud, and network intrusions.
- **Heterogeneous anomaly classes.** Anomalies are irregular, and thus, one class of anomalies may demonstrate completely different abnormal characteristics from another class of anomalies. For example, in video surveillance, the abnormal events robbery, traffic accidents and burglary are visually highly different.
- **Rarity and class imbalance.** Anomalies are typically rare data instances, contrasting to normal instances that often account for an overwhelming proportion of the data. Therefore, it is difficult, if not impossible, to collect a large amount of labeled abnormal instances. This results in the unavailability of large-scale labeled data in most applications. The class imbalance is also due to the fact that misclassification of anomalies is normally much more costly than that of normal instances.
- **Diverse types of anomaly.** Three completely different types of anomaly have been explored [28]. *Point anomalies* are individual instances that are anomalous w.r.t. the majority of other individual instances, e.g., the abnormal health indicators of a patient. *Conditional anomalies*, a.k.a. contextual anomalies, also refer to individual anomalous instances but in a specific context, i.e., data instances are anomalous in the specific context, otherwise normal. The contexts can be highly different in real-world applications, e.g., sudden temperature drop/increase in a particular temporal context, or rapid credit card transactions in unusual spatial contexts. *Group anomalies*, a.k.a. collective anomalies, are a subset of data instances anomalous as a whole w.r.t. the other data instances; the individual members of the collective anomaly may not be anomalies, e.g., exceptionally dense subgraphs formed by fake accounts in social network are anomalies as a collection, but the individual nodes in those subgraphs can be as normal as real accounts.

2.2 Main Challenges Tackled by Deep Anomaly Detection

The above complex problem nature leads to a number of detection challenges. Some challenges, such as scalability w.r.t. data size, have been well addressed in recent years, while the following are largely unsolved, to which deep anomaly detection can play some essential roles.

- **CH1: Low anomaly detection recall rate.** Since anomalies are highly rare and heterogeneous, it is difficult to identify all of the anomalies. Many normal instances are wrongly reported as anomalies while true yet sophisticated anomalies are missed. Although a plethora of anomaly detection methods have been introduced over the years, the current state-of-the-art methods, especially unsupervised methods (e.g., References [17, 84]), still often incur high false positives on real-world datasets [20, 115]. How to reduce false positives and enhance detection recall rates is one of the most important and yet difficult challenges, particularly for the significant expense of failing to spotting anomalies.
- **CH2: Anomaly detection in high-dimensional and/or not-independent data.** Anomalies often exhibit evident abnormal characteristics in a low-dimensional space yet become hidden and unnoticeable in a high-dimensional space. High-dimensional anomaly detection has been a long-standing problem [178]. Performing anomaly detection in a reduced lower-dimensional space spanned by a small subset of original features or newly constructed features is a straightforward solution, e.g., in subspace-based [70, 77, 84, 123] and feature selection-based methods [12, 109, 111]. However, identifying intricate (e.g., high-order, nonlinear and heterogeneous) feature interactions and couplings [22] may be essential in high-dimensional data, but it remains a major challenge for anomaly detection.

Further, how to guarantee the new feature space preserved proper information for specific detection methods is critical to downstream accurate anomaly detection, but it is challenging due to the aforementioned unknowns and heterogeneities of anomalies. Also, it is challenging to detect anomalies from instances that may be dependent on each other such as by temporal, spatial, graph-based and other interdependency relationships [2, 4, 22, 53].

- **CH3: Data-efficient learning of normality/abnormality.** Due to the difficulty and cost of collecting large-scale labeled anomaly data, *fully supervised anomaly detection* is often impractical as it assumes the availability of labeled training data with both normal and anomaly classes. In the last decade, major research efforts have been focused on *unsupervised anomaly detection* that does not require any labeled training data. However, unsupervised methods do not have any prior knowledge of true anomalies. They rely heavily on their assumption on the distribution of anomalies. However, it is often not difficult to collect labeled normal data and some labeled anomaly data. In practice, it is often suggested to leverage such readily accessible labeled data as much as possible [2]. Thus, utilizing those labeled data to learn expressive representations of normality/abnormality is crucial for accurate anomaly detection. *Semi-supervised anomaly detection*, which assumes a set of labeled normal training data,[1] is a research direction dedicated to this problem. Another research line is *weakly supervised anomaly detection* that assumes we have some labels for anomaly classes yet the class labels are partial/incomplete (i.e., they do not span the entire set of anomaly class), inexact (i.e., coarse-grained labels), or inaccurate (i.e., some given labels can be incorrect). Two major challenges are how to learn expressive normality/abnormality representations with a small amount of labeled anomaly data and how to learn detection models that are generalized to novel anomalies uncovered by the given labeled anomaly data.
- **CH4: Noise-resilient anomaly detection.** Many weakly/semi-supervised anomaly detection methods assume the labeled training data are clean, which can be vulnerable to noisy instances that are mistakenly labeled as an opposite class label. In such cases, we may use unsupervised methods instead, but this fails to utilize the genuine labeled data. Additionally, there often exists large-scale anomaly-contaminated unlabeled data. Noise-resilient models can leverage those unlabeled data for more accurate detection. Thus, the noise here can be either mislabeled data or unlabeled anomalies. The main challenge is that the amount of noises can differ significantly from datasets and noisy instances may be irregularly distributed in the data space.
- **CH5: Detection of complex anomalies.** Most of existing methods are for point anomalies, which cannot be used for conditional anomaly and group anomaly, since they exhibit completely different behaviors from point anomalies. One main challenge here is to incorporate the concept of conditional/group anomalies into anomaly measures/models. Also, current methods mainly focus on detect anomalies from single data sources, while many applications require the detection of anomalies with multiple heterogeneous data sources, e.g., multidimensional data, graph, image, text, and audio data. One main challenge is that some anomalies can be detected only when considering two or more data sources.
- **CH6: Anomaly explanation.** In many safety-critical domains there may be some major risks if anomaly detection models are directly used as black-box models. For example, the *rare* data instances reported as anomalies may lead to possible algorithmic bias against the

[1]There have been some studies that refer the methods trained with purely normal training data to be unsupervised approach. However, this setting is different from the general sense of an unsupervised setting. To avoid unnecessary confusion, following References [2, 28], these methods are referred to as semi-supervised methods hereafter.

Table 1. Deep Learning Methods vs. Traditional Methods in Anomaly Detection

Method	End-to-end Optimization	Tailored Representation Learning	Intricate Relation Learning	Heterogeneity Handling
Traditional	×	×	Weak	Weak
Deep	✓	✓	Strong	Strong
Challenges	CH1-6	CH1-6	CH1, CH2, CH3, CH5	CH3, CH5

minority groups presented in the data, such as under-represented groups in fraud detection and crime detection systems. An effective approach to mitigate this type of risk is to have anomaly explanation algorithms that provide straightforward clues about why a specific data instance is identified as anomaly. Human experts can then look into and correct the bias. Providing such explanation can be as important as detection accuracy in some applications. However, most anomaly detection studies focus on detection accuracy only, ignoring the capability of providing explanation of the identified anomalies. To derive anomaly explanation from specific detection methods is still a largely unsolved problem, especially for complex models. Developing inherently interpretable anomaly detection models is also crucial, but it remains a main challenge to well balance the model's interpretability and effectiveness.

Deep methods enable end-to-end optimization of the whole anomaly detection pipeline, and they also enable the learning of representations specifically tailored for anomaly detection. These two capabilities are crucial to tackle the above six challenges, but traditional methods do not have. Particularly they help largely improve the utilization of labeled normal data or some labeled anomaly data regardless of the data type, reducing the need of large-scale labeled data as in fully supervised settings (CH2, CH3, CH4, and CH5). This subsequently results in more informed models and thus better recall rate (CH1). For the anomaly explanation challenge, although deep methods are often black-box models, they offer options to unify anomaly detection and explanation into single frameworks, resulting in more genuine explanation of the anomalies spotted by specific models (see Section 8.5). Deep methods also excel at learning intricate structures and relations from diverse types of data, such as high-dimensional data, image data, video data, graph data, and so on. This capability is important to address various challenges, such as CH1, CH2, CH3, and CH5. Further, they offer many effective and easy-to-use network architectures and principled frameworks to seamlessly learn unified representations of heterogeneous data sources. This empowers the deep models to tackle some key challenges such as CH3 and CH5. Although there are shallow methods for handling those complex data, they are generally substantially weaker and less adaptive than the deep methods. A summary of this discussion is presented in Table 1.

3 ADDRESSING THE CHALLENGES WITH DEEP ANOMALY DETECTION

3.1 Preliminaries

Deep neural networks leverage complex compositions of linear/non-linear functions that can be represented by a computational graph to learn expressive representations [49]. Two basic building blocks of deep learning are activation functions and layers. *Activation functions* determine the output of computational graph nodes (i.e., neurons in neural networks) given some inputs. They can be linear or non-linear functions. Some popular activation functions include linear, sigmoid, tanh, Rectified Linear Unit (ReLU) and its variants. A *layer* in neural networks refers to a set of neurons stacked in some forms. Commonly used layers include fully connected, convolutional and pooling, and recurrent layers. These layers can be leveraged to build different popular neural

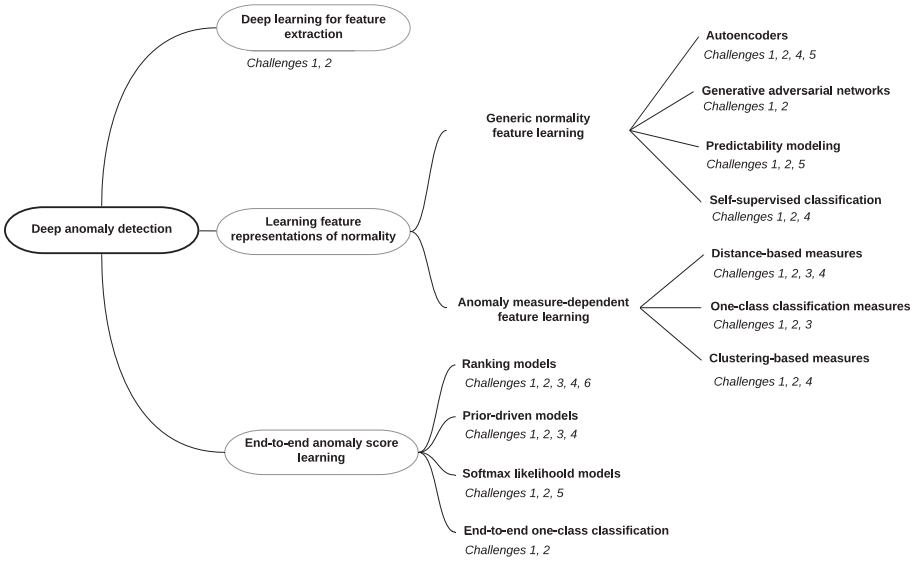

Fig. 1. The proposed taxonomy of current deep anomaly detection techniques. The detection challenges that each category of methods can address are also presented.

networks. For example, multilayer perceptron (MLP) networks are composed by fully connected layers, convolutional neural networks (CNN) are featured by varying groups of convolutional and pooling layers, and recurrent neural networks (RNN), e.g., vanilla RNN, gated recurrent units, and long short term memory (LSTM), are built upon recurrent layers. See Reference [49] for detailed introduction of these neural networks.

Given a dataset $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ with $\mathbf{x}_i \in \mathbb{R}^D$, let $\mathcal{Z} \in \mathbb{R}^K$ ($K \ll N$) be a representation space, then **deep anomaly detection** aims at learning a feature representation mapping function $\phi(\cdot) : \mathcal{X} \mapsto \mathcal{Z}$ or an anomaly score learning function $\tau(\cdot) : \mathcal{X} \mapsto \mathbb{R}$ in a way that anomalies can be easily differentiated from the normal data instances in the space yielded by the ϕ or τ function, where both ϕ and τ are a neural network-enabled mapping function with $H \in \mathbb{N}$ hidden layers and their weight matrices $\Theta = \{\mathbf{M}^1, \mathbf{M}^2, \ldots, \mathbf{M}^H\}$. In the case of learning the feature mapping $\phi(\cdot)$, an additional step is required to calculate the anomaly score of each data instance in the new representation space, while $\tau(\cdot)$ can directly infer the anomaly scores with raw data inputs. Larger τ outputs indicate greater degree of being anomalous.

3.2 Categorization of Deep Anomaly Detection

To have a thorough understanding of the area, we introduce a hierarchical taxonomy to classify deep anomaly detection methods into three main categories and 11 fine-grained categories from the modeling perspective. An overview of the taxonomy of the methods is shown in Figure 1. Specifically, deep anomaly detection consists of three conceptual paradigms: *Deep Learning for Feature Extraction*, *Learning Feature Representations of Normality*, and *End-to-end Anomaly Score Learning*.

The procedure of these three paradigms is presented in Figure 2. As shown in Figure 2(a), deep learning and anomaly detection are fully separated in the first main category (Section 4), so deep learning techniques are used as some independent feature extractors only. The two modules are dependent on each other in some form in the second main category (Section 5) presented

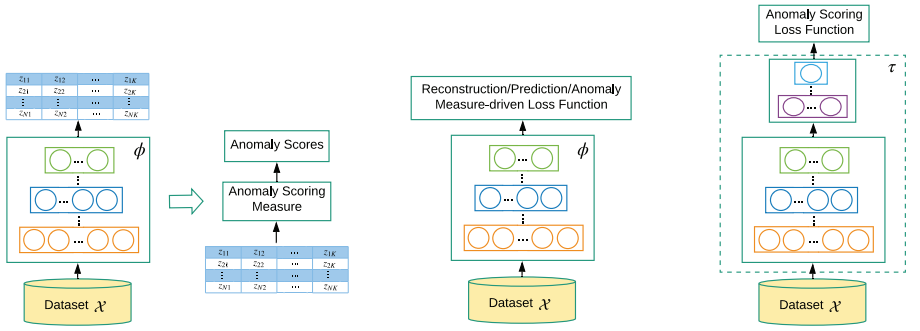

Fig. 2. Conceptual frameworks of three main deep anomaly detection approaches.

in Figure 2(b), with an objective of learning expressive representations of normality. This category of methods can be further divided into two subcategories based on how the representations are learned, i.e., whether using existing shallow anomaly measures (e.g., distance- and clustering-based measures) to guide the learning or not. These two subcategories encompass seven fine-grained categories of methods, with each category taking a different approach to formulate its objective function. The two modules are fully unified in the third main category (Section 6) presented in Figure 2(c), in which the methods are dedicated to learning anomaly scores via neural networks in an end-to-end fashion. This category is further broken down into four subcategories based on the formulation of the anomaly scoring learning. In the following three sections we review these three paradigms in detail.

4 DEEP LEARNING FOR FEATURE EXTRACTION

This category of methods aims at leveraging deep learning to extract low-dimensional feature representations from high-dimensional and/or non-linearly separable data for downstream anomaly detection. The feature extraction and the anomaly scoring are fully disjointed and independent from each other. Thus, the deep learning components work purely as dimensionality reduction only. Formally, the approach can be represented as

$$\mathbf{z} = \phi(\mathbf{x}; \Theta), \tag{1}$$

where $\phi : \mathcal{X} \mapsto \mathcal{Z}$ is a deep neural network-based feature mapping function, with $\mathcal{X} \in \mathbb{R}^D$, $\mathcal{Z} \in \mathbb{R}^K$ and normally $D \gg K$. An anomaly scoring method f that has no connection to the feature mapping ϕ is then applied onto the new space to calculate anomaly scores.

Compared to the dimension reduction methods that are popular in anomaly detection, such as principal component analysis [21, 140, 180] and random projection [80, 112, 123], deep learning techniques have been demonstrating substantially better capability in extracting semantic-rich features and non-linear feature relations [14, 49].

Assumptions. The feature representations extracted by deep learning models preserve the discriminative information that helps separate anomalies from normal instances.

One research line is to directly uses popular pre-trained deep learning models, such as AlexNet [75], VGG [143], and ResNet [58], to extract low-dimensional features. This line is explored in anomaly detection in complex high-dimensional data such as image data and video data. One interesting work of this line is the unmasking framework for online anomaly detection [66]. The key idea is to iteratively train a binary classifier to separate one set of video frames from its subsequent video frames in a sliding window, with the most discriminant features removed in each

iteration step. This is analogous to an unmasking process. The framework assumes the first set of video frames as normal and evaluates its separability from its subsequent video frames. Thus, the training classification accuracy is expected to be high if the subsequent video frames are abnormal, and low otherwise. Clearly the power of the unmasking framework relies heavily on the quality of the features, so it is essential to have quality features to represent the video frames. The VGG model pre-trained on the ILSVRC benchmark [134] is shown to be effective to extract expressive appearance features for this purpose [66]. In Reference [88], the masking framework is formulated as a two-sample test task to understand its theoretical foundation. They also show that using features extracted from a dynamically updated sampling pool of video frames is found to improve the performance of the framework. Additionally, similar to many other tasks, the feature representations extracted from the deep models pre-trained on a source dataset can be transferred to fine-tune anomaly detectors on a target dataset. As shown in Reference [6], one-class support vector machines (SVM) can be first initialized with the VGG models pre-trained on ILSVRC and then fine-tuned to improve anomaly classification on the MNIST data [78]. Similarly, the ResNet models pre-trained on MNIST can empower abnormal event detection in various video surveillance datasets [117, 176].

