RESEARCH ARTICLE

Xilian YANG, Kanru CHENG, Qunfei ZHAO, Yuzhang WANG

Unknown fault detection for EGT multi-temperature signals based on self-supervised feature learning and unary classification

© Higher Education Press 2023

Abstract Intelligent power systems can improve operational efficiency by installing a large number of sensors. Data-based methods of supervised learning have gained popularity because of available Big Data and computing resources. However, the common paradigm of the loss function in supervised learning requires large amounts of labeled data and cannot process unlabeled data. The scarcity of fault data and a large amount of normal data in practical use pose great challenges to fault detection algorithms. Moreover, sensor data faults in power systems are dynamically changing and pose another challenge. Therefore, a fault detection method based on self-supervised feature learning was proposed to address the above two challenges. First, self-supervised learning was employed to extract features under various working conditions only using large amounts of normal data. The self-supervised representation learning uses a sequencebased Triplet Loss. The extracted features of large amounts of normal data are then fed into a unary classifier. The proposed method is validated on exhaust gas temperatures (EGTs) of a real-world 9F gas turbine with sudden, progressive, and hybrid faults. A comprehensive comparison study was also conducted with various feature extractors and unary classifiers. The results show that the proposed method can achieve a relatively high recall for all kinds of typical faults. The model can detect progressive faults very quickly and achieve improved results for comparison without feature extractors in terms of F1 score.

Received Dec. 31, 2022; accepted Apr. 18, 2023; online Jun. 20, 2023

Xilian YANG, Qunfei ZHAO School of Electronic Information and Electrical Engineering, Shanghai

Jiao Tong University, Shanghai 200240, China

Kanru CHENG, Yuzhang WANG (🖂)

School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China E-mail: yuzhangwang@sjtu.edu.cn **Keywords** fault detection, unary classification, selfsupervised representation learning, multivariate nonlinear time series

1 Introduction

1.1 Motivation

Nowadays, a large number of commercial power generation gas turbines and aircraft engines are in service and need regular maintenance which is called condition monitoring. Therefore, the automatic detection technology to reduce manpower is very necessary [1]. Timely and accurate detection of faults in power systems can not only avoid blackouts and losses, but also help provide stable power guarantees for people living in extreme weather [2]. Condition monitoring is conducive to precise control and decreases maintenance costs [3]. Fault detection is the first step in the four steps of condition monitoring [4], i.e. anomaly detection, fault classification, fault isolation, and fault mitigation. Fault detection also helps to achieve intelligent control [5] and prevent attacks [6]. Timely and accurate fault detection is conducive to realizing the condition monitoring and safe operation of the power system.

A large number of state sensing-sensors are installed for intelligent condition monitoring. Among the faults statistics of three 9F gas turbines from a power plant located in a eastern coastal city in China, the number of sensor faults is the highest during use, as shown in Fig. 1. For the availability of a large amount of operational data and advances in computing power, data-based detection technologies have become popular. Data-based detection technologies generally require four steps: data acquisition, data cleaning, feature extraction, and fault classification [7]. These installed sensors can provide a large amount of operating data, but considering the catastrophic losses caused by gas turbine failures, few power plants will allow gas turbines to operate with fault. Collecting fault data is therefore time-consuming and expensive [8]. In addition, a complex system such as a gas turbine contains various components that interact with each other, making the possible types of faults innumerable [9]. Ordinary classification-based fault detection methods may not work because the number of classifications needs to be determined in advance.



Fig. 1 Statistics of sensor fault number from the average of three 9F gas turbines in power plants.

In this study, multiple exhaust gas temperatures (EGTs) are taken as an example to describe and discuss the multisensor fault detection problem. Sensor fault has two primary reasons, the first being the power system itself malfunction, and the second being the sensing elements malfunction. As combustion has the most adverse environment in the power system, EGTs indicate that the combustion state need detection urgently. Additionally, installing more sensors in the outlet of the turbine will also increase the probability of sensor fault.

1.2 Related work

Sensor data fault detection has two main research directions, physical-based fault detection [10,11] and databased fault detection [12]. Physical-based methods obtain detection results through physical modeling [13]. The most prominent advantage of physical-based methods is highly interpretable. But physical modeling cannot take into account all factors, and the error between the final modeling results and the real results is unknown. Databased methods do not generalize well, but are gaining popularity due to the recent emergence of Big Data, convenient models, and computation power [14,15]. Data-based fault detection methods learn model parameters by using the collected historical information, and make judgments for the present, which mainly include classification-based [16,17], statistics-based [18], clustering-based [19] and reconstruction-based detection [20]. The latter three are unsupervised methods, which need to manually set the number of cluster categories or the value of outlier thresholds, and is not easily scalable to

multivariate sensor processing. For example, the reconstruction-based methods need to set a threshold to the difference between the regressed reconstructed value and the truth value [21].

Classification-based methods may not work well in practical applications due to the cost and limited amount of fault data. Treating fault detection as a general binary classification problem is somewhat inappropriate, because the class of normal data has defined boundaries and the class of abnormal data has uncertain boundaries. These uncertain boundaries during classification pose challenge to the accuracy of results. Some scenarios even have no fault samples [7], thus the boundaries between normal and abnormal samples are unknown. If the generated fault data serves to build boundaries [22], this generated fault data may miss some corner cases. Deep learning methods become popular in recent years [23,24]. Deep network learning for classification uses the crossentropy loss [25,26], but sets a too-high confidence level for each class. Out-of-distribution [27] improves the cross-entropy loss, but still draws the boundaries between known and determined classes. Positive unlabeled learning [28] defines the boundary of positive and negative samples from the construction of local similarity, avoiding the dilemma of incomplete labels of positive samples. Therefore, representation learning in combination with some traditional algorithms like unary classification can avoid the use of cross-entropy loss.

To perform the task of sensor fault detection, the sensor output data are taken as a time series data structure. Time series representation is an important research field in time series data mining because it is difficult to directly apply data mining algorithms to raw time series data due to its natural characteristics. Therefore, a fast and effective time-series representation is needed [25]. Since a variable is treated as a dimension, the dimension of a large amount of raw data are usually huge [29], because the control system of power engine will have hundreds or thousands of sensors installed. The costs of high-dimension data processing and storage are high. Therefore, it is necessary to extract the original data, but to guarantee that possible key information is not lost. Usually, the original collected signals in power systems will be accompanied by white noise. Signals with white noise will affect the recognition of features to a certain extent. Therefore, time series data mining algorithms need to be robust to noise [30].

There are many methods for extracting features, and what kind of feature to extract depends on experts' prior knowledge, e.g., geometric shape information [31], statistical feature information, frequency information [32,33], regression model information, matrix decomposition [34,35]. Fault detection involves a large number of installed sensors. Therefore, fault detection is a case of multivariate time series mining. The popularity of motif discovery techniques that can handle multivariate time

series continues to grow. Therefore, more complex representation learning emerges, such as fully connected networks [36], recurrent neural networks [37], and convolution networks [29]. After obtaining the trained model, transfer learning can be conducted for a specific domain [38,39]. Then, first of all, it is necessary to obtain a strong expressive model for the multi-sensor fault detection task.

