Neural Computing and Applications (2023) 35:211–252
https://doi.org/10.1007/s00521-022-08017-3

REVIEW

Challenges and opportunities of deep learning-based process fault detection and diagnosis: a review

Jianbo Yu[1] · Yue Zhang[1]

Received: 22 December 2021 / Accepted: 27 October 2022 / Published online: 18 November 2022

Abstract

Process fault detection and diagnosis (FDD) is a predominant task to ensure product quality and process reliability in modern industrial systems. Those traditional FDD techniques are largely based on diagnostic experience. These methods have met significant challenges with immense expansion of plant scale and large numbers of process variables. Recently, deep learning has become the newest trends in process control. The upsurge of deep neural networks (DNNs) in leaning highly discriminative features from complicated process data has provided practitioners with effective process monitoring tools. This paper is to present a review and full developing route of deep learning-based FDD in complex process industries. Firstly, the nature of traditional data projection-based and machine learning-based FDD methods is discussed in process FDD. Secondly, the characteristics of deep learning and their applications in process FDD are illustrated. Thirdly, these typical deep learning techniques, e.g., transfer learning, generative adversarial network, capsule network, graph neural network, are presented for process FDD. These DNNs will effectively solve these problems of fault detection, fault classification, and fault isolation in process. Finally, the developing route of DNN-based process FDD techniques is highlighted for future work.

Keywords Process fault detection · Process fault diagnosis · Deep learning · Feature learning

1 Introduction

Manufacturing industry refers to the use of specific resources (e.g., materials, equipment, tools) to create products that can be used by people through the manufacturing process [1]. It is an important pillar industry of national economy and social development, with process safety and product quality being two critical issues in modern industries [2]. In general, the manufacturing industry is divided into two categories: discrete industry and process industry. Typical discrete manufacturing industry mainly includes mechanical processing and assembly, electronic and electrical appliances, automobile manufacturing, etc. [1]. Process industry mainly includes chemical industry, metallurgy, pharmacy, etc. [2]. The manufacturing process refers to the entire process from raw material input to finished product production. For the process industry, the actual industrial processes are mostly complex processes that will involve complex physical and chemical reactions, and each subsystem is interconnected. The main data characteristics in the complex process industries are high dimensionality, non-Gaussian distribution, nonlinear relationships, time-varying and multimode behaviors, data autocorrelations, and other data characteristics [2]. With the expansion of production scale and the rapid development of technology, fault detection and diagnosis (FDD) in modern industry becomes more and more complex and important. A small fault in the industrial process may spread through the system, eventually leading to equipment damage or product quality degradation. Effective FDD models can detect process faults in the early stage of production and classify faults accurately for manufacturing process improvement [3]. Process fault refers to the abnormal operation of production, which means that at least one feature or variable appears some unexpected deviations for process industrial system, while fault detection method has been employed to monitor

✉ Jianbo Yu
jbyu@tongji.edu.cn

[1] School of Mechanical Engineering, Tongji University, Shanghai 201804, China

process data and determine whether some faults happened [4]. Fault diagnosis is to determine which kind of fault occurs, specifically to determine the type of fault, fault magnitude, fault location, and time to mitigate potential risks. MSPC has long been recognized as one of the most essential tools for FDD in processes industries [5–8].

Currently, prevalent data-driven techniques include projection-based methods, traditional machine learning methods, and deep learning methods. Over the past few decades, these methods have been intensively investigated for solving various process FDD issues. The most widely used data-driven methods, e.g., principle component analysis (PCA) [9], independent component analysis (ICA) [10, 11], partial least squares (PLS) [12], fisher discriminant analysis (FDA) [13], subspace-aided approach (SAP) [14] as well as their variants, have shown great significance in process FDD because their simplicity. Among these techniques, PCA and ICA are two typical unsupervised methods. PCA is a data projection-based feature extraction method for the process data with Gaussian distribution. ICA is able to extract independent components from process measurements. PLS is a supervised method that can extract the correlation model of the process inputs for prediction. FDA is a typical dimensionality reduction technique and has been widely applied for process FDD. SAP offers a unique way for data-driven design of observer-based process monitoring system without identification of the complete process model.

The nonlinear extensions of the aforementioned models (e.g., kernel PCA (KPCA), kernel PLS (KPLS), kernel FDA (KFDA)) were developed about two decades ago and have achieved wide applications in process monitoring systems [15–18]. These methods basically utilize a kernel function to map the original data into a higher-dimensional space in which they vary linearly. The nonlinear structure of input data space is more likely to be linear after high-dimensional nonlinear mapping [16]. Then, the latent variables can be extracted from the higher-dimensional space. Although kernel-based methods are attractive in nonlinear nonstationary process monitoring, there are still a lot of critical issues. For instance, conventional KPCA only considers normal operation data for statistical modeling and ignores prior fault data in the historical database. A wealth of surveys given by [19–22] provided the readers with more comprehensive information on popular data-driven methods and their applications in process FDD. These methods rely heavily on statistical models to determine the existence of process faults. However, the explosion of data volume and dimension makes these traditional data-driven methods limited in process feature extraction.

Modern industrial processes are usually featured with complexity and distributed. Manifold learning is widely acknowledged as an effective technology to enhance the performance of regular classifiers because it can find the distribution of input data and preserve their local and global manifold information [23]. Typical manifold learning algorithms including isometric feature mapping (IsoMap) [24] local linear embedding (LLE) [25], Laplacian eigenmaps (LE) [26], local tangent space alignment (LTSA) [27], and locality preserving projections (LPP) [28, 29] have achieved remarkable successes in diverse applications. More recently, global/local variance-preserving algorithms are embedded in conventional projection-based methods to [30–32] discover the intrinsic manifold in the data for more precise feature extraction. Manifold learning exhibits great popularity in dimensionality reduction and learning geometric distribution of data. It has been intensively studied in different fields. Other manifold learning approaches can refer to [33–36].

The ever-increasing amount of big data produced in modern process systems, coupled with the complexities of correlating process variables, could result in barriers that were not anticipated by the practitioners in the manufacturing process. This eventually results in high-level of uncertainty during process fault diagnosis [37]. The advent of artificial intelligence (AI) accelerates the development of process FDD techniques. AI in a broad sense refers to the realization of human mind thinking through computers (machines), so that machines like human decision-making. Machine learning is a method to implement AI, and deep learning is a subarea of machine learning. The concept of deep learning stems from the study of artificial neural networks (ANNs), and multi-layer perceptron with multiple hidden layers is a deep learning structure. Deep learning generates abstract high-level representations of the given data. The most traditional machine learning models are usually simple in structure and have only a few shallow layers. The features for traditional machine learning methods are usually predetermined and selected manually according to specific scenarios, which bring many difficulties for practitioners. Moreover, these methods mainly focus on various classification tasks.

In this context, it is essential to exploit effective process FDD models that can automatically extract highly representative features from complicated process data. In recent years, deep learning has attracted many attentions in process FDD due to its unparalleled feature learning ability. Different from conventional machine learning-based feature learners, a deep neural network (DNN) usually consists of multiple layers that can hierarchically convert input data into hidden abstractions and use these extracted features for FDD tasks. The most popular DNNs, e.g., autoencoder (AE) [38] and its variant, convolutional neural network (CNN) [39], deep belief network (DBN) [40], recurrent neural network (RNN) [41], residual network (ResNet) [42], have been widely employed as representation learners to extract comprehensive features for different pattern recognition tasks in the fields of computer vision, natural language processing, and speech recognition. These DNNs now have become promising techniques for handling complicated process signals with high nonlinearity and correlations. There is a wealth of literature that investigated the effectiveness of DNN-based models for solving process FDD issues. Their excellent performance in learning high-level features remarkably increases the process FDD accuracy of the classifiers.

Different from the model-based approaches that require prior knowledge of the process, the data-driven methods only need the availability of a large amount of historical process data. Feature extraction can transform these data and present them to the FDD system as prior knowledge. This extraction process can be either qualitative or quantitative. The three main symbolic AI-based methods for extracting qualitative historical information are expert systems, fault tree, and signed diagraph. These machine learning-based quantitative extraction methods can be divided into four categories: unsupervised learning, supervised learning, semi-supervised learning, and reinforcement learning. Supervised learning and unsupervised learning have been widely adopted in process control, while semi-supervised learning methods and reinforcement learning methods have rarely been used in this field. The data-driven-based machine learning methods for process FDD are shown in Fig. 1.