Another research line in this category is to explicitly train a deep feature extraction model rather than a pre-trained model for the downstream anomaly scoring [44, 65, 163, 168]. Particularly, in Reference [163], three separate autoencoder networks are trained to learn low-dimensional features for respective appearance, motion, and appearance-motion joint representations for video anomaly detection. An ensemble of three one-class SVMs is independently trained on each of these learned feature representations to perform anomaly scoring. Similarly to Reference [163], a linear one-class SVM is used to enable anomaly detection on low-dimensional representations of high-dimensional tabular data yielded by deep belief networks [44]. Instead of one-class SVM, unsupervised classification approaches are used in Reference [65] to enable anomaly scoring in the projected space. Specially, they first cluster the low-dimensional features of video frames yielded by convolutional autoencoders, and then treat the cluster labels as pseudo class labels to perform one-vs.-the-rest classification. The classification probabilities are used to define frame-wise anomaly scores. Similar approaches can also be found in graph anomaly detection [168], in which unsupervised clustering-based anomaly measures are used in the latent representation space to calculate the abnormality of graph vertices or edges. To learn expressive representations of graph vertices, the vertex representations are optimized by minimizing autoencoder-based reconstruction loss and pairwise distances of neighbored graph vertices, taking one-hot encoding of graph vertices as input.

Advantages. The advantages of this approach are as follows. (i) A large number of state-of-the-art (pre-trained) deep models and off-the-shelf anomaly detectors are readily available. (ii) Deep feature extraction offers more powerful dimensionality reduction than popular linear methods. (iii) It is easy-to-implement given the public availability of the deep models and detection methods.

Disadvantages. Their disadvantages are as follows. (i) The fully disjointed feature extraction and anomaly scoring often lead to suboptimal anomaly scores. (ii) Pre-trained deep models are typically limited to specific types of data.

Challenges Targeted. This category of methods projects high-dimensional/non-independent data onto substantially lower-dimensional space, enabling existing anomaly detection methods to work on simpler data space. The lower-dimensional space often helps reveal hidden anomalies and reduces false positives (CH2). However, it should be noted that these methods may not preserve sufficient information for anomaly detection as the data projection is fully decoupled with anomaly detection. In addition, this approach allows us to leverage multiple types of features and learn

semantic-rich detection models (e.g., various predefined image/video features in References [65, 66, 163]), which also helps reduce false positives (CH1).

5 LEARNING FEATURE REPRESENTATIONS OF NORMALITY

The methods in this category couple feature learning with anomaly scoring in some ways, rather than fully decoupling these two modules as in the last section. These methods generally fall into two groups: generic feature learning and anomaly measure-dependent feature learning.

5.1 Generic Normality Feature Learning

This category of methods learns the representations of data instances by optimizing a generic feature learning objective function that is not primarily designed for anomaly detection, but the learned representations can still empower the anomaly detection, since they are forced to capture some key underlying data regularities. Formally, this framework can be represented as

$$\{\Theta^*, \mathbf{W}^*\} = \arg\min_{\Theta,\, \mathbf{W}} \sum_{\mathbf{x}\in\mathcal{X}} \ell\Big(\psi\big(\phi(\mathbf{x};\Theta);\mathbf{W}\big)\Big), \tag{2}$$

$$s_{\mathbf{x}} = f(\mathbf{x}, \phi_{\Theta^*}, \psi_{\mathbf{W}^*}), \tag{3}$$

where ϕ maps the original data onto the representation space $\mathcal{Z}$, ψ parameterized by $\mathbf{W}$ is a surrogate learning task that operates on the $\mathcal{Z}$ space and is dedicated to enforcing the learning of underlying data regularities, ℓ is a loss function relative to the underlying modeling approach, and f is a scoring function that utilizes ϕ and ψ to calculate the anomaly score s.

This approach include methods driven by several perspectives, including data reconstruction, generative modeling, predictability modeling and self-supervised classification. Both predictability modeling and self-supervised classification are built upon self-supervised learning approaches, but they have different assumptions, advantages and flaws, and thus they are reviewed separately.

5.1.1 Autoencoders. This type of approach aims to learn some low-dimensional feature representation space on which the given data instances can be well reconstructed. This is a widely used technique for data compression or dimension reduction [61, 69, 150]. The heuristic for using this technique in anomaly detection is that the learned feature representations are enforced to learn important regularities of the data to minimize reconstruction errors; anomalies are difficult to be reconstructed from the resulting representations and thus have large reconstruction errors.

Assumptions. Normal instances can be better restructured from compressed space than anomalies.

Autoencoder (AE) networks are the commonly used techniques in this category. An AE is composed of an encoding network and an decoding network. The encoder maps the original data onto low-dimensional feature space, while the decoder attempts to recover the data from the projected low-dimensional space. The parameters of these two networks are learned with a reconstruction loss function. A bottleneck network architecture is often used to obtain low-dimensional representations than the original data, which forces the model to retain the information that is important in reconstructing the data instances. To minimize the overall reconstruction error, the retained information is required to be as much relevant as possible to the dominant instances, e.g., the normal instances. As a result, the data instances such as anomalies that deviate from the majority of the data are poorly reconstructed. The data reconstruction error can therefore be directly used as anomaly score. The basic formulation of this approach is given as follows:

$$\mathbf{z} = \phi_e(\mathbf{x};\Theta_e),\ \hat{\mathbf{x}} = \phi_d(\mathbf{z};\Theta_d), \tag{4}$$

$$\{\Theta_e^*, \Theta_d^*\} = \arg\min_{\Theta_e,\Theta_d} \sum_{\mathbf{x}\in\mathcal{X}} \Big\|\mathbf{x} - \phi_d\big(\phi_e(\mathbf{x};\Theta_e);\Theta_d\big)\Big\|^2, \tag{5}$$

$$s_{\mathbf{x}} = \left\| \mathbf{x} - \phi_d\left(\phi_e(\mathbf{x}; \Theta_e^*); \Theta_d^*\right) \right\|^2, \tag{6}$$

where ϕ_e is the encoding network with the parameters Θ_e and ϕ_d is the decoding network with the parameters Θ_d. The encoder and the decoder can share the same weight parameters to reduce parameters and regularize the learning. $s_{\mathbf{x}}$ is a reconstruction error-based anomaly score of $\mathbf{x}$.

Several types of regularized autoencoders have been introduced to learn richer and more expressive feature representations [38, 95, 128, 153]. Particularly, sparse AE is trained in a way that encourages sparsity in the activation units of the hidden layer, e.g., by keeping the top-K most active units [95]. Denoising AE [153] aims at learning representations that are robust to small variations by learning to reconstruct data from some predefined corrupted data instances rather than original data. Contractive AE [128] takes a step further to learn feature representations that are robust to small variations of the instances around their neighbors. This is achieved by adding a penalty term based on the Frobenius norm of the Jacobian matrix of the encoder's activations. Variational AE [38] instead introduces regularization into the representation space by encoding data instances using a prior distribution over the latent space, preventing overfitting and ensuring some good properties of the learned space for enabling generation of meaningful data instances.

AEs are easy-to-implement and have straightforward intuitions in detecting anomalies. As a result, they have been widely explored in the literature. Replicator neural network [57] is the first piece of work in exploring the idea of data reconstruction to detect anomalies, with experiments focused on static multidimensional/tabular data. The Replicator network is built upon a feed-forward multi-layer perceptron with three hidden layers. It uses *parameterized* hyperbolic tangent activation functions to obtain different activation levels for different input values, which helps discretize the intermediate representations into some predefined bins. As a result, the hidden layers naturally cluster the data instances into a number of groups, enabling the detection of clustered anomalies. After this work there have been a number of studies dedicated to further enhance the performance of this approach. For instance, RandNet [29] further enhances the basic AEs by learning an ensemble of AEs. In RandNet, a set of independent AEs are trained, with each AE having some randomly selected constant dropout connections. An adaptive sampling strategy is used by exponentially increasing the sample size of the mini-batches. RandNet is focused on tabular data. The idea of autoencoder ensembles is extended to time series data in Reference [71]. Motivated by robust principal component analysis (RPCA), RDA [175] attempts to improve the robustness of AEs by iteratively decomposing the original data into two subsets, normal instance set and anomaly set. This is achieved by adding a sparsity penalty ℓ_1 or grouped penalty $\ell_{2,1}$ into its RPCA-alike objective function to regularize the coefficients of the anomaly set.

AEs are also widely leveraged to detect anomalies in data other than tabular data, such as sequence data [91], graph data [37], and image/video data [163]. In general, there are two types of adaptions of AEs to those complex data. The most straightforward way is to follow the same procedure as the conventional use of AEs by adapting the network architecture to the type of input data, such as CNN-AE [56, 172], LSTM-AE [96], Conv-LSTM-AE [92], and graph convolutional network-AE [37]. This type of AEs embeds the encoder-decoder scheme into the full procedure of these methods. Another type of AE-based approaches is to first use AEs to learn low-dimensional representations of the complex data and then learn to predict these learned representations. The learning of AEs and representation prediction is often two separate steps. These approaches are different from the first type of approaches in that the prediction of representations are wrapped around the low-dimensional representations yielded by AEs. For example, in Reference [91], denoising AE is combined with RNNs to learn normal patterns of multivariate sequence data, in which a denoising AE with two hidden layers is first used to learn representations of multidimensional data inputs in each time step and a RNN with a single hidden layer is then trained to predict

the representations yielded by the denoising AE. A similar approach is also used for detecting acoustic anomalies [97], in which a more complex RNN, bidirectional LSTMs, is used.

Advantages. The advantages of data reconstruction-based methods are as follows. (i) The idea of AEs is straightforward and generic to different types of data. (ii) Different types of powerful AE variants can be leveraged to perform anomaly detection.

Disadvantages. Their disadvantages are as follows. (i) The learned feature representations can be biased by infrequent regularities and the presence of outliers or anomalies in the training data. (ii) The objective function of the data reconstruction is designed for dimension reduction or data compression, rather than anomaly detection. As a result, the resulting representations are a generic summarization of underlying regularities, which are not optimized for detecting irregularities.

Challenges Targeted. Different types of neural network layers and architectures can be used under the AE framework, allowing us to detect anomalies in high-dimensional data, as well as non-independent data such as attributed graph data [37] and multivariate sequence data [91, 97] (CH2). These methods may reduce false positives over traditional methods built upon handcrafted features if the learned representations are more expressive (CH1). AEs are generally vulnerable to data noise presented in the training data as they can be trained to remember those noise, leading to severe overfitting and small reconstruction errors of anomalies. The idea of RPCA may be used into AEs to train more robust detection models [175] (CH4).

5.1.2 Generative Adversarial Networks. GAN-based anomaly detection emerges quickly as one popular deep anomaly detection approach after its early use in Reference [138]. This approach generally aims to learn a latent feature space of a generative network G so that the latent space well captures the normality underlying the given data. Some form of residual between the real instance and the generated instance are then defined as anomaly score.

Assumption. Normal data instances can be better generated than anomalies from the latent feature space of the generative network in GANs.

One of the early methods is AnoGAN [138]. The key intuition is that, given any data instances $\mathbf{x}$, it aims to search for an instance $\mathbf{z}$ in the learned latent feature space of the generative network G so that the corresponding generated instance $G(\mathbf{z})$ and $\mathbf{x}$ are as similar as possible. Since the latent space is enforced to capture the underlying distribution of training data, anomalies are expected to be less likely to have highly similar generated counterparts than normal instances. Specifically, a GAN is first trained with the following conventional objective:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_X}\Big[\log D(\mathbf{x})\Big] + \mathbb{E}_{\mathbf{z}\sim p_Z}\Big[\log\Big(1 - D\Big(G(\mathbf{z})\Big)\Big)\Big], \tag{7}$$

where G and D are respectively the generator and discriminator networks parameterized by Θ_G and Θ_D (the parameters are omitted for brevity), and V is the value function of the two-player minimax game. After that, for each $\mathbf{x}$, to find its best $\mathbf{z}$, two loss functions—residual loss and discrimination loss—are used to guide the search. The residual loss is defined as

$$\ell_R(\mathbf{x}, \mathbf{z}_\gamma) = \Big\|\mathbf{x} - G(\mathbf{z}_\gamma)\Big\|_1, \tag{8}$$

while the discrimination loss is defined based on the feature matching technique [136]:

$$\ell_{fm}(\mathbf{x}, \mathbf{z}_\gamma) = \Big\|h(\mathbf{x}) - h\Big(G(\mathbf{z}_\gamma)\Big)\Big\|_1, \tag{9}$$

where γ is the index of the search iteration step and h is a feature mapping from an intermediate layer of the discriminator. The search starts with a randomly sampled $\mathbf{z}$, followed by updating $\mathbf{z}$ based on the gradients derived from the overall loss $(1-\alpha)\ell_R(\mathbf{x}, \mathbf{z}_\gamma) + \alpha\ell_{fm}(\mathbf{x}, \mathbf{z}_\gamma)$, where α is a hyperparameter. Throughout this search process, the parameters of the trained GAN are fixed;

the loss is only used to update the coefficients of $\mathbf{z}$ for the next iteration. The anomaly score is accordingly defined upon the similarity between $\mathbf{x}$ and $\mathbf{z}$ obtained at the last step γ^*:

$$s_{\mathbf{x}} = (1-\alpha)\ell_R(\mathbf{x}, \mathbf{z}_{\gamma^*}) + \alpha\ell_{fm}(\mathbf{x}, \mathbf{z}_{\gamma^*}). \tag{10}$$

One main issue with AnoGAN is the computational inefficiency in the iterative search of $\mathbf{z}$. One way to address this issue is to add an extra network that learns the mapping from data instances onto the latent space, i.e., an inverse of the generator, resulting in methods like EBGAN [170] and fast AnoGAN [137]. These two methods share the same spirit. Here we focus on EBGAN that is built upon the bi-directional GAN (BiGAN) [39]. Particularly, BiGAN has an encoder E to map $\mathbf{x}$ to $\mathbf{z}$ in the latent space and simultaneously learn the parameters of G, D and E. Instead of discriminating $\mathbf{x}$ and $G(\mathbf{z})$, BiGAN aims to discriminate the pair of instances $(\mathbf{x}, E(\mathbf{x}))$ from the pair $(G(\mathbf{z}), \mathbf{z})$:

$$\min_{G,E}\max_{D} \quad \mathbb{E}_{\mathbf{x}\sim p_X}\Big[\mathbb{E}_{\mathbf{z}\sim p_E(\cdot|\mathbf{x})}\log\Big[D(\mathbf{x},\mathbf{z})\Big]\Big] + \mathbb{E}_{\mathbf{z}\sim p_{\mathcal{Z}}}\Big[\mathbb{E}_{\mathbf{x}\sim p_G(\cdot|\mathbf{z})}\Big[\log\Big(1-D(\mathbf{x},\mathbf{z})\Big)\Big]\Big], \tag{11}$$

After the training, inspired by Equation (10) in AnoGAN, EBGAN defines the anomaly score as:

$$s_{\mathbf{x}} = (1-\alpha)\ell_G(\mathbf{x}) + \alpha\ell_D(\mathbf{x}), \tag{12}$$

where $\ell_G(\mathbf{x}) = \Big\|\mathbf{x} - G\Big(E(\mathbf{x})\Big)\Big\|_1$ and $\ell_D(\mathbf{x}) = \Big\|h\Big(\mathbf{x}, E(\mathbf{x})\Big) - h\Big(G\Big(E(\mathbf{x})\Big), E(\mathbf{x})\Big)\Big\|_1$. This eliminates the need to iteratively search $\mathbf{z}$ in AnoGAN. EBGAN is extended to a method called ALAD [171] by adding two more discriminators, with one discriminator trying to discriminate the pair $(\mathbf{x}, \mathbf{x})$ from $(\mathbf{x}, G(E(\mathbf{x})))$ and another one trying to discriminate the pair $(\mathbf{z}, \mathbf{z})$ from $(\mathbf{z}, E(G(\mathbf{z})))$.

GANomaly [3] further improves the generator over the previous work by changing the generator network to an encoder-decoder-encoder network and adding two more extra loss functions. The generator can be conceptually represented as: $\mathbf{x} \xrightarrow{G_E} \mathbf{z} \xrightarrow{G_D} \hat{\mathbf{x}} \xrightarrow{E} \hat{\mathbf{z}}$, in which G is a composition of the encoder G_E and the decoder G_D. In addition to the commonly used feature matching loss:

$$\ell_{fm} = \mathbb{E}_{\mathbf{x}\sim p_X}\Big\|h(\mathbf{x}) - h\Big(G(\mathbf{x})\Big)\Big\|_2, \tag{13}$$

the generator includes a contextual loss and an encoding loss to generate more realistic instances:

$$\ell_{con} = \mathbb{E}_{\mathbf{x}\sim p_X}\Big\|\mathbf{x} - G(\mathbf{x})\Big\|_1, \tag{14}$$

$$\ell_{enc} = \mathbb{E}_{\mathbf{x}\sim p_X}\Big\|G_E(\mathbf{x}) - E\Big(G(\mathbf{x})\Big)\Big\|_2. \tag{15}$$

The contextual loss in Equation (14) enforces the generator to consider the contextual information of the input $\mathbf{x}$ when generating $\hat{\mathbf{x}}$. The encoding loss in Equation (15) helps the generator to learn how to encode the features of the generated instances. The overall loss is then defined as

$$\ell = \alpha\ell_{fm} + \beta\ell_{con} + \gamma\ell_{enc}, \tag{16}$$

where α, β, and γ are the hyperparameters to determine the weight of each individual loss. Since the training data contains mainly normal instances, the encoders G and E are optimized toward the encoding of normal instances, and thus, the anomaly score can be defined as

$$s_{\mathbf{x}} = \Big\|G_E(\mathbf{x}) - E\Big(G(\mathbf{x})\Big)\Big\|_1, \tag{17}$$

in which $s_{\mathbf{x}}$ is expected to be large if $\mathbf{x}$ is an anomaly.