The existing problems in sensor fault detection of power systems are as follows: First, the original sampled data are large amounts of normal data, and the real fault data are little for fault detection of power systems. General supervised learning may have trouble dealing with untrained fault data during practical usage. Then, it would be more difficult to represent the raw EGT data from multi-sensors in the control system of power systems if the fault data and normal data overlap with each other in the original dimension. Finally, simple accuracy evaluation criterion cannot reach satisfied detection task. For example, if there is only 1% of fault data, the final detection accuracy is at least 99%, which is obviously not competent for fault detection tasks.

1.3 Contributions

The proposed multi-sensor fault detection method is contributive because, first, this study intends to solve the dilemma of unknown abnormal data in the practical usage of multi-sensor fault detection using zero fault data during training. The proposed method can extract features of various high-dimensional fault signals without manual annotation for convenient and quick online fault detection. Then, a self-supervised representation learning based on Triplet Loss is introduced to train on a large amount of normal data and to quickly obtain representations of original multi-signal data including various complex untrained fault signals during the test. Those extracted features from multi-signals are used as input for unary classification to realize unknown fault detection. Finally, several datasets from real-world power systems are collected for experimental verification. Various faults are superimposed on the multi-sensor dataset. Fault detection results are analyzed thoroughly using comprehensive evaluation metrics. The comparisons with stateof-art methods are demonstrated qualitatively and quantitively for evaluating the validity of the proposed method.

2 Problem definition

A large number of state-aware sensors in power systems can be roughly divided into physical redundancy and spatial redundancy. Physical redundancy refers to sensors installed on the same part but of different types, such as the fuel-given signal, inlet temperature, outlet pressure, and rotational speed of the compressor in the left part of the data acquisition module in Fig. 2. They have a certain physical relationship due to being installed on the same part. Spatial redundancy refers to the installation of multiple sensors of the same type in the same space, typically a ring of EGT installed at the turbine outlet, as shown in the right part of the data acquisition module in Fig. 2.

Except for the data acquisition module, Fig. 2 also shows the self-supervised representation learning and unary classification, the evaluation metrics in the processing flow. The direction of the blue arrow on the left side of Fig. 2 is the training process for the proposed method. The input data are a large number of multivariate EGT data in a normal state. First, the model RNN for representation learning is obtained through Triplet Loss, and then the representation results after training are input into a unary classifier. The direction of the orange arrow on the right is the test process for the proposed method. The test dataset contains unbalanced normal and fault data. First, an intermediate result is obtained through the representation learning model, and then the intermediate result is input into the trained unary classifier. Finally, the test results are evaluated with comprehensive evaluation metrics.

The goal of multivariate sensor fault detection is to detect the abnormal state timely given observed value from history sensors on the signal network. The recorded exhaust gas temperature signal sequences are denoted as multivariate time series. Supposing the number of exhaust gas temperature signals is N, the sensor data recorded at time t is represented by $\mathbf{x}_t^{\mathrm{T}} \triangleq [\mathbf{x}_t^1, \mathbf{x}_t^2, \dots, \mathbf{x}_t^N]$, $t \in [1, L]$. L is the recorded time period of the dataset. A subsequence of $\mathbf{x}_{1:L}$ with a length of τ is represented as $\mathbf{x}_{t_0-\tau+1:t_0} \triangleq [\mathbf{x}_{t_0-\tau+1}, \mathbf{x}_{t_0-\tau+2}, \dots, \mathbf{x}_{t_0}]^{\mathrm{T}}$. The fault signal detection problem is modeled as

$$\boldsymbol{x}_{t_0-\tau+1:t_0} \stackrel{\mathcal{G}(\phi)}{\to} \boldsymbol{H}^{t_0} \stackrel{\mathcal{F}(\theta)}{\to} pos, \qquad (1)$$

where $\mathbf{x}_{t_0-\tau+1:t_0}$ is the input subsequence of the EGT data and model \mathcal{G} is the feature extractor RNN learned in Section 3. Model \mathcal{G} could process a large amount of unlabeled data through self-supervised representation learning. Model \mathcal{F} is the unary classifier learned in Fig. 2. The normal data distribution is as prior knowledge for a fully unknown fault. The final output result *pos* has two integer symbols representing inliers and outliers.

3 Multi-signal fault detection methods

Recently, data-based detection methods have been widely used because of their high accuracy and Big Data availability. However, supervised networks employ cross-entropy loss, using each known and determined number of classification labels during training. Then, the cross-entropy loss is unable to deal with unseen samples



Fig. 2 Schematic of the processing flow of the proposed method.

during testing. When a sensor fault occurs, there is no guarantee that the fault sample must have appeared in the historical data. Therefore, the task of a classifier is to detect faults by comparing the difference between normal data and fault data in the feature domain. This unary classifier is only trained on normal samples, then tests different types of fault samples. The whole process during the online test is shown in Fig. 3.

The recorded samples from real industry contain no fault data. Therefore, different kinds of fault data are superimposed on real-world data to make an imbalanced fault dataset as shown in Section 4.1. The backbone of self-supervised learning is an RNN as shown in Section 3.1.2. In addition, a comparative analysis is conducted for one-class classifiers including extracted features and original data.

3.1 Representation learning

A large amount of normal data are labor-intensive to label, while self-supervised representation learning can automatically extract the effective features of the input multivariate sequence without labeling while exploiting a large amount of normal data. Extracted features can serve for subsequent machine learning tasks, like fault detection. Before sequenced-based Triplet Loss is investigated, Triplet Loss with labels is first introduced to get the preliminary idea of embedding learning.

3.1.1 Triplet Loss with labels

In supervised learning networks, the learning of fixed classes generally uses cross-entropy loss. But in some cases, the number of categories is unknown. We can compare two samples to see if they are similar to learn the representation of the sensor data. The purpose of Triplet Loss is that in the representation space, samples of the same class with the same label should be very close; samples of different classes with different labels should be far away from each other.

In Triplet Loss [40], in order to define the loss function, the samples are divided into anchors, positive

samples, and negative samples. Positive samples and anchors belong to the same class, while negative samples and anchors belong to different classes. The selection of triplets is introduced in Algorithms 1. Supposing the label of the subsequence $\mathbf{x}_{t_0-\tau+1:t_0}$ is y_{t_0} . y_{t_0} is denoted as y for simplicity, hence y^a denotes the label of anchors, y^p denotes positive the labels of samples, and y^n denotes the labels of negative samples. The loss based on some distance metric d is expressed as

$$\mathcal{L} = \max(d(y^{a}, y^{p}) - d(y^{a}, y^{n}) + margin, 0).$$
(2)

In the process of minimizing the loss \mathcal{L} , $d(y^a, y^p)$ tends to be zero, and $d(y^a, y^n)$ is larger than $d(y^a, y^p) + margin$. The purpose of setting the margin is to increase the distance between positive and negative samples. This margin is conducive to distinguishing positive and negative samples during testing. In training, each element of the triplet needs to be set according to the difficulty levels of the label. According to the distance between $d(y^{a}, y^{p})$ and $d(y^{a}, y^{n})$, the triplets are divided into three difficulty levels. $d(y^a, y^p) > d(y^a, y^n)$ is difficult-to-mine triplet. $d(y^{a}, y^{p}) < d(y^{a}, y^{n}) < d(y^{a}, y^{p}) + margin$ is moderately-difficult-to-mine triplet. $d(y^a, y^p) + margin < d(y^a, y^n)$ is easy-to-mine triplet. It is further extended to divide the negative samples into three categories: hard negative samples, semi-hard negative samples, and easy negative samples. When all negative samples become easy negative samples, the loss will tend to be zero.