Anupam et al. [43] presented a general framework common to all the process monitoring fault detection (PMFD) and discussed the future challenges of this research. Ge et al. [2] provided a review about data-based process monitoring. Yin et al. [22] reviewed the wide applications of data-driven methods in process monitoring and fault diagnosis, which provides a reference for industrial process monitoring of large-scale industrial processes. Qin et al. [44] provided a review about data-driven FDD methods and their applications in industrial processes from the perspective of multivariate statistical methods and finally summarized the challenges, opportunities, and expansion. From the perspective of machine learning, Ge et al. [45] reviewed the application of data mining and analysis in process industry in recent decades. Nor et al. [46] reviewed a comprehensive literature review on applications of data-driven methods in FDD systems for chemical process systems. Taqvi et al. [47] reviewed the methods of chemical process FDD based on supervised and unsupervised learning technology and presented the challenges in this field. Lei et al. [48] presented a review and roadmap to deep learning-based machinery fault diagnosis and offered a future perspective of intelligent fault diagnosis (IFD). Wang et al. [1] presented a comprehensive survey of commonly used deep learning algorithms and discussed their applications in smart manufacturing. Although numerous studies and reviews have been dedicated to process detection and diagnosis, the coverage about data-driven methods using deep learning is still rather limited. For examples, Refs. [2, 43–47] just reviewed applications of traditional machine learning to process FDD. Refs. [1, 22, 48] mainly focused on the applications on specific application areas using machine learning models. Therefore, it is still a blank to systematically review the development of process FDD from the past to the future. Furthermore, these reviews have not given a detailed process FDD developing route for forecasting future trends, which is very meaningful for readers. In recent years, a large number of new deep learning methods have emerged, which have not been explained in detail in other review papers. Therefore, it needs a new review to summarize the current research progress of process FDD.

This paper provides a systematic review on the development of process (especially complex industrial processes) FDD by using machine learning (especially deep learning) methods and presents a future development on this field. It is a special and completely different topic. The developments of deep network, their applications, and future prospects in process FDD in this paper are quite different from the previous review articles. The discussions on the applications of these deep learning methods (i.e., deep residual shrinkage network (DRSN), transformer, large-scale neural network, edge computing, etc.) in process FDD are presented in this review. The aims of this review are refined as follows: (1) The developments of data-driven process FDD are summarized into three periods from traditional machine learning to deep learning. In the future, these new deep learning techniques, e.g., transfer learning, graph neural network (GNN), deep Gaussian process (DGP), are viewed to promote the further development of process FDD; (2) A developing route of process FDD is presented in this review. The developing route includes potential research trends and can provide direction and valuable guidance for researchers.

The rest of the study is structured as follows. Section 2 presents a comprehensive review on the development of process FDD in the past. Section 3 reviews the application of deep learning, which are considered as the present period in the development of process FDD. Section 4 argues applications of transfer learning to process FDD. Section 5 displays a developing route in deep learning-based process FDD. Conclusions are drawn in Sect. 6.

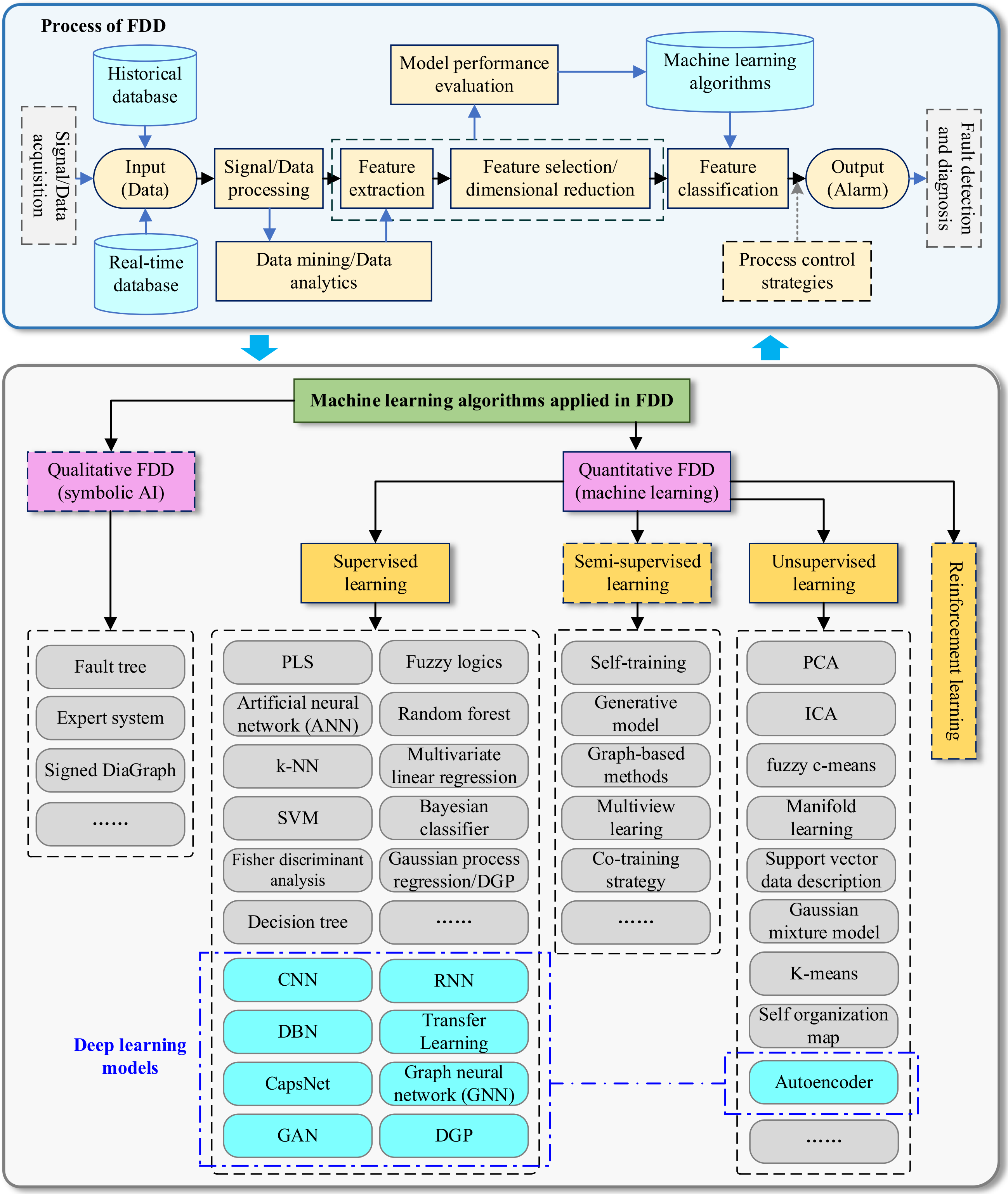


Fig. 1 Applications of machine learning methods in process FDD

2 Traditional methods of process fault detection and diagnosis in the past

This section reviews the traditional machine learning-based methods for process FDD. The process of process FDD mainly includes three steps: data acquisition and preprocessing, feature extraction and selection, and model training and feature classification. Different models have their own characteristics, and they are applicable to different industrial scenarios. In general, there are these representative processes in industrial processes, e.g., continuous processes, batch processes, multimode processes [2]. (1) The continuous process always operates through a continuous way [49]. After the process has been started up, it runs around the best state most of the time and produces constant output. Continuous process is a traditional industrial process, which has been widely existed in chemical, petrochemical and metallurgical industries. (2) Batch process is a discontinuous process with a limited operation duration [50]. Compared with continuous process, the set point of batch process always changes, which means that the process usually operates under different process conditions. Thus, batch process can produce various grades of products in a single batch process. It is inherently nonlinear, time varying, and often has a strong dynamic data behavior. The batch processes exist in the plastic engineering, food engineering and biochemical industries. (3) Multimode process refers to an industrial process with multiple modes, and its operating conditions are always switched from one operating mode to another [51]. There are many multimode methods for process monitoring [8, 52–54]. In these methods, the predefined model matches the corresponding operation mode of the process.

The process FDD methods based on machine learning are mainly based on supervised learning or unsupervised learning [45]. The data in supervised learning method consist of input samples and their corresponding labels. By learning the relationship between samples and labels, the model predicts the labels of unknown samples for process fault classification. Common supervised learning models include PLS, k-nearest neighbor (kNN), ANN, etc. Accordingly, the data in the unsupervised learning method consist of only inputs without any corresponding labels. The goal of this unsupervised learning problem may be to find a group of similar samples in the data (i.e., clustering problem), or determine the distribution of the data in the input space (i.e., density estimation), or project the data from the high-dimensional space to the low-dimensional space (i.e., dimensionality reduction and data visualization). Common unsupervised learning models include PCA, KPCA, ICA, etc.

The effectiveness of these machine learning methods needs to be verified in real industrial processes. Most of the proposed FDD methods are applied to chemical process benchmarks, e.g., Tennessee Eastman process (TEP) [49], Fed-batch fermentation penicillin process (FBFP) [55], continuous stirred tank reactor (CSTR) [56]. Other industrial processes include semiconductor manufacturing process [57], air separation process [58], grinding process [59], boring processes [60], laminar cooling process [61], assembly process [62], and aluminum smelting process [63]. In these modern industrial processes, sensors are often used to monitor and evaluate process variables and obtain a large number of process history data. Industrial process data usually involve the following main characteristics, i.e., high dimensionality, non-Gaussian distribution, nonlinear relationships, time varying, autocorrelations, and multivariate [2]. According to different data characteristics, different FDD models should be reasonably selected. For example, PLS is a linear estimation method, which is not suitable for the monitoring of nonlinear processes. In addition, although GMM can handle the nonlinearity of the process, it may not be able to model all types of non-Gaussian data. According to the characteristics of different methods, the advantages and disadvantages of these traditional machine learning methods and their applications in process FDD are summarized in Table 1.

2.1 Overview

Process monitoring refers to the continuous monitoring of the industrial process to detect abnormal conditions or abnormal behavior. Once the fault is detected in the industrial process, the fault diagnosis is needed to determine the root cause of the fault. Through the fault detection and diagnosis technology to eliminate the fault causes, it is helpful to maintain the smooth operation of the process industry.