There have been a number of other GANs introduced over the years such as Wasserstein GAN [10] and Cycle GAN [177]. They may be used to further enhance the anomaly detection performance of the above methods, such as replacing the standard GAN with Wasserstein GAN [137].

Another relevant research line is to adversarially learn end-to-end one-class classification models, which is categorized into the end-to-end anomaly score learning framework and discussed in Section 6.4.

Advantages. The advantages of these methods are as follows. (i) GANs have demonstrated superior capability in generating realistic instances, especially on image data, empowering the detection of abnormal instances that are poorly reconstructed from the latent space. (ii) A large number of existing GAN-based models and theories [32] may be adapted for anomaly detection.

Disadvantages. Their disadvantages are as follows. (i) The training of GANs can suffer from multiple problems, such as failure to converge and mode collapse [99], which leads to to large difficulty in training GANs-based anomaly detection models. (ii) The generator network can be misled and generates data instances out of the manifold of normal instances, especially when the true distribution of the given dataset is complex or the training data contain unexpected outliers. (iii) The GANs-based anomaly scores can be suboptimal, since they are built upon the generator network with the objective designed for data synthesis rather than anomaly detection.

Challenges Targeted. Similarly to AEs, GAN-based anomaly detection is able to detect high-dimensional anomalies by examining the reconstruction from the learned low-dimensional latent space (CH2). When the latent space preserves important anomaly discrimination information, it helps improve detection accuracy over that in the original data space (CH1).

5.1.3 Predictability Modeling. Predictability modeling-based methods learn feature representations by predicting the current data instances using the representations of the previous instances within a temporal window as the context. In this section data instances are referred to as individual elements in a sequence, e.g., video frames in a video sequence. This technique is widely used for sequence representation learning and prediction [63, 82, 98, 146]. To achieve accurate predictions, the representations are enforced to capture the temporal/sequential and recurrent dependence within a given sequence length. Normal instances are normally adherent to such dependencies well and can be well predicted, whereas anomalies often violate those dependencies and are unpredictable. Therefore, the prediction errors can be used to define the anomaly scores.

Assumption. Normal instances are temporally more predictable than anomalies.

This research line is popular in video anomaly detection [1, 86, 167]. Video sequence involves complex high-dimensional spatial-temporal features. Different constraints over appearance and motion features are needed in the prediction objective function to ensure a faithful prediction of video frames. This deep anomaly detection approach is initially explored in Reference [86]. Formally, given a video sequence with consecutive t frames $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t$, then the learning task is to use all these frames to generate a future frame $\hat{\mathbf{x}}_{t+1}$ so that $\hat{\mathbf{x}}_{t+1}$ is as close as possible to the ground truth $\mathbf{x}_{t+1}$. Its general objective function can be formulated as

$$\alpha \ell_{pred}\left(\hat{\mathbf{x}}_{t+1}, \mathbf{x}_{t+1}\right) + \beta \ell_{adv}\left(\hat{\mathbf{x}}_{t+1}\right), \tag{18}$$

where $\hat{\mathbf{x}}_{t+1} = \psi\left(\phi(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_t; \Theta); \mathbf{W}\right)$, ℓ_{pred} is the frame prediction loss measured by mean squared errors, ℓ_{adv} is an adversarial loss. The popular network architecture named U-Net [129] is used to instantiate the ψ function for the frame generation. ℓ_{pred} is composed by a set of three separate losses that respectively enforce the closeness between $\hat{\mathbf{x}}_{t+1}$ and $\mathbf{x}_{t+1}$ in three key image feature descriptors: intensity, gradient and optical flow. ℓ_{adv} is due to the the use of adversarial training to enhance the image generation. After training, for a given video frame $\mathbf{x}$, a normalized Peak Signal-to-Noise Ratio [98] based on the prediction difference $||\mathbf{x}_i - \hat{\mathbf{x}}_i||_2$ is used to define the anomaly score. Under the same framework, an additional autoencoder-based reconstruction

network is added in Reference [167] to further refine the predicted frame quality, which helps to enlarge the anomaly score difference between normal and abnormal frames.

Another research line in this direction is based on the autoregressive models [50] that assume each element in a sequence is linearly dependent on the previous elements. The autoregressive models are leveraged in Reference [1] to estimate the density of training samples in a latent space, which helps avoid the assumption of a specific family of distributions. Specifically, given $\mathbf{x}$ and its latent space representation $\mathbf{z} = \phi(\mathbf{x}; \Theta)$, the autoregressive model factorizes $p(\mathbf{z})$ as

$$p(\mathbf{z}) = \prod_{j=1}^{K} p(z_j | z_{1:j-1}), \tag{19}$$

where $z_{1:j-1} = \{z_1, z_2, \ldots, z_{j-1}\}$, $p(z_j|z_{1:j-1})$ represents the probability mass function of z_j conditioned on all the previous instances $z_{1:j-1}$ and K is the dimensionality size of the latent space. The objective in Reference [1] is to jointly learn an autoencoder and a density estimation network $\psi(\mathbf{z}; \mathbf{W})$ equipped with autoregressive network layers. The overall loss can be represented as

$$L = \mathbb{E}_{\mathbf{x}}\left[\left\|\mathbf{x} - \phi_d\left(\phi_e(\mathbf{x}; \Theta_e); \Theta_d\right)\right\|_2 - \lambda \log\left(\psi(\mathbf{z}; \mathbf{W})\right)\right], \tag{20}$$

where the first term is a reconstruction error measured by MSE while the second term is an autoregressive loss measured by the log-likelihood of the representation under an estimated conditional probability density prior. Minimizing this loss enables the learning of the features that are common and easily predictable. At the evaluation stage, the reconstruction error and the log-likelihood are combined to define the anomaly score.

Advantages. The advantages of this category of methods are as follows. (i) A number of sequence learning techniques can be adapted and incorporated into this approach. (ii) This approach enables the learning of different types of temporal and spatial dependencies.

Disadvantages. Their disadvantages are as follows. (i) This approach is limited to anomaly detection in sequence data. (ii) The sequential predictions can be computationally expensive. (iii) The learned representations may suboptimal for anomaly detection as its underlying objective is for sequential predictions rather than anomaly detection.

Challenges Targeted. This approach is particularly designed to learn expressive temporally-dependent low-dimensional representations, which helps address the false positives of anomaly detection in high-dimensional and/or temporal datasets (CH1 and CH2). The prediction here is conditioned on some elapsed temporal instances, so this category of methods is able to detect temporal context-based conditional anomalies (CH5).

5.1.4 Self-supervised Classification. This approach learns representations of normality by building self-supervised classification models and identifies instances that are inconsistent to the classification models as anomalies. This approach is rooted in traditional methods based on cross-feature analysis or feature models [64, 105, 149]. These shallow methods evaluate the normality of data instances by their consistency to a set of predictive models, with each model learning to predict one feature based on the rest of the other features. The consistency of a test instance can be measured by the average prediction results [64], the log loss-based surprisal [105], or the majority voting of binary decisions [149] given the classification/regression models across all features. Unlike these studies that focus on tabular data and build the feature models using the original data, deep consistency-based anomaly detection focuses on image data and builds the predictive models by using feature transformation-based augmented data. To effectively discriminate the transformed instances, the classification models are enforced to learn features that are highly important

to describe the underlying patterns of the instances presented in the training data. Therefore, normal instances generally have stronger agreements with the classification models.

Assumptions. Normal instances are more consistent to self-supervised classifiers than anomalies.

This approach is initially explored in Reference [48]. To build the predictive models, different compositions of geometric transformation operations, including horizontal flipping, translations, and rotations, is first applied to normal training images. A deep multi-class classification model is trained on the augmented data, treating data instances with a specific transformation operation comes from the same class, i.e., synthetic class. At inference, test instances are augmented with each of transformation compositions, and their normality score is defined by an aggregation of all softmax classification scores to the augmented test instance. Its loss function is defined as

$$L_{cons} = CE\Big(\psi\big(\mathbf{z}_{T_j}; \mathbf{W}\big), \mathbf{y}_{T_j}\Big), \tag{21}$$

where $\mathbf{z}_{T_j} = \phi\big(T_j(\mathbf{x}); \Theta\big)$ is a low-dimensional feature representation of instance $\mathbf{x}$ augmented by the transformation operation type T_j, ψ is a multi-class classifier parameterized with $\mathbf{W}$, $\mathbf{y}_{T_j}$ is a one-hot encoding of the synthetic class for instances augmented using the transformation operation T_j, and CE is a standard cross-entropy loss function.

By minimizing Equation (21), we obtain the representations that are optimized for the classifier ψ. We then can apply the feature learner $\phi(\cdot, \Theta^*)$ and the classifier $\psi(\cdot, \mathbf{W}^*)$ to obtain a classification score for each test instance augmented with a transformation operation T_j. The classification scores of each test instance w.r.t. different T_j are then aggregated to compute the anomaly score. To achieve that, the classification scores conditioned on each T_j is assumed to follow a Dirichlet distribution in Reference [48] to estimate the consistency of the test instance to the classification model ψ; a simple average of the classification scores associated with different T_j also works well.

A semi-supervised setting, i.e., training data contain normal instances only, is assumed in Reference [48]. A similar idea is explored in the unsupervised setting in Reference [157], with the transformation sets containing four transformation operations, i.e., rotation, flipping, shifting and path re-arranging. Two key insights revealed in Reference [157] is that (i) the gradient magnitude induced by normal instances is normally substantially larger than outliers during the training of such self-supervised multi-class classification models; and (ii) the network updating direction is also biased toward normal instances. As a result of these two properties, normal instances often have stronger agreement with the classification model than anomalies. Three strategies of using the classification scores to define the anomaly scores are evaluated, including average prediction probability, maximum prediction probability, and negative entropy across all prediction probabilities [157]. Their results show that the negative entropy-based anomaly scores perform generally better than the other two strategies.

Advantages. The advantages of deep consistency-based methods are as follows. (i) They work well in both the unsupervised and semi-supervised settings. (ii) Anomaly scoring is grounded by some intrinsic properties of gradient magnitude and its updating.

Disadvantages. Their disadvantages are as follows. (i) The feature transformation operations are often data dependent. The above transformation operations are applicable to image data only. (ii) Although the classification model is trained in an end-to-end manner, the consistency-based anomaly scores are derived upon the classification scores rather than an integrated module in the optimization, and thus they may be suboptimal.

Challenges Targeted. The expressive low-dimensional representations of normality this approach learns help detect anomalies better than in the original high-dimensional space (CH1 and CH2).

Due to some intrinsic differences between anomalies and normal instances presented in the self-supervised classifiers, this approach is also able to work in an unsupervised setting [157], demonstrating good robustness to anomaly contamination in the training data (CH4).

5.2 Anomaly Measure-dependent Feature Learning

Anomaly measure-dependent feature learning aims at learning feature representations that are specifically optimized for one particular existing anomaly measure. Formally, the framework for this group of methods can be represented as

$$\{\Theta^*, \mathbf{W}^*\} = \arg\min_{\Theta,\, \mathbf{W}} \sum_{\mathbf{x}\in\mathcal{X}} \ell\Big(f\Big(\phi(\mathbf{x};\Theta);\mathbf{W}\Big)\Big), \tag{22}$$

$$s_{\mathbf{x}} = f\Big(\phi(\mathbf{x};\Theta^*);\mathbf{W}^*\Big), \tag{23}$$

where f is an existing anomaly scoring measure operating on the representation space. Note that whether f may involve trainable parameters $\mathbf{W}$ or not is dependent on the anomaly measure used. Different from the generic feature learning approach as in Equations (2) and (3) that calculates anomaly scores based on some heuristics after obtaining the learned representations, this research line incorporates an existing anomaly measure f into the feature learning objective function to optimize the feature representations specifically for f. Below we review representation learning specifically designed for three types of popular anomaly measures, including distance-based measure, one-class classification measure and clustering-based measure.

5.2.1 Distance-based Measure. Deep distance-based anomaly detection aims to learn feature representations that are specifically optimized for a specific type of distance-based anomaly measures. Distance-based methods are straightforward and easy-to-implement. There have been a number of effective distance-based anomaly measures introduced, e.g., DB outliers [72, 73], k-nearest neighbor distance [125, 126], average k-nearest neighbor distance [9], relative distance [173], and random nearest neighbor distance [116, 144]. One major limitation of these traditional distance-based anomaly measures is that they fail to work effectively in high-dimensional data due to the curse of dimensionality. Since deep distance-based anomaly detection techniques project data onto low-dimensional space before applying the distance measures, it can well overcome this limitation.

Assumption. Anomalies are distributed far from their closest neighbors while normal instances are located in dense neighborhoods.

This approach is first explored in Reference [112], in which the random neighbor distance-based anomaly measure [116, 144] is leveraged to drive the learning of low-dimensional representations out of ultrahigh-dimensional data. The key idea is that the representations are optimized so that the nearest neighbor distance of pseudo-labeled anomalies in random subsamples is substantially larger than that of pseudo-labeled normal instances. The pseudo labels are generated by some off-the-shelf anomaly detectors. Let $\mathcal{S} \in \mathcal{X}$ be a subset of data instances randomly sampled from the dataset $\mathcal{X}$, $\mathcal{A}$ and $\mathcal{N}$ respectively be the pseudo-labeled anomaly and normal instance sets, with $\mathcal{X} = \mathcal{A} \cup \mathcal{N}$ and $\emptyset = \mathcal{A} \cap \mathcal{N}$, its loss function is built upon the hinge loss function [130]:

$$L_{query} = \frac{1}{|\mathcal{X}|} \sum_{\mathbf{x}\in\mathcal{A},\mathbf{x}'\in\mathcal{N}} \max\Big\{0, m + f(\mathbf{x}',\mathcal{S};\Theta) - f(\mathbf{x},\mathcal{S};\Theta)\Big\}, \tag{24}$$

where m is a predefined constant for the margin between two distances yielded by $f(\mathbf{x},\mathcal{S};\Theta)$, which is a random nearest neighbor distance function operated in the representation space:

$$f(\mathbf{x},\mathcal{S};\Theta) = \min_{\mathbf{x}'\in\mathcal{S}} \Big\|\phi(\mathbf{x};\Theta), \phi(\mathbf{x}';\Theta)\Big\|_2. \tag{25}$$

Minimizing the loss in Equation (24) guarantees that the random nearest neighbor distance of anomalies are at least m greater than that of normal instances in the ϕ-projected representation space. At the evaluation stage, the random distance in Equation (25) is used directly to obtain the anomaly score for each test instance. Following this approach, we might also derive similar representation learning tailored for other distance-based measures by replacing Equation (25) with the other measures, such as the k-nearest neighbor distance [126] or the average k-nearest neighbor distance [9]. However, these measures are significantly more computationally costly than Equation (25). Thus, one major challenging for such adaptions would be the prohibitively high computational cost.

Compared to Reference [112] that requires to query the nearest neighbor distances in random data subsets, inspired by Reference [19], a simpler idea explored in Reference [155] uses the distance between optimized representations and randomly projected representations of the same instances to guide the representation learning. The objective of the method is as follows:

$$\Theta^* = \arg\min_{\Theta} \sum_{\mathbf{x}\in\mathcal{X}} f\Big(\phi(\mathbf{x};\Theta), \phi'(\mathbf{x})\Big), \tag{26}$$

where ϕ' is a random mapping function that is instantiated by the neural network used in ϕ with fixed random weights and f is a measure of distance between the two representations of the same data instance. As discussed in Reference [19], solving Equation (26) is equivalent to have a knowledge distillation from a random neural network and helps learn the frequency of different underlying patterns in the data. However, Equation (26) ignores the relative proximity between data instances and is sensitive to the anomalies presented in the data. As shown in Reference [155], such proximity information may be learned by a pretext task, in which we aim to predict the distance between random instance pairs. A boosting process can also be used to iteratively filter potential anomalies and build robust detection models. At the evaluation stage, $f(\phi(\mathbf{x};\Theta^*), \phi'(\mathbf{x}))$ is used to compute the anomaly scores.

Advantages. The advantages of this category of methods are as follows. (i) The distance-based anomalies are straightforward and well defined with rich theoretical supports in the literature. Thus, deep distance-based anomaly detection methods can be well grounded due to the strong foundation built in previous relevant work. (ii) They work in low-dimensional representation spaces and can effectively deal with high-dimensional data that traditional distance-based anomaly measures fail. (iii) They are able to learn representations specifically tailored for themselves.