3.1.2 Sequenced-based Triplet Loss

The Triplet Loss training process in Eq. (2) requires labels. However, in fault detection, some faults may not have occurred before and labels are unavailable. When learning from unlabeled data, self-supervised learning is required. As in the work of word2vec [41], the label is not used in the process of selecting triplets, instead the negative sampling strategy is used. The specific idea is that, in the embedding space, the representation of a sentence is closer to the words in the sentence, and is farther away from other randomly selected words that are not in the sentence.



Fig. 3 Proposed fault detection processing steps.

Applied to the time series in the sensor fault detection task, any subsequence $x_{t_0-r+1:t_0}$ is abbreviated as x. As shown in Fig. 4, considering an arbitrary subsequence x^a , on the one hand, x^a should be very close to the representation of its subsequence x^{pos} . On the other hand, x^a should be far away from the randomly selected subsequence x^{neg} of the rest of the sensors, or far away from the subsequence x^{neg} of the same sensor in different working conditions. To improve the stability and convergence speed of the learned representation, a set of independent K negative samples is randomly selected. The self-supervised Triplet Loss function based on time series is expressed as

$$\mathcal{L} = -\log\left(\sigma\left(\mathcal{G}(\boldsymbol{x}^{a}, \boldsymbol{\phi})^{\mathrm{T}} \mathcal{G}(\boldsymbol{x}^{pos}, \boldsymbol{\phi})\right)\right) - \sum_{k=1}^{K} \log\left(\sigma\left(-\mathcal{G}(\boldsymbol{x}^{a}, \boldsymbol{\phi})^{\mathrm{T}} \mathcal{G}(\boldsymbol{x}^{\mathrm{neg}}_{k}, \boldsymbol{\phi})\right)\right), \quad (3)$$

where $\mathcal{G}(\cdot, \phi)$ is a deep network with parameters ϕ , σ is the sigmoid function and the purpose of the loss function is to make the representations of x^a and x^{pos} as similar as possible, and to make the representations of x^a and x^{neg} as different as possible. In the training process, the specific steps of the selection process of the anchor, positive sample, and negative sample triplet are

Algorithm 1: Strategies of choosing x^a , x^{pos} and x_k^{neg} on sensor data of length L

Let's define $[m,n] \triangleq [m,m+1,\ldots,n], m,n \in \mathbb{N}$. For $i \in [1,N], l_i = len(\mathbf{x}^i)$.

1) Determine the length of anchor and positive samples, uniformly and randomly select l^{pos} in $[1, l_i]$, and l^a in $[l^{pos}, l_i]$;

2) Determine the subsequence of anchor, uniformly and randomly select x^a in the subsequence of x^i , $l^a = len(x^a)$;

3) Determine the positive sample subsequence, uniformly and randomly select x^{pos} in the subsequence of x^a , $l^{pos} = len(x^{pos})$; and

4) Determine the negative sample subsequence, uniformly and randomly select $k \in [1, N]$, l_k^{neg} in $[1, len(\mathbf{x}^k)]$, then $\mathbf{x}_k^{\text{neg}}$ is in the subsequence of \mathbf{x}^k , $l_k^{\text{neg}} = len(\mathbf{x}_k^{\text{neg}})$.

The deep network $\mathcal{G}(\cdot, \phi)$ in Eq. (3) is a recurrent

neural network (RNN), especially a gated recurrent unit (GRU), a simple yet powerful type of RNN. GRU can efficiently model nonlinear multivariate time series and has achieved good results on many tasks. The structure of the RNN in Fig. 4 is

$$\boldsymbol{r}^{(t_0)} = sigmoid(\boldsymbol{\phi}_{\mathrm{r}}[\boldsymbol{x}, \boldsymbol{h}^{(t_0-1)}] + \boldsymbol{b}_{\mathrm{r}}), \qquad (4)$$

$$\boldsymbol{u}^{(t_0)} = sigmoid\left(\boldsymbol{\phi}_{u}[\boldsymbol{x}, \boldsymbol{h}^{(t_0-1)}] + \boldsymbol{b}_{u}\right), \quad (5)$$

$$\boldsymbol{C}^{(t_0)} = tanh(\boldsymbol{\phi}_{c}[\boldsymbol{x}, (\boldsymbol{r}^{(t_0)} \odot \boldsymbol{h}^{(t_0-1)})] + \boldsymbol{b}_{c}), \quad (6)$$

$$\boldsymbol{h}^{(t_0)} = \boldsymbol{u}^{(t)} \odot \boldsymbol{h}^{(t_0-1)} + (1 - \boldsymbol{u}^{(t_0)}) \odot \boldsymbol{C}^{(t_0)}, \quad (7)$$

where at time t_0 , as before, any subsequence $x_{t_0-r+1:t_0}$ is abbreviated as x, and treated as anchor x^a . Positive and negative sample sequences x^{pos} and x_k^{neg} are selected according to **Algorithm 1**. x^a , x^{pos} , and x_k^{neg} are respectively input into the RNN encoder, and finally the loss is obtained according to Eq. (3). $h^{(t_0-1)}$ denotes output at the previous step. $r^{(t_0)}$, $u^{(t_0)}$, and $C^{(t_0)}$ denote the reset gate, update gate, and candidate gate. ϕ_r , ϕ_u , ϕ_c , b_r , b_u , and b_c are the parameters for the corresponding network layer in each gate. \odot is Hadamard product denoting element-wise product.

3.2 Local outlier factor

Local outlier factor (LOF) is introduced for the significance of prior knowledge from a large amount of normal data. LOF mainly compares the difference between local information and neighbor information, and judges the abnormality according to the proportion of abnormal data [42]. Therefore, LOF is more suitable for diagnosing progressive and difficult-to-diagnose faults such as drift. Unlike the classification and clustering method, they require a balanced number of data samples of different classes, otherwise, there will be an imbalanced classification problem, or the statistical method needs to assume that the samples are Gaussian distribution, and the distribution characteristics of fault samples should be significantly different from normal samples.

