CSCAD: Correlation Structure-Based Collective Anomaly Detection in Complex System

Huiling Qin, Xianyuan Zhan, *Member, IEEE*, and Yu Zheng, *Fellow, IEEE*

Abstract—Detecting anomalies in large complex systems is a critical and challenging task. The difficulties arise from several aspects. First, collecting ground truth labels or prior knowledge for anomalies is hard in real-world systems, which often lead to limited or no anomaly labels in the dataset. Second, anomalies in large systems usually occur in a collective manner due to the underlying dependency structure among devices or sensors. Lastly, real-time anomaly detection for high-dimensional data requires efficient algorithms that are capable of handling different types of data (i.e. continuous and discrete). We propose a correlation structure-based collective anomaly detection (CSCAD) model for high-dimensional anomaly detection problem in large systems, which is also generalizable to semi-supervised or supervised settings. Our framework utilize graph convolutional network combining a variational autoencoder to jointly exploit the feature space correlation and reconstruction deficiency of samples to perform anomaly detection. We propose an extended mutual information (EMI) metric to mine the internal correlation structure among different data features, which enhances the data reconstruction capability of CSCAD. The reconstruction loss and latent standard deviation vector of a sample obtained from reconstruction network can be perceived as two natural anomalous degree measures. An anomaly discriminating network can then be trained using low anomalous degree samples as positive samples, and high anomalous degree samples as negative samples. Experimental results on five public datasets demonstrate that our approach consistently outperforms all the competing baselines.

Index Terms—Anomaly detection, complex system, correlation mining, unsupervised learning, urban computing, variational autoencoder

1 Introduction

Efficiently detecting faults or anomalies is vital to safeguard the daily operation of many large complex systems. Typical applications including manufacturing system fault detection, network intrusion detection, abnormal bio-activities discovery and so on [1], [2]. Although extensive anomaly detection studies have been conducted in the past, developing a robust anomaly detection technique for large systems with complex internal structures is still a challenging task to solve. The challenges arise from several aspects.

First, unlike other data-driven tasks, collecting ground truth labels or prior knowledge for anomalies in complex systems is much harder. Anomalies are typically rare in the population, leading to highly unbalanced datasets. In many real-world systems, anomaly labels are collected through manual inspection, leading to very limited or even no anomaly labels being collected. In extreme cases, it is even impossible to know the basic information about the anomalies in the system, which forbids the use of many anomaly detection algorithms with a predetermined threshold (e.g. anomalous degree threshold or the proportion of anomalies in the data) [3], [4], [5]. Due to these facts, unsupervised anomaly detection algorithms [6] that can adaptively learn the discriminating boundaries of normal and anomaly samples are particularly desirable.

Second, most sensors in large complex systems do not work independently. Many anomaly detection scenarios involve complicated internal dependency structures among sensors or devices. For example, sensors monitoring different parts of a manufacturing system typically have a complex interdependent relationship. Fault from a sensor can propagate to other dependent sensors, causing cascading failures in parts or the entire system. This is also referred to as *collective anomaly* [7], [8], [9], [10]. Under such circumstances, an anomaly might not be that anomalous by checking a single data feature but could be identified as an anomaly when checking multiple related features simultaneously. Considering the feature space correlation to analyze the change in the underlying correlation structural pattern facilitates better modeling the operation characteristics of the system, leading to more accurate and robust results. This is particularly important in detecting early-stage anomalies when the anomalous pattern is not significant.

Lastly, sensory data collected in large systems typically involve data with different properties (e.g. static and time-series) or types, such as continuous (e.g. temperature reading from a sensor) and discrete (e.g. on/off status of a switch) data. A unified approach is needed to handle different types of data and generalizable to time-series settings.

In this paper, we propose a highly adaptive correlation structure-based anomaly detection (CSCAD) framework to address the aforementioned challenges. We consider the behavior of anomalies from three aspects: (1) break down of

- *Huiling Qin and Yu Zheng are with Xidian University and JD iCity, JD Technology, Beijing 100176, China, and also with JD Intelligent Cities Research, Beijing 100080, China. Email: orekinana@gmail.com and msyuzheng@outlook.com.*
- *Xianyuan Zhan is with the Institute for AI Industry Research (AIR), Tsinghua University, Beijing 100190, China. Email: zhanxianyuan@gmail.com.*

Manuscript received 2 July 2020; revised 18 December 2021; accepted 12 January 2022. Date of publication 22 March 2022; date of current version 3 April 2023.
This work was supported in part by the National Key R&D Program of China under Grant 2019YFB2101801, in part by the Public Service Platform for Industrial Technology under Grant 2021-0166-1-1, and in part by the National Natural Science Foundation of China under Grant 62076191.
(Corresponding author: Huiling Qin.)
Recommended for acceptance by L. Xiong.
Digital Object Identifier no. 10.1109/TKDE.2022.3154166

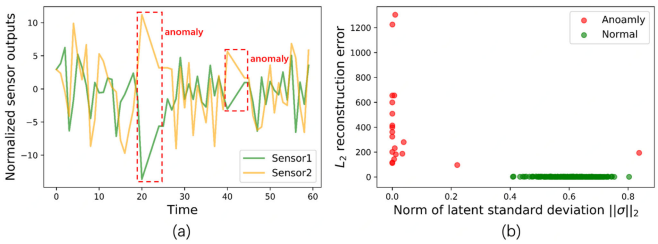


Fig. 1. Sensory data and anomaly status from an human activity dataset: (a) Normalized readings of two sensors over time. (b) Reconstruction error and the norm of latent standard deviation vector of samples evaluated using a VAE.

the original feature space correlation patterns (e.g. correlated features exhibit different correlation pattern); (2) anomalous samples are harder to reconstruct compared with normal samples, as they possess different statistical patterns; and (3) anomalous samples have larger variance when explaining using a model trained with the complete dataset. We illustrate the above anomalous behavior using a simple human activity dataset [11]. In Fig. 1a, anomalies occur when two positively correlated sensors suddenly become uncorrelated; and in Fig. 1b, the reconstruction error and latent standard deviation evaluated using a variational autoencoder (VAE) are observed to be highly discriminative features for the anomaly detection task.

Based on above observations, we use a graph convolutional variational autoencoder as the reconstruction network to capture the feature space correlation and reconstruction deficiency of samples, and further perform detection using a feedforward neural network as the discriminating network. Graph convolutional layers are used in the reconstruction network to mine the hidden structure in the feature space of data, which is constructed using a new correlation evaluation metric, the extended mutual information (EMI). It is capable of handling different types of data (continuous and categorical) and generalizable to time-series data. The feature space correlation provides extra information about the internal correlation structure among different features, thus enhances the data reconstruction capability of the model. Two anomalous degree measures, the reconstruction loss (measures reconstruction difficulty) and the latent standard deviation (measures internal variance) of samples can be obtained from the trained reconstruction network. The discriminating network is then trained using low anomalous degree samples as positive samples, and high anomalous degree samples or samples with anomaly labels as negative samples. This design allows the training process of the framework can be performed in an unsupervised manner without the need of introducing any predetermined anomaly threshold. The combination of the aforementioned machine learning approaches and the resulting anomaly measures is new in the fields of collective anomaly detection.

The contributions of our work is fivefold:

- We propose an extended version of mutual information to measure the correlation between data features with different characteristics. It is capable of handling different types of data (continuous and categorical) and generalizable to time-series data.
- We propose a novel graph-based deep learning framework, named CSCAD, and the resulting anomaly measures to detect collective anomalies in complex systems. CSCAD leverages the correlation structure among features and the reconstruction deficiency of samples to perform anomaly detection.
- We use a discriminating network to adaptively identify anomalous samples by utilizing anomalous degree measures mined from the data in an unsupervised or semi-supervised manner, without the need of introducing any predetermined anomaly threshold.
- CSCAD provides a highly flexible framework, which is applicable to fully unsupervised, semi-supervised or supervised settings depending on the availability of anomaly labels, and can be easily extended to model high-dimensional time-series data.
- Experiments on five public datasets show that the proposed framework, as well as its time-series extension, consistently outperforms the baselines in terms of precision, recall and F_1 score, which demonstrate the effectiveness and extensibility of our framework.