Disadvantages. Their disadvantages are as follows. (i) The extensive computation involved in most of distance-based anomaly measures may be an obstacle to incorporate distance-based anomaly measures into the representation learning process. (ii) Their capabilities may be limited by the inherent weaknesses of the distance-based anomaly measures.

Challenges Targeted. This approach is able to learn low-dimensional representations tailored for existing distance-based anomaly measures, addressing the notorious curse of dimensionality in distance-based detection [178] (CH1 and CH2). As shown in Reference [112], an adapted triplet loss can be devised to utilize a few labeled anomaly examples to learn more effective normality representations (CH3). Benefiting from pseudo anomaly labeling, the methods [112, 155] are also robust to potential anomaly contamination and work effectively in the fully unsupervised setting (CH4).

5.2.2 One-class Classification-based Measure. This category of methods aims to learn feature representations customized to subsequent one-class classification-based anomaly detection. One-class classification is referred to as the problem of learning a description of a set of data instances to detect whether new instances conform to the training data or not. It is one of the most popular

approaches for anomaly detection [101, 131, 139, 148]. Most one-class classification models are inspired by SVM [31], such as the two widely used one-class models: one-class SVM (or v-SVC) [139] and Support Vector Data Description (SVDD) [148]. One main research line here is to learn representations that are specifically optimized for these traditional one-class classification models. This is the focus of this section. Another line is to learn an end-to-end adversarial one-class classification model, which will be discussed in Section 6.4.

Assumption. All normal instances come from a single (abstract) class and can be summarized by a compact model, to which anomalies do not conform.

There are a number of studies dedicated to combine one-class SVM with neural networks [27, 104, 161]. Conventional one-class SVM is to learn a hyperplane that maximize a margin between training data instances and the origin. The key idea of deep one-class SVM is to learn the one-class hyperplane from the neural network-enabled low-dimensional representation space rather than the original input space. Let $\mathbf{z} = \phi(\mathbf{x}; \Theta)$, then a generic formulation of the key ideas in References [27, 104, 161] can be represented as

$$\min_{r,\Theta,\mathbf{w}} \frac{1}{2}||\Theta||^2 + \frac{1}{vN}\sum_{i=1}^{N} \max\left\{0, r - \mathbf{w}^\intercal \mathbf{z}_i\right\} - r, \tag{27}$$

where r is the margin parameter, Θ are the parameters of a representation network, and $\mathbf{w}^\intercal \mathbf{z}$ (i.e., $\mathbf{w}^\intercal \phi(\mathbf{x}; \Theta)$) replaces the original dot product $\left\langle \mathbf{w}, \Phi(\mathbf{x}) \right\rangle$ that satisfies $k(\mathbf{x}_i, \mathbf{x}_j) = \left\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \right\rangle$. Here Φ is a Reproducing Kernel Hilbert Space associated mapping and $k(\cdot, \cdot)$ is a kernel function; v is a hyperparameter that can be seen as an upper bound of the fraction of the anomalies in the training data. Any instances that have $r - \mathbf{w}^\intercal \mathbf{z}_i > 0$ can be reported as anomalies.

This formulation brings two main benefits: (i) it can leverage (pretrained) deep networks to learn more expressive features for downstream anomaly detection, and (iii) it also helps remove the computational expensive pairwise distance computation in the kernel function. As shown in [104, 161], the reconstruction loss in AEs can be added into Equation (27) to enhance the expressiveness of representations $\mathbf{z}$. As shown in Reference [124], many kernel functions can be approximated with random Fourier features. Thus, before $\mathbf{w}^\intercal \mathbf{z}$, some form of random mapping h may be applied to $\mathbf{z}$ to generate Fourier features, resulting in $\mathbf{w}^\intercal h(\mathbf{z})$, which may further improve one-class SVM models. Another research line studies deep models for SVDD [132, 133]. SVDD aims to learn a minimum hyperplane characterized by a center $\mathbf{c}$ and a radius r so that the sphere contains all training data instances, i.e.,

$$\min_{r,\mathbf{c},\xi} r^2 + \frac{1}{vN}\sum_{i=1}^{N} \xi_i, \tag{28}$$

$$\text{s.t. } ||\Phi(\mathbf{x}_i) - \mathbf{c}||^2 \leq r^2 + \xi_i,\ \xi_i \geq 0,\ \forall i. \tag{29}$$

Similarly to deep one-class SVM, deep SVDD [132] also aims to leverage neural networks to map data instances into the sphere of minimum volume, and then employs the hinge loss function to guarantee the margin between the sphere center and the projected instances. The feature learning and the SVDD objective can then be jointly trained by minimizing the following loss:

$$\min_{r,\Theta} r^2 + \frac{1}{vN}\sum_{i=1}^{N} \max\{0, ||\phi(\mathbf{x}_i; \Theta) - \mathbf{c}||^2 - r^2\} + \frac{\lambda}{2}||\Theta||^2. \tag{30}$$

This assumes the training data contain a small proportion of anomaly contamination in the unsupervised setting. In the semi-supervised setting, the loss function can be simplified as

$$\min_{\Theta} \frac{1}{N}||\phi(\mathbf{x}_i; \Theta) - \mathbf{c}||^2 + \frac{\lambda}{2}||\Theta||^2, \tag{31}$$

which directly minimizes the mean distance between the representations of training data instances and the center $\mathbf{c}$. Note that including $\mathbf{c}$ as trainable parameters in Equation (31) can lead to trivial solutions. It is shown in Reference [132] that $\mathbf{c}$ can be fixed as the mean of the feature representations yielded by performing a single initial forward pass. Deep SVDD can also be further extended to address another semi-supervised setting where a small number of both labeled normal instances and anomalies are available [133]. The key idea is to minimize the distance of labeled normal instances to the center while at the same time maximizing the distance of known anomalies to the center. This can be achieved by adding $\sum_{j=1}^{M}\left(||\phi(\mathbf{x}'_j;\Theta)-\mathbf{c}||^2\right)^{y_j}$ into Equation (31), where $\mathbf{x}'_j$ is a labeled instance, $y_j=+1$ when it is a normal instance and $y_j=-1$ otherwise.

Advantages. The advantages of this category of methods are as follows. (i) The one-class classification-based anomalies are well studied in the literature and provides a strong foundation of deep one-class classification-based methods. (ii) The representation learning and one-class classification models can be unified to learn tailored and more optimal representations. (iii) They free the users from manually choosing suitable kernel functions in traditional one-class models.

Disadvantages. Their disadvantages are as follows. (i) The one-class models may work ineffectively in datasets with complex distributions within the normal class. (ii) The detection performance is dependent on the one-class classification-based anomaly measures.

Challenges Targeted. This category of methods enhances detection accuracy by learning lower-dimensional representation space optimized for one-class classification models (CH1 and CH2). A small number of labeled normal and abnormal data can be leveraged by these methods [133] to learn more effective one-class description models, which can not only detect known anomalies but also novel classes of anomaly (CH3).

5.2.3 Clustering-based Measure. Deep clustering-based anomaly detection aims at learning representations so that anomalies are clearly deviated from the clusters in the newly learned representation space. The task of clustering and anomaly detection is naturally tied with each other, so there have been a large number of studies dedicated to using clustering results to define anomalies, e.g., cluster size [67], distance to cluster centers [59], distance between cluster centers [68], and cluster membership [141]. Gaussian mixture model-based anomaly detection [43, 94] is also included into this category due to some of its intrinsic relations to clustering, e.g., the likelihood fit in the Gaussian mixture model (GMM) corresponds to an aggregation of the distances of data instances to the centers of the Gaussian clusters/components [2].

Assumptions. Normal instances have stronger adherence to clusters than anomalies.

Deep clustering, which aims to learn feature representations tailored for a specific clustering algorithm, is the most critical component of this anomaly detection method. A number of studies have explored this problem in recent years [25, 36, 47, 151, 162, 165, 166]. The main motivation is due to the fact that the performance of clustering methods is highly dependent on the input data. Learning feature representations specifically tailored for a clustering algorithm can well guarantee its performance on different datasets [5]. In general, there are two key intuitions here: (i) Good representations enables better clustering and good clustering results can provide effective supervisory signals to representation learning, and (ii) representations that are optimized for one clustering algorithm is not necessarily useful for other clustering algorithms due to the difference of the underlying assumptions made by the clustering algorithms.

The deep clustering methods typically consist of two modules: performing clustering in the forward pass and learning representations using the cluster assignment as pseudo class labels in the backward pass. Its loss function is often the most critical part, which can be generally formulated as

$$\alpha\ell_{clu}\Big(f\Big(\phi(\mathbf{x};\Theta);\mathbf{W}\Big),y_x\Big)+\beta\ell_{aux}(\mathcal{X}), \tag{32}$$

where ℓ_{clu} is a clustering loss function, within which ϕ is the feature learner parameterized by Θ, f is a clustering assignment function parameterized by $\mathbf{W}$ and y_x represents pseudo class labels yielded by the clustering; ℓ_{aux} is a non-clustering loss function used to enforce additional constrains on the learned representations; and α and β are two hyperparameters to control the importance of the two losses. ℓ_{clu} can be instantiated with a k-means loss [25, 162], a spectral clustering loss [151, 166], an agglomerative clustering loss [165], or a GMM loss [36], enabling the representation learning for the targeted clustering algorithm. ℓ_{aux} is often instantiated with an autoencoder-based reconstruction loss [47, 166] to learn robust and/or local structure preserved representations.

After the deep clustering, the cluster assignments in the resulting f function can then be utilized to compute anomaly scores based on References [59, 67, 68, 141]. However, it should be noted that the deep clustering may be biased by anomalies if the training data are anomaly contaminated. Therefore, the above methods can be applied to the semi-supervised setting where the training data are composed by normal instances only. In the unsupervised setting, some extra constrains are required in ℓ_{clu} and/or ℓ_{aux} to eliminate the impact of potential anomalies.

The aforementioned deep clustering methods are focused on learning optimal clustering results. Although their clustering results are applicable to anomaly detection, the learned representations may not be able to well capture the abnormality of anomalies. It is important to utilize clustering techniques to learn representations so that anomalies have clearly weaker adherence to clusters than normal instances. Some promising results for this type of approach are shown in References [83, 179], in which they aim to learn representations for a GMM-based model with the representations optimized to highlight anomalies. The general formation of these two studies is similar to Equation (32) with ℓ_{clu} and ℓ_{aux} respectively specified as a GMM loss and an autoencoder-based reconstruction loss, but to learn deviated representations of anomalies, they concatenate some handcrafted features based on the reconstruction errors with the learned features of the autoencoder to optimize the combined features together. Since the reconstruction error-based handcrafted features capture the data normality, the resulting representations are more optimal for anomaly detection than that yielded by other deep clustering methods.

Advantages. The advantages of deep clustering-based methods are as follows. (i) A number of deep clustering methods and theories can be utilized to support the effectiveness and theoretical foundation of anomaly detection. (ii) Compared to traditional clustering-based methods, deep clustering-based methods learn specifically optimized representations that help spot the anomalies easier than on the original data, especially when dealing with intricate datasets.

Disadvantages. Their disadvantages are as follows. (i) The performance of anomaly detection is heavily dependent on the clustering results. (ii) The clustering process may be biased by contaminated anomalies in the training data, which in turn leads to less effective representations.

Challenges Targeted. The clustering-based anomaly measures are applied to newly learned low-dimensional representations of data inputs; when the new representation space preserves sufficient discrimination information, the deep methods can achieve better detection accuracy than that in the original data space (CH1 and CH2). Some clustering algorithms are sensitive to outliers, so the deep clustering and the subsequent anomaly detection can be largely misled when the given data are contaminated by anomalies. Deep clustering using handcrafted features from the reconstruction errors of autoencoders [179] may help learn more robust models w.r.t. the contamination (CH4).

6 END-TO-END ANOMALY SCORE LEARNING

This research line aims at learning scalar anomaly scores in an end-to-end fashion. Compared to anomaly measure-dependent feature learning, the anomaly scoring in this type of approach is not dependent on existing anomaly measures; it has a neural network that directly learns the anomaly

scores. Novel loss functions are often required to drive the anomaly scoring network. Formally, this approach aims at learning an end-to-end anomaly score learning network: $\tau(\cdot;\Theta) : \mathcal{X} \mapsto \mathbb{R}$. The underlying framework can be represented as

$$\Theta^* = \arg\min_{\Theta} \sum_{\mathbf{x}\in\mathcal{X}} \ell\Big(\tau(\mathbf{x};\Theta)\Big), \tag{33}$$

$$s_{\mathbf{x}} = \tau(\mathbf{x};\Theta^*). \tag{34}$$

Unlike those methods in Section 5.1 that use some sort of heuristics to calculate anomaly scores after obtaining the learned representations, the methods in this category simultaneously learn the feature representations and anomaly scores. This greatly optimizes the anomaly scores and/or anomaly ranking. In this perspective they share some similarities as the methods in Section 5.2. However, the anomaly measure-dependent feature learning methods are often limited by the inherent disadvantages of the incorporated anomaly measures, whereas the methods here do not have such weakness; they also represent two completely different directions of designing the models: one focuses on how to synthesize existing anomaly measures and neural network models, while another focuses on devising novel loss functions for direct anomaly score learning.

Below we review four main approaches in this category: ranking models, prior-driven models, softmax likelihood models and end-to-end one-class classification models. The key to this framework is to incorporate order or discriminative information into the anomaly scoring network.

6.1 Ranking Models

This group of methods aims to directly learn a ranking model, such that data instances can be sorted based on an observable ordinal variable associated with the absolute/relative ordering relation of the abnormality. The anomaly scoring neural network is driven by the observable ordinal variable.

Assumptions. There exists an observable ordinal variable that captures some data abnormality.

One research line of this approach is to devise ordinal regression-based loss functions to drive the anomaly scoring neural network [114, 117]. In Reference [117], a self-trained deep ordinal regression model is introduced to directly optimize the anomaly scores for unsupervised video anomaly detection. Particularly, it assumes an observable ordinal variable $\mathbf{y} = \{c_1, c_2\}$ with $c_1 > c_2$, let $\tau(\mathbf{x};\Theta) = \eta(\phi(\mathbf{x};\Theta_t);\Theta_s)$, $\mathcal{A}$ and $\mathcal{N}$, respectively, be pseudo anomaly and normal instance sets and $\mathcal{G} = \mathcal{A} \cup \mathcal{N}$, then the objective function is formulated as

$$\arg\min_{\Theta} \sum_{\mathbf{x}\in\mathcal{G}} \ell\Big(\tau(\mathbf{x};\Theta), y_{\mathbf{x}}\Big), \tag{35}$$

where $\ell(\cdot,\cdot)$ is a MSE/MAE-based loss function and $y_{\mathbf{x}} = c_1, \forall \mathbf{x} \in \mathcal{A}$ and $y_{\mathbf{x}} = c_2, \forall \mathbf{x} \in \mathcal{N}$. Here y takes two scalar ordinal values only, so it is a two-class ordinal regression.

The end-to-end anomaly scoring network takes $\mathcal{A}$ and $\mathcal{N}$ as inputs and learns to optimize the anomaly scores such that the data inputs of similar behaviors as those in $\mathcal{A}$ ($\mathcal{N}$) receive large (small) scores as close c_1 (c_2) as possible, resulting in larger anomaly scores assigned to anomalous frames than normal frames. Due to the superior capability of capturing appearance features of image data, ResNet-50 [58] is used to specify the feature network ϕ, followed by the anomaly scoring network η built with a fully connected two-layer neural network. η consists of a hidden layer with 100 units and an output layer with a single linear unit. Similarly to Reference [112], $\mathcal{A}$ and $\mathcal{N}$ are initialized by some existing anomaly measures. The anomaly scoring model is then iteratively updated and enhanced in a self-training manner. The MAE-based loss function is employed in Equation (35) to reduce the negative effects brought by false pseudo labels in $\mathcal{A}$ and $\mathcal{N}$.