Some traditional machine learning methods (e.g., ANN, SVM, PCA) are applied to process FDD. The procedure includes three steps, e.g., data acquisition and preprocessing, feature extraction, feature selection and model selection, training and validation, as shown in Fig. 2. Each step will be detailed in the following subsections.

2.2 Step 1: data acquisition and preprocessing

Data include history data and real-time data in process industry. Data acquisition is the first step of process monitoring. There are several main types of data collected through sensors: vibration signal, speed, pressure, temperature, force, current signal, audio signal, images, etc. In this step, the process data structure is checked, different data features are analyzed, the operation area of the current

Table 1 Summary of applications of the traditional machine learning methods in process FDD

Methods	Advantages	Disadvantages	Applications
Supervised learning methods			
PLS	It is commonly used in Gaussian process monitoring	It is very difficult to be applied in nonlinear process	Industrial processes [64–71]
kNN	(1) It is simple and effective in classification (2) It is more suitable for automatic classification with large sample size (3) It can deal with multiple fault classification problems	(1) When the class labels are unbalanced, the classification accuracy will decrease significantly (2) It is difficult to deal with multi-dimensional data	Semiconductor manufacturing process [57, 72–75]
ANN	(1) It shows strong fault tolerance to noise data (2) It can fully approximate the complex nonlinear relation in the data	(1) A large number of parameters need to be learned on a big dataset (2) It is difficult to observe the learning process	Engineering processes [76], air separation process [58], grinding process [59], boring processes [60], multivariate processes [77–80], laminar cooling process [61], continuous process [81], nonlinear process [82, 83], batch process [84], assembly processes [62]
Support vector machine (SVM)	It can solve high-dimensional problems and has good generalization performance	(1) It is sensitive to missing data and has no general solution to nonlinear problems (2) It is difficult to deal with large-scale training samples	Chemical process [85], TEP [49, 86], FBFP [55], batch process [87]
FDA	It can effectively reduce the dimension of high-dimensional data and is conducive to fault detection	It is difficult to solve complex process control problems	Complex chemical process [88], batch process [17, 89, 90], TEP [91–94], multimode process [95]
Unsupervised learning methods			
PCA	(1) It is used for dimension reduction and feature extraction from the data (2) It can reduce the computational cost of the classifiers	(1) It is difficult to deal with nonlinear data (2) It is very difficult to be applied in non-Gaussian process	Batch processes [96, 97], multivariate processes [98], industrial boiler process [99], multivariable continuous processes [100], TEP [31], multivariate statistics process [101–103], chemical process [104]
KPCA	It is suitable for solving the problem of nonlinear feature extraction from the data	The calculation of KPCA is more complex than that of PCA	Nonlinear multimodal process [105], nonlinear processes [106], chemical process [107]
ICA	(1) It can extract high-order information from the data (2) It can deal with non-Gaussian process	(1) It is difficult to determine the control limit (2) It cannot be used to estimate Gaussian distribution of the data	Multivariate statistical process [108–110], TEP [111–113]
Gaussian mixture model (GMM)	(1) It is easy to estimate the distribution of the data with multimodal characteristics (2) It can handle the nonlinearity of the process (3) It is commonly used in non-Gaussian process monitoring	(1) The model training is complicated (2) It may not be able to estimate very complex distributions of the data	CSTR [56], TEP [56], semiconductor manufacturing processes [114, 115], manufacturing process [32, 116], chemical processes [117]
Hidden Markov model (HMM)	(1) It can fully mine historical data information and can be used for process fault prediction (2) It has the ability to process dynamic data	It requires large-scale training data	Multimode processes [52–54], nonlinear multimodal process [118]
Support vector data description (SVDD)	(1) It can handle both of the linear and nonlinear process data (2) The trained model can be used for process monitoring directly based on a threshold	(1) It is difficult to setup the threshold of SVDD that could trigger many false alarms (2) It is difficult to analyze and interpret the process	Multivariate processes [119], batch process [120], non-Gaussian process [121]

Table 1 (continued)

Methods	Advantages	Disadvantages	Applications
K-means	(1) It is an un-supervised clustering algorithm (2) The calculation cost is very small, and it is easy to use in applications	(1) The number of clusters needs to be given in advance. In many cases, it is very difficult to estimate it (2) It is vulnerable to noise in the data	Batch process [122], aluminum smelting process [63], multimode process [51, 123]
Self-organizing map (SOM)	(1) It is a kind of unsupervised learning method with self-organization and visualization (2) It is very effective to estimate the distribution of the data	(1) The convergence of the network is greatly affected by the model parameters (2) It requires long running time	Manufacturing process [124], industrial gas fractionation process [125], TEP [126–128]

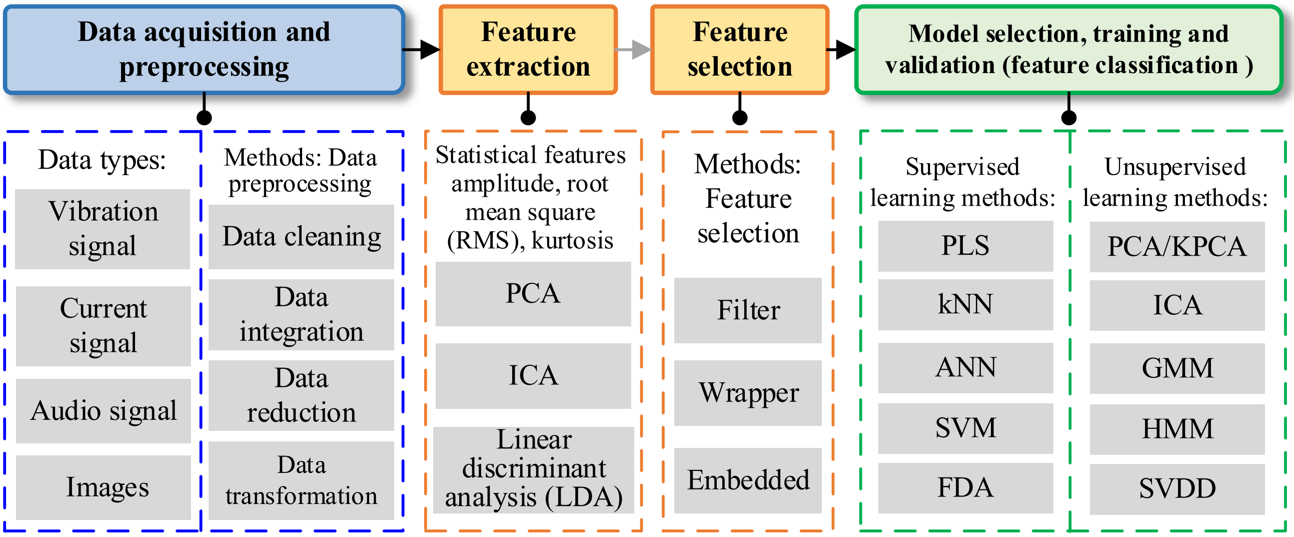

Data acquisition and preprocessing
Feature extraction
Feature selection
Model selection, training and validation (feature classification)
Data types:
Vibration signal
Current signal
Audio signal
Images
Methods: Data preprocessing
Data cleaning
Data integration
Data reduction
Data transformation
Statistical features amplitude, root mean square (RMS), kurtosis
PCA
ICA
Linear discriminant analysis (LDA)
Methods: Feature selection
Filter
Wrapper
Embedded
Supervised learning methods:
PLS
kNN
ANN
SVM
FDA
Unsupervised learning methods:
PCA/KPCA
ICA
GMM
HMM
SVDD

Fig. 2 Detection and diagnosis procedure of process FDD using traditional machine learning methods

process is identified, and the modeling and evaluation datasets are determined. After data collection, data preprocessing is needed to improve the quality of data. The methods of data preprocessing include data cleaning, data integration, data transformation, data reduction, etc. Data cleaning is to process the missing data through sample deletion, missing value estimation, Bayesian inference, and other methods to eliminate the inconsistency of data. Data integration removes outliers and gross errors in datasets through data combination and unified storage. Data transformation is to transform data into a form suitable for data mining by means of smooth aggregation, data generalization, and normalization.

2.3 Step 2: feature extraction/feature selection

Feature extraction includes two steps. Firstly, some common features are extracted from the collected data. Secondly, feature selection methods, such as filter, wrapper, and embedding method, are used to select features that are sensitive to process status from the extracted features.

2.3.1 Feature extraction

Feature extraction generates a subset of new features through the combination of existing features. Its purpose is to obtain the essential features of process data, remove useless noise and realize data visualization. These common feature extraction methods consist of PCA, ICA, linear discriminant analysis (LDA) and manifold learning. In general, LDA is used to reduce the dimension if the data have class labels. Otherwise, PCA is used if the training data have no class labels.

2.3.2 Feature selection

These feature selection methods, e.g., filters, wrappers, and embedded methods, are used to select sensitive features to process state from the extracted features. It is beneficial to remove the redundant information and further improve the process FDDs.

Filter-based methods. The filter directly preprocesses the collected features, which are independent of the training of the classifier [129]. Firstly, feature selection is carried out, and then, the learner is trained, so the process of feature selection has nothing to do with the learner. It is equivalent to filtering the features first and then training the classifier with feature subset.