2 Related Work

There is extensive literature related to anomaly detection. Our focus is mainly restricted to high-dimensional collective anomaly detection problems for multiple types of data with limited or no anomaly labels (see [12], [13], [14], [15] for a wider scope survey). In this scope, most works evaluate the anomaly measure/score with a predetermined or learned threshold to solve the anomaly detection problem, with recent works tending to use reconstruction-based approaches.

2.1 Anomaly Score-Based Anomaly Detection

Most methods detects anomaly based on some anomaly representation score by conventional or deep learning models. A predetermined or learned threshold is used on the score to determine whether a sample is an anomaly.

Predetermined threshold. In most unsupervised anomaly detection methods, data samples are considered as anomalous when they are located in a low density/probability region of the training data. In these settings, the proportion of anomalies in data is often used as a form of threshold to identify the anomalous samples. Some traditional methods (e.g. kernel density estimation [16] and Robust-KDE [17]) are known to be problematic in dealing with high-dimensional data due to the curse of dimensionality. To mitigate this problem, some studies [3] compress the high-dimensional data into low-dimensional latent representations using deep autoencoders, and then apply a density-based model on the low-dimensional space to detect anomaly. Some recent works combine these two steps and directly learn an anomaly score that perform density estimation. Zhai *et al.* [4] utilize an energy-based autoencoder model to map each data sample to an energy score. Zong *et al.* [5] use a compression network combined with the Gaussian mixture model to estimate the density of each sample and further detect the anomaly. Some methods [18], [19], [20] also use variational autoencoder (VAE) or its variants to learn the probability density of each sample and identify the low-probability samples as anomalies. In addition to the probability density-based anomaly scores, other methods also consider more complex normality measures. Zhang *et al.* [10] detect traffic volume anomaly

using pairwise similarity matrix for each region and data source. The anomaly score is evaluated as the degree of change in similarity matrix between consecutive time steps. Oh *et al.* [21] use inverse reinforcement learning to learn a reward function as a form of anomaly score, and then use a predetermined threshold for time-series anomaly detection.

Learned threshold. A major drawback of aforementioned anomaly score-based methods is the need to specify a threshold for the anomaly score or the proportion of anomalies in data to discriminate normal or anomalous samples. To overcome this drawback, other techniques are developed to learn the threshold adaptively using a data-driven approach. Ramakrishnan *et al.* [22] use cross-validation on the test set and choose the threshold that maximizes the recall at a minimum precision level. OmniAnomaly [23] determines the anomaly threshold offline following the principle of Extreme Value Theory (EVT) [24]. The threshold is obtained by fitting the tail proportion of a generalized Pareto distribution. These methods are fully unsupervised, which are difficult to incorporate additional information or prior knowledge (e.g. partially available anomaly labels) in real-world applications. To address this issue, Pang *et al.* [25] leverage limited anomaly labels and a prior probability to enforce statistically significant deviations of the anomaly scores of anomalies from that of normal samples in the upper tail. Ren *et al.* [26] propose a SR-CNN model for time-series anomaly detection, which uses a CNN discriminating network trained on well-designed synthetic data instead of a threshold to identify anomalies. The synthetic data is generated by injecting anomalous samples into a collection of saliency maps that are not included in the evaluated data.

2.2 Reconstruction-Based Anomaly Detection

The main assumption of the reconstruction-based approach is that anomalous samples have different patterns compared with normal samples, hence are more difficult to be reconstructed. The anomalous degree of a data sample can be reflected by the loss or distance between the original and reconstructed samples generated by some statistical or neural models. Classic methods include principal component analysis (PCA) with explicit linear projections, and the improved version, robust principal component analysis (RPCA) [3], [27], which improve the robustness of PCA by enforcing sparse structures. Inspired by recent advances in deep learning research, autoencoders have become another popular data reconstruction technique in anomaly detection studies. Araya *et al.* [8] use an autoencoder to capture the internal relationship among feature while recognizing normal patterns in the training data. Zhou and Paffenroth [28] introduce a robust deep autoencoder (RDA) model which split the input data into reconstructed part and noise to improve the robustness of standard deep autoencoders. Chalapathy *et al.* [29] focuses on mining irregular group distributions utilizing adversarial autoencoder (AAE) and variational autoencoder (VAE) for group anomaly detection. Finally, generative adversarial network (GAN) [30] is also adopted by several methods to detect anomalies [31], [32], [33]. The idea is that anomalies have different distribution compared with normal samples, which makes it difficult to generate similar non-anomalous samples through GANs. The weakness of the reconstruct-based approach lies in the need of a reliable data reconstruction model. A low capacity model is unable to capture complex patterns in the data, resulting in model-induced reconstruction deficiency.

In this work, we combine the merits of the aforementioned approaches while overcomes their drawbacks. We exploit the internal correlation structure in data features and use a high-capacity model for data reconstruction. We further train a small discriminative network to perform detection using the data reconstruction loss and latent standard deviation. The proposed framework is highly adaptive, which can be used with limited or no anomaly labels, and easily generalizable to high-dimensional time-series data.

3 Overall Framework

First, we give the definition of the collective anomaly. If a collection of related data instances is anomalous with respect to the entire data set, it is termed as a collective anomaly. The individual data instances in a collective anomaly may not be anomalies by themselves, but their occurrence together as a collection is anomalous.

Here, we consider the problem of collective anomaly detection in large complex systems. Let $\mathcal{X}$ be the space of all samples in a system, with the ith sample denoted as $X_i = (x_{i1}, \ldots, x_{im})^T$, with x_{ij} as the feature or sensor of sample X_i in dimension j. We judge a sample X_i to be normal or anomalous based on three criteria: (1) break down of usual feature space correlation pattern; (2) difficulty in reconstructing using a model trained using the complete dataset, and (3) latent space embedding exhibits high variance.

Our framework adopts the following design to incorporate the three criteria. First, the hidden structure of the feature space is mined using a new extended mutual information (EMI) metric, which constructs a correlation structure graph among features. A reconstruction network modeled as graph convolutional variational autoencoder is then trained to generate two sample anomalous degree measures (reconstruction loss d and latent standard deviation σ_z) by capturing the feature space correlation and perform robust sample reconstruction. The two anomaly degree measures are used as the inputs of a discriminative network to predict the final anomalous probability. The training of the discriminative network adopts a self-learning mechanism, which uses low anomalous degree samples as positive samples, and high anomalous degree samples or samples with anomaly labels as negative samples. An illustration of the proposed framework can be found in Fig. 2.

3.1 Correlation Structure Mining

Large complex systems typically have large amount of sensors involving both continuous (e.g. temperature readings from a thermometer) and discrete (on/off status of a switch) data; some may be time varying and others are static. Most existing correlation measures, such as Pearson correlation coefficient, cross-variance and mutual information (MI), only work for data with identical types. In order to mine the correlation structure across multiple types and properties of data features, a generic metric is needed. We proposed the extended mutual information (EMI) metric extending the work of Galka *et al.* [34] to measure the correlation between features with different characteristics.