Different from Reference [117] that addresses an unsupervised setting, a weakly supervised setting is assumed in References [114, 145]. A very small number of labeled anomalies, together with large-scale unlabeled data, is assumed to be available during training in Reference [114]. To leverage the known anomalies, the anomaly detection problem is formulated as a pairwise relation prediction task. Specifically, a two-stream ordinal regression network is devised to learn the relation of randomly sampled pairs of data instances, i.e., to discriminate whether the instance pair contains two labeled anomalies, one labeled anomaly, or just unlabeled data instances. Let $\mathcal{A}$ be the small labeled anomaly set, $\mathcal{U}$ be the large unlabeled dataset and $\mathcal{X} = \mathcal{A} \cup \mathcal{U}$, $\mathcal{P} = \left\{ \{\mathbf{x}_i, \mathbf{x}_j, y_{\mathbf{x}_i\mathbf{x}_j}\} \,|\, \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X} \text{ and } y_{\mathbf{x}_i\mathbf{x}_j} \in \mathbb{N} \right\}$ is first generated. Here $\mathcal{P}$ is a set of random instance pairs with *synthetic* ordinal class labels, where $\mathbf{y} = \{y_{\mathbf{x}_{a_i}\mathbf{x}_{a_j}}, y_{\mathbf{x}_{a_i}\mathbf{x}_{u_i}}, y_{\mathbf{x}_{u_i}\mathbf{x}_{u_j}}\}$ is an ordinal variable. The synthetic label $y_{\mathbf{x}_{a_i}\mathbf{x}_{u_i}}$ means an ordinal value for any instance pairs with the instances $\mathbf{x}_{a_i}$ and $\mathbf{x}_{u_i}$ respectively sampled from $\mathcal{A}$ and $\mathcal{U}$. $y_{\mathbf{x}_{a_i}\mathbf{x}_{a_j}} > y_{\mathbf{x}_{a_i}\mathbf{x}_{u_i}} > y_{\mathbf{x}_{u_i}\mathbf{x}_{u_j}}$ is predefined such that the pairwise prediction task is equivalent to anomaly score learning. The method can then be formally framed as

$$\Theta^* = \arg\min_{\Theta} \frac{1}{|\mathcal{P}|} \sum_{\{\mathbf{x}_i, \mathbf{x}_j, y_{ij}\} \in \mathcal{P}} \left| y_{\mathbf{x}_i\mathbf{x}_j} - \tau\Big((\mathbf{x}_i, \mathbf{x}_j); \Theta\Big) \right|, \tag{36}$$

which is trainable in an end-to-end fashion. By minimizing Equation (36), the model is optimized to learn larger anomaly scores for the pairs of two anomalies than the pairs with one anomaly or none. At inference, each test instance is paired with instances from $\mathcal{A}$ or $\mathcal{U}$ to obtain the anomaly scores.

The weakly supervised setting in Reference [145] addresses frame-level video anomaly detection, but only video-level class labels are available during training, i.e., a video is normal or contains abnormal frames somewhere—we do not know which specific frames are anomalies. A multiple instance learning– (MIL) based ranking model is introduced in Reference [145] to harness the high-level class labels to directly learn the anomaly score for each video segment (i.e., a small number of consecutive video frames). Its key objective is to guarantee that the maximum anomaly score for the segments in a video that contains anomalies somewhere is greater than the counterparts in a normal video. To achieve this, each video is treated as a bag of instances in MIL, the videos that contains anomalies are treated as positive bags, and the normal videos are treated as negative bags. Each video segment is an instance in the bag. The ordering information of the anomaly scores is enforced as a relative pairwise ranking order via the hinge loss function. The overall objective function is defined as

$$\begin{aligned} \arg\min_{\Theta} \sum_{\mathcal{B}_p, \mathcal{B}_n \in \mathcal{X}} \max\left\{0, 1 - \max_{\mathbf{x} \in \mathcal{B}_p} \tau(\mathbf{x}; \Theta) + \max_{\mathbf{x} \in \mathcal{B}_n} \tau(\mathbf{x}; \Theta)\right\} \\ + \lambda_1 \sum_{i=1}^{|\mathcal{B}_p|} \Big(\tau(\mathbf{x}_i; \Theta) - \tau(\mathbf{x}_{i+1}; \Theta)\Big)^2 + \lambda_2 \sum_{\mathbf{x} \in \mathcal{B}_p} \tau(\mathbf{x}; \Theta), \end{aligned} \tag{37}$$

where $\mathbf{x}$ is a video segment, $\mathcal{B}$ contains a bag of video segments, and $\mathcal{B}_p$ and $\mathcal{B}_n$, respectively, represents positive and negative bags. The first term is to guarantee the relative anomaly score order, i.e., the anomaly score of the most abnormal video segment in the positive instance bag is greater than that in the negative instance bag. The last two terms are extra optimization constraints, in which the former enforces score smoothness between consecutive video segments while the latter enforces anomaly sparsity, i.e., each video contains only a few abnormal segments.

Advantages. The advantages of deep ranking model-base methods are as follows. (i) The anomaly scores can be optimized directly with adapted loss functions. (ii) They are generally free from the

definitions of anomalies by imposing a weak assumption of the ordinal order between anomaly and normal instances. (iii) This approach may build upon well-established ranking techniques and theories from areas like learning to rank [85, 87, 158].

Disadvantages. Their disadvantages are as follows. (i) At least some form of labeled anomalies are required in these methods, which may not be applicable to applications where such labeled anomalies are not available. The method in Reference [117] is fully unsupervised and obtains some promising performance but there is still a large gap compared to semi-supervised methods. (ii) Since the models are exclusively fitted to detect the few labeled anomalies, they may not be able to generalize to unseen anomalies that exhibit different abnormal features to the labeled anomalies.

Challenges Targeted: Using weak supervision such as pseudo labels or noisy class labels provide some important knowledge of suspicious anomalies, enabling the learning of more expressive low-dimensional representation space and better detection accuracy (CH1 and CH2). The MIL scheme [145] and the pairwise relation prediction [114] provide an easy way to incorporate coarse-grained/limited anomaly labels to detection model learning (CH3). More importantly, the end-to-end anomaly score learning offers straightforward anomaly explanation by backpropagating the activation weights or the gradient of anomaly scores to locate the features that are responsible for large anomaly scores [117] (CH6). In addition, the methods in Reference [114, 117] also work well in data with anomaly contamination or noisy labels (CH4).

6.2 Prior-driven Models

This approach uses a prior distribution to encode and drive the anomaly score learning. Since the anomaly scores are learned in an end-to-end manner, the prior may be imposed on either the internal module or the learning output (i.e., anomaly scores) of the score learning function τ.

Assumptions. The imposed prior captures the underlying (ab)normality of the dataset.

The incorporation of the prior into the internal anomaly scoring function is exemplified by a recent study on the Bayesian inverse reinforcement learning– (IRL) based method [107]. The key intuition is that given an agent that takes a set of sequential data as input, the agent's normal behavior can be understood by its latent reward function, and thus a test sequence is identified as anomaly if the agent assigns a low reward to the sequence. IRL approaches [102] are used to infer the reward function. To learn the reward function more efficiently, a sample-based IRL approach is used. Specifically, the IRL problem is formulated as the following posterior optimization problem:

$$\max_{\Theta} \mathbb{E}_{\mathbf{s}\sim\mathcal{S}}\Big[\log p(\mathbf{s}|\Theta) + \log p(\Theta)\Big], \tag{38}$$

where $p(\mathbf{s}|\Theta) = \frac{1}{Z}\exp\Big(\sum_{(o,a)\in\mathbf{s}} \tau_\Theta(o,a)\Big)$, $\tau_\Theta(o,a)$ is a latent reward function parameterized by Θ, (o,a) is a pair of state and action in the sequence $\mathbf{s}$, Z represents the partition function that is the integral of $\exp\Big(\sum_{(o,a)\in\mathbf{s}} \tau_\Theta(o,a)\Big)$ over all the sequences consistent with the underlying Markov decision process dynamics, $p(\Theta)$ is a prior distribution of Θ, and $\mathcal{S}$ is a set of observed sequences. Since the inverse of the reward yielded by τ is used as the anomaly score, maximizing Equation (38) is equivalent to directly learning the anomaly scores.

At the training stage, a Gaussian prior distribution over the weight parameters of the reward function learning network is assumed, i.e., $\Theta \sim \mathcal{N}(0,\sigma^2)$. The partition function Z is estimated using a set of sequences generated by a sample-generating policy π,

$$Z = \mathbb{E}_{\mathbf{s}\sim\pi}\left[\sum_{(o,a)\in\mathbf{s}} \tau_\Theta(o,a)\right]. \tag{39}$$

The policy π is also represented as a neural network. τ and π are alternatively optimized, i.e., to optimize the reward function τ with a fixed policy π and to optimize π with the updated reward

function τ. Note that τ is instantiated with a bootstrap neural network with multiple output heads in Reference [107]; Equation (38) presents a simplified τ for brevity.

The idea of enforcing a prior on the anomaly scores is explored in Reference [115]. Motivated by the extensive empirical results in Reference [74] that show the anomaly scores in a variety of real-world datasets fits Gaussian distribution very well, the work uses a Gaussian prior to encode the anomaly scores and enable the direct optimization of the scores. That is, it is assumed that the anomaly scores of normal instances are clustered together while that of anomalies deviate far away from this cluster. The prior is leveraged to define a loss function, called deviation loss, which is built upon the well-known contrastive loss [55],

$$L_{dev} = (1 - y_{\mathbf{x}})|dev(\mathbf{x})| + y_{\mathbf{x}} \max\left\{0, m - dev(\mathbf{x})\right\} \quad \text{and} \quad dev(\mathbf{x}) = \frac{\tau(\mathbf{x};\Theta) - \mu_b}{\sigma_b}, \tag{40}$$

where μ_b and σ_b are respectively the estimated mean and standard deviation of the prior $\mathcal{N}(\mu, \sigma)$, $y_{\mathbf{x}} = 1$ if $\mathbf{x}$ is an anomaly and $y_{\mathbf{x}} = 0$ if $\mathbf{x}$ is a normal object, and m is equivalent to a Z-Score confidence interval parameter. μ_b and σ_b are estimated using a set of values $\{r_1, r_2, \ldots, r_l\}$ drawn from $\mathcal{N}(\mu, \sigma)$ for each batch of instances to learn robust representations of normality and abnormality.

The detection model is driven by the deviation loss to push the anomaly scores of normal instances as close as possible to μ while guaranteeing at least m standard deviations between μ and the anomaly scores of anomalies. When $\mathbf{x}$ is an anomaly and it has a negative $dev(\mathbf{x})$, the loss would be particularly large, resulting in large *positive deviations* for all anomalies. As a result, the deviation loss is equivalent to enforcing a statistically significant deviation of the anomaly score of the anomalies from that of normal instances in the upper tail. Further, this Gaussian prior-driven loss also results in well interpretable anomaly scores, i.e., given any anomaly score $\tau(\mathbf{x})$, we can use the Z-score confidence interval $\mu \pm z_p\sigma$ to explain the abnormality of the instance $\mathbf{x}$. This is an important and very practical property that existing methods do not have.

Advantages. The advantages of prior-driven models are as follows. (i) The anomaly scores can be directly optimized w.r.t. a given prior. (ii) It provides a flexible framework for incorporating different prior distributions into the anomaly score learning. Different Bayesian deep learning techniques [156] may be adapted for anomaly detection. (iii) The prior can also result in more interpretable anomaly scores than the other methods.

Disadvantages. Their disadvantages are as follows. (i) It is difficult, if not impossible, to design a universally effective prior for different anomaly detection application scenarios. (ii) The models may work less effectively if the prior does not fit the underlying distribution well.

Challenges Targeted: The prior empowers the models to learn informed low-dimensional representations of different complex data such as high-dimensional data and sequential data (CH1 and CH2). By imposing a prior over anomaly scores, the deviation network method [115] shows promising performance in leveraging a limited amount of labeled anomaly data to enhance the representations of normality and abnormality, substantially boosting the detection recall (CH1 and CH3). The detection models here are driven by a prior distribution w.r.t. anomaly scoring function and work well in data with anomaly contamination in the training data (CH4).

6.3 Softmax Likelihood Models

This approach aims at learning anomaly scores by maximizing the likelihood of events in the training data. Since anomaly and normal instances respectively correspond to rare and frequent patterns, from the probabilistic perspective, normal instances are presumed to be high-probability events whereas anomalies are prone to be low-probability events. Therefore, the negative of the event likelihood can be naturally defined as anomaly score. Softmax likelihood models are shown effective and efficient in achieving this goal via tools like noise contrastive estimation (NCE) [54].

Assumptions. Anomalies and normal instances are respectively low- and high-probability events.

The idea of learning anomaly scores by directly modeling the event likelihood is introduced in [30]. Particularly, the problem is framed as

$$\Theta^* = \arg\max_{\Theta} \sum_{\mathbf{x}\in\mathcal{X}} \log p(\mathbf{x};\Theta), \tag{41}$$

where $p(\mathbf{x};\Theta)$ is the probability of the instance $\mathbf{x}$ (i.e., an event in the event space) with the parameters Θ to be learned. To easy the optimization, $p(\mathbf{x};\Theta)$ is modeled with a softmax function:

$$p(\mathbf{x};\Theta) = \frac{\exp\Big(\tau(\mathbf{x};\Theta)\Big)}{\sum_{\mathbf{x}\in\mathcal{X}} \exp\Big(\tau(\mathbf{x};\Theta)\Big)}, \tag{42}$$

where $\tau(\mathbf{x};\Theta)$ is an anomaly scoring function designed to capture pairwise feature interactions:

$$\tau(\mathbf{x};\Theta) = \sum_{i,j\in\{1,2,\ldots,K\}} w_{ij}\mathbf{z}_i\mathbf{z}_j, \tag{43}$$

where $\mathbf{z}_i$ is a low-dimensional embedding of the ith feature value of $\mathbf{x}$ in the representation space $\mathcal{Z}$, w_{ij} is the weight added to the interaction and is a trainable parameter. Since $\sum_{\mathbf{x}\in\mathcal{X}} \exp\Big(\tau(\mathbf{x};\Theta)\Big)$ is a normalization term, learning the likelihood function p is equivalent to directly optimizing the anomaly scoring function τ. The computation of this explicit normalization term is prohibitively costly, the well-established NCE is used in Reference [30] to learn the following approximated likelihood:

$$\log p(d=1|\mathbf{x};\Theta) + \log \sum_{j=1}^{k} p(d=0|\mathbf{x}'_j;\Theta), \tag{44}$$

where $p(d=1|\mathbf{x};\Theta) = \frac{\exp\Big(\tau(\mathbf{x};\Theta)\Big)}{\exp\Big(\tau(\mathbf{x};\Theta)\Big)+kQ(\mathbf{x}')}$ and $p(d=0|\mathbf{x}';\Theta) = \frac{kQ(\mathbf{x}')}{\exp\Big(\tau(\mathbf{x};\Theta)\Big)+kQ(\mathbf{x}')}$; for each instance $\mathbf{x}$, k noise samples $\mathbf{x}'_{1,\ldots,k} \sim Q$ are generated from some synthetic known 'noise' distribution Q. In Reference [30], a context-dependen method is used to generate the k negative samples by univariate extrapolation of the observed instance $\mathbf{x}$.

The method is primarily designed to detect anomalies in categorical data [30]. Motivated by this application, a similar objective function is adapted to detect abnormal events in heterogeneous attributed bipartite graphs [45]. The problem in Reference [45] is to detect anomalous paths that span both partitions of the bipartite graph. Therefore, $\mathbf{x}$ in Equation (43) is a graph path containing a set of heterogeneous graph nodes, with $\mathbf{z}_i$ and $\mathbf{z}_j$ be the representations of every pair of the nodes in the path. To map attributed nodes into the representation space $\mathcal{Z}$, multilayer perceptron networks and autoencoders are respectively applied to the node features and the graph topology.

Advantages. The advantages of softmax model-based methods are as follows. (i) Different types of interactions can be incorporated into the anomaly score learning process. (ii) The anomaly scores are faithfully optimized w.r.t. the specific abnormal interactions we aim to capture.

Disadvantages. Their disadvantages are as follows. (i) The computation of the interactions can be very costly when the number of features/elements in each data instance is large, i.e., we have $O(D^n)$ time complexity per instance for nth order interactions of D features/elements. (ii) The anomaly score learning is heavily dependent on the quality of the generation of negative samples.

Challenges Targeted: The formulation in this category of methods provides a promising way to learn low-dimensional representations of datasets with heterogeneous data sources (CH2 and CH5). The learned representations often capture more normality/abnormality information from different data sources and thus enable better detection than traditional methods (CH1).

6.4 End-to-end One-class Classification

This category of methods aims to train a one-class classifier that learns to discriminate whether a given instance is normal or not in an end-to-end manner. Different from the methods in Section 5.2.2, this approach does not rely on any existing one-class classification measures such as one-class SVM or SVDD. This approach emerges mainly due to the marriage of GANs and the concept of one-class classification, i.e., adversarially learned one-class classification. The key idea is to learn a one-class discriminator of the normal instances so that it well discriminates those instances from adversarially generated pseudo anomalies. This approach is also very different from the GAN-based methods in Section 5.1.2 due to two key differences. First, the GAN-based methods aim to learn a generative distribution to maximally approximate the real data distribution, achieving a generative model that well captures the normality of the training normal instances; while the methods in this section aim to optimize a discriminative model to separate normal instances from adversarially generated fringe instances. Second, the GAN-based methods define the anomaly scores based on the residual between the real instances and the corresponding generated instances, whereas the methods here directly use the discriminator to classify anomalies, i.e., the discriminator D acts as τ in Equation (33). This section is separated from Sections 5.1.2 and 5.2.2 to highlight the above differences.

Assumptions. (i) Data instances that are approximated to anomalies can be effectively synthesized. (ii) All normal instances can be summarized by a discriminative one-class model.