LOF gives a comparison score by comparing the local density of a sample with the *k*-nearest neighbor density of



the sample. The reachable distance of sample x is defined as

$$rd_k(\boldsymbol{x}, \boldsymbol{z}) = \max\{k - d(\boldsymbol{z}), d(\boldsymbol{x}, \boldsymbol{z})\}.$$
(8)

When x and z are far apart, the reachable distance is itself; when x and z are very close, the reachable distance is the farthest point in the *k*-nearest neighbors to z, represented by k-d(z). The set of *k*-nearest neighbors is denoted as $N_k(x)$. The local reachability density of sample x is defined by

$$\operatorname{lrd}_{k}(\boldsymbol{x}) = 1 \left| \left(\frac{\sum_{z \in N_{k}(\boldsymbol{x})} r d_{k}(\boldsymbol{x}, \boldsymbol{z})}{|N_{k}(\boldsymbol{x})|} \right).$$
(9)

The final LOF is defined as

$$\operatorname{LOF}_{k}(\boldsymbol{x}) = \frac{\sum_{z \in N_{k}(\boldsymbol{x})} \frac{\operatorname{lrd}_{k}(z)}{\operatorname{lrd}_{k}(\boldsymbol{x})}}{|N_{k}(\boldsymbol{x})|}.$$
(10)

LOF_k is the ratio of the local reachable distance of sample x to the local reachable distance of the k-nearest neighbors of x. LOF is tough to interpret because it is a ratio. There is no specific threshold value above which a sample is defined as an outlier. The detection of an outlier is dependent on the problem. Despite the disadvantage, LOF can effectively and robustly identify the local outliers. A sample will be treated as an outlier even if the sample is at a small distance from the extremely dense cluster samples, while the global approach may not consider that sample as an outlier.

4 Implement details

4.1 Fault data sources

The power systems often work for a long time and do not stop frequently. Therefore, a large amount of normal data can be collected. In real working conditions, the general strategy for sensor fault is shutting down immediately. If the sensor data can be monitored and diagnosed on the fly, an intelligent response plan can help avoid the inconvenience and property loss caused by sudden blackouts. First, the feasibility of the proposed algorithm should be verified on a public simulated gas turbine dataset, and then the effectiveness of the proposed algorithm is further verified on the real gas turbine dataset.

The simulation dataset comes from the NASA Turbofan Jet Engine Dataset [43], which contains 21 measurement sensors, and 4 sets of training and testing data. The length of each training set and test set is 30000 sampling points. Regularization processing is performed before data training. The second and fourth sets of data were used to verify the proposed fault detection method, and various types of faults were added to the test sets of these two sets of test data.

The real-world experimental data comes from the 9F gas turbine control system of an in-service power plant

located in an eastern coastal city in China. The total sensor variable number is 33 including 20 exhausted gas temperatures. The collected data from a power plant are 3 sections of 10079 sampling points, sampling at every 1 min. The most complex first section is used as the training dataset. The training set includes the continuous working conditions of fueling and stopping. The second section is regarded as the verification dataset, and the full load operation process in the third section is regarded as the test dataset. All data do regularization with the mean and variance of the training dataset before inputting to the network.

Five types of faults are superimposed on the real operation data, including the short, step, drift, noise, and periodic faults [44] generated by TimeSynth (from github) as shown in Fig. 5. These five kinds of faults sensor fault can be due to humidity, pressure, temperature, sensor fixed bias, drifting bias, precision degradation, and complete failure [45]. To simulate the imbalanced classification problem in real situations, the fault data only occupies less than 5% of the entire test dataset. To show the distribution law of fault data and normal data, the histogram specially adjusts the ratio of fault data and normal data to a balanced 1:1 in Fig. 6.

The fluctuations of generated faults with random white noise are more in line with the real sampling situation. Different faults have a certain typical representation because of different amplitudes, different trends, and different distributions as shown in Fig. 6. The distribution characteristics of short and step faults are similar. They are relatively clustered, the distance from the normal data are far away, and their respective boundaries are very clear. Due to the different magnitudes, the distribution of short and step faults is just opposite to that of the normal data. However, drift faults, noise faults, and periodic faults are mixed with the original data, and there is no clear dividing line. Although the distribution of drift faults and periodic faults is very similar, due to the amplitude, the data of periodic faults are evenly distributed on both sides of the normal data, while the drift faults are only located on one side of the normal data. The distribution of noise data is also located on both sides of the normal data, and it is light at both ends and heavy in the middle. These characteristics of various faults require the model in representation learning to have strong representation ability.

4.2 Evaluation metrics

4.2.1 Classification evaluation metrics

The confusion matrix consists of truth label and predicted label. Positive is defined as abnormal, while negative as normal. True positive (TP) means abnormal detected as abnormal, false negative (FN) means abnormal detected



Fig. 5 Real original EGT signals imposed with various fault data.

as normal, false positive (FP) means normal detected as abnormal, and true negative (TN) means normal detected as normal. Intuitively, the higher the proportion of TP and TN on the diagonal matrix indicates the better results. For fault detection, it is vitally important to detect all positive samples, otherwise the algorithm is inferior. The proposed fault detection approach is to be evaluated using the following metrics.

1) **Recall** represents what percentage of the true positive samples are detected:

$$TP/(TP + FN) = TP/P.$$
 (11)

2) Precision represents what percentage of the detected positive samples are really positive:

$$TP/(TP + FP) = TP/PP.$$
 (12)

3) F-measure is the harmonic mean of precision and recall, called F1-score:

$$F1-score = (2 \times Precision \times Recall)/(Precision + Recall).$$
(13)

4) Balanced accuracy is suitable for unbalanced dataset evaluation:

$$\frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} + \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}\right) = \frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{P}} + \frac{\mathrm{TN}}{\mathrm{N}}\right).$$
 (14)

5) Matthew correlation coefficient:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}.$$
(15)

6) False alarm represents what percentage of the real negative samples are not detected:

$$\frac{FP}{FP+TN} = \frac{FP}{N}.$$
(16)

The false alarm is different from the previous five evaluation metrics. A higher the first fifth metrics indicate better results, while a lower the false alarm rate indicates better results.

4.2.2 Clustering evaluation metrics

Clustered samples can be evaluated using intra-class and inter-class distance between classes. Later cluster evaluation metrics are used to measure self-supervised representation learning models.

1) Silhouette coefficient:

For a single sample *x*:

$$s = \frac{q - p}{\max(p, q)},\tag{17}$$

where p is the average distance between the self-sample and all other samples in the same class. q is the average distance between the self-sample and all samples in the nearest class. For a series of samples, the silhouette



Fig. 6 Various real data with fault data visualization (left: original scale; right: scatter with histogram).

coefficient is the *s* average of all samples. The metric is between -1 and +1, with -1 for incorrect clusters, +1 for very dense clusters, and 0 for overlapping clusters.

4.3 Hyperparameters

All mentioned methods in this research are implemented in Python with deep learning under PyTorch. According to previous prediction research [46,47], 36 is a suitable equidistant window length for real-world 9F gas turbine dataset. Because too long a window length brings a burden for RNN and becomes less accurate while too small a window length carries not enough information to extract features. The window lengths do not overlap each other and the moments of windows are adjacent. The number of input dimensions is fixed to 20 by the number of recorded signals. The parameters of the intermediate hidden layer should be appropriately larger to map to a high-dimensional space, which is set to 100. The number of RNN layers is set to 1 and bidirectional parameter is set to true in order to obtain a lightweight model for online detection. Besides, the number of output features should not be too large because representation learning aims to reduce high dimensional Big Data, which is set to 10. Some important experimental hyperparameters of corresponding dataset are shown in Table 1. NASA [39] represents the public dataset, while 9F EGTs represents the collected real-world data from a power plant.