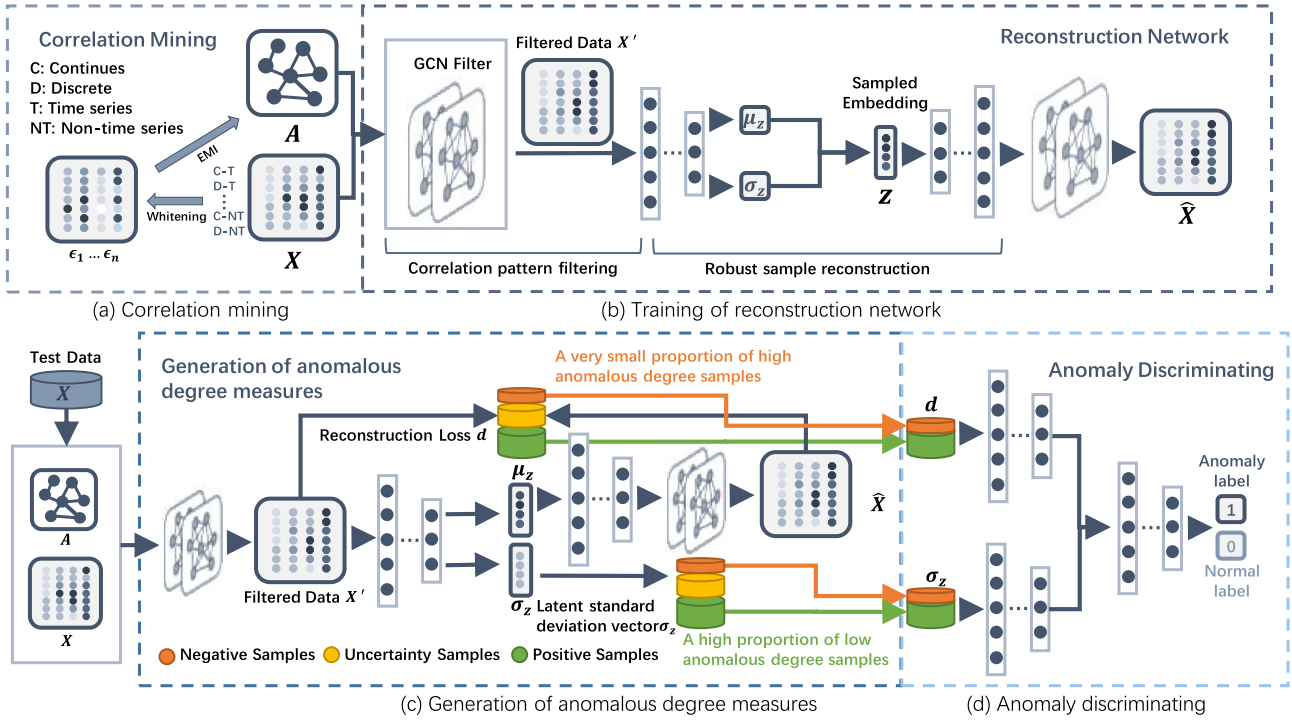


Fig. 2. Proposed framework for collective anomaly detection.

EMI is based on information theory and directly works with probabilistic distributions of variables, which can be applied to both continuous and categorical data, and generalizable to time-series data. A illustration of the correlation mining process using EMI see Fig. 3

Time-series MI. The MI between two random variables $\mathbf{x}$ and $\mathbf{y}$ is defined as the Kullback-Leibler (KL) divergence between the joint distribution $p(\mathbf{x}, \mathbf{y})$ and the product of their marginals $p(\mathbf{x})$ and $p(\mathbf{y})$: $I(\mathbf{x}, \mathbf{y}) = KL(p(\mathbf{x}, \mathbf{y})||p(\mathbf{x})p(\mathbf{y}))$. Galka *et al.* [34] proposed an improved version of MI, which generalizes the MI for time-series data. Let x_t and y_t be two time-series random variables at time step t, the generalized MI can be written as:

$$I(\mathbf{x}, \mathbf{y}) = \log p((x_1, y_1), \ldots, (x_{N_t}, y_{N_t})) - \log\left(p(x_1, \ldots, x_{N_t}) p(y_1, \ldots, y_{N_t})\right) \quad (1)$$

As the high-dimensional joint distributions $p((x_1, y_1), \ldots, (x_{N_t}, y_{N_t}))$, $p(x_1, \ldots, x_{N_t})$ and $p(y_1, \ldots, y_{N_t})$ are complicated to estimate, Galka *et al.* [34] suggests removing the self-temporal correlation of the data by a whitening operation which utilizing a predictive model $\mathcal{E}(\cdot)$ to approximate the conditional means of variables, such that the whitened data satisfy the I.I.D. property (see [34] for details). Let $\epsilon_{(\cdot)}$ represent the whitened form of data, which is the residual between the original and predictive conditional mean of data. The time-series MI can thus be evaluated as:

$$I(\mathbf{x}, \mathbf{y}) = \log\left(p(\epsilon_{x_1}, \epsilon_{y_1}) \cdots p(\epsilon_{x_{N_t}}, \epsilon_{y_{N_t}})\right) - \log\left(p(\epsilon_{x_1}) \cdots p(\epsilon_{x_{N_t}}) p(\epsilon_{y_1}) \cdots p(\epsilon_{y_{N_t}})\right) \quad (2)$$

Extended MI (EMI). With the nice statistical property of the whitened residuals, the key to compute the mutual information between two variables with different types lies in finding a unified predictive model to approximate the conditional means of the variable $\mathcal{E}(\cdot)$, as well as defining the residual for discrete variables.

Although TMI solves the problem of calculating the correlation between time-series data, the calculation of correlation between different types of data (discrete, continuous, time-series, etc.) is still a challenge. Based on time-series MI, we propose an extended version of MI to cover different types of data. There are two key ingredients of EMI: (1) a unified predictive model to perform temporal whitening for different types of data; and (2) a new scheme to properly define the "residual" of whitened discrete variables.

For both continuous and discrete variables, we can perceive the conditional mean $\mathcal{E}(\cdot)$ as a temporal predictor $x_t = f(x_{t-1}, x_{t-2}, \cdots)$ that predict the data value at time t given historical time-series information. In order to unify the predictive model for different types of data, we calculate the conditional mean using a regression tree (for continuous data) or a decision tree (for discrete data) given historical time-series data from a sliding time window. And for the conditional mean of the joint of two variables $\mathcal{E}((x_t, y_t)|(x_{t-1}, y_{t-1}), (x_{t-2}, y_{t-2}), \cdots)$, we can still use the regression tree regardless the types of x_t and y_t, as the

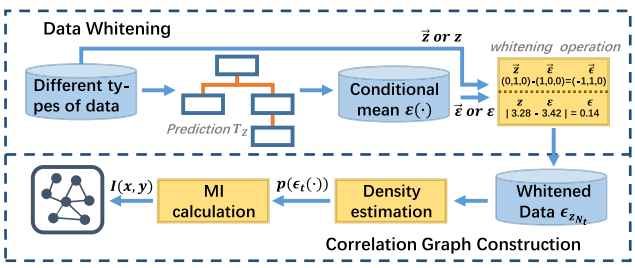


Fig. 3. Illustration of the correlation mining process.

regression tree can handle both the continues and discrete data. The reasons that we choose the tree-based model to estimate the conditional means are because: (1) tree-based model can deal with different type of variable; (2) it can capture both the linear and non-linear relationship in the temporal data; (3) it is effective and simple, which help to reduce the computational cost during correlation evaluation for high-dimensional data.

Another challenge of evaluating mutual information between two variables across different data types is to properly define the residual for discrete variables, as simple subtraction is no longer well-defined for discrete variables. To address this issue while guarantee the linear transformation for the variable, we encode the discrete variables as one-hot encoded vector (m-dimensional vector if has m discrete states), and calculate vector subtraction results $\vec{y}_i - \vec{y}_j$. There are a total of $m^2 - m + 1$ possible outcomes of $\vec{y}_i - \vec{y}_j$ (each element in the m-dimensional vector $\vec{y}_i - \vec{y}_j$ takes value of -1, 0, 1), therefore, we encode different outcomes of the vector subtraction as a new $m^2 - m + 1$ dimensional one-hot encoded vector $\vec{z}$ and use it as the residual for discrete variables. The residual is a lossless recording of the transition between different discrete states of the variables and is well-defined.

Finally, we can unify the whitening operation as follows:

$$\epsilon_{z_{N_t}} = z_{N_t} \ominus T_Z(z_{N_t}|z_{N_t} \cdots z_1) \tag{3}$$

where $T_Z(\cdot)$ is the trained tree-based predictive model for variable Z; $\ominus$ is the subtraction operation when z is continuous and is the previously defined discrete residual operation if z is discrete.