The idea of adversarially learned one-class (ALOCC) classification is first studied in Reference [135]. The key idea is to train two deep networks, with one network trained as the one-class model to separate normal instances from anomalies while the other network trained to enhance the normal instances and generate distorted outliers. The two networks are instantiated and optimized through the GANs approach. The one-class model is built upon the discriminator network and the generator network is based on a denoising AE [153]. The objective of the AE-empower GAN is defined as

$$\min_{AE}\max_{D} V(D,G) = \mathbb{E}_{\mathbf{x}\sim p_X}\Big[\log D(\mathbf{x})\Big] + \mathbb{E}_{\hat{\mathbf{x}}\sim p_{\hat{X}}}\Big[\log\Big(1 - D\Big(AE(\hat{\mathbf{x}})\Big)\Big)\Big], \tag{45}$$

where $p_{\hat{X}}$ denotes a data distribution of $\mathcal{X}$ corrupted by a Gaussian noise, i.e., $\hat{\mathbf{x}} = \mathbf{x} + \mathbf{n}$ with $\mathbf{n} \sim \mathcal{N}(0, \sigma^2\mathbf{I})$. This objective is jointly optimized with the following data construction error in AE,

$$\ell_{ae} = \|\mathbf{x} - AE(\hat{\mathbf{x}})\|^2. \tag{46}$$

The intuition in Equation (45) is that AE can well reconstruct (and even enhance) normal instances, but it can be confused by input outliers and consequently generates distorted outliers. Through the minimax optimization, the discriminator D learns to better discriminate normal instances from the outliers than using the original data instances. Thus, $D\Big(AE(\hat{\mathbf{x}})\Big)$ can be directly used to detect anomalies. In Reference [135] the outliers are randomly drawn from some classes other than the classes where the normal instances come from.

However, obtaining the reference outliers beyond the given training data as in Reference [135] may be unavailable in many domains. Instead of taking random outliers from other datasets, we can generate fringe data instances based on the given training data and use them as negative reference instances to enable the training of the one-class discriminator. This idea is explored in [103, 174]. One-class adversarial networks (OCAN) is introduced in Reference [174] to leverage the idea of bad GANs [33] to generate fringe instances based on the distribution of the normal training data. Unlike conventional generators in GANs, the generator network in bad GANs is trained to generate data instances that are complementary, rather than matching, to the training data. The

objective of the complement generator is as follows:

$$\min_{G} -\mathcal{H}(p_{\mathcal{Z}}) + \mathbb{E}_{\hat{\mathbf{z}}\sim p_{\mathcal{Z}}} \log p_{\mathcal{X}}(\hat{\mathbf{z}})\mathbb{I}[p_{\mathcal{X}}(\hat{\mathbf{z}}) > \epsilon] + \|\mathbb{E}_{\hat{\mathbf{z}}\sim p_{\mathcal{Z}}} h(\hat{\mathbf{z}}) - \mathbb{E}_{\mathbf{z}\sim p_{\mathcal{X}}} h(\mathbf{z})\|_2, \tag{47}$$

where $\mathcal{H}(\cdot)$ is the entropy, $\mathbb{I}[\cdot]$ is an indicator function, ϵ is a threshold hyperparameter, and h is a feature mapping derived from an intermediate layer of the discriminator. The first two terms are devised to generate low-density instances in the original feature space. However, it is computationally infeasible to obtain the probability distribution of the training data. Instead the density estimation $p_{\mathcal{X}}(\hat{\mathbf{z}})$ is approximated by the discriminator of a regular GAN. The last term is the widely used feature matching loss that helps better generate data instances within the original data space. The objective of the discriminator in OCAN is enhanced with an extra conditional entropy term to enable the detection with high confidence:

$$\max_{D} \mathbb{E}_{\mathbf{x}\sim p_{\mathcal{X}}}\Big[\log D(\mathbf{z})\Big] + \mathbb{E}_{\hat{\mathbf{z}}\sim p_{\mathcal{Z}}}\Big[\log\Big(1 - D(\hat{\mathbf{z}})\Big)\Big] + \mathbb{E}_{\mathbf{x}\sim p_{\mathcal{X}}}\Big[D(\mathbf{x})\log D(\mathbf{x})\Big], \tag{48}$$

In Reference [103], Fence GAN is introduced with the objective to generate data instances tightly lying at the boundary of the distribution of the training data. This is achieved by introducing two loss functions into the generator that enforce the generated instances to be evenly distributed along a sphere boundary of the training data. Formally, the objective of the generator is defined as

$$\min_{G} \mathbb{E}_{\mathbf{z}\sim p_{\mathcal{Z}}}\Big[\log\Big[\Big|\alpha - D\Big(G(\mathbf{z})\Big)\Big|\Big]\Big] + \beta\frac{1}{\mathbb{E}_{\mathbf{z}\sim p_{\mathcal{Z}}}\|G(\mathbf{z}) - \boldsymbol{\mu}\|_2}, \tag{49}$$

where $\alpha \in (0, 1)$ is a hyperparameter used as a discrimination reference score for the generator to generate the boundary instances and $\boldsymbol{\mu}$ is the center of the generated data instances. The first term is called encirclement loss that enforces the generated instances to have the same discrimination score, ideally resulting in instances tightly enclosing the training data. The second term is called dispersion loss that enforces the generated instances to evenly cover the whole boundary.

There have been some other methods introduced to effectively generate the reference instances. For example, uniformly distributed instances can be generated to enforce the normal instances to be distributed uniformly across the latent space [120]; an ensemble of generators is used in Reference [89], with each generator synthesizing boundary instances for one specific cluster of normal instances.

Advantages. The advantages of this category of methods is as follows. (i) Its anomaly classification model is adversarially optimized in an end-to-end fashion. (ii) It can be developed and supported by the affluent techniques and theories of adversarial learning and one-class classification.

Disadvantages. Their disadvantages are as follows. (i) It is difficult to guarantee that the generated reference instances well resemble the unknown anomalies. (ii) The instability of GANs may lead to generated instances with diverse quality and consequently unstable anomaly classification performance. This issue is recently studied in Reference [169], which shows that the performance of this type of anomaly detectors can fluctuate drastically in different training steps. (iii) Its applications are limited to semi-supervised anomaly detection scenarios.

Challenges Targeted: The adversially learned one-class classifiers learn to generate realistic fringe/boundary instances, enabling the learning of expressive low-dimensional normality representations (CH1 and CH2).

7 ALGORITHMS AND DATASETS

7.1 Representative Algorithms

To gain a more in-depth understanding of methods in this area, in Table 2 we summarize some key characteristics of representative algorithms from each category of methods. Since these methods

Table 2. Key Characteristics of 30 Representative Algorithms

Method	Ref.	Sup.	Objective	DA	DP	PT	Archit.	Activation	# layers	Loss	Data
OADA	[65] (4)	Semi	Reconstruction	Yes	No	No	AE	ReLU	3	MSE	Video
Replicator	[57] (5.1.1)	Unsup.	Reconstruction	No	No	No	AE	Tanh	2	MSE	Tabular
RandNet	[29] (5.1.1)	Unsup.	Reconstruction	No	Yes	Yes	AE	ReLU	3	MSE	Tabular
RDA	[175] (5.1.1)	Semi	Reconstruction	No	No	No	AE	Sigmoid	2	MSE	Tabular
UODA	[91] (5.1.1)	Semi	Reconstruction	No	No	Yes	AE & RNN	Sigmoid	4	MSE	Sequence
AnoGAN	[138] (5.1.2)	Semi	Generative	No	No	No	Conv.	ReLU	4	MAE	Image
EBGAN	[170] (5.1.2)	Semi	Generative	No	No	No	Conv. & MLP	ReLU/lReLU	3-4	GAN	Image & Tabular
FFP	[86] (5.1.3)	Semi	Predictive	Yes	No	Yes	Conv.	ReLU	10	MAE/MSE	Video
LSA	[1] (5.1.3)	Semi	Predictive	No	No	No	Conv.	lReLU	4-7	MSE & KL	video
GT	[48] (5.1.4)	Semi	Classification	Yes	Yes	No	Conv.	ReLU	10-16	CE	Image
E^3Outlier	[157] (5.1.4)	Semi	Classification	Yes	Yes	No	Conv.	ReLU	10	CE	Image
REPEN	[112] (5.2.1)	Unsup.	Distance	No	No	No	MLP	ReLU	1	Hinge	Tabular
RDP	[155] (5.2.1)	Unsup.	Distance	No	No	No	MLP	lReLU	1	MSE	Tabular
AE-1SVM	[104] (5.2.2)	Unsup.	One-class	No	No	No	AE & Conv.	Sigmoid	2-5	Hinge	Tabular & image
DeepOC	[161] (5.2.2)	Semi	One-class	No	No	No	3D Conv.	ReLU	5	Hinge	Video
Deep SVDD	[132] (5.2.2)	Semi	One-class	No	No	Yes	Conv.	lReLU	3-4	Hinge	Image
Deep SAD	[133] (5.2.2)	Semi	One-class	No	No	Yes	Conv. & MLP	lReLU	3-4	Hinge	Image & Tabular
DEC	[162] (5.2.3)	Unsup.	Clustering	No	Yes	Yes	MLP	ReLU	4	KL	Image & Tabular
DAGMM	[179] (5.2.3)	Unsup.	Clustering	No	Yes	No	AE & MLP	Tanh	4-6	Likelihood	Tabular
SDOR	[117] (6.1)	Unsup.	Anomaly scores	No	No	Yes	ResNet & MLP	ReLU	50 + 2	MAE	Video
PReNet	[114] (6.1)	Weak	Anomaly scores	Yes	No	No	MLP	ReLU	2-4	MAE	Tabular
MIL	[145] (6.1)	Weak	Anomaly scores	No	Yes	Yes	3DConv. & MLP	ReLU	18/34 + 3	Hinge	Video
PUP	[107] (6.2)	Unsup.	Anomaly scores	No	No	No	MLP	ReLU	3	Likelihood	Sequence
DevNet	[115] (6.2)	Weak	Anomaly scores	No	No	No	MLP	ReLU	2-4	Deviation	Tabular
APE	[30] (6.3)	Unsup.	Anomaly scores	No	No	No	MLP	Sigmoid	3	Softmax	Tabular
AEHE	[45] (6.3)	Unsup.	Anomaly scores	No	No	No	AE & MLP	ReLU	4	Softmax	Graph
ALOCC	[135] (6.4)	Semi	Anomaly scores	Yes	No	No	AE & CNN	lReLU	5	GANs	Image
OCAN	[174] (6.4)	Semi	Anomaly scores	No	No	Yes	LSTM-AE & MLP	ReLU	4	GANs	Sequence
Fence GAN	[103] (6.4)	Semi	Anomaly scores	No	Yes	No	Conv. & MLP	lReLU/Sigmoid	4-5	GANs	Image & Tabular
OCGAN	[120] (6.4)	Semi	Anomaly scores	No	No	No	Conv.	ReLU/Tanh	3	GANs	Image

DA, DP, PT, and Archit. are short for data augmentation, dropout, pre-training, and architecture, respectively. # layers account for all layers except the input layer. lReLU represents leaky ReLU.

are evaluated on diverse datasets, it is difficult to have an universal meta-analysis of their empirical performance. Instead, some main observations w.r.t. the model design are summarized as follows: (i) most methods operate in an unsupervised or semi-supervised mode; (ii) deep learning tricks like data augmentation, dropout and pre-training are under-explored; (iii) the network architecture used is not that deep, with a majority of the methods having no more than five network layers; (iv) (leaky) ReLU is the most popular activation function; and (v) diverse backbone networks can be used to handle different types of input data. The source code of most of these algorithms is publicly accessible. We summarize those source codes in Table A1 in Appendix A to facilitate the access.

7.2 Datasets with Real Anomalies

One main obstacle to the development of anomaly detection is the lack of real-world datasets with real anomalies. Many studies (e.g., References [3, 48, 103, 132, 157, 170, 175]) evaluate the performance of their presented methods on datasets converted from popular classification data for this reason. This way may fail to reflect the performance of the methods in real-world anomaly detection applications. We summarize a collection of 21 publicly available real-world datasets with real anomalies in Table 3 to promote the performance evaluation on these datasets. The datasets cover a wide range of popular application domains presented in a variety of data types. Only large-scale and/or high-dimensional complex datasets are included here to provide challenging

Table 3. 21 Publicly Accessible Real-world Datasets with Real Anomalies

Domain	Data	Size	Dimension	Anomaly (%)	Type	Reference
Intrusion detection	KDD Cup 99 [13]	4,091-567,497	41	0.30%-7.70%	Tabular	[57, 103, 104, 179]
Intrusion detection	UNSW-NB15 [100]	257,673	49	≤9.71%	Streaming	[114, 115]
Excitement prediction	KDD Cup 14	619,326	10	6.00%	Tabular	[114, 115]
Dropout prediction	KDD Cup 15	35,091	27	0.10%-0.40%	Sequence	[91]
Malicious URLs detection	URL [93]	2.4m	3.2m	33.04%	Streaming	[112]
Spam detection	Webspam [160]	350,000	16.6m	39.61%	Tabular/text	[112]
Fraud detection	Credit-card-fraud [34]	284,807	30	0.17%	Streaming	[114, 115, 174]
Vandal detection	UMDWikipedia [76]	34,210	N/A	50.00%	Sequence	[174]
Mutant activity detection	p53 Mutants [13]	16,772	5,408	0.48%	Tabular	[112]
Internet ads detection	AD [13]	3,279	1,555	14.00%	Tabular	[112]
Disease detection	Thyroid [13]	7,200	21	7.40%	Tabular	[114, 115, 133, 179]
Disease detection	Arrhythmia [13]	452	279	14.60%	Tabular	[116, 133, 179]
Defect detection	MVTec AD	5,354	N/A	35.26%	Image	[15]
Video surveillance	UCSD Ped 1 [81]	14,000 frames	N/A	28.6%	Video	[117, 161]
Video surveillance	UCSD Ped 2 [81]	4,560 frames	N/A	35.9%	Video	[117, 161]
Video surveillance	UMN [106]	7,739 frames	N/A	15.5%-18.1%	Video	[117]
Video surveillance	Avenue [90]	30,652 frames	N/A	12.46%	Video	[161]
Video surveillance	ShanghaiTech Campus	317,398 frames	N/A	5.38%	Video	[86]
Video surveillance	UCF-Crime	1,900 videos (13.8m frames)	N/A	13 crimes	Video	[145]
System log analysis	HDFS Log [164]	11.2m	N/A	2.90%	Sequence	[40]
System log analysis	OpenStack log	1.3m	N/A	7.00%	Sequence	[40]

testbeds for deep anomaly detection. In addition, a continuously updated collection of widely used anomaly detection datasets (including some pre-processed datasets from Table 3) is available at https://git.io/JTs93.

8 CONCLUSIONS AND FUTURE OPPORTUNITIES

In this work, we review 12 diverse modeling perspectives on harnessing deep learning techniques for anomaly detection. We also discuss how these methods address some notorious anomaly detection challenges to demonstrate the importance of deep anomaly detection. Through such a review, we identify some exciting opportunities as follows.

8.1 Exploring Anomaly-supervisory Signals

Informative supervisory signals are the key for deep anomaly detection to learn accurate anomaly scores or expressive representations of normality/abnormality. While a wide range of unsupervised or self-supervised supervisory signals have been explored, as discussed in Section 5.1, to learn the representations, a key issue for these formulations is that their objective functions are generic but not optimized specifically for anomaly detection. Anomaly measure-dependent feature learning in Section 5.2 helps address this issue by imposing constraints derived from traditional anomaly measures. However, these constraints can have some inherent limitations, e.g., implicit assumptions in the anomaly measures. It is critical to explore *new sources of anomaly-supervisory signals* that lie beyond the widely used formulations such as data reconstruction and GANs, and have weak assumptions on the anomaly distribution. Another possibility is to develop *domain-driven anomaly detection* by leveraging domain knowledge [23] such as application-specific knowledge of anomaly and/or expert rules as the supervision source.

8.2 Deep Weakly Supervised Anomaly Detection

Deep weakly supervised anomaly detection [114] aims at leveraging deep neural networks to learn anomaly-informed detection models with some weakly supervised anomaly signals, e.g.,

partially/inexactly/inaccurately labeled anomaly data. These labeled data provide important knowledge of anomaly and can be a major driving force to lift detection recall rates [112, 114, 115, 145, 147]. One exciting opportunity is to utilize a small number of accurate labeled anomaly examples to enhance detection models as they are often available in real-world applications, e.g., some intrusions/frauds from deployed detection systems/end-users and verified by human experts. However, since anomalies can be highly heterogeneous, there can be unknown/novel anomalies that lie beyond the span set of the given anomaly examples. Thus, one important direction here is *unknown anomaly detection*, in which we aim to build detection models that are generalized from the limited labeled anomalies to unknown anomalies. Some recent studies [113–115, 133] show that deep detection models are able to learn abnormality that lie beyond the scope of the given anomaly examples. It would be important to further understand and explore the extent of the generalizability and to develop models to further improve the accuracy performance.