NASA 128 21 128	2 True 10	
9F EGTs 36 20 100	1 True 10	

Table 1 Experiment parameters

5 Results and discussion

5.1 Fault detection results

The unary classification algorithm divides normal and abnormal according to the paradigm and rules of the normal state, and can flexibly handle various unknown types of faults. A total of four unary classification algorithms based on different principles are compared to show the robustness of the proposed detection strategy, the LOF, and three others.

1) Robust covariance [48] determines the multivariate minimum determinant by assuming the proportion of abnormal samples. Robust covariance randomly selects a certain number of samples to calculate distribution parameters. The covariance with the smallest determinant is regarded as the mean and covariance of the sample distribution, so that the mean and covariance will not be affected by abnormal samples. This unary classifier is robust against outlier values. The abnormal samples are determined according to the Mahalanobis distance which represents the distance from the sample to the normal distribution. This algorithm fits exactly for multivariate Gaussian distribution.

2) One-Class SVM [49] uses a kernel function to map samples to high-dimensional space. One-Class SVM determines the outliers using the hyperplane boundary of normal samples, so that samples without labels can be processed. The proportion of abnormal samples is achieved by adjusting the relaxation factor.

3) Isolation Forest [50] randomly selects a value between the maximum and minimum values of a feature and recursively divides the dataset. An Isolation tree is used to represent the divided data. The number of divisions required to separate a sample is called a path. The sample with the shortest path is considered abnormal and the sample with a longer path is normal. The method need not calculate the parameters of a dataset distribution.

The general supervised multi-class classification method is not adopted because of the scarcity and unavailability of abnormal data in real work conditions. Instead, a unary classification algorithm that only requires training normal samples is adopted. A total of three state-of-art self-supervised representation learning methods are compared, the Triplet Loss and two others.

1) CPC [51] is short for contrastive predictive coding. CPC learns representation by predicting the hidden space and uses a probability contrast loss, so that the hidden space can capture as much useful information as possible to predict future samples. Specifically, the noise-contrast loss maximizes the mutual information between encoded representations. A nonlinear encoder is used to encode local information, and a regression model is used to summarize the encoded hidden space sequence.

2) TNC [52] is short for temporal neighborhood coding. To solve the problem of negative sample sampling bias in contrastive learning, TNC adopts the idea of positive-unlabeled learning. TNC treats unlabeled data as negative samples with a small weight. The small weight is reflected in the loss function to deal with the situation that samples are not neighboring pairs but may also be positive samples. The idea is reflected in the loss function that the features of neighboring samples are similar, and the features of non-neighboring samples are adjusted according to the weight.

To ensure a fair comparison, the three self-supervised representation learning methods use the same encoder to ensure that the results are not influenced by different encoders.

5.1.1 Test on sudden faults, progressive faults, and hybrid faults

F1-score is a common comprehensive evaluation metric because it balances precision and recall, and thus is crucial for the evaluation of imbalanced classification datasets. Fault detection is a typical imbalanced classification problem. The overall comparison between the proposed method and the original data detection results is shown in Fig. 7. The proposed method couples Triplet Loss with LOF and is able to mine high dimensional data efficiently. The proposed method performs well under various fault situations while the result of original data using unary classification may collapse for unknown faults. The effect of different representation learning models for different faults will be discussed in detail in combination with the real-world 9F EGTs dataset.

A unary classifier can detect serious sudden faults, such as short or step faults, when using only original signal data. As shown in Fig. 6, the distribution of short fault and normal data are easily distinguished, because they are two multivariate Gaussian distributions with a large difference between their means. This case is favorable for robust covariance. Because statistics-based robust covariance is particularly suitable when the fault distribution and the normal data distributions. As the difference between the fault distribution and the normal



Fig. 7 Comparison of Triplet Loss coupling LOF of proposed method with original data using only unary classification under F1-score.

data distribution becomes smaller and overlaps with each other, the performance of robust covariance begins to decline. Therefore, as shown in Table 2, the original data perform significantly well in terms of the F1-score, and self-supervised feature extractors also perform well in terms of other metrics.

Table 2Short fault, F1-score of various unary classificationalgorithms coupling with various feature extractors

Unary classifier	CPC	TNC	Triplet Loss	Original data
Robust covariance	0.615	0.696	0.889	0.963
One-Class SVM	0.421	0.421	0.085	0.548
Isolation Forest	0.410	0.457	0.500	0.537
LOF	0.727	0.696	0.762	0.059

As shown in Fig. 6, the step fault features are not as obvious as the short fault, and the deviation from the mean of the normal data are only half of the short fault. Thus, the performance of original data detection is not as good as the self-supervised representation learning features as shown in Table 3, but still acceptable. Since the step fault is also a multivariate Gaussian distribution, robust covariance performs the best among the four unary

Table 3 Step fault, F1-score of various unary classification algorithms coupling with various feature extractors

• . •				
Unary classifier	CPC	TNC	Triplet Loss	Original data
Robust covariance	0.593	0.696	0.889	0.877
One-Class SVM	0.421	0.421	0.085	0.548
Isolation Forest	0.410	0.457	0.500	0.537
LOF	0.727	0.696	0.762	0.076

classification algorithms. The original data performance is inferior to the Triplet Loss feature extractor when using robust covariance due to the lower recall.

After the features are extracted by self-supervised representation learning from the original data, the features of the fault signal become closer in feature space and easier to a classifier. Thus, the unary classifier can deal with more common complex faults, such as drift faults, noise faults, and periodic faults with the help of a self-supervised representation learning based feature extractor. As shown in Fig. 6, the distribution of drift fault is completely different from step or short fault. Drift fault is a uniform distribution with gradual changes. This gradual change overlapping with original data makes the drift fault completely undetectable. As shown in Table 4, the F1-scores of the four unary classifiers are all close to 0 when using original data without any feature extraction process. On the contrary, features extracted by Triplet Loss obtain the best F1-score of 0.8 with the help of LOF. Because LOF is not affected by the overall probability distribution and relies on the comparison between the local density and the k-nearest neighbor density of the sample point.

The distribution characteristics of noise faults and periodic faults are somewhat similar, and normal data and

 Table 4
 Drift fault, F1-score of various unary classification algorithms coupling with various feature extractors

Unary classifier	CPC	TNC	Triplet Loss	Original data
Robust covariance	0.583	0.667	0.714	0.000
One-Class SVM	0.432	0.432	0.085	0.013
Isolation Forest	0.421	0.471	0.516	0.073
LOF	0.727	0.727	0.800	0.244

abnormal data are overlapping with each other as shown in the scatter diagram in Fig. 6. The original data cannot detect those two faults as shown in Tables 5 and 6. Even combined with the feature extractor, a classificationbased algorithm such as One-Class SVM still fails. At the same time, robust covariance performs a little better than the LOF. The feature extractor only strengthens some statistical features of the original distribution, such as the variance of the sample distribution, but cannot extract all the features completely, such as the period. A period and slope extraction modules are further needed to classify fault types.