After obtain the whitened residuals for the time-series data of the system, the marginal distribution of $p(\epsilon_t(x|x))$ and $p(\epsilon_t(x|(x,y)))$ can be estimated either by non-parametric statistics or methods such as kernel density estimation(KDE) (for continuous residuals) or simply count the proportion of each discrete residual state (for discrete residuals). For the joint distribution $p(\epsilon_t(x|x,y), \epsilon_t(y|x,y))$ involving both continuous and discrete variables, it is calculate as:

$$p(\epsilon_t(x|x,y), \epsilon_t(y|x,y)) = p(\epsilon_t(x|x,y)|\epsilon_t(y|x,y))p(\epsilon_t(y|x,y)) \tag{4}$$

where x is a continuous variable, and y is a discrete variable. The conditional probability of $p(\epsilon_t(x|x,y)|\epsilon_t(y|x,y))$ can be estimated by a set of non-parametric statistics conditioned on each discrete state of y. Finally, we can use Eqs. 1, 2 to obtain the MI-based correlation for two time-series variables with arbitrary types.

With the extended data whitening scheme, the MI of data with different types can be evaluated in a similar procedure as discussed in [34]. The proposed EMI allows evaluating correlation for both continuous and categorical data, and generalizable to time-series data. We use EMI to evaluate the pairwise correlation between different features of data, from which a correlation structure graph can be constructed, with nodes represent data features, and edges represent the pairwise correlation of the two features. We perform the pairwise EMI calculations in parallel to speed up the correlation graph construction. In our experiment, to preserve the complete hidden relationships among features, we construct a fully connected graph by keeping all the correlation edges computed by EMI. In practical applications, we can set a correlation threshold according to the actual situation, and only the highly correlated edges are retained to reduce the computational consumption.

3.2 Reconstruction Network

A notable characteristic of anomalies is that they are harder to reconstruct compared with normal samples [5]. Inspired by this observation, we design a graph convolutional variational autoencoder as the reconstruction network for sample reconstruction, which is a combination of graph convolutional neural network (GCN) and variational autoencoder (VAE) [35] (see Fig. 2b). We use GCN layers to capture the correlation structure of features mined using the EMI. Hence when data features exhibit distinct correlation patterns, the outputs filtered by the GCN layers will facilitate enlarging the reconstruction error evaluated by the VAE. Furthermore, we use the VAE rather than a conventional autoencoder. As VAE can learn a latent mean and a latent standard deviation vector, which can be perceived as the denoised mean and the internal variance of the sample embedding, thus extracts more detailed information about the anomalous behavior of a sample.

We use the GCN layer proposed in Defferrard *et al*. [36] to model the correlation structure in the feature space. Consider a spectral convolution on graph defined as the multiplication of a graph signal $X \in \mathbb{R}^{m \times c}$ with a filter g_θ parameterized by θ in the Fourier domain:

$$g_\theta^\star \mathcal{G} X = g_\theta(L)X = g_\theta(U\Lambda U^T)X = U g_\theta(\Lambda)U^T X \tag{5}$$

where $U \in \mathbb{R}^{m \times m}$ is the matrix of eigenvectors, and $\Lambda \in \mathbb{R}^{m \times m}$ is the diagonal matrix of eigenvalues of the normalized graph Laplacian $L = I_N - D^{\frac{1}{2}}\Lambda D^{\frac{1}{2}} = U\Lambda U^T$, where I_N is the identity matrix, $D \in \mathbb{R}^{m \times m}$ is the diagonal degree matrix with $D_{ii} = \sum_j A_{ij}$ and A is the adjacency matrix of the correlation structure graph. A non-parametric filter, i.e. a filter whose parameters are all free, would be defined as:

$$g_\theta(\Lambda) = diag(\theta) \tag{6}$$

where the parameter $\theta \in \mathbb{R}^n$ is a vector of Fourier coefficients.

Directly perform convolution operation using above formulation is computationally expensive. Defferrard *et al*. [36] used the Kth-order polynomial in the Laplacian to enable fast evaluation, which restricts the GCN to capture the information at maximum K step away from the central node (K-localized). The corresponding graph convolutional operator and the lth layer output of GCN $H^{(l)} \in \mathbb{R}^{m \times d}$ given activation function $f(\cdot)$ are given as:

$$g_\theta(L)X = \sum_{k=0}^{K-1} \theta_k L^k X, \quad H^{(l+1)} = f\left(\sum_{k=0}^{K-1} \theta_k L^k H^{(l)}\right) \tag{7}$$

The subsequent VAE component of the reconstruction network takes the outputs from the GCN layer to perform a robust reconstruction. VAE forces its encoder to generate a latent vector z that roughly follow a Gaussian distribution,

which is parameterized by a latent mean vector μ_z, and a latent standard deviation vector σ_z. μ_z and σ_z can be perceived as embeddings of the denoised mean and the internal uncertainty level of a sample, which provides important information about the anomalous degree of the sample. Given a dataset of N samples, the reconstruction network can be trained in a similar manner as VAE using variational inference [35] by minimizing the following objective function:

$$J(\theta, \phi) = \frac{1}{N}\sum_{i=1}^{N} L(X_i, \hat{X}_i) + \frac{\lambda}{N}\sum_{i=1}^{N} D_{KL}(q_\phi(z_i|X_i)||p_\theta(z_i)) \tag{8}$$

where $L(X_i, \hat{X}_i) = ||X_i - \hat{X}_i||_2^2$ is the loss between original sample X_i and reconstructed sample $\hat{X}_i$. The KL divergence $D_{KL}(q_\phi(z|X)||p_\theta(z))$ forces the approximated posterior distribution $q_\phi(z|X)$ to be similar to the prior distribution $p_\theta(z)$, which improves the robustness of reconstruction when separating the internal noise from input data.

The proposed reconstruction network can be easily extended to model high-dimensional time-series data. This is done by introducing the recurrent neural network layer (e.g. LSTM, GRU) after GCN layer to capture temporal information across different time steps.

3.3 Discriminating Network and Detection Strategy

Generation of anomalous degree measures. Utilizing the trained reconstruction network, we derive two anomalous degree measures to support anomaly detection:

- The reconstruction loss $\vec{d_i} = [(x_{i1} - \hat{x}_{i1})^2, \ldots, (x_{im} - \hat{x}_{im})^2]^T$ is the element-wise reconstruction loss calculated by original sample X_i and reconstructed sample $\hat{X}_i$. We generate $\hat{X}_i$ by applying the decoder of GCVAE on the latent mean vector μ_z rather than the sampled latent vector z. As $\hat{X}_i$ generated from μ_z is deterministic and can be perceived as a denoised version of X_i, which helps reconstruction loss d more accurately reflect the reconstruction difficulty of a sample. As a result, a sample with a larger d indicates a higher anomalous degree.
- The latent standard deviation σ_z represents the internal variance level of a sample. It is another anomalous degree measure which reflects the uncertainty of the input data. A sample with high uncertainty or significantly deviate from the regular patterns of the data is likely to be an anomaly.

These two anomalous degree measures provide more detailed anomalous information of a sample to allow users to "interpret" the results. The reconstruction loss and latent standard deviation are given as a vector indicating the reconstruction difficulty and uncertainty level of each feature (e.g. each sensor), which can also help locate the most problematic sensors in the system.

Anomaly discriminating. We construct a discriminating network using reconstruction loss d and latent standard deviation σ_z as inputs (see Fig. 2c, d). In the discriminating network, d and σ_z are first fed into two sets of fully connected layers. Their outputs are then concatenated and fed into two fully connected layers to output the final anomalous probability of a sample.