To detect anomalies that belong to the same classes of the given anomaly examples can be as important as the detection of novel/unknown anomalies. Thus, another important direction is to develop *data-efficient anomaly detection* or *few-shot anomaly detection*, in which we aim at learning highly expressive representations of the known anomaly classes given only limited anomaly examples [112, 114, 115, 152]. It should be noted that the limited anomaly examples may come from different anomaly classes, and thus, exhibit completely different manifold/class features. This scenarios is fundamentally different from the general few-shot learning [159], in which the limited examples are class-specific and assumed to share the same manifold/class structure. Additionally, as shown in Table 2, the network architectures are mostly not as deep as that in other machine learning tasks. This may be partially due to the limitation of the labeled training data size. It is important to explore the possibility of leveraging those small labeled data to learn more powerful detection models with deeper architectures. Also, inexact or inaccurate (e.g., coarse-grained) anomaly labels are often inexpensive to collect in some applications [145]; learning deep detection models with this weak supervision is important in these scenarios.

8.3 Large-scale Normality Learning

Large-scale unsupervised/self-supervised representation learning has gained tremendous success in enabling downstream learning tasks [35, 122]. This is particular important for learning tasks, in which it is difficult to obtain sufficient labeled data, such as anomaly detection (see Section 2.1). The goal is to first learn transferable pre-trained representation models from large-scale unlabeled data in an unsupervised/self-supervised mode, and then fine-tune detection models in a semi-supervised mode. The self-supervised classification-based methods in Section 5.1.3 may provide some initial sources of supervision for the normality learning. However, precautions must be taken to ensure that (i) the unlabeled data are free of anomaly contamination and/or (ii) the representation learning methods are robust w.r.t. possible anomaly contamination. This is because most methods in Section 5 implicitly assume that the training data are clean and do not contain any noise/anomaly instances. This robustness is important in both the pre-trained modeling and the fine-tuning stage. Additionally, anomalies and datasets in different domains vary significantly, so the large-scale normality learning may need to be domain/application specific.

8.4 Deep Detection of Complex Anomalies

Most deep anomaly detection methods focus on point anomalies, showing substantially better performance than traditional methods. However, deep models for conditional/group anomalies have been significantly less explored. Deep learning has superior capability in capturing complex temporal/spatial dependence and learning representations of a set of unordered data points; it

is important to explore whether deep learning could also gain similar success in detecting such complex anomalies. Novel neural network layers or objectives functions may be required.

Similar to traditional methods, current deep anomaly detection mainly focus on single data sources. *Multimodal anomaly detection* is a largely unexplored research area. It is difficult for traditional approaches to bridge the gap presented by those multimodal data. Deep learning has demonstrated tremendous success in learning feature representations from different types of raw data for anomaly detection [37, 65, 91, 112, 135]; it is also able to concatenate the representations from different data sources to learn unified representations [49], so deep approaches present important opportunities of multimodal anomaly detection.

8.5 Interpretable and Actionable Deep Anomaly Detection

Current deep anomaly detection mainly focuses on the detection accuracy aspect. *Interpretable deep anomaly detection* and *actionable deep anomaly detection* are essential for understanding model decisions and results, mitigating any potential bias/risk against human users and enabling decision-making actions. In recent years, there have been some studies [7, 8, 42, 142, 154] that explore the anomaly explanation issues by searching for a subset of features that makes a reported anomaly most abnormal. The abnormal feature selection methods [12, 110, 111] may also be utilized for anomaly explanation purpose. The anomalous feature searching in these methods is independent from the anomaly detection methods, and thus, may be used to provide explanation of anomalies identified by any detection methods, including deep models. However, this model-agnostic approach may render the explanation less useful, because they cannot provide a genuine understanding of the mechanisms underlying specific detection models, resulting in weak interpretability and actionability (e.g., quantifying the impact of detected anomalies and mitigation actions). Deep models with inherent capability to provide anomaly explanation is important, such as Reference [117]. To achieve this, methods for deep model explanation [41] and actionable knowledge discovery [23] could be explored with deep anomaly detection models.

8.6 Novel Applications and Settings

There have been some exciting emerging research applications and problem settings, into which there could be some important opportunities of extending deep detection methods. First, out-of-distribution (OOD) detection [60, 79, 127] is a closely related area, which detects data instances that are drawn far away from the training distribution. This is an essential technique to enable machine learning systems to deal with instances of novel classes in open-world environments. OOD detection is also an anomaly detection task, but in OOD detection it is generally assumed that fine-grained normal class labels are available during training, and we need to retain the classification accuracy of these normal classes while performing accurate OOD detection. Second, *curiosity learning* [18, 19, 118] aims at learning a bonus reward function in reinforcement learning with sparse rewards. Particularly, reinforcement learning algorithms often fail to work in an environment with very sparse rewards. Curiosity learning addresses this problem by augmenting the environment with a bonus reward in addition to the original sparse rewards from the environment. This bonus reward is defined typically based on the novelty or rarity of the states, i.e., the agent receives large bonus rewards if it discovers novel/rare states. The novel/rare states are concepts similar to anomalies. Therefore, it would be interesting to explore how deep anomaly detection could be utilized to enhance this challenging reinforcement learning problem; conversely, there can be opportunities to leverage curiosity learning techniques for anomaly detection, such as the method in Reference [155]. Third, most shallow and deep models for anomaly detection assume that the abnormality of data instances is independent and identically distributed (IID), while the abnormality in real applications may suffer from some non-IID characteristics,

e.g., the abnormality of different instances/features is interdependent and/or heterogeneous [108]. For example, the abnormality of multiple synchronized disease symptoms is mutually reinforced in early detection of diseases. This requires *non-IID anomaly detection* [108] that is dedicated to learning such non-IID abnormality. This task is crucial in complex scenarios, e.g., where anomalies have only subtle deviations and are masked in the data space if not considering these non-IID abnormality characteristics. Last, other interesting applications include detection of adversarial examples [51, 119], anti-spoofing in biometric systems [46, 121], and early detection of rare catastrophic events (e.g., financial crisis [24] and other black swan events [11]).

REFERENCES

[1] Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. 2019. Latent space autoregression for novelty detection. In *CVPR*. 481–490.

[2] Charu C. Aggarwal. 2017. *Outlier Analysis*. Springer.

[3] Samet Akcay, Amir Atapour-Abarghouei, and Toby P. Breckon. 2018. GANomaly: Semi-supervised anomaly detection via adversarial training. In *ACCV*. Springer, 622–637.

[4] Leman Akoglu, Hanghang Tong, and Danai Koutra. 2015. Graph based anomaly detection and description: A survey. *Data Min. Knowl. Discov.* 29, 3 (2015), 626–688.

[5] Elie Aljalbout, Vladimir Golkov, Yawar Siddiqui, Maximilian Strobel, and Daniel Cremers. 2018. Clustering with deep learning: Taxonomy and new methods. *arXiv:1801.07648*. Retrieved from https://arxiv.org/abs/1801.07648.

[6] J. Andrews, Thomas Tanay, Edward J. Morton, and Lewis D. Griffin. 2016. Transfer representation-learning for anomaly detection. In *PMLR*.

[7] Fabrizio Angiulli, Fabio Fassetti, Giuseppe Manco, and Luigi Palopoli. 2017. Outlying property detection with numerical attributes. *Data Min. Knowl. Discov.* 31, 1 (2017), 134–163.

[8] Fabrizio Angiulli, Fabio Fassetti, and Luigi Palopoli. 2009. Detecting outlying properties of exceptional objects. *ACM Trans. Database Syst.* 34, 1 (2009), 1–62.

[9] Fabrizio Angiulli and Clara Pizzuti. 2002. Fast outlier detection in high dimensional spaces. In *PKDD*. Springer, 15–27.

[10] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *ICML*. 214–223.

[11] Terje Aven. 2016. Risk assessment and risk management: Review of recent advances on their foundation. *Eur. J. Operat. Res.* 253, 1 (2016), 1–13.

[12] Fatemeh Azmandian, Ayse Yilmazer, Jennifer G. Dy, Javed A. Aslam, and David R. Kaeli. 2012. GPU-accelerated feature selection for outlier detection using the local kernel density ratio. In *ICDM*. IEEE, 51–60.

[13] Kevin Bache and Moshe Lichman. 2013. UCI machine learning repository. Retrieved from http://archive.ics.uci.edu/ml.

[14] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 8 (2013), 1798–1828.

[15] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. 2019. MVTec AD–A comprehensive real-world dataset for unsupervised anomaly detection. In *CVPR*. 9592–9600.

[16] Azzedine Boukerche, Lining Zheng, and Omar Alfandi. 2020. Outlier detection: Methods, models and classifications. *Comput. Surv.* (2020).

[17] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. LOF: Identifying density-based local outliers. *ACM SIGMOD Rec.* 29, 2 (2000), 93–104.

[18] Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A. Efros. 2019. Large-scale study of curiosity-driven learning. In *ICLR*.

[19] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. 2019. Exploration by random network distillation. In *ICLR*.

[20] Guilherme O. Campos, Arthur Zimek, Jörg Sander, Ricardo J. G. B. Campello, Barbora Micenková, Erich Schubert, Ira Assent, and Michael E. Houle. 2016. On the evaluation of unsupervised outlier detection: Measures, datasets, and an empirical study. *Data Min. Knowl. Discov.* 30, 4 (2016), 891–927.

[21] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. 2011. Robust principal component analysis? *J. ACM* 58, 3 (2011), 1–37.

[22] Longbing Cao. 2015. Coupling learning of complex interactions. *Inf. Process. Manage.* 51, 2 (2015), 167–186.

[23] Longbing Cao, Philip S. Yu, Chengqi Zhang, and Yanchang Zhao. 2010. *Domain Driven Data Mining*. Springer.

[24] Wei Cao and Longbing Cao. 2015. Financial crisis forecasting via coupled market state analysis. *IEEE Intell. Syst.* 30, 2 (2015), 18–25.

[25] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *ECCV*. 132–149.
[26] Raghavendra Chalapathy and Sanjay Chawla. 2019. Deep learning for anomaly detection: A survey. *arXiv:1901.03407*. Retrieved from https://arxiv.org/abs/1901.03407.
[27] Raghavendra Chalapathy, Aditya Krishna Menon, and Sanjay Chawla. 2018. Anomaly detection using one-class neural networks. *arXiv:1802.06360*. Retrieved from https://arxiv.org/abs/1802.06360.
[28] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *Comput. Surv.* 41, 3 (2009), 15.
[29] Jinghui Chen, Saket Sathe, Charu Aggarwal, and Deepak Turaga. 2017. Outlier detection with autoencoder ensembles. In *SDM*. 90–98.
[30] Ting Chen, Lu-An Tang, Yizhou Sun, Zhengzhang Chen, and Kai Zhang. 2016. Entity embedding-based anomaly detection for heterogeneous categorical events. In *IJCAI*. 1396–1403.
[31] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Mach. Learn.* 20, 3 (1995), 273–297.
[32] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A. Bharath. 2018. Generative adversarial networks: An overview. *IEEE Sign. Process. Mag.* 35, 1 (2018), 53–65.
[33] Zihang Dai, Zhilin Yang, Fan Yang, William W. Cohen, and Russ R. Salakhutdinov. 2017. Good semi-supervised learning that requires a bad gan. In *NeurIPS*. 6510–6520.
[34] Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, and Gianluca Bontempi. 2017. Credit card fraud detection: A realistic modeling and a novel learning strategy. *IEEE Trans. Neural Netw. Learn. Syst.* 29, 8 (2017), 3784–3797.
[35] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*. Retrieved from https://arxiv.org/abs/1810.04805.
[36] Nat Dilokthanakul, Pedro A. M. Mediano, Marta Garnelo, Matthew C. H. Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. 2017. Deep unsupervised clustering with gaussian mixture variational autoencoders. In *ICLR*.
[37] Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. 2019. Deep anomaly detection on attributed networks. In *SDM*. 594–602.
[38] Carl Doersch. 2016. Tutorial on variational autoencoders. *arXiv:1606.05908*. Retrieved from https://arxiv.org/abs/1606.05908.
[39] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. 2017. Adversarial feature learning. In *ICLR*.
[40] Min Du, Feifei Li, Guineng Zheng, and Vivek Srikumar. 2017. Deeplog: Anomaly detection and diagnosis from system logs through deep learning. In *CCS*. 1285–1298.
[41] Mengnan Du, Ninghao Liu, and Xia Hu. 2019. Techniques for interpretable machine learning. *Commun. ACM* 63, 1 (2019), 68–77.
[42] Lei Duan, Guanting Tang, Jian Pei, James Bailey, Akiko Campbell, and Changjie Tang. 2015. Mining outlying aspects on numeric data. *Data Min. Knowl. Discov.* 29, 5 (2015), 1116–1151.
[43] Andrew F. Emmott, Shubhomoy Das, Thomas Dietterich, Alan Fern, and Weng-Keen Wong. 2013. Systematic construction of anomaly detection benchmarks from real data. In *KDD Workshop*. 16–21.
[44] Sarah M. Erfani, Sutharshan Rajasegarar, Shanika Karunasekera, and Christopher Leckie. 2016. High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recogn.* 58 (2016), 121–134.
[45] Shaohua Fan, Chuan Shi, and Xiao Wang. 2018. Abnormal event detection via heterogeneous information network embedding. In *CIKM*. 1483–1486.
[46] Soroush Fatemifar, Shervin Rahimzadeh Arashloo, Muhammad Awais, and Josef Kittler. 2019. Spoofing attack detection by anomaly detection. In *ICASSP*. IEEE, 8464–8468.
[47] Kamran Ghasedi Dizaji, Amirhossein Herandi, Cheng Deng, Weidong Cai, and Heng Huang. 2017. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *ICCV*. 5736–5745.
[48] Izhak Golan and Ran El-Yaniv. 2018. Deep anomaly detection using geometric transformations. In *NeurIPS*. 9758–9769.
[49] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press.
[50] Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. 2014. Deep AutoRegressive networks. In *ICML*. 1242–1250.
[51] Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. 2017. On the (statistical) detection of adversarial examples. *arXiv:1702.06280*. Retrieved from https://arxiv.org/abs/1702.06280.
[52] Frank E Grubbs. 1969. Procedures for detecting outlying observations in samples. *Technometrics* 11, 1 (1969), 1–21.
[53] Manish Gupta, Jing Gao, Charu C. Aggarwal, and Jiawei Han. 2013. Outlier detection for temporal data: A survey. *IEEE Trans. Knowl. Data Eng.* 26, 9 (2013), 2250–2267.

[54] Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*. 297–304.
[55] R. Hadsell, S. Chopra, and Y. LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *CVPR*, Vol. 2. 1735–1742.
[56] Mahmudul Hasan, Jonghyun Choi, Jan Neumann, Amit K. Roy-Chowdhury, and Larry S. Davis. 2016. Learning temporal regularity in video sequences. In *CVPR*. 733–742.
[57] Simon Hawkins, Hongxing He, Graham Williams, and Rohan Baxter. 2002. Outlier detection using replicator neural networks. In *DaWaK*.
[58] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.
[59] Zengyou He, Xiaofei Xu, and Shengchun Deng. 2003. Discovering cluster-based local outliers. *Pattern Recogn. Lett.* 24, 9–10 (2003), 1641–1650.
[60] Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*.
[61] Geoffrey E. Hinton and Ruslan R. Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *Science* 313, 5786 (2006), 504–507.
[62] Victoria Hodge and Jim Austin. 2004. A survey of outlier detection methodologies. *Artif. Intell. Rev.* 22, 2 (2004), 85–126.
[63] Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Li F Fei-Fei, and Juan Carlos Niebles. 2018. Learning to decompose and disentangle representations for video prediction. In *NeurIPS*. 517–526.
[64] Yi-an Huang, Wei Fan, Wenke Lee, and Philip S. Yu. 2003. Cross-feature analysis for detecting ad-hoc routing anomalies. In *ICDCS*. IEEE, 478–487.
[65] Radu Tudor Ionescu, Fahad Shahbaz Khan, Mariana-Iuliana Georgescu, and Ling Shao. 2019. Object-centric auto-encoders and dummy anomalies for abnormal event detection in video. In *CVPR*. 7842–7851.
[66] Radu Tudor Ionescu, Sorina Smeureanu, Bogdan Alexe, and Marius Popescu. 2017. Unmasking the abnormal events in video. In *ICCV*. 2895–2903.
[67] Mon-Fong Jiang, Shian-Shyong Tseng, and Chih-Ming Su. 2001. Two-phase clustering process for outliers detection. *Pattern Recogn. Lett.* 22, 6–7 (2001), 691–700.
[68] ShengYi Jiang, Xiaoyu Song, Hui Wang, Jian-Jun Han, and Qing-Hua Li. 2006. A clustering-based method for unsupervised intrusion detections. *Pattern Recogn. Lett.* 27, 7 (2006), 802–810.
[69] Xinwei Jiang, Junbin Gao, Xia Hong, and Zhihua Cai. 2014. Gaussian processes autoencoder for dimensionality reduction. In *PAKDD*. Springer, 62–73.
[70] Fabian Keller, Emmanuel Muller, and Klemens Bohm. 2012. HiCS: High contrast subspaces for density-based outlier ranking. In *ICDE*. IEEE, 1037–1048.
[71] Tung Kieu, Bin Yang, Chenjuan Guo, and Christian S. Jensen. 2019. Outlier detection for time series with recurrent autoencoder ensembles. In *IJCAI*.
[72] Edwin M. Knorr and Raymond T. Ng. 1999. Finding intensional knowledge of distance-based outliers. In *VLDB*, Vol. 99. 211–222.
[73] Edwin M. Knorr, Raymond T. Ng, and Vladimir Tucakov. 2000. Distance-based outliers: Algorithms and applications. *VLDB J.* 8, 3–4 (2000), 237–253.
[74] Hans-Peter Kriegel, Peer Kroger, Erich Schubert, and Arthur Zimek. 2011. Interpreting and unifying outlier scores. In *SDM*. 13–24.
[75] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *NeurIPS*. 1097–1105.
[76] Srijan Kumar, Francesca Spezzano, and V. S. Subrahmanian. 2015. Vews: A wikipedia vandal early warning system. In *KDD*. 607–616.
[77] Aleksandar Lazarevic and Vipin Kumar. 2005. Feature bagging for outlier detection. In *KDD*. ACM, 157–166.
[78] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.
[79] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. 2018. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *ICLR*.
[80] Ping Li, Trevor J. Hastie, and Kenneth W. Church. 2006. Very sparse random projections. In *KDD*. 287–296.
[81] Weixin Li, Vijay Mahadevan, and Nuno Vasconcelos. 2013. Anomaly detection and localization in crowded scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* 36, 1 (2013), 18–32.
[82] Binbing Liao, Jingqing Zhang, Chao Wu, Douglas McIlwraith, Tong Chen, Shengwen Yang, Yike Guo, and Fei Wu. 2018. Deep sequence learning with auxiliary information for traffic prediction. In *KDD*. 537–546.