 Table 5
 Noise fault, F1-score of various unary classification algorithms coupling with various feature extractors

Unary classifier	CPC	TNC	Triplet Loss	Original data
Robust covariance	0.593	0.636	0.824	0.094
One-Class SVM	0.333	0.286	0.085	0.473
Isolation Forest	0.368	0.457	0.452	0.468
LOF	0.727	0.696	0.700	0.319

 Table 6
 Periodic fault, F1-score of various unary classification algorithms coupling with various feature extractors

Unary classifier	CPC	TNC	Triplet Loss	Original data
Robust covariance	0.615	0.696	0.889	0.000
One-Class SVM	0.378	0.378	0.085	0.489
Isolation Forest	0.410	0.457	0.500	0.485
LOF	0.545	0.696	0.762	0.308

By comparing different self-supervised feature learning and unary classification algorithms, it is found that Triplet Loss combined with LOF is competent for various types of faults. This combination achieves the highest overall F1-score. In addition to the F1-score, the Matthews correlation coefficient (MCC) is also considered as a balanced measure, as it considers all the four elements in the confusion matrix. Even in the case of an extremely unbalanced dataset, the MCC can still fully reflect all the information on binary classification results. Moreover, the F1-score is affected by the definition of whether the fault data are a positive or negative sample, while the MCC is not affected by this definition. As shown in Fig. 8, the MCCs of five types of faults using the unary classifier with various self-supervised representation learning or original data are compared. First, the accuracy of most original data is low, except for short faults under the robust covariance. The reason for this is that the robust covariance assumes that the sample is based on a Gaussian distribution, and subsequently checks whether the mean and variance of the sample deviate from the mean and variance of the normal sample. At this time, the short fault is just a Gaussian distribution that deviates far from the normal sample. Therefore, it is easy to detect in such a special case. The LOF performance changes the most after passing through the feature extractor, especially for the Triplet Loss. MCC is a more discriminating metric than balanced accuracy. As a result, balanced accuracy is omitted here.

5.1.2 Performance on recall and false alarm

Although the previous overall F1-score and MCC have shown that combining LOF with Triplet Loss achieves the best results, there are other perspectives that reflect the intermediate process, such as the confusion matrix. However, the confusion matrix is more complex than recall or false alarm. As shown in Fig. 9, inputting various self-supervised representation learning and original data respectively into four unary classifiers, the recall comparison results of five faults are obtained. The recall represents the number of real positive samples (fault data) that are detected, and the higher recall indicates the improved result. Self-supervised representation learning Triplet Loss has the highest recall in the five faults. It shows that the features extracted by Triplet Loss



Fig. 8 Matthews correlation coefficient (MCC) of four unary classifiers with self-supervised representation learning and original data (horizontal tick label denotation: 1 – Robust covariance; 2 – One-Class SVM; 3 – Isolation Forest; 4 – LOF).
 (a) Original data; (b) TNC; (c) CPC; (d) Triplet.

are more obvious than other representation learning. The clustering results slightly reflect the evaluation of classification results. the advantages As and disadvantages of these three kinds of self-supervised representation learning can be seen from the cluster visualization, the performances of the extracted features are significantly different for downstream tasks. Feature extractors TNC and CPC have significantly lower recall rates for noisy and periodic faults. Isolation Forest and LOF with TNC and Triplet Loss perform well, and CPC looks good with isolation forest. It shows that even though MCC has a great relationship with recall, the recall rate alone cannot evaluate the fault detection problem very well.

As shown in Fig. 10, various self-supervised representation learning and original data are respectively inputted into four unary classifiers, and the false alarm comparison results of five types of faults are obtained. The false alarm represents the number of real negative samples that are not detected, and the lower false alarm rate indicates the improved result. Recall and false alarm are important evaluation criteria for fault detection. Similar to the recall, the LOF with both TNC and Triplet Loss performs well. Figure 10 also shows that even though MCC is strongly related to the false alarm, the false alarm alone is not a good evaluation for the fault detection problem. In addition, in Figs. 9(d) and 10(d), the trade-off between recall and false alarm is well reflected. Although the false alarm of robust covariance in Fig. 10(d) is smaller than LOF, the recall in Fig. 9(d) is obviously inferior to LOF. Fault detection needs to ensure high recall then select a small false alarm.

To further analyze the performance of TNC and Triplet Loss, the comparison of the confusion matrix for noise fault is shown in Figs. 11 and 12, and the comparison of the confusion matrix for drift fault is shown in Figs. 13 and 14. However, there are two ways to regularize the confusion matrix along the row or the column, and non-regularization is more intuitive. For brevity, only parts of the confusion matrix results are listed. When using Triplet Loss to detect noise faults, the FN of four unary classifiers is all 0. When cooperating with the LOF, the FP of Triplet Loss is smaller than TNC. The same is true for progressive faults like drift in terms of FN and FP.



Fig. 9 Recall of four unary classifiers with self-supervised representation learning and original data (horizontal tick label denotation: 1 – Robust covariance; 2 – One-Class SVM; 3 – Isolation Forest; 4 – LOF).





Fig. 10 False alarm of diverse coupling methods (horizontal tick label: 1 – Robust covariance; 2 – One-Class SVM; 3 – Isolation Forest; 4 – LOF).

(a) Original data; (b) TNC; (c) CPC; (d) Triplet.



Fig. 11 Confusion matrix of noise fault using TNC (-1 denoting abnormal while 1 denoting normal). (a) Robust covariance; (b) One-Class SVM; (c) Isolation Forest; (d) Local outlier factor.



Fig. 12 Confusion matrix of noise fault using Triplet Loss (-1 denoting abnormal, while 1 denoting normal).
(a) Robust covariance; (b) One-Class SVM; (c) Isolation Forest; (d) Local outlier factor.



Fig. 13 Confusion matrix of drift fault using TNC (-1 denoting abnormal while 1 denoting normal). (a) Robust covariance; (b) One-Class SVM; (c) Isolation Forest; (d) Local outlier factor.

5.2 Clustering of extracted features

The different working conditions are denoted by numbers as shown in Fig. 15. The proportions of different working conditions are different in the training set and the test set. The heat maps of corresponding encoding features under different working conditions are also shown in Fig. 15. The extracted features closely correlate and change with different working conditions.