Training the discriminating network requires anomaly labels in the data, which can be difficult to acquire. We take an alternative approach by utilizing the already obtained anomalous degree measures d and σ_z. The anomaly discrimination can be viewed as a semi-supervised positive-unlabeled (PU) learning problem [37]. Assuming that the data points are separable and smooth, then we can solve this problem with a two-step technique. **Step 1:** Identifying reliable negatives (and positives). In the first step, unlabeled examples that are very different from the positive examples are selected as reliable negatives. **Step 2:** Semi-supervised learning. In the second step, the labeled positive examples and reliable negatives are used to train a classifier.

We first evaluate d and σ_z of all samples in the training set, and rank the samples according to their norms $(||d||_2, ||\sigma_z||_2)$. Considering that the anomalies are typically rare in the population, and a sample with large d or σ_z might be anomalous. We label 50% (the proportion can be flexibly adjusted according to different real-world datasets) of low anomalous degree samples as positive samples. The negative samples are selected based on a very conservative criterion, that we only select a very small proportion of high anomalous degree sample (e.g. top 2.5%) as negative samples. The discriminating network is trained only using selected positive and negative samples. In our experiments, we conservatively oversample the negative samples to maintain the anomalous and normal sample ratio to be 1:5 in each training epoch, thus to prevent the class imbalance issue (in case of too few negative samples) in the final classification problem, while accounting for the overfitting on abnormal samples caused by oversampling.

When parts or full anomaly labels are available, the actual anomalies can be used as negative samples, which enables the proposed framework adaptable to semi-supervised or supervised settings. The proposed strategy does not need to specify a predetermined anomaly threshold that typically used in anomaly score-based methods [4], [5], [18], [19], [20], and capable of learning complex high-dimensional separation boundaries between normal and anomalous samples.

At the detection stage, the reconstruction loss d and latent standard deviation vector σ_z are obtained using the trained reconstruction network. They are fed into the trained discriminating network to evaluate the final anomalous probability of the given sample (see Fig. 2d for detailed illustration).

3.4 Time-Series Extension

The proposed CSCAD framework can also be generalized to time-series settings. We focus on the collective anomaly detection problem that finds anomalies at time step $t+1$ based on previous k time steps' data $(t-k+1, \ldots, t)$. Let the sample at time step t be $X_t = (x_{t1}, x_{t2}, \ldots, x_{tm})^T$. We use EMI to construct the feature space correlation structure graph of the time-series data. EMI can capture the correlation between features while eliminating the historical influence of the sequence. The reconstruction network is modified to include an LSTM layer to capture the temporal characteristics of the data.

The time-series data X_t at each time step is first fed into the GCN layer, and the sequence of the GCN outputs are

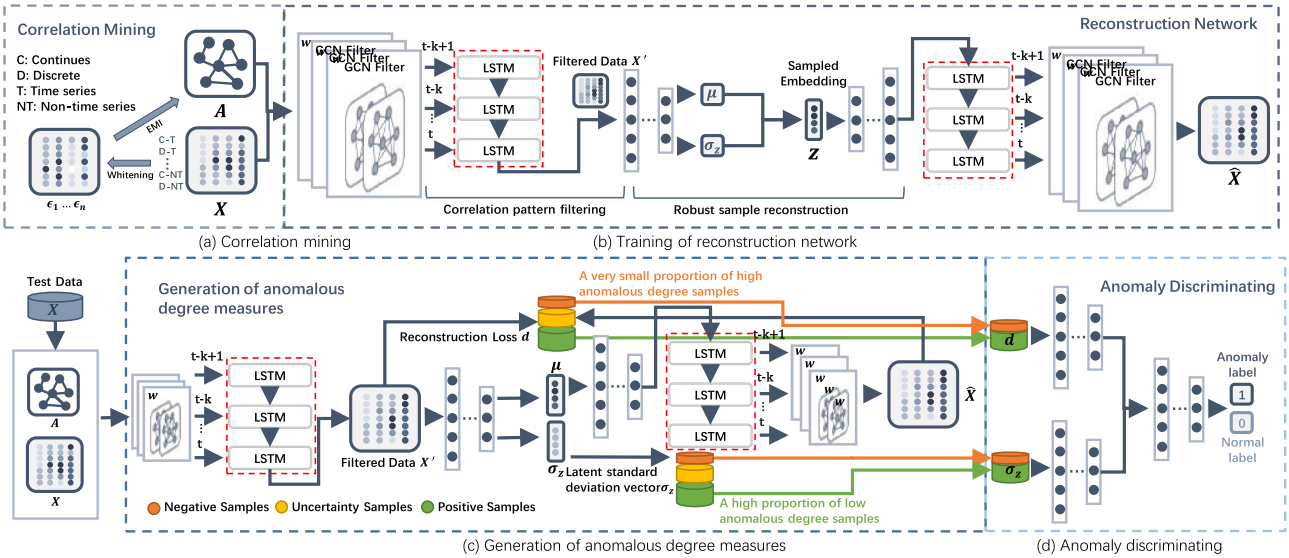


Fig. 4. Proposed framework for collective anomaly detection in time-series settings.

then modeled using the LSTM layer. The VAE component takes the final output of the LSTM layer and generates the latent mean and standard deviation of samples to support anomaly detection. The discriminating network and detection strategy remain the same as in the previous section. A illustration of the time-series extension framework see Fig. 4

4 Experimental Result

In this section, we use public benchmark datasets to demonstrate the effectiveness of the proposed CSCAD framework against several competing baselines.

4.1 Dataset

We use five public datasets from UCI machine learning repository [11] to evaluate the proposed framework, including KDDCUP, Thyroid, MoCap, UJIIndoorLoc for the static version of the framework, and Heterogeneity for the time-series extension of the framework. Detailed statistics of each dataset is presented in Table 1.

- **KDDCUP**. The KDDCUP99 dataset consists of 34 continuous and 7 categorical features. The categorical features are encoded using one-hot encoding, resulting in a dataset of 121 dimensions. As only 20% of samples are labeled as "normal" and 80% are labeled as "attack", therefore, we treat "normal" samples as anomalies in this task.
- **Thyroid**. The Thyroid dataset is a disease dataset in which "negative" samples are treated as normal and others are anomalies. 7 continuous and 2 categorical relevant features are used in this task. Again, the categorical features are encoded using one-hot encoding.
- **MoCap**. The MoCap dataset consists of 36 continuous features. Representing the hand postures (such as 'one,' 'two,' 'three' etc.) by 12 measuring points in the three-dimensional space. We construct a dataset suitable for anomaly detection by sampling a small number of instances of 'five' postures from the dataset as anomalies and keeping all instances of 'one' postures as normal samples.
- **UJIIndoorLoc**. UJIIndoorLoc is a multi-building indoor location dataset containing 522 WiFi fingerprints as continuous features. Since most of the collected samples are located in building "2" and a small number of samples are located in building "0," here we treat "BUILDING ID" labeled "2" as normal samples and "0" as abnormal samples.
- **Heterogeneity**. The Heterogeneity is a human activity recognition dataset which contains records of gyro sensor and accelerometer from smartphones and smartwatches, including 6 continuous time-series features. We consider "Stand" activities as normal samples, and "Stair Up" activities as anomalies.

TABLE 1
Statistics of the Four Public Datasets

	Dimensions	Instances	Anomaly ratio
KDDCUP99	121	494,021	20%
Thyroid	13	19,016	10%
MoCap	36	15,963	8.5%
UJIIndoorLoc	522	10,000	7.5%
Heterogeneity	6	54,000	20%

4.2 Baseline

We consider several state-of-the-art deep learning approaches and a few widely used anomaly detection methods as baselines. And to explore the contribution of each component of the proposed framework, we conduct the ablation study including VAE-DN, CSCAD(no DN), CSCAD(no /sigma) to verify the effectiveness of graph convolution networks, discriminative networks, and anomaly measures, respectively.

- **DAGMM**. The deep autoencoding Gaussian mixture model (DAGMM) [5] uses latent low-dimensional representation and a Gaussian mixture model to perform density estimation of data, and further predict anomalies using a predetermined threshold. It is a state-of-the-art method in the high-dimensional anomaly detection problem. We used the exact same

parameters as in the original paper (including the structure of the neural network, the number of nodes, the number of layers, dimensions, etc.)