Wrapper-based methods. The final classifier is directly used as the evaluation function of feature selection, and the feature subset is selected for a specific classifier. Different from the filter-based method, wrapper focuses on the interaction between feature selection and training classifier [129].

Embedded methods. Embedded method combines the process of feature selection with the process of classifier learning and selects features in the process of learning. The most common feature selection method is L1 regularization or L2 regularization [129].

2.4 Step 3: model selection, training, and validation

Different models can be selected according to the characteristics of data relationship and process variables. For example, if the data relationship is linear and most process variables are Gaussian distribution, PCA or PLS can be selected for process FDD. If there are process variables that are non-Gaussian, ICA and other non-Gaussian modeling methods can be selected. If the relationship between different process variables is nonlinear, the nonlinear modeling methods, e.g., ANN and SVM, can be selected for classification.

The next step is to train the model and evaluate its effectiveness. Based on the training model of labeled samples, input unlabeled samples can achieve the purpose of feature classification. Several typical process FDD methods using traditional machine learning are briefly introduced in the following section.

2.4.1 Supervised learning methods

The data in supervised learning methods must be classified and labeled with tags that indicate the system conditions, such as health, fault, and fault type. In supervised learning-based process FDD, the labeled data are used to train the machine learning model, and the trained model can classify the unlabeled data [22].

2.4.1.1 PLS-based approaches PLS is a basic multivariate statistical method and extensively used for process FDD. It establishes a linear regression model by simultaneously projecting the predicted variables and observable variables into the latent variable space. PLS needs to use label information in modeling procedure, which is a commonly used in supervised learning. Collect the data under normal operation to generate an input matrix $X = [x_1, x_2, \ldots, x_n] \in R^{n\times m}$ and an output matrix $Y = [y_1, y_2 \ldots, y_n] \in R^{n\times p}$ with p process variables. PLS projects X and Y to a low-dimensional space defined by $\boldsymbol{l}$ latent variables as follows:

$$\begin{cases} X = \sum_{i=1}^{l} t_i p_i^T + E = TP^T + E \\ Y = \sum_{i=1}^{l} t_i q_i^T + F = TQ^T + F \end{cases} \tag{1}$$

where $T = [t_1, \ldots, t_l]$ denote the latent score vectors, $P = [p_1, \ldots, p_l]$ and $Q = [q_1, \ldots, q_l]$ denote the loadings for X and Y, respectively, E and F denote the residuals of PLS corresponding to X and Y, respectively, and l is generally determined cross-validation. The details of the PLS algorithm can refer to [130, 131].

The latent vectors t_i are computed sequentially from the data such that the covariance between the deflated input data, $X_i = X_{i-1} - t_{i-1} p_{i-1}^T; X_1 = X$, and output data Y for each factor can be maximized, and w_i is the weight vectors to compute the scores $t_i = X_i w_i$. The scores can be denoted as:

$$T = XR \tag{2}$$

where $R = W(P^T W)^{-1}$ with the following relation [132]:

$$P^T R = R^T P = I_l \tag{3}$$

PLS uses an oblique projection toward the input data space, such that the model estimate and residual on the new sample x can be obtained:

$$\hat{x} = PR^T x \tag{4}$$

$$\hat{y} = QR^T x \tag{5}$$

$$\tilde{x} = (I - PR^T)x \tag{6}$$

where $\hat{x}$ and $\tilde{x}$ denote the oblique projections of x [44].

Although the FDD technology based on PLS has been widely used in industrial process, there are still two problems: (1) PLS needs to select more principal components to describe the process related changes, which makes the interpretation of the model very difficult; (2) PLS does not extract principal components according to the order of variance in the process variable matrix. It is not suitable to monitor the residual subspace with Q statistic. In order to solve the above problems, the following extended models are proposed based on the basic PLS model.

Zhou et al. [133] proposed a PLS-based scheme, total projection to latent structure (TPLS), to deal with skew decomposition in standard PLS. However, TPLS does not clearly explain the reason that the principal component space of PLS contains the change independent of the fault in the practical application, and the principal component space does not need to be decomposed into four subspaces, which can be completely decomposed into the subspace related to the process fault and the subspace related to the input. Qin et al. [134] proposed concurrent projection of latent structure (CPLS), which simplified the structure of TPLS. Based on the consistent projection of input and output data spaces, CPLS provides complete monitoring of process faults occurring in predictable output subspace and unpredictable output residual subspace. In order to solve the dynamic problem of industrial process, dynamic principal component analysis (DPCA) and dynamic PLS (DPLS) method are proposed [64–68]. Similar to TPLS, CPLS does not change the prediction ability of PLS for fault variables, but further decomposes the measurement variable space according to the fault variable space. Ding et al. [69] and Yin et al. [69, 70] constructed the modified PLS (MPLS), which cleverly used SVD to decompose the process variable space into two subspaces, but required that the principal component subspace should not contain components orthogonal to the fault variables. MPLS avoids the complex iterative calculation process of CPLS in practical application, which is conducive to the prediction of process faults, but the residual space has no contribution to its prediction. In order to ensure the completeness of spatial decomposition, Peng et al. [71] proposed an efficient latent structure projection (EPLS) based on MPLS. Das et al. [135] combined cluster analysis with multiple multi-block PLS (MBPLS) for process monitoring.

2.4.1.2 kNN-based approaches kNN is one of the simplest machine learning algorithms, which are often used to complete classification tasks [136]. In this method, a distance metric is used to search for k samples near a given unlabeled sample. As shown in Fig. 3, in the decision-making of classification, kNN only determines the category of the sample to be classified according to the category of the nearest one or several samples.

The kNN-based method is widely used in semiconductor manufacturing process. He et al. [57] proposed a fault detection method based on k-nearest neighbor rule (FD-kNN) for semiconductor manufacturing process. In order to enable FD-kNN to monitor online process, He et al. [72] proposed a principal component-based kNN (PC-kNN) for fault detection. For nonlinear, multimodal, and non-Gaussian batch processes, Guo et al. [73] proposed an improved fault detection method based on KNN. The results show that the proposed method can achieve better fault detection performance compared with MPCA, FD-kNN and PC-kNN. Li et al. [74] proposed a diffusion mapping-based kNN rule (DM-kNN) technology, which can reduce the cost of data storage and improve the performance of fault detection. Zhang et al. [75] proposed a new fault detection model based on weighted distance of kNNs (FD-wkNNs), which is more suitable for multimodal process monitoring than kNN. kNN and its extensions are simple and can be applied in the process FDD effectively. When the number of features increases, however, the amount of calculation for kNNs increases significantly. Thus, it is difficult to effectively solve the problem of sample imbalance.

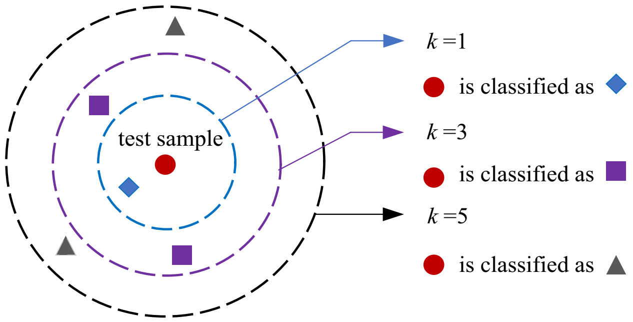


Fig. 3 Illustration of the kNN algorithm

2.4.1.3 ANN-based approaches ANN generally consists of a complex network structure formed by the interconnection of a large number of processing units (neurons). It is a kind of abstraction, simplification, and simulation of human brain organization structure and operation mechanism.

Multi-layer perceptron (MLP) trained by back propagation (BP) algorithm is the most successful neural network model. An ANN consists of three components: input layer, hidden layer, and output layer. The signal propagates forward, and the error propagates backward. ANN is based on numbers of simple processors and neurons, as shown in Fig. 4. Given the training dataset $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$, where $x_m \in R^d$ includes d features and $y_m \in R^l$ includes l health states, the output of the hth hidden layer is expressed as:

$$\left(x_i^h\right)_j = f^h\left(\sum_{i=1}^{n_{h-1}} \omega_j^h \bullet x_j^{h-1} + b_j^h\right), j = 1, 2, \ldots, n_h, h = 1, 2, \ldots, H \tag{7}$$

where $\left(x_i^h\right)_j$ is the output of the jth neuron in the hth hidden layer, and $x_i^0 = x_i$, n_h is the number of neurons in the hth hidden layer, f^h is called the activation function of the hth hidden layer, often chosen to be the sigmoid function, n_{h-1} is the number of neurons in the $(h-1)$th hidden layer, ω_j^h is the weights between the neurons in the previous layer and the jth neuron in the hth hidden layer, and b_j^h is the bias

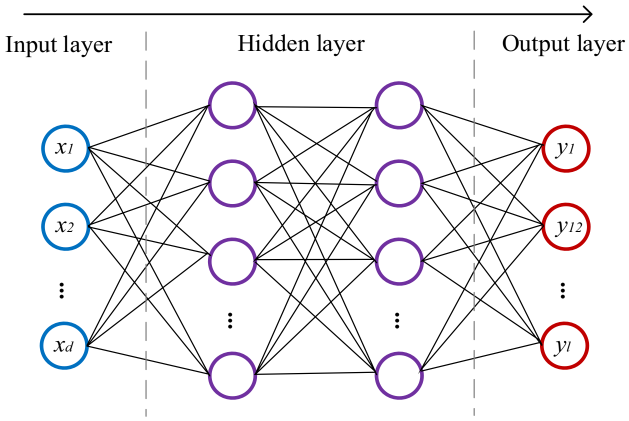


Fig. 4 An artificial neural network with two hidden layers

of the hth hidden layer. The output of BPNN can be obtained by:

$$y_k = f^{out}\left(\sum_{i=1}^{n_H} \omega_j^{out} \bullet x_j^H + b_j^{out}\right), k = 1, 2, \ldots, l \qquad (8)$$

where y_k is the predicted output of the kth neuron in the output layer, f^{out} is the activation function of the output layer, and ω_j^{out} and b_j^{out} are the weights and bias of the output layer, respectively.