Clustering results of representation learning are also visualized using t-SNE [53]. T-SNE is a nonlinear

dimensionality reduction technique suitable for visualizing high-dimensional data in two-dimensional spaces. The comparison between the clustering visualization of original data and those of features extracted by Triplet Loss, CPC, and TNC is shown in Fig. 16. State denotations under various working conditions are shown in Fig. 15. State 10 denotes all kinds of faults in Fig. 5. After the original data pass through the feature extractor, the densities of clusters change from scattered to dense, and the boundary shapes of clusters change from irregular to more regular. Triplet Loss tends to represent



Fig. 14 Confusion matrix of drift fault using Triplet Loss (-1 denoting abnormal while 1 denoting normal).(a) Robust covariance; (b) One-Class SVM; (c) Isolation Forest; (d) Local outlier factor.



Fig. 15 Denotation of different working states and corresponding encoded features of all sensor data. (a) Sampled sensor data; (b) corresponding encoded features.

fault data in closer feature space than the other two.

The *k*-means is for clustering features learned by selfsupervised representation models, and the Silhouette Score on the test set is shown in Fig. 17. The difference between CPC and TNC models is not very significant. But TNC has the longest training time because the hypothesis test is performed in each batch, and the remaining two training processes are faster. Triplet Loss performs better than the other two, as do the fault detection results.

5.3 Computing resources

The operating environment consists of an NVIDIA TITAN Xp GPU and an Intel(R) Xeon(R) CPU. The algorithm is designed using Python and PyTorch frameworks. The software platform of the algorithm is Visual Studio Code. When the original data are input, the computation time and CPU memory required by different classifiers during testing are shown in Table 7. The test set contains 20 variables and 9000 sampling points, and the fault data rate is 3%. When using a feature extractor, the main computational time is consumed in the feature extraction process. All after encoded computation time of unary classifiers are significantly less than inputting original data time. The encoders in different feature extractors are set to the same. Therefore, the required parameters are the same. Due to the differences in the calculation process, the consumption of GPU memory is slightly different, and so is the calculation time, as shown in Table 8. It can be seen that the unsupervised learning model designed in this work is very lightweight and fast.

6 Conclusions

With the tricky problem of a large amount of normal sensing data lacking fault information in practical fault detection tasks, this work proposes to adopt a selfsupervised feature extractor to extract features from the original data of multi-sensors. It then treats a large amount of normal data as prior knowledge to train the feature extractor. It detects anomalies in the extracted features domain based on a simple fast classifier.



Fig. 16 Clustering visualization of original signals and extracted features using t-SNE.(a) Original signals; (b) Triplet Loss; (c) contrastive predictive coding (CPC); (d) temporal neighborhood coding (TNC).



Fig. 17 Influence of self-supervised clustering evaluation metric with the change of the number of categories under Silhouette Score[↑].

Comparing three self-supervised representation learning feature extraction methods and not using feature extraction, it presents the results of four unary classifiers.

 Table 7
 Comparison of unary classifier computation time

1	2	1	
Unary classifier	Original time/s	Original CPU memory/GB	After encoded time/s
Robust covariance	2.99	0.31	0.0550
One-Class SVM	0.53	0.32	0.0010
Isolation Forest	0.58	0.32	0.2100
LOF	2.67	0.35	0.0036

Under the background of industrial intelligence, it investigates the proposed method on a real-world gas turbine dataset including multi-temperature sensors. It imposes five kinds of typical faults with different distributions only on the testing data, abrupt faults, such as short, step, complex faults, such as drift, noise, and periodic, and compares the computational resources required by various feature extractors. It is found that self-supervised representation learning can quickly obtain

 Table 8
 Comparison of feature extractor computation time

Feature	CPC	TNC	Triplet Loss
Parameters	75210	75210	75210
GPU memory/MB	2.22	2.22	2.22
Time/s	0.025	0.021	0.025

features from original data, and use the extracted features to obtain better one-class classification results in the face of complex faults. The reason is also shown through the cluster visualization. The Triplet Loss coupling with LOF can significantly achieve an F1-score of 0.8 for drift faults that are generally considered the most difficult to detect. This coupling method also has a recall rate of 100% and a false alarm rate of less than 1%.

7 Future work

Since this work only involves fault detection, it can be applied to compound fault detection of various fault superpositions, including sensors on renewable energy systems. It can also handle various unknown faults of multi-sensors. But additional step is needed to locate the fault sensor space position at the time of fault arisen. However, the scope of this work is very broad, and it can even be used in online characterization and detection of false data injection attacks.

Acknowledgements This work was supported by the National Science and Technology Major Project of China (Grant No. 2017-V-0011-0063).

Competing interests The authors declare that they have no competing interests.

References

- Tahan M, Tsoutsanis E, Muhammad M, et al. Performance-based health monitoring, diagnostics and prognostics for conditionbased maintenance of gas turbines: A review. Applied Energy, 2017, 198: 122–144
- Jufri F H, Widiputra V, Jung J. State-of-the-art review on power grid resilience to extreme weather events: Definitions, frameworks, quantitative assessment methodologies, and enhancement strategies. Applied Energy, 2019, 239: 1049–1065
- Fink O, Wang Q, Svensén M, et al. Potential, challenges and future directions for deep learning in prognostics and health management applications. Engineering Applications of Artificial Intelligence, 2020, 92: 103678
- Yan C, Chen J, Liu H, et al. Health management for PEM fuel cells based on an active fault tolerant control strategy. IEEE Transactions on Sustainable Energy, 2021, 12(2): 1311–1320
- Jain T, Yamé J J. Fault-tolerant economic model predictive control for wind turbines. IEEE Transactions on Sustainable Energy, 2019, 10(4): 1696–1704
- 6. Zhang D, Ye Z, Dong X. Co-design of fault detection and

consensus control protocol for multi-agent systems under hidden DoS attack. IEEE Transactions on Circuits and Systems. I, Regular Papers, 2021, 68(5): 2158–2170

- Feng L, Zhao C. Fault description based attribute transfer for zero-sample industrial fault diagnosis. IEEE Transactions on Industrial Informatics, 2021, 17(3): 1852–1862
- Gao D W, Wang Q, Zhang F, et al. Application of AI techniques in monitoring and operation of power systems. Frontiers in Energy, 2019, 13(1): 71–85
- Michau G, Fink O. Unsupervised transfer learning for anomaly detection: Application to complementary operating condition transfer. Knowledge-Based Systems, 2021, 216: 106816
- Chen Y, Zuo M J. A sparse multivariate time series model-based fault detection method for gearboxes under variable speed condition. Mechanical Systems and Signal Processing, 2022, 167: 108539
- Gallo M, Costabile C, Sorrentino M, et al. Development and application of a comprehensive model-based methodology for fault mitigation of fuel cell powered systems. Applied Energy, 2020, 279: 115698
- Singla P, Duhan M, Saroha S. A comprehensive review and analysis of solar forecasting techniques. Frontiers in Energy, 2022, 16(2): 187–223
- Fu F, Wang D, Ding S X, et al. Fault identifiability analysis of linear discrete time-varying systems. IEEE Transactions on Circuits and Systems. I, Regular Papers, 2019, 66(6): 2371–2381
- Li Y, Zhang M, Chen C. A deep-learning intelligent system incorporating data augmentation for short-term voltage stability assessment of power systems. Applied Energy, 2022, 308: 118347
- Waqar Akram M, Li G, Jin Y, et al. Failures of photovoltaic modules and their detection: A review. Applied Energy, 2022, 313: 118822
- Ajagekar A, You F. Quantum computing based hybrid deep learning for fault diagnosis in electrical power systems. Applied Energy, 2021, 303: 117628
- Yang C C, Soh C S, Yap V V. A systematic approach in load disaggregation utilizing a multi-stage classification algorithm for consumer electrical appliances classification. Frontiers in Energy, 2019, 13(2): 386–398
- Sun Z, Han Y, Wang Z, et al. Detection of voltage fault in the battery system of electric vehicles using statistical analysis. Applied Energy, 2021, 307: 118172
- 19. Dey M, Rana S P, Simmons C V, et al. Solar farm voltage anomaly detection using high-resolution μ PMU data-driven unsupervised machine learning. Applied Energy, 2021, 303: 117656
- Zhao Y, Li D, Lu T, et al. Collaborative fault detection for largescale photovoltaic systems. IEEE Transactions on Sustainable Energy, 2020, 11(4): 2745–2754
- Wei L, Qian Z, Zareipour H. Wind turbine pitch system condition monitoring and fault detection based on optimized relevance vector machine regression. IEEE Transactions on Sustainable Energy, 2020, 11(4): 2326–2336
- 22. Zhuo Y, Ge Z. Auxiliary information-guided industrial data augmentation for any-shot fault learning and diagnosis. IEEE