- **LOF**. The local outlier factor (LOF) [38] finds anomalous data points by measuring the local deviation of a given data point with respect to its neighbors. Since taking $n_{neighbors} = 20$ appears to work well in general and the anomaly rate in the datasets is not too high, so we choose 20 as the number of neighbors. Although $n_{neighbors}$ should be larger when the percentage of outliers is high (i.e., greater than 10%), it will cause LOF to fail to run the dataset.
- **AE-LOF**. This method adopts a two-step approach. It first trains a deep autoencoder to generate a latent representation of a sample, then uses LOF to detect the anomaly. The structure and parameters of the autoencoder are the same as those of the CSCAD, except for the sampling layer (generation of μ and σ). On the other hand, the LOF parameters in AE-LOF are the same as those of the single LOF we mentioned above.
- **IF**. Isolation Forest (IF) [39] is a decision tree-based ensemble method. It partitions the sample points by randomly selecting a split value between the maximum and minimum values of a feature. Anomalies are more easily separated under random splits. This baseline is also used in the experiment on time-series data. In order to ensure the efficient operation and the effectiveness of the algorithm, we set the $n_{estimators}$ to 100, and limit the maximum height of the tree by setting the number of samples drawn from the dataset to train each tree as 256.
- **OC-SVM**. One-class support vector machine (OC-SVM) [40] is a kernel-based method which learns a decision function between normal and anomalous samples. Considering the number of samples and the feature dimension of the dataset used in the experiments, we use RBF as the kernel function.
- **VAE-DN**. This is a variant of the proposed CSCAD framework, which uses VAE as the reconstruction network without considering the feature space correlation.
- **CSCAD(no DN)**. This variant only uses GVAE as the reconstruction network to generate the anomalous measure, and directly use a threshold on the anomalous measure to detect anomalies without using the discriminative network.
- **CSCAD(no σ)**. This variant of the proposed framework only uses the reconstruction loss d in the discriminating network, without considering the latent standard deviation vector σ_z. We conservatively choose 2.5% of high anomalous degree samples as negative samples.
- **CSCAD(p%)**. These models are the proposed framework of which the discriminating network is trained by selecting different percentages (i.e. 2.5%, 5%, 7.5%, smaller than the actual anomaly ratio) of high anomalous degree samples as negative samples.
- **AE(LSTM)-IF**. AE(LSTM)-IF introduces LSTM layer in the autoencoder, and uses IF to perform detection on the encoded representation.
- **VAE(LSTM)-DN**. VAE(LSTM)-DN is a variant of the time-series extension of CSCAD, which removes GCN layers in the reconstruction network.
- **SR-CNN**. SR-CNN [26] borrows the SR model from visual saliency detection domain to time-series anomaly detection and then uses CNN as a discriminating network to improve the performance of the SR model. It is the state-of-the-art method in time-series anomaly detection problems. In the experiment, we use the different channel to capture the pattern of different features.

For the anomaly score-based (with predetermined threshold) methods, the threshold we choose for KDDCUP99 is 0.2 and for Thyroid, MoCap, UJIIndoorLoc is 0.1 which is close to the true anomaly rate of the dataset. Other parameters used in these algorithms remain the default. And network architecture of the deep learning-based approaches and the variants of CSCAD is the same as CSCAD. The model configurations of the proposed CSCAD is introduced as the next part.

4.3 Model Configurations

The detailed model configurations used on each individual datasets are summarized below. Let m be the dimension of data features. For a fairer comparison, CSCAD and some variants are modeled with a similar structural parameters as state-of-the-art baseline for the reconstructed network, thus avoiding the accuracy discrimination caused by tuning parameters.

- **Reconstruction Network**. For all datasets, the reconstruction network runs with GCN(m, k=2, ReLU)-FC(m, 60, ReLU)-FC(60, 30, ReLU)-FC(30, 10, ReLU)-2 FC(10, 5, none)-reparameterize-FC(5, 10, ReLU)-FC(10, 30, ReLU)-FC(30, 60, ReLU)-FC(60, m, ReLU)-GCN(m, k=2, none).
- **Discriminating Network**. The discriminating network contains 2 compression sub-networks that encode d and σ_z. The outputs of the two compression sub-networks are concatenated and fed into a sub-discriminating network to output the final anomalous probability of a sample. For all datasets, the compression sub-network that encodes the d runs with BN(m, elu)-FC(m, $m/2$, elu)-FC($m/2$, $m/4$, elu)-FC($m/4$, 10, elu)-FC(10, 5, elu); and the compression sub-network for σ_z runs with BN(m, elu)-FC(m, $m/2$, elu)-FC($m/2$, 2, elu). Finally, the sub-discriminating network runs with BN(7, elu)-FC(7, 4, elu)-FC(4, 2, softmax).
- **Time-series extension framework**. We add an LSTM layer in the reconstruction network runs with LSTM (m, layers=2, ReLU), other configurations are the same with CSCAD.

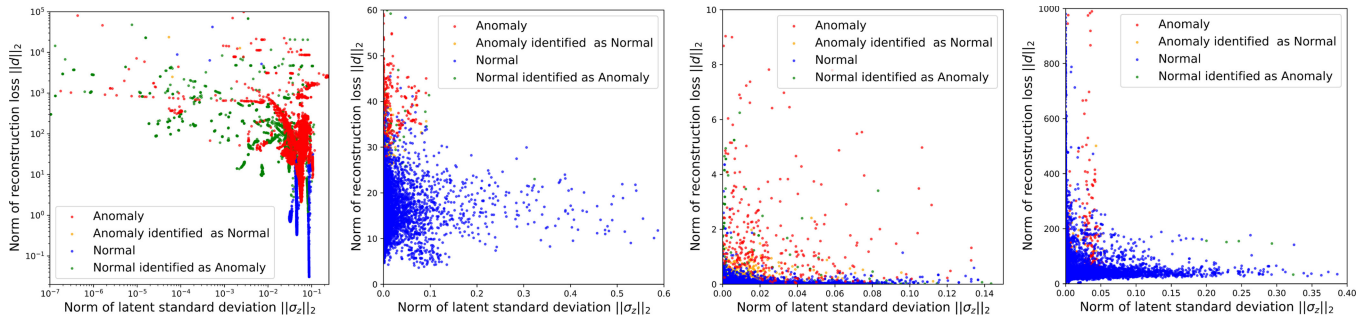
4.4 Evaluation Metrics

We use the precision, recall and F_1 score to evaluate the performance of the proposed CSCAD framework as well as the baselines, which are calculated as follows:

$$Precision = \frac{TP}{TP + FP}$$

TABLE 2
Experiment Results of Our Framework and the Baseline Methods for Four Static Datasets

Methods	KDDCUP			Thyroid			MoCap			UJIIndoorLoc		
	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1
DAGMM	0.830	0.839	0.835	0.273	0.257	0.265	0.431	**1**	0.603	0.720	0.709	0.714
AE-LOF	0.456	0.461	0.458	0.269	0.507	0.351	0.122	0.283	0.171	0.067	0.264	0.107
LOF	-	-	-	0.248	0.467	0.324	0.123	0.286	0.172	0.134	0.528	0.214
IF	0.449	0.454	0.451	0.280	0.526	0.365	0	0	0	0.508	0.998	0.673
OC-SVM	-	-	-	0.331	0.340	0.335	0.001	0.002	0.001	-	-	-
VAE-DN	0.852	0.954	0.9	0.268	0.345	0.302	0.256	0.522	0.344	0.112	0.701	0.211
CSCAD(no DN)	0.841	0.850	0.845	0.100	0.105	0.102	0.391	0.337	0.362	0.179	0.091	0.121
CSCAD(no σ)	0.734	**1**	0.846	0.440	0.586	0.503	**0.557**	0.684	0.614	0.059	**1**	0.112
CSCAD(7.5%)	0.857	0.994	0.920	**0.496**	**0.795**	**0.611**	0.486	**0.991**	0.652	**0.924**	0.856	**0.889**
CSCAD(5%)	0.862	0.921	0.890	0.480	0.662	0.557	0.496	0.956	**0.653**	0.916	0.858	0.886
CSCAD(2.5%)	**0.881**	0.996	**0.934**	0.495	0.553	0.522	0.507	0.792	0.618	0.706	0.894	0.789



(a) KDDCUP99 (b) MoCap (c) Thyroid (d) UJIndoorLoc

Fig. 5. Visualization of the detection results with respect to norms of reconstruction loss d and latent standard deviation σ_z.

$$Recall = \frac{TP}{TP + FN}$$

$$F_1 = \frac{2 * Precision * Recall}{Precision + Recall}$$

where true positive (TP) is the number of anomalous samples that are correctly identified; false negative (FN) is the number of normal samples that are wrongly identified as anomalies; and false positive (FP) is the number of anomalous samples that are wrongly identified as normal samples. The higher values of F_1 score, recall, and precision correspond to more accurate anomaly detection.