Numerous research activities have shown that ANN has powerful pattern classification and recognition. As a result, ANN is one of the classifiers commonly used in intelligent fault diagnosis [137, 138]. MLP is an ANN made of units arranged in layers with only forward connections to units in subsequent layers [139]. This model has been successfully applied to fault detection and identification of turning processes [76], air separation process [58], grinding process [59], boring processes [60], multivariate attribute process [78].

The radial basis function (RBF) network has a feedforward structure, consisting of only one hidden layer with no weighted connections and fully interconnected to the output layer. Compared with MLP, RBF is faster to train [140]. RBF has been used in several applications [61, 77, 81]. Probabilistic neural network (PNN) [141] is similar with MLP in structure, but due to the smaller number of connections, PNN is normally easier to train than MLP. PNN and its variants have been used at fault diagnosis of nonlinear process [82, 83] and batch process [84]. In addition, Yu et al. [78, 79] developed a selective neural network ensemble method (DPSOEN, discrete particle swarm optimization) to accurately locate the source of runaway signals in multivariate manufacturing processes. Du et al. [62] explored a selective neural network ensemble algorithm for detecting and isolating process fault in assembly processes.

2.4.1.4 SVM-based approaches

SVM is a computational learning method for small samples classification [142]. A hyperplane $f(x) = 0$ is expected to be found to separate the given datasets into two classes, and the hyperplane is defined as:

$$f(x) = \omega^T x + b = \sum_{i=1}^{m} \omega_i x_i + b = 0 \qquad (9)$$

where ω is a m-dimensional vector and b is a scalar. As shown in Fig. 5, SVM constructs two parallel hyperplanes as the interval boundary, that is, the maximum margin, to distinguish the classification of samples.

Due to the data classification ability of SVM, it has been used for process FDD in the past years. Peng et al. [85] proposed a novel process fault detection and classification approach via non-negative matrix factorization with sparseness constraints (NMFSC) and structural SVMs. Yin et al. [86] reviewed the research and development of FDD based on SVM in complicated industrial processes. Onel et al. [49] proposed a new feature selection algorithm based on nonlinear (kernel correlation) SVM and applied it to continuous process monitoring and fault detection. Yang et al. [55] combined PCA and recursive feature elimination (RFE) with SVM for process FDD. Onel et al. [87] presented a novel data-driven framework for process monitoring in batch processes, where a feature selection algorithm based on nonlinear SVM is used to exploit high-dimensional process data.

2.4.1.5 FDA-based approaches

FDA is an optimal dimension reduction method that has been intensively studied in process fault diagnosis. It can separate process faults from normal samples by searching for the directions with the maximized discrimination between faults and

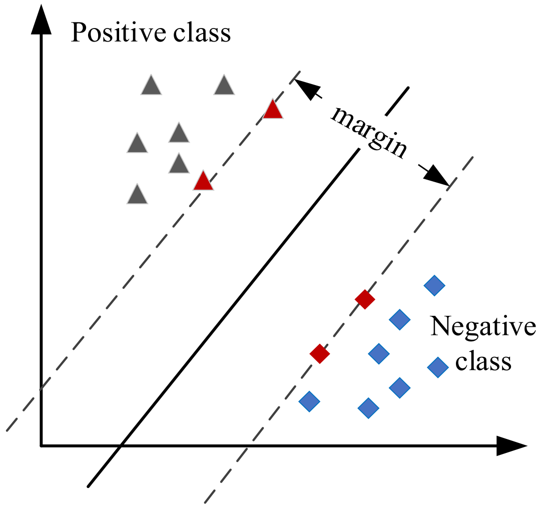


Fig. 5 Classification by the linear SVM

normal data instead of the largest variability only within normal samples [88]. The objective of FDA is to seek a mapping to maximize the scatter between the classes and minimize the scatter within each class.

In the past few years, FDA is mainly used in process monitoring, fault classification and fault diagnosis in process industry. Xi et al. [17] proposed a novel nonlinear biological batch process monitoring and fault identification approach based on kernel FDA. In order to deal with highly correlated, complex, and noisy high-dimensional databases effectively, Nor et al. [91] proposed a process FDD method based on wavelet analysis, kernel FDA (KFDA), and SVM. Nor et al. [94] proposed a multi-scale KFDA adaptive neuro-fuzzy inference system (KFDA-ANFIS) framework. Yu [88] proposed a novel localized FDA (LFDA)-based process monitoring approach to monitor the processes containing multiple types of steady-state or dynamic faults. Yu [89] proposed a multiway kernel localized FDA (MKLFDA) for batch bioprocess monitoring. Zhao et al. [90] proposed a FDD method based on extreme learning machine (ELM) and multiway FDA (MFDA). Ren et al. [95] integrated local consistency Gaussian mixture model (DLCGMM) with modified local FDA (MLFDA) for multimode process monitoring. Tang et al. [92] proposed a novel data-driven process monitoring method named Fisher discriminant global–local preserving projection (FDGLPP) and applied it to diagnosis fault in industrial process. Yang et al. [93] presented a class-incremental scheme of FDA to improve the performance of process fault diagnosis.

2.4.2 Unsupervised learning methods

The difference between supervised learning and unsupervised learning is whether the training data of the model need labels. Unsupervised learning models can be constructed on training data without class labels [22].

2.4.2.1 PCA-based approaches PCA is a typical statistical approach that has been widely used in process monitoring field. PCA does not use label information in the modeling procedure and is a unsupervised learning method for dimension reduction [45]. It is capable of projecting the high-dimension data onto a lower-dimension space that contains the most variance of the original data and accounts for correlation among variables.

Denote the dataset $X = [x_1, x_2, \ldots, x_n]$, PCA seeks a mapping axis a, such that the mean square of the Euclidean distance between all pairs of the projected samples $Y = [y_1, y_2 \ldots, y_n]$, i.e.,$y_i = a^T x_i (i = 1, \ldots, n)$, is maximized as follows:

$$J_g(a) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \overline{y})^2 = \frac{1}{n}\sum_{i=1}^{n} a^T(x_i - \overline{x})(x_i - \overline{x})^T a = a^T C a \tag{10}$$

where $\overline{y} = \frac{1}{n}\sum y_i$, $\overline{x} = \frac{1}{n}\sum x_i$ and $C = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})(x_i - \overline{x})^T$ is the covariance matrix. The eigenvectors of the data covariance matrix associated with the largest eigenvalues are the basis functions of PCA. The output of principal vectors $a = a_1, \ldots a_l$ is an orthonormal set of vectors representing the eigenvectors of the sample covariance matrix associated with $l < m$ (l is the selected number of principal components (PCs) and m is the number of all PCs).

PCA and PLS are two typical multivariate statistical approaches in FDD [143]. Some literatures reviewed the applications of PCA and PLS in process analysis and control, FDD [96–98, 144]. The first attempt of applying PCA in FDD can be found in [99]. This method was extended to batch processes by using multiway PCA [97]. To deal with nonlinearity, a nonlinear PCA [97] and a neural net PLS [145] were proposed. In [100], an integral statistical methodology combining PCA and discrimination analysis techniques was proposed. Yu [31] presented a local and global principal component analysis (LGPCA), which is a linear dimensionality reduction technique through preserving both of local and global information in the observation data. Chaouch et al. [146] developed an off-line monitoring method based on multilayer neural PCA and nonlinear gain scheduling and applied it to a photovoltaic system.

For high-dimensional-related process variables, the main data information can be retained, and the data information can be greatly compressed through the dimensionality reduction in PCA. After the dimension of process variables is reduced, further data analysis will become easier. In the past few years, PCA has many applications for dimensionality reduction and data visualization [101–103].

2.4.2.2 KPCA-based approaches Kernel methods, e.g., kernel PCA (KPCA), KPLS, are utilized to avoid nonlinear optimization and has been widely applied in nonlinear process monitoring. The basic idea of kernel methods is first projecting the process data into a high-dimensional feature space and then performing linear PCA or PLS in the feature space [105]. Take KPCA as an example, the specific implementation of KPCA is presented as follows:

Denote a dataset $X = [x_1, x_2, \ldots, x_n]$, $x_i = [x_{i1}, x_{i2}, \ldots, x_{ip}]$,$i = 1, 2, \ldots, n$ with p measured process variables. The covariance matrix of the dataset is calculated as follows:

$$\frac{1}{n}\sum_{j=1}^{n} x_j^T x_j \tag{11}$$

where n is the number of samples. Suppose the nonlinear mapping:

$$\Phi_x \rightarrow F \tag{12}$$

Thus, F is generated as $\Phi(x_1),\Phi(x_2),\ldots,\Phi(x_n)$. Denote the kernel function as:

$$K = \Phi(X)^T \Phi(X) = [k(x_i, x_j)]_{n\times n} \tag{13}$$

The kernel function must satisfy the Mercer theorem. In other words, the kernel is the inner product of the feature space. Through the kernel function, the calculation in the feature space can be calculated in the original input space, without knowing the high-dimensional transformation:

$$k(x_i, x_j) = \left\langle \Phi(x_i)^T, \Phi(x_j)\right\rangle = \Phi(x_i)^T, \Phi(x_j) \tag{14}$$

Then, the covariance matrix in the feature space is calculated:

$$C = \frac{1}{n}\sum_{j=1}^{n} \Phi(x_j)^T, \Phi(x_j) \tag{15}$$

Thus, the operation of PCA in the feature space is:

$$\lambda V = CV \tag{16}$$

where λ is the eigenvalue and V is the eigenvector,$V \in F\{0\}$. Since V belongs to the generated space of $\{\Phi(x_1),\Phi(x_2),\ldots,\Phi(x_n)\}$, the following formula can be obtained:

$$\lambda(\Phi(x_k)\cdot V) = \Phi(x_k)\cdot CV(k = 1,2,\ldots,n) \tag{17}$$

There exists a parameter $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$, such that V can be expressed linearly by $\Phi(x_k)(k = 1,2,\ldots,n)$, namely:

$$V = \sum_{i=1}^{n} \alpha_i \Phi(x_i) \tag{18}$$

By combining Eqs. (1) and (2), the following expression is obtained:

$$\lambda \sum_{i=1}^{n} \alpha_i[\Phi(x_k)\Phi(x_i)] = \frac{1}{n}\sum_{i=1}^{n} \alpha_i \left[\Phi(x_k)\sum_{j=1}^{n} \alpha_i[\Phi(x_j)]\right][\Phi(x_j)^T \Phi(x_i)] \tag{19}$$

Then, the kernel function is introduced:

$$n\lambda\alpha = K\alpha \tag{20}$$

Finally, the eigenvalues and eigenvectors of K are obtained, and the principal components are extracted by performing PCA.

KPCA has been used to monitor nonlinear processes, but it is not very suitable for process fault diagnosis. Thus, it often combines other methods to solve the problems of FDD. Li et al. [106] presented a new method of FDD for nonlinear processes based on KPCA and least squares support vector machine (LSSVM). A new detection and diagnosis system combining the decision directed acyclic graph (DDAG) with KPCA is proposed in [147]. Xu et al. [107] used an improved multi-scale KPCA to analyze the multi-scale and nonlinear property of chemical data.

2.4.2.3 ICA-based approaches

ICA is a multivariate statistical approach that can extract statistically independent components (ICs) from the given data. Because it is suitable for process measurement with non-Gaussian distribution, it has been intensively applied for solving process fault detection issues during the past few decades [148].

Denote a given input vector $x(i) = [x_1(i),x_2(i),\ldots,x_m(i)]^T$ with m observed measurements at sample i, which can be described as a linear combinations of k unknown independent components (IC) $s_1, s_2, \ldots, s_k$, then [11]:

$$x(i) = \sum_{j=1}^{k} a_j s_j(i) = As(i) \tag{21}$$

The relationship between the original data and ICs is expressed as:

$$X = AS + E \tag{22}$$

where $X = [x(1),x(2),\ldots,x(n)] \in R^{m\times n}$ denotes the input matrix with n samples, $A = [a_1, a_2, \ldots, a_k] \in R^{m\times k}$ denotes the mixing matrix, $S = [s(1),s(2),\ldots,s(n)] \in R^{k\times n}$ denotes the IC matrix. $E \in R^{m\times n}$ denotes the residual matrix. ICA aims to calculate a separating matrix W to enable the components of the reconstructed data matrix $\hat{S}$ to be as independent of each other as possible. $\hat{S}$ is expressed as follows:

$$\hat{S} = WX \tag{23}$$

There are many types of ICA algorithm. The fixed-point algorithm (also known as Fast ICA) algorithm has fast convergence speed and has achieved remarkable performance. Fast ICA is widely used in signal processing because it is easy to converge, and it has good separation performance. The algorithm can estimate the original signals that are statistically independent and mixed by unknown factors from the observed signals.

ICA is a multivariate statistical tool to extract statistically independent components from observed data, which has drawn considerable attention in FDD. Lee et al. [108] analyzed the defects of the original ICA algorithm and presented a novel multivariate statistical process monitoring (MSPM) method based on modified ICA. Zhang et al.

[109] proposed a modified ICA algorithm based on particle swarm optimization (PSO-ICA) for the purpose of MSPM. Hsu et al. [110] integrated ICA and SVM (ICA-SVM) for monitoring multivariate processes, where ICA was used to extract the hidden information of a non-Gaussian process. In order to solve the problem of complex industrial process monitoring, Zhang et al. [111] combined kernel ICA (KICA) and LSSVM to establish the industrial process monitoring model. Li et al. [112] advised a correlated and weakly correlated process fault detection approach based on variable division and ICA. Wang et al. [113] proposed a totally data-driven ICA model to improve the ability of monitoring non-Gaussian process.

2.4.2.4 GMM-based approaches Gaussian model is to use Gaussian probability density function (normal distribution curve) to accurately quantify things, and decompose a thing into several models based on Gaussian probability density function. Gaussian mixture model (GMM) is a clustering algorithm, which belongs to unsupervised learning. It organizes itself according to the nature of input data with complex distribution (e.g., multimodal, or nonlinear distribution) and can monitor health status of industrial process without prior knowledge of abnormal patterns.

GMM can be used in general process applications, e.g., data clustering analysis, process monitoring, dimension reduction, data visualization. Yu et al. [56] proposed a multi-modal process monitoring method based on finite Gaussian mixture modes and Bayesian inference strategy for complex multi condition industrial processes. In order to solve the problems of the particularity of semiconductor process, the nonlinearity of most batch processes, and the multipeak intermittent trajectory under multiworking conditions, Yu [114] proposed a principal component-based GMM (PCGMM). After that, in order to find meaningful low-dimensional information hidden in high-dimensional observations, Yu [115] proposed a manifold learning feature extraction algorithm based on local and nonlocal preserving projection (LNPP) and then used GMM to process data with nonlinear or multimodal features. Yu [32] proposed local/nonlocal manifold regularization-based GMM (LNGMM) to estimate process data distributions with nonlinear and multimodal characteristics. Yu [116] presented a GMM-based process patterns recognition (PPR) model, which employs a collection of several GMMs trained for process pattern recognition. Jie [117] proposed a nonlinear kernel Gaussian mixture model-based inferential monitoring approach for FDD of chemical processes.

2.4.2.5 HMM-based approaches Hidden Markov model (HMM) is a statistical model, which is used to describe a Markov process with hidden unknown parameters. The difficulty is to determine the hidden parameters of the process from the observable parameters and then use these parameters for further analysis, e.g., pattern recognition. HMM contains a limited number of states, in which each state generates an observation at a certain time point. HMM performs two random processes: one is a random transition from one state to another, and the other is to generate a random output symbol in each state. Thus, these models can only be observed through another set of random processes. The actual state sequence is hidden and cannot be observed directly.

For HMM with N states and M observations, the observation sequence up to time T is denoted as $O \in \{O_1, O_2, \ldots, O_T\}$, where $O_t \in \{v_1, v_2, \ldots, v_M\}$, the corresponding state sequence is denoted as $O_T \in \{q_1, q_2, \ldots, q_T\}$, where $q_t \in \{S_1, S_2, \ldots, S_N\}$. Assume that the underlying state sequence is transferred based on Markov process, the transition probability matrix is expressed as$A = \left\{a_{ij}\right\}, 1 \leq i, j \leq N$, where $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$, and q_t represents the hidden state at time t. For a HMM, these states are observable. On the contrary, the observed value depends on whether it is visible in this state and is represented by a matrix according to the conditional probability distribution. In addition, if the initial state probability distribution is $b_j(k) = P(O_k | q_t = S_j)$, an HMM can be represented by $B = \{b_j(k)\}(1 \leq j \leq N)$. HMM adopts these basic algorithms $\pi_i = P(q_t = S_j)$, namely forward backward algorithm $\lambda = \{A, B, \pi\}$, Baum Welch algorithm, and Viterbi algorithm, which are used λ for model parameter learning and recognition.

In the past years, HMMs have been applied to the process industry. In order to effectively monitor multimode processes, Wang et al. [52] proposed a process monitoring scheme based on orthogonal nonnegative matrix factorization (ONMF) and HMM. Afzal et al. [53] proposed a monitoring method based on HMM for multimodal processes with mode reachability constraints. Lou et al. [54] combined hidden semi-Markov model (HSMM) with PCA (HSMM-PCA) and introduced mode duration probability into HMM. Regarding the complex nonlinear industrial process, Peng et al. [118] combined HMM and KPCA for fault detection of nonlinear multimodal process.

2.4.2.6 SVDD-based approaches Support vector data description (SVDD) is a single-value classification algorithm, which can distinguish target samples from non-target samples, and is usually applied in anomaly detection and fault detection. In this method, there is no restriction that the process data should be assumed to be Gaussian. Thus, SVDD is considered as a promising method for non-Gaussian process monitoring. Cho [149] dealt with the data description and noise filtering issues based on SVDD and applied it for process fault detection. Ge et al. [119] further proposed an SVDD-based reconstruction algorithm for sensor fault identification and isolation of multivariate processes. Based on FDD, Yao et al. [120] extended the single class classification method of SVDD to functional SVDD (FSVDD) and applied it to batch process monitoring. For further improving the competence of ICA, Chen et al. [121] integrated ICA, Durbin–Watson (DW) criterion, and SVDD to monitor non-Gaussian process for detecting faults.

2.4.2.7 *K*-means-based approaches *K*-means is an iterative clustering analysis algorithm, and the calculation steps are as follows: (1) Divide the data into *k* groups; (2) Randomly select *k* objects as the initial clustering center; (3) Calculate the distance between each object and each seed clustering center; (4) Assign each object to the nearest clustering center. *K*-means is a widely used unsupervised machine learning clustering algorithm. The main application of this method in process industry is to divide the process data into different operation modes, different fault types, or different grades of products. For example, Lv et al. [122] employed *K*-means to derive the segmentation rule of variable subspace for batch process monitoring. Majid et al. [63] used *K*-means as a data mining tool to isolate different types of faults in aluminum smelting process. Zhou et al. [123] combined *K*-means and PCA for fault detection and identification in multiples processes. Tong et al. [51] proposed an adaptive multimode process monitoring strategy based on *K*-means.

2.4.2.8 SOM-based approaches Self-organizing map (SOM) is an unsupervised learning algorithm derived from competitive learning. It uses neighborhood function to maintain the topological properties of input space. It is usually represented by low dimensional discretization to train the input space of samples. SOM can reduce dimension and model for highly discrete and nonlinear data.

When SOM is used to monitor the production process, a certain amount of controlled data is collected for the first time to create and train the SOM model and generate the data feature space model of the controlled process. Then, the observation vector is collected online from the manufacturing process, and the best match unit (BMU) is obtained by comparing with the weight vector of all primitives in SOM. The distance between the input vector and BMU is defined as MQE [124] as follows:

$$\mathrm{MQE} = ||D - W_{\mathrm{EMU}}|| \tag{24}$$

where D is the input vector and W_{BMU} is the weight vector of BMU. The size of MQE represents the distance between the input vector and the normal state space. MQE can be used as a process monitoring indicator.

In the past few years, SOM has been widely used in process industry. Yu et al. [124] proposed a process monitoring method based on SOM and developed a novel minimum quantization error (MQE) chart for monitoring process changes. Corona et al. [125] introduced the advantages of SOM in combination with industrial data and its application in a series of process measurements in industrial gas processing plants. In order to clearly show the occurrence of complex chemical process fault, Chen et al. [128, 150] proposed a novel fault diagnosis method that combines SOM with correlative component analysis (CCA). Yu et al. [126] proposed a SOM-based methodology for process FDD with nonlinear and non-Gaussian features. In order to solve the problems of highly dimensional input variables, high correlation among some input variables, overlap among the input variable spaces of different fault classes, and invisible distribution of fault classes, Song et al. [127] integrated canonical variate analysis (CVA) with multiple SOM (multi-SOM) for process monitoring and fault diagnosis.

3 Process fault detection and diagnosis based on deep learning at present

The generalization performance and self-learning performance of traditional machine learning-based process FDD models are insufficient because it is difficult for them to extract effective features from the process data. Thus, it is necessary to extract features from the original data, and then achieve effective process FDD. Process control in industry refers to the automatic control using the real-time data collected by computer and taking the process parameters (e.g., temperature, pressure, flow, liquid level and composition) as the controlled variables. It usually includes control of the whole industrial process. This section reviews the typical deep learning methods and their applications in process FDD in recent years. First, this section presents detection and diagnosis procedure of industrial process by using deep learning. The process FDD procedures are mainly divided into data collection, deep learning model construction, and process monitoring. Second, the application of deep learning in industrial processes is reviewed.

3.1 Overview

With the rapid development of internet technologies and internet of things (IOT), the amount of data collection is unprecedented. Due to the constraints of the number of model parameters and operation speed, the traditional machine learning-based process FDD method is not suitable for industrial big data scenarios. At this time, the deep learning-based process FDD method is proposed in recent years. It uses deep hierarchical structure to automatically represent abstract features from process data and then directly establishes the relationship between learning features and target output, which can effectively solve the data explosion. As shown in Fig. 6, the deep learning-based process FDD procedure mainly includes three steps, namely big data collection, deep learning-based model construction, and process monitoring. Each step is presented in the following subsections.

3.2 Step 1: big data collection and preprocessing

Big data are massive information. Massive information in modern industrial process comes from the data generated by equipment operation. With the development of sensor and computing technology, the amount of data increases almost exponentially. Different sensors are usually employed, e.g., vibration, acoustic emission, temperature, current transformer. There are four main types of big data collected based on these sensors (i.e., vibration signal, current signal, audio signal, and images.) [151]. However, the big data are not fully utilized in knowledge mining or intelligent decision-making. Big data have the following four characteristics, i.e., volume, variety, velocity, and value [152, 153]: (1) Volume. Due to the development of data storage technology, in the long-term operation of the machine, the amount of information collected continues to grow, forming a huge historical database. (2) Variety. Multisource data collected by different types of sensors, a wide range of data sources, determine the diversity of the form of big data. (3) Velocity. Data generation is very fast; mainly through the Internet transmission, high-speed transmission channel can immediately collect data from the machine. (4) Value. There is incomplete information in the collected big data. In addition, a large number of data are mixed with some poor-quality data.

With the big data era coming, machine learning has attracted increasing interest for various applications. Lavasani et al. [154] discussed the latest applications of big data in the chemical industry and stressed the necessity of big data analysis in various fields of process engineering. Shu et al. [155] proposed a new abnormal situation management (ASM) framework to solve the big data problem in the cloud computing environment of a big chemical corporation. Onel et al. [87] presented a new data-driven batch process monitoring model, which uses the nonlinear SVM-based feature selection for big data fault detection and diagnosis. Aiming at plant-wide processes with big data, Yao et al. [156] proposed a distributed parallel modeling and monitoring framework for process fault detection and diagnosis. Jiang et al. [157] proposed a local–global modeling and distributed computing method to achieve efficient fault detection and isolation for nonlinear plant-wide processes.

After obtaining big data, data preprocessing is needed to improve data quality, because the quality of data determines the effectiveness of deep learning model. If the quality of process data cannot be well guaranteed, the deep learning model may produce misleading results. If the data are too noisy, a data processing step is required. For example, Guo et al. [158] used the improved local entropy method to eliminate the multimodal and non-Gaussian characteristics of the raw data and improve the fault detection performance of locality preserving projection (LPP) in multimodal industrial processes. Geng et al. [159]

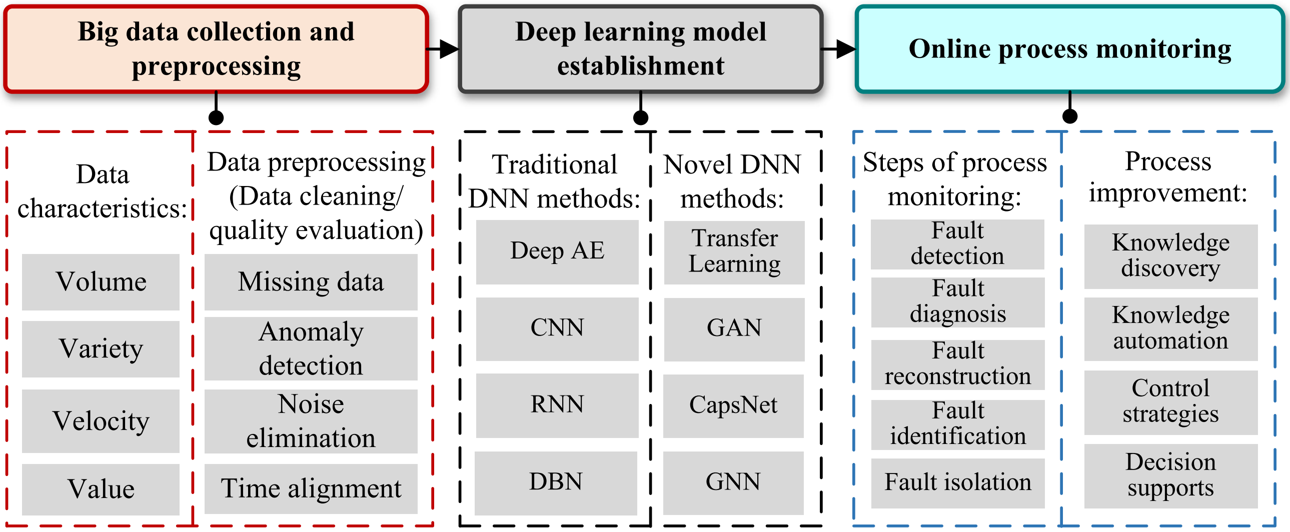


Fig. 6 Fault detection and diagnosis procedure of deep learning-based methods

used the parameters of the variable modulus decomposition for data preprocessing to suppress the data noise of nonlinear complex chemical process and fed the data into a sparse principal component analysis (SPCA). Alrifaey et al. [160] used wavelet packet transform (WPT) to process the collected photovoltaic voltage signal and fed the preprocessed signal into long short-term memory (LSTM) model. Thus, the data cleaning and quality assessment should be carried out before they are fed to those deep learning models [45]. Xu et al. [161] reviewed data cleaning in process industry and introduced different data cleaning methods (i.e., missing data interpolation [162], anomaly detection [163], noise elimination [159], and time alignment [164]). Ding et al. [163] proposed a system for industrial time series cleaning, which can discover knowledge from high-quality time series and be used for industrial production optimization and anomaly detection. To sum up, data cleaning and quality assessment mainly include the following aspects: (1) Appropriate methods should be used to deal with missing data (e.g., sample deletion, missing value estimation, Bayesian inference). (2) The outliers in the data should be detected, and the problem of different sampling rates between process variables should also be considered. (3) The noise from the process itself and measuring equipment should be fully considered. The noise can be removed by filtering methods. (4) The scale difference between process variables needs to be considered. In the face of data imbalance, the performance of the learners can be improved by data scaling and data conversion.

3.3 Step 2: deep learning model construction

The process control method based on deep learning can effectively deal with big data scenarios, and the development of deep learning is shown in Fig. 7. Overall, the development of deep learning can be divided into four stages: embryonic stage, the first climax, the second climax, and the third climax [150].

In embryonic stage, the McCulloch-Pitts (MP) model was proposed in 1943 by Warren McCulloch and Walter Pitts. As the origin of ANN, MP model not only creates a new era of ANN, but also lays the foundation of neural network model. In the late 1950s, based on the research of MP model and Hebb learning rules, a learning algorithm similar to human learning process, perceptron learning, was proposed. In 1958, a neural network composed of two layers of neurons was formally proposed, which is called "perceptron." The proposal of perceptron has attracted a large number of scientists' interest in the research of ANN, which is of milestone significance to the development of neural network.

In the second climax, the famous physicist John Hopfield invented Hopfield neural network in 1982. However, due to the defect that it is easy to fall into local minimum, this algorithm did not cause a great sensation at that time. Until 1986, a back-propagation algorithm for multilayer perceptron, BP algorithm, was proposed. The BP algorithm perfectly solves the nonlinear classification problem, so that ANN has attracted extensive attention again. However, due to the limited hardware level of computers in the 1980s, the problem of "vanishing gradient" will appear when the scale of neural network increases. This has greatly limited the development of BP algorithm, and the development of ANN has entered the bottleneck period again.

In the third climax, the concept of deep learning was formally proposed in 2006. This method describes the solution of the problem of vanishing gradient in detail. The proposal of deep learning immediately aroused great repercussions in the academic circle and then quickly spread to the industry.

In recent years, many novel deep learning methods have been proposed for complex real production and manufacturing systems, e.g., GAN, GNN, DGPs, which solve the problems of unbalanced data types and model migration under different datasets.

3.3.1 AE-based approaches

3.3.1.1 A brief introduction to AE AE is an unsupervised deep learning model, which consists of the encoder and decoder [165]. As depicted in Fig. 8, an AE is a special neural network consists of three layers: input layer, hidden layer, and output layer. The difference is that in the structure of AE, the input and output layers have the same number of neurons. Given the dataset $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$ with m samples, the represented features h_i are defined as:

$$h_i = f_e(\omega^T \bullet x_i + b) \tag{25}$$

The decoding step tries to reconstruct input values from hidden values; the reconstructed sample $\hat{x}_i$ is expressed as follows:

$$\hat{x}_i = f_d(\omega'^T \bullet h_i + b') \tag{26}$$

where f_e,f_d are the activation function of the encoder and decoder network, respectively, and $\{\omega, b\}, \{\omega', b'\}$ represent the training parameters of the encoder and decoder network, respectively. The loss function is defined the following equation with squared error as:

$$J\left(\omega, \omega^{'}, b, b^{'}\right) = \frac{1}{2m}\sum_{i=1}^{m}(\hat{x}_i - x_i)^2 \tag{27}$$

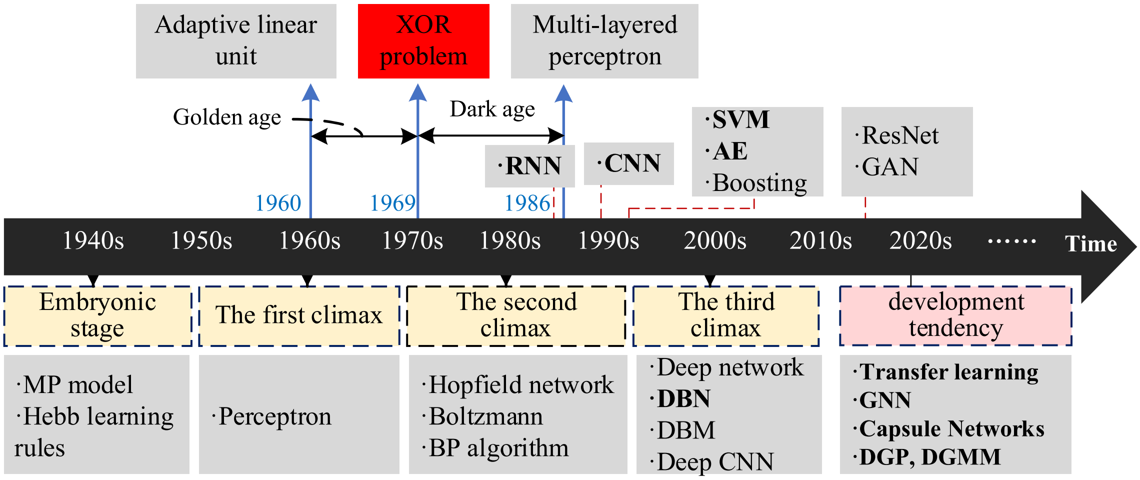


Fig. 7 Development of deep learning

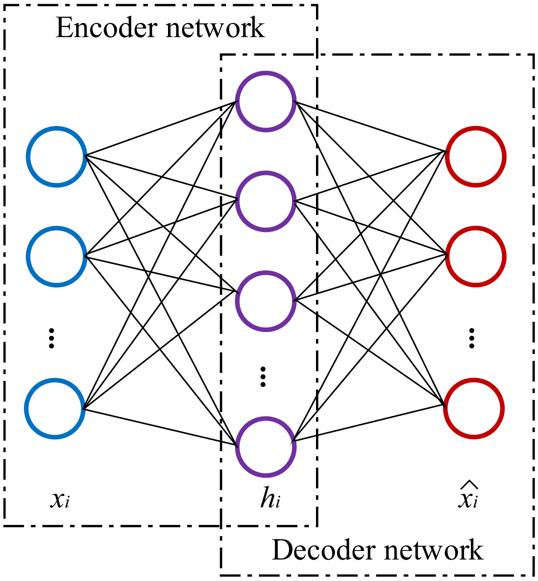


Fig. 8 Structure of AE

3.3.1.2 Application of AE and its variations-based approaches to fault detection and diagnosis AE and its common varieties have been applied to process FDD. Gavneet et al. [166] compared deep stacking networks and sparse stacked AE for process fault detection. Yan et al. [167] separated the feature extraction and model construction by designing teacher and supervise dual stacked auto-encoder (TSSAE) for quality-relevant fault detection in industrial process. Yu et al. [168] presented an effective deep learning method known as stacked denoising AE (SDAE) for process pattern recognition (PPR) in manufacturing processes. Yu et al. [169] concentrated on developing a SDAE model for multivariate process pattern recognition to learn effective discriminative features from the process signals through deep network architecture. Zhang et al. [170] integrated one-dimensional CNN (1-DCNN) and SDAE to extract high level features from complex process signals. Li et al. [171] proposed a distributed ensemble stacked AE (DE-SAE) model based on deep learning technology for monitoring non-linear, large-scale, multi-unit processes. Liu et al. [172] proposed a new DNN, residual attention convolutional AE (RACAE) for complex nonlinear process monitoring. Yu et al. [173] suggested a one-dimension residual convolutional AE (1DRCAE) model, which used unsupervised learning to extract representative features from complex industrial processes. Li et al. [174] proposed a slow feature analysis-aided autoencoder (SFA-AE) for interpretable process monitoring. It enables the learning of deep slow variation patterns from the high-level features extracted by the AE.

Zhang et al. [175] presented a new DNN, manifold regularized stacked AE (MRSAE) for fault detection in complex industrial processes. In this study, the fault detection process based on MRSAE is divided into two stages: offline modeling and online monitoring, as shown in Fig. 9. Their testing results reveal that MRSAE is effective in learning representative features from the complicated data for process fault detection.

The off-line modeling phase includes six steps. Step1 is to collect the training dataset under the normal operation. Step 2 is to normalize samples in the dataset within 0 and 1. Step 3 is to train a MRSAE model in an unsupervised way with greedy layer-wise training algorithm. Step 4 is to generate the feature and residual spaces from the well-trained model. Step 5 is to calculate the monitoring statistic (i.e., T-squared (T^2) and squared prediction error (SPE)), respectively. Step 6 is to setup the thresholds by using the kernel density estimation (KDE) method.