[83] Weixian Liao, Yifan Guo, Xuhui Chen, and Pan Li. 2018. A unified unsupervised gaussian mixture variational autoencoder for high dimensional outlier detection. In *IEEE Big Data*. IEEE, 1208–1217.
[84] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2012. Isolation-based anomaly detection. *ACM Trans. Knowl. Discov. Data* 6, 1 (2012), 3.
[85] Tie-Yan Liu et al. 2009. Learning to rank for information retrieval. *Found. Trends Inf. Retriev.* 3, 3 (2009), 225–331.
[86] Wen Liu, Weixin Luo, Dongze Lian, and Shenghua Gao. 2018. Future frame prediction for anomaly detection–A new baseline. In *CVPR*. 6536–6545.
[87] Xialei Liu, Joost van de Weijer, and Andrew D. Bagdanov. 2018. Leveraging unlabeled data for crowd counting by learning to rank. In *CVPR*. 7661–7669.
[88] Yusha Liu, Chun-Liang Li, and Barnabás Póczos. 2018. Classifier two sample test for video anomaly detection. In *BMVC*.
[89] Yezheng Liu, Zhe Li, Chong Zhou, Yuanchun Jiang, Jianshan Sun, Meng Wang, and Xiangnan He. 2019. Generative adversarial active learning for unsupervised outlier detection. *IEEE Trans. Knowl. Data Eng.* (2019).
[90] Cewu Lu, Jianping Shi, and Jiaya Jia. 2013. Abnormal event detection at 150 fps in matlab. In *ICCV*. 2720–2727.
[91] Weining Lu, Yu Cheng, Cao Xiao, Shiyu Chang, Shuai Huang, Bin Liang, and Thomas Huang. 2017. Unsupervised sequential outlier detection with deep architectures. *IEEE Trans. Image Process.* 26, 9 (2017), 4321–4330.
[92] Weixin Luo, Wen Liu, and Shenghua Gao. 2017. Remembering history with convolutional lstm for anomaly detection. In *ICME*. IEEE, 439–444.
[93] Justin Ma, Lawrence K. Saul, Stefan Savage, and Geoffrey M. Voelker. 2009. Identifying suspicious URLs: An application of large-scale online learning. In *ICML*. ACM, 681–688.
[94] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. 2010. Anomaly detection in crowded scenes. In *CVPR*. 1975–1981.
[95] Alireza Makhzani and Brendan Frey. 2014. K-sparse autoencoders. In *ICLR*.
[96] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. 2016. LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv:1607.00148*. Retrieved from https://arxiv.org/abs/1607.00148.
[97] Erik Marchi, Fabio Vesperini, Felix Weninger, Florian Eyben, Stefano Squartini, and Björn Schuller. 2015. Non-linear prediction with LSTM recurrent neural networks for acoustic novelty detection. In *IJCNN*. IEEE, 1–7.
[98] Michael Mathieu, Camille Couprie, and Yann LeCun. 2016. Deep multi-scale video prediction beyond mean square error. In *ICLR*.
[99] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. 2017. Unrolled generative adversarial networks. In *ICLR*.
[100] Nour Moustafa and Jill Slay. 2015. UNSW-NB15: A comprehensive data set for network intrusion detection systems. In *MilCIS*. 1–6.
[101] Mary M. Moya, Mark W. Koch, and Larry D. Hostetler. 1993. *One-class Classifier Networks for Target Recognition Applications*. Technical Report. NASA STI/Recon Technical Report N.
[102] Andrew Y. Ng and Stuart J. Russell. 2000. Algorithms for inverse reinforcement learning. In *ICML*. Morgan Kaufmann Publishers Inc., 663–670.
[103] Cuong Phuc Ngo, Amadeus Aristo Winarto, Connie Kou Khor Li, Sojeong Park, Farhan Akram, and Hwee Kuan Lee. 2019. Fence GAN: Towards better anomaly detection. *arXiv:1904.01209*. Retrieved from https://arxiv.org/abs/1904.01209.
[104] Minh-Nghia Nguyen and Ngo Anh Vien. 2018. Scalable and interpretable one-class svms with deep learning and random fourier features. In *ECML-PKDD*. Springer, 157–172.
[105] Keith Noto, Carla Brodley, and Donna Slonim. 2012. FRaC: A feature-modeling approach for semi-supervised and unsupervised anomaly detection. *Data Min. Knowl. Discov.* 25, 1 (2012), 109–133.
[106] University of Minnesota. 2020. UMN Unusual Crowd Activity data set. Retrieved May 30, 2020 from http://mha.cs.umn.edu/Movies/Crowd-Activity-All.avi.
[107] Min-hwan Oh and Garud Iyengar. 2019. Sequential anomaly detection using inverse reinforcement learning. In *KDD*. 1480–1490.
[108] Guansong Pang. 2019. *Non-IID Outlier Detection with Coupled Outlier Factors*. Ph.D. Dissertation.
[109] 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 *AAAI*. 3892–3899.
[110] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. 2016. Unsupervised feature selection for outlier detection by modelling hierarchical value-feature couplings. In *ICDM*. IEEE, 410–419.
[111] 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 *IJCAI*. 2585–2591.

[112] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. 2018. Learning representations of ultrahigh-dimensional data for random distance-based outlier detection. In *KDD*. 2041–2050.
[113] Guansong Pang, Anton van den Hengel, Chunhua Shen, and Longbing Cao. 2020. Deep reinforcement learning for unknown anomaly detection. *arXiv:2009.06847*. Retrieved from https://arxiv.org/abs/2009.06847.
[114] Guansong Pang, Chunhua Shen, Huidong Jin, and Anton van den Hengel. 2019. Deep weakly-supervised anomaly detection. *arXiv:1910.13601*. Retrieved from https://arxiv.org/abs/1910.13601.
[115] Guansong Pang, Chunhua Shen, and Anton van den Hengel. 2019. Deep anomaly detection with deviation networks. In *KDD*. 353–362.
[116] Guansong Pang, Kai Ming Ting, and David Albrecht. 2015. LeSiNN: Detecting anomalies by identifying least similar nearest neighbours. In *ICDM Workshop*. IEEE, 623–630.
[117] Guansong Pang, Cheng Yan, Chunhua Shen, Anton van den Hengel, and Xiao Bai. 2020. Self-trained deep ordinal regression for end-to-end video anomaly detection. In *CVPR*. 12173–12182.
[118] Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, and Trevor Darrell. 2017. Curiosity-driven exploration by self-supervised prediction. In *ICML*. 2778–2787.
[119] Andrea Paudice, Luis Muñoz-González, Andras Gyorgy, and Emil C. Lupu. 2018. Detection of adversarial training examples in poisoning attacks through anomaly detection. *arXiv:1802.03041*. Retrieved from https://arxiv.org/abs/1802.03041.
[120] Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. 2019. OCGAN: One-class novelty detection using gans with constrained latent representations. In *CVPR*. 2898–2906.
[121] Daniel Pérez-Cabo, David Jiménez-Cabello, Artur Costa-Pazo, and Roberto J. López-Sastre. 2019. Deep anomaly detection for generalized face anti-spoofing. In *CVPR Workshops*.
[122] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*. 2227–2237.
[123] Tomáš Pevnỳ. 2016. Loda: Lightweight on-line detector of anomalies. *Mach. Learn.* 102, 2 (2016), 275–304.
[124] Ali Rahimi and Benjamin Recht. 2008. Random features for large-scale kernel machines. In *NeurIPS*. 1177–1184.
[125] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. 2000. Efficient algorithms for mining outliers from large data sets. In *SIGMOD*. 427–438.
[126] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. 2000. Efficient algorithms for mining outliers from large data sets. *ACM SIGMOD Rec.* 29, 2 (2000), 427–438.
[127] Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. 2019. Likelihood ratios for out-of-distribution detection. In *NeurIPS*. 14680–14691.
[128] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. 2011. Contractive auto-encoders: Explicit invariance during feature extraction. In *ICML*. 833–840.
[129] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*. Springer, 234–241.
[130] Lorenzo Rosasco, Ernesto De Vito, Andrea Caponnetto, Michele Piana, and Alessandro Verri. 2004. Are loss functions all the same? *Neural Comput.* 16, 5 (2004), 1063–1076.
[131] Volker Roth. 2005. Outlier detection with one-class kernel fisher discriminants. In *NeurIPS*. 1169–1176.
[132] Lukas Ruff, Nico Görnitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Robert Vandermeulen, Alexander Binder, Emmanuel Müller, and Marius Kloft. 2018. Deep one-class classification. In *ICML*. 4390–4399.
[133] Lukas Ruff, Robert A. Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. 2020. Deep semi-supervised anomaly detection. In *ICLR*.
[134] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* 115, 3 (2015), 211–252.
[135] Mohammad Sabokrou, Mohammad Khalooei, Mahmood Fathy, and Ehsan Adeli. 2018. Adversarially learned one-class classifier for novelty detection. In *CVPR*. 3379–3388.
[136] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans. In *NeurIPS*. 2234–2242.
[137] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. 2019. f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. *Med. Image Anal.* 54 (2019), 30–44.
[138] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *IPMI*. Springer, Cham, 146–157.
[139] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. 2001. Estimating the support of a high-dimensional distribution. *Neural Comput.* 13, 7 (2001), 1443–1471.

[140] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. 1997. Kernel principal component analysis. In *ICANN*. 583–588.
[141] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. 2017. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Trans. Database Syst.* 42, 3 (2017), 1–21.
[142] Md Amran Siddiqui, Alan Fern, Thomas G. Dietterich, and Weng-Keen Wong. 2019. Sequential feature explanations for anomaly detection. *ACM Trans. Knowl. Discov. Data* 13, 1 (2019), 1–22.
[143] Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *ICLR*.
[144] Mahito Sugiyama and Karsten Borgwardt. 2013. Rapid distance-based outlier detection via sampling. In *NeurIPS*. 467–475.
[145] Waqas Sultani, Chen Chen, and Mubarak Shah. 2018. Real-world anomaly detection in surveillance videos. In *CVPR*. 6479–6488.
[146] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NeurIPS*. 3104–3112.
[147] Acar Tamersoy, Kevin Roundy, and Duen Horng Chau. 2014. Guilt by association: Large scale malware detection by mining file-relation graphs. In *KDD*. 1524–1533.
[148] David M. J. Tax and Robert P. W. Duin. 2004. Support vector data description. *Mach. Learn.* 54, 1 (2004), 45–66.
[149] Lena Tenenboim-Chekina, Lior Rokach, and Bracha Shapira. 2013. Ensemble of feature chains for anomaly detection. In *MCS*. 295–306.
[150] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. 2017. Lossy image compression with compressive autoencoders. In *ICLR*.
[151] Fei Tian, Bin Gao, Qing Cui, Enhong Chen, and Tie-Yan Liu. 2014. Learning deep representations for graph clustering. In *AAAI*. 1293–1299.
[152] Yu Tian, Gabriel Maicas, Leonardo Zorron Cheng Tao Pu, Rajvinder Singh, Johan W. Verjans, and Gustavo Carneiro. 2020. Few-shot anomaly detection for polyp frames from colonoscopy. In *MICCAI*.
[153] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* 11 (Dec. 2010), 3371–3408.
[154] 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.
[155] Hu Wang, Guansong Pang, Chunhua Shen, and Congbo Ma. 2020. Unsupervised representation learning by predicting random distances. In *IJCAI*.
[156] Hao Wang and Dit-Yan Yeung. 2016. Towards Bayesian deep learning: A framework and some existing methods. *IEEE Trans. Knowl. Data Eng.* 28, 12 (2016), 3395–3408.
[157] Siqi Wang, Yijie Zeng, Xinwang Liu, En Zhu, Jianping Yin, Chuanfu Xu, and Marius Kloft. 2019. Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network. In *NeurIPS*. 5960–5973.
[158] Xuanhui Wang, Nadav Golbandi, Michael Bendersky, Donald Metzler, and Marc Najork. 2018. Position bias estimation for unbiased learning to rank in personal search. In *WSDM*. 610–618.
[159] Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *Comput. Surv.* 53, 3 (2020), 1–34.
[160] Steve Webb, James Caverlee, and Calton Pu. 2006. Introducing the webb spam corpus: Using email spam to identify web spam automatically. In *CEAS*.
[161] Peng Wu, Jing Liu, and Fang Shen. 2019. A deep one-class neural network for anomalous event detection in complex scenes. *IEEE Trans. Neural Netw. Learn. Syst.* (2019).
[162] Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *ICML*. 478–487.
[163] Dan Xu, Elisa Ricci, Yan Yan, Jingkuan Song, and Nicu Sebe. 2015. Learning deep representations of appearance and motion for anomalous event detection. In *BMVC*.
[164] Wei Xu, Ling Huang, Armando Fox, David Patterson, and Michael Jordan. 2009. Online system problem detection by mining patterns of console logs. In *ICDM*. IEEE, 588–597.
[165] Jianwei Yang, Devi Parikh, and Dhruv Batra. 2016. Joint unsupervised learning of deep representations and image clusters. In *CVPR*. 5147–5156.
[166] Xu Yang, Cheng Deng, Feng Zheng, Junchi Yan, and Wei Liu. 2019. Deep spectral clustering using dual autoencoder network. In *CVPR*. 4066–4075.
[167] Muchao Ye, Xiaojiang Peng, Weihao Gan, Wei Wu, and Yu Qiao. 2019. Anopcn: Video anomaly detection via deep predictive coding network. In *ACM MM*. 1805–1813.

[168] Wenchao Yu, Wei Cheng, Charu C. Aggarwal, Kai Zhang, Haifeng Chen, and Wei Wang. 2018. Netwalk: A flexible deep embedding approach for anomaly detection in dynamic networks. In *KDD*. 2672–2681.

[169] Muhammad Zaigham Zaheer, Jin-ha Lee, Marcella Astrid, and Seung-Ik Lee. 2020. Old is gold: Redefining the adversarially learned one-class classifier training paradigm. In *CVPR*. 14183–14193.

[170] Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, and Vijay Ramaseshan Chandrasekhar. 2018. Efficient gan-based anomaly detection. *arXiv:1802.06222*. Retrieved from https://arxiv.org/abs/1802.06222.

[171] Houssam Zenati, Manon Romain, Chuan-Sheng Foo, Bruno Lecouat, and Vijay Chandrasekhar. 2018. Adversarially learned anomaly detection. In *ICDM*. IEEE, 727–736.

[172] Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V. Chawla. 2019. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *AAAI*, Vol. 33. 1409–1416.

[173] Ke Zhang, Marcus Hutter, and Huidong Jin. 2009. A new local distance-based outlier detection approach for scattered real-world data. In *PAKDD*. Springer, 813–822.

[174] Panpan Zheng, Shuhan Yuan, Xintao Wu, Jun Li, and Aidong Lu. 2019. One-class adversarial nets for fraud detection. In *AAAI*. 1286–1293.

[175] Chong Zhou and Randy C. Paffenroth. 2017. Anomaly detection with robust deep autoencoders. In *KDD*. ACM, 665–674.

[176] Joey Tianyi Zhou, Jiawei Du, Hongyuan Zhu, Xi Peng, Yong Liu, and Rick Siow Mong Goh. 2019. Anomalynet: An anomaly detection network for video surveillance. *IEEE Trans. Inf. Forens. Secur.* 14, 10 (2019), 2537–2550.

[177] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*. 2223–2232.

[178] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. 2012. A survey on unsupervised outlier detection in high-dimensional numerical data. *Stat. Anal. Data Min.* 5, 5 (2012), 363–387.

[179] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. 2018. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *ICLR*.

[180] Hui Zou, Trevor Hastie, and Robert Tibshirani. 2006. Sparse principal component analysis. *J. Comput. Graph. Stat.* 15, 2 (2006), 265–286.

Received July 2020; revised October 2020; accepted November 2020