4.5 Result

4.5.1 Detection Accuracy

Table 2 presents the precision, recall and F_1 score of the baseline methods and our proposed framework for different datasets. It is shown that CSCAD demonstrates superior performance over all the baseline methods. Several baseline methods such as LOF and OC-SVM even failed within an acceptable running time due to the large data size (e.g., KDDCUP, UJIndoorLoc). It is observed that even training by selecting only the most conservative 2.5% high anomalous degree samples as negative samples, the proposed framework (CSCAD(2.5%)) achieves 10.2%, 15.7%, 1.5% and 7.5% improvement on F_1 score on different datasets over the best baseline methods (excluding VAE+DN and other CSCAD variant models). Moreover, comparing the results of CSCAD(2.5%, 5% and 7.5%), it is observed that conservatively selecting a very small proportion (much smaller than the anomaly ratio of the dataset) of high anomalous degree samples as negative samples for training still enables the discriminating network to maintain reasonable anomaly detection accuracy. This is important, as in many real-world scenarios, the actual anomaly ratio in the dataset is not known. It is desired to have a model that works with a very conservative estimate of the anomaly ratio while still achieves reasonable accuracy.

Several interesting observations can be made by analyzing the results in Table 2. It is observed that considering feature space correlation clearly improves anomaly detection accuracy. More specifically, compared with the VAE-DN (without GCN to perform correlation pattern filtering), the proposed framework achieved almost 30-60% improvement in terms of F_1 score on Thyroid, MoCap and UJIIndoorLoc datasets. This shows that considering feature space correlation makes a significant improvement on the detection accuracy when data features have strong internal correlation structure (e.g. interdependency of the physiological features under disease for Thyroid dataset, the correlated relative movement of the measuring points in the hand posture dataset MoCap and the underlying pattern of the relative position of the WiFi fingerprint signals in UJIIndoorLoc dataset.).

4.5.2 Impact of the Generated Anomalous Degree Measures

It is found that the CSCAD consistently achieves higher F_1 score compared with the variant method CSCAD(no σ) that

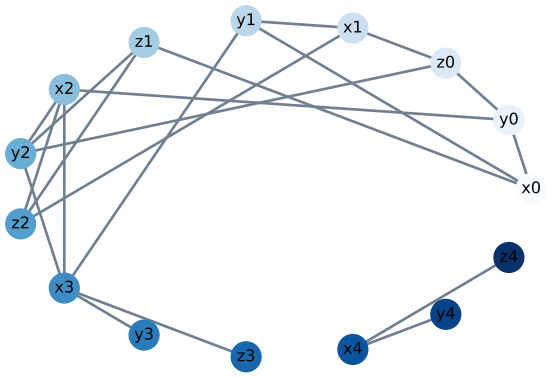


Fig. 6. Illustration of part of the correlation structure of MoCap dataset.

does not consider the latent standard deviation vector σ_z. This shows that considering only reconstruction loss may not fully reflect the anomalous behavior of a sample. Incorporating the internal uncertainty level information reflected in σ_z is also important. This observation is also confirmed in Fig. 5, which presents the visualization of detection results with respect to the L2-norms of reconstruction loss d and latent standard deviation vector σ_z. It is observed that there does not exist simple boundary values to separate the normal and anomalous samples by solely inspecting d or σ_z. However, by joint modeling both d and σ_z, we can learn a more robust discriminative boundary to detect anomalies.

4.5.3 Correlation Structure Mining

To demonstrate the necessity of the graph-based approach, we visualize the correlation graph constructed (see Fig. 6) from the features in the MoCap data. To observe the correlation structure between features more clearly, we keep only the connected edges of each node with its highest correlation node, and zoom in to show the correlation of some nodes. The x, y, z nodes with the same subscripts are the three directional axes of the same sensor. From the figure, we can see that the features with the same subscripts have high correlation. When collective anomalies occur, multiple nodes with high correlation (e.g., different features from the same sensor) may not appear to find anomalies when only one of them is observed, while observing multiple ones at the same time will reveal patterns that differ from the history. The graph convolutional layer in CSCAD takes into account multiple correlated features to better mine collective anomalous behavior.

4.5.4 Results Under Semi-Supervised Settings

CSCAD can be easily adapted to semi-supervised or supervised settings by using actual anomalous or normal samples in the positive and negative sample set during training the discriminative network. We conduct an experiment on the KDDCUP dataset to test the performance of our framework when limited amount of anomaly labels are known. Table 3 presents the results of CSCAD(2.5%) model when replacing 0% to 1% of the 2.5% high anomalous degree negative samples with the actual anomalies. It is observed that with the introduction of higher amount of labeled anomalies, the F_1 score increases from 0.934 to 0.937. Even without labeled data, the framework trained under unsupervised setting already achieved reasonable accuracy (0.934), with only 0.003 decrease in F_1 score compared with the case introducing 1% of actual anomalies during training. This demonstrates the robustness of the proposed framework.

TABLE 3
Experiment Results With Limited Anomaly Label Available for KDDCUP Dataset

Labeled Rate	Precision	Recall	F_1
0%	0.881	0.996	0.934
0.50%	0.883	0.990	0.934
0.75%	0.883	0.997	0.936
1.00%	0.881	1	0.937

TABLE 4
Experiment Results of Our Framework and the Baseline Methods for Heterogeneity Dataset

Methods	Heterogeneity		
	Precision	Recall	F_1
IF	0.830	**1**	0.907
SR-CNN	0.834	**1**	0.909
AE(LSTM)-IF	0.825	0.994	0.902
VAE(LSTM)-DN	0.953	**1**	0.976
CSCAD(LSTM)	**0.980**	**1**	**0.990**

4.5.5 Time-Series Extension

To demonstrate the extensibility of our framework, we also present the experiment results on the time-series dataset Heterogeneity in Table 4. We compare our framework (CSCAD(LSTM)) with a classic method (IF) and two deep learning-based approaches (AE(LSTM)-IF and VAE+DN (LSTM)). Our framework outperforms all the baseline methods in terms of precision, recall and F_1 score. Compared with IF and AE(LSTM)-IF, the proposed framework achieved 8-9% improvement on F_1 score. Again, we observe the feature space correlation helps to improve detection accuracy, which accounts for 1.4% increase on F_1 score compared with the VAE-DN (LSTM) model. These results show the effectiveness and extensibility of our framework.

5 Discussion and Conclusion

We propose a new adaptive framework (CSCAD) for collective anomaly detection on high-dimensional data for a large complex system with limited or no anomaly labels. Unlike the state-of-the-art approaches so far, CSCAD jointly considers the correlation structure in the feature space and robust sample reconstruction, which leads to superior performance in high-dimensional collective anomaly detection tasks. To the best of our knowledge, this is the first attempt to use the graph convolutional network as a filter to capture the variation in correlation structure among features, which enables more expressive modeling of the normal patterns in data. In this framework, we propose a new EMI metric which is capable of evaluating the correlation of data with different types (continuous and categorical) and properties (static and time-series). A reconstruction network is developed to perform sample reconstruction and evaluates two

natural anomalous degree measures of each sample: the reconstruction loss and the latent standard deviation. These two anomalous degree measures serve as inputs to a discriminating network to perform final anomaly detection, which is trained using high anomalous degree samples as positive samples, and low anomalous degree samples or samples with anomaly labels as negative samples. This scheme allows the discriminating network can be trained without a predetermined threshold and adaptable to semi- or supervised settings, which is a perfect fit for many real-world applications. Numerical results on five public datasets show that our framework consistently outperforms the baseline methods, which proves the effectiveness of CSCAD.

We believe the proposed CSCAD can build a bridge between the fields of correlation mining and the practical application of collective anomaly detection. The generated feature level anomalous degree measures can also be used in anomaly propagation tracing and anomaly source identification. The proposed CSCAD framework is a scalable and highly flexible approach that can extend to other scenarios such as time-series and image anomaly detection tasks.

REFERENCES

[1] B. Mukherjee, L. T. Heberlein, and K. N. Levitt, "Network intrusion detection," *IEEE Netw.*, vol. 8, no. 3, pp. 26–41, May/Jun. 1994.

[2] N. Abe, B. Zadrozny, and J. Langford, "Outlier detection by active learning," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2006, pp. 504–509.

[3] E. J. Candés, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *J. ACM*, vol. 58, no. 3, pp. 11:1–11:37, 2011.

[4] S. Zhai, Y. Cheng, W. Lu, and Z. Zhang, "Deep structured energy based models for anomaly detection," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1100–1109.

[5] B. Zong *et al.*, "Deep autoencoding gaussian mixture model for unsupervised anomaly detection," in *Proc. Int. Conf. Learn. Representations*, 2018.

[6] K. Tian, S. Zhou, J. Fan, and J. Guan, "Learning competitive and discriminative reconstructions for anomaly detection," 2019, *arXiv:1903.07058*.

[7] L. Bontemps *et al.*, "Collective anomaly detection based on long short-term memory recurrent neural networks," in *Proc. Int. Conf. Future Data Secur. Eng.*, Springer, 2016, pp. 141–152.

[8] D. B. Araya, K. Grolinger, H. F. ElYamany, M. A. Capretz, and G. Bitsuamlak, "Collective contextual anomaly detection framework for smart buildings," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2016, pp. 511–518.

[9] Y. Zheng, H. Zhang, and Y. Yu, "Detecting collective anomalies from multiple spatio-temporal datasets across different domains," in *Proc. 23rd SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2015, pp. 2:1–2:10.

[10] H. Zhang, Y. Zheng, and Y. Yu, "Detecting urban anomalies using multiple spatio-temporal data sources," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 1, p. 54, 2018.

[11] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[12] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *J. Netw. Comput. Appl.*, vol. 60, pp. 19–31, 2016.

[13] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1–15:58, 2009.

[14] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Process.*, vol. 99, pp. 215–249, 2014.

[15] M. Zhang, T. Li, Y. Yu, Y. Li, P. Hui, and Y. Zheng, "Urban anomaly analytics: Description, detection and prediction," *IEEE Trans. Big Data*, early access, Apr. 28, 2020, doi: 10.1109/TBDATA.2020.2991008.

[16] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, 1962.

[17] J. Kim and C. D. Scott, "Robust kernel density estimation," *J. Mach. Learn. Res.*, vol. 13, no. Sep., pp. 2529–2565, 2012.

[18] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special Lecture IE*, vol. 2, pp. 1–18, 2015.

[19] Y. Ikeda, K. Tajiri, Y. Nakano, K. Watanabe, and K. Ishibashi, "Estimation of dimensions contributing to detected anomalies with variational autoencoders," 2018, *arXiv:1811.04576*.

[20] H. Xu *et al.*, "Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications," in *Proc. World Wide Web Conf. World Wide Web. Int. World Wide Web Conf. Steering Committee*, 2018, pp. 187–196.

[21] M.-h. Oh and G. Iyengar, "Sequential anomaly detection using inverse reinforcement learning," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 1480–1490.

[22] J. Ramakrishnan, E. Shaabani, C. Li, and M. A. Sustik, "Anomaly detection for an e-commerce pricing system," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 1917–1926.

[23] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 2828–2837.

[24] A. Siffer, P.-A. Fouque, A. Termier, and C. Largouet, "Anomaly detection in streams with extreme value theory," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2017, pp. 1067–1075.

[25] G. Pang, C. Shen, and A. van denHengel, "Deep anomaly detection with deviation networks," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 353–362.

[26] H. Ren *et al.*, "Time-series anomaly detection service at microsoft," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 3009–3017.

[27] P. J. Huber, *Robust Statistics*. Berlin, Germany: Springer, 2011.

[28] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2017, pp. 665–674.

[29] R. Chalapathy, E. Toth, and S. Chawla, "Group anomaly detection using deep generative models," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, Springer, 2018, pp. 173–189.

[30] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[31] L. Deecke, R. A. Vandermeulen, L. Ruff, S. Mandt, and M. Kloft, "Image anomaly detection with generative adversarial networks," in *Proc. Mach. Learn. Knowl. Discov. Databases*, 2018, pp. 3–17.

[32] D. Li, D. Chen, J. Goh, and S.-k. Ng, "Anomaly detection with generative adversarial networks for multivariate time series," 2018, *arXiv:1809.04758*.

[33] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Proc. Int. Conf. Inf. Process. Med. Imag.*, Springer, 2017, pp. 146–157.

[34] A. Galka, T. Ozaki, J. B. Bayard, and O. Yamashita, "Whitening as a tool for estimating mutual information in spatiotemporal data sets," *J. Stat. Phys.*, vol. 124, no. 5, pp. 1275–1315, 2006.

[35] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014, *arXiv:1312.6114*.

[36] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.

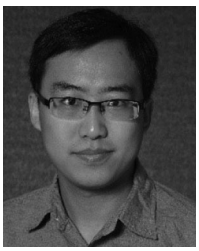
[37] J. Bekker and J. Davis, "Learning from positive and unlabeled data: A survey," *Mach. Learn.*, vol. 109, no. 4, pp. 719–760, 2020.

[38] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM Sigmod Record*, 2000, vol. 29, no. 2, pp. 93–104.

[39] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining.*, 2008, pp. 413–422.

[40] Y. Chen, X. S. Zhou, and T. S. Huang, "One-class SVM for learning in image retrieval," in *Proc. Int. Conf. Image Process.*, 2001, pp. 34–37.

Huiling Qin received the BE degree from Xidian University, where she is currently working toward the PhD degree with the School of Computer Science and Technology. Her current research interests include urban computing, spatio-temporal data mining, and graph-based model.



Xianyuan Zhan (Member, IEEE) received the dual master's degree in computer science and transportation engineering, and the PhD degree in transportation engineering from Purdue University. He is currently a research assistant professor with the Institute for AI Industry Research (AIR), Tsinghua University. Before joining AIR, he was a data scientist with JD Technology and also a researcher with Microsoft Research Asia (MSRA). His research interests include deep reinforcement learning, urban computing, and big data analytics in transportation.

Yu Zheng (Fellow, IEEE) is currently the vice president of JD.COM and the chief data scientist of JD Technology. He also leads the Intelligent Cities Business Unit, as the president and is the managing director of JD Intelligent Cities Research. Before joining JD Technology, he was a senior research manager with Microsoft Research and is also a chair professor with Shanghai Jiao Tong University, an adjunct professor with the Hong Kong University of Science and Technology. His research interests include big data analytics, spatio-temporal data mining, machine learning, and artificial intelligence. He was named an ACM distinguished Scientist in 2014 and elevated to an IEEE Fellow in 2020 for his contributions to spatio-temporal data mining and urban computing.