Transactions on Industrial Informatics, 2021, 17(11): 7535-7545

- Zhao Y, Li T, Zhang X, et al. Artificial intelligence-based fault detection and diagnosis methods for building energy systems: Advantages, challenges and the future. Renewable & Sustainable Energy Reviews, 2019, 109: 85–101
- Li B, Delpha C, Diallo D, et al. Application of artificial neural networks to photovoltaic fault detection and diagnosis: A review. Renewable & Sustainable Energy Reviews, 2021, 138: 110512
- 25. Lu X, Lin P, Cheng S, et al. Fault diagnosis model for photovoltaic array using a dual-channels convolutional neural network with a feature selection structure. Energy Conversion and Management, 2021, 248: 114777
- Zuo B, Zhang Z, Cheng J, et al. Data-driven flooding fault diagnosis method for proton-exchange membrane fuel cells using deep learning technologies. Energy Conversion and Management, 2022, 251: 115004
- Nandy J, Hsu W, Lee M L. Towards maximizing the representation gap between in-domain & out-of-distribution examples. In: 34th Conference on Neural Information Processing Systems, 2020
- Nguyen M N, Li X L, Ng S K. Positive unlabeled learning for time series classification. In: Proceedings of the 22nd International Joint Conference on Artificial Intelligence—Volume Two, Barcelona, Catalonia, Spain, 2011
- Wang Y, Liu R, Lin D, et al. Coarse-to-fine: Progressive knowledge transfer-based multitask convolutional neural network for intelligent large-scale fault diagnosis. IEEE Transactions on Neural Networks and Learning Systems, 2021, 34(2): 761–774
- Sun S, Wang T, Yang H, et al. Condition monitoring of wind turbine blades based on self-supervised health representation learning: A conducive technique to effective and reliable utilization of wind energy. Applied Energy, 2022, 313: 118882
- Chen J, Xu X, Yan Z, et al. Data-driven distribution network topology identification considering correlated generation power of distributed energy resource. Frontiers in Energy, 2022, 16(1): 121–129
- 32. Zhao X, Yao J, Deng W, et al. Intelligent fault diagnosis of gearbox under variable working conditions with adaptive intraclass and interclass convolutional neural network. IEEE Transactions on Neural Networks and Learning Systems, 2022, online, https://doi.org/10.1109/TNNLS.2021.3135877
- Patnaik B, Mishra M, Bansal R C, et al. MODWT-XGBoost based smart energy solution for fault detection and classification in a smart microgrid. Applied Energy, 2021, 285: 116457
- 34. Shi H, Li Y, Bai X, et al. A two-stage sound-vibration signal fusion method for weak fault detection in rolling bearing systems. Mechanical Systems and Signal Processing, 2022, 172: 109012
- Liang J, Zhang K, Al-Durra A, et al. A novel fault diagnostic method in power converters for wind power generation system. Applied Energy, 2020, 266: 114851
- Sapountzoglou N, Lago J, De Schutter B, et al. A generalizable and sensor-independent deep learning method for fault detection and location in low-voltage distribution grids. Applied Energy, 2020, 276: 115299
- 37. Van Gompel J, Spina D, Develder C. Satellite based fault

diagnosis of photovoltaic systems using recurrent neural networks. Applied Energy, 2022, 305: 117874

- Bai M, Yang X, Liu J, et al. Convolutional neural network-based deep transfer learning for fault detection of gas turbine combustion chambers. Applied Energy, 2021, 302: 117509
- Feng Y, Chen J, He S, et al. Globally localized multisource domain adaptation for cross-domain fault diagnosis with category shift. IEEE Transactions on Neural Networks and Learning Systems, 2021, 1–15
- Schroff F, Kalenichenko D, Philbin J. FaceNet: A unified embedding for face recognition and clustering. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015
- Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems, 2013
- Breunig M M, Kriegel H P, Ng R T, et al. LOF: Identifying density-based local outliers. SIGMOD Record, 2000, 29(2): 93–104
- Saxena A, Goebel K, Simon D, et al. Damage propagation modeling for aircraft engine run-to-failure simulation. In: 2008 International Conference on Prognostics and Health Management, Denver, CO, USA, 2008
- Sun R, Shi L, Yang X, et al. A coupling diagnosis method of sensors faults in gas turbine control system. Energy, 2020, 205: 117999
- 45. Chen J, Zhang L, Li Y, et al. A review of computing-based automated fault detection and diagnosis of heating, ventilation and air conditioning systems. Renewable & Sustainable Energy Reviews, 2022, 161: 112395
- Zhou H, Zhang S, Peng J, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting. In: The 35th Conference on Artificial Intelligence, 2021
- Yang X, Zhao Q, Wang Y, et al. Fault signal reconstruction for multi-sensors in gas turbine control systems based on prior knowledge from time series representation. Energy, 2023, 262: 124996
- Rousseeuw P J, Driessen K V. A fast algorithm for the minimum covariance determinant estimator. Technometrics, 1999, 41(3): 212–223
- Tax D M J, Duin R P W. Support vector data description. Machine Learning, 2004, 54(1): 45–66
- Liu F T, Ting K M, Zhou Z H. Isolation-based anomaly detection. ACM Transactions on Knowledge Discovery from Data, 2012, 6(1): 3
- van den Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding. arXiv:1807.03748 [cs.LG], 2019
- Tonekaboni S, Eytan D, Goldengerg A. Unsupervised representation learning for time series with temporal neighborhood coding. In: International Conference on Learning Representations, 2021
- van der Maaten L, Hinton G. Visualizing data using t-SNE. Journal of Machine Learning Research, 2008, 9: 2579–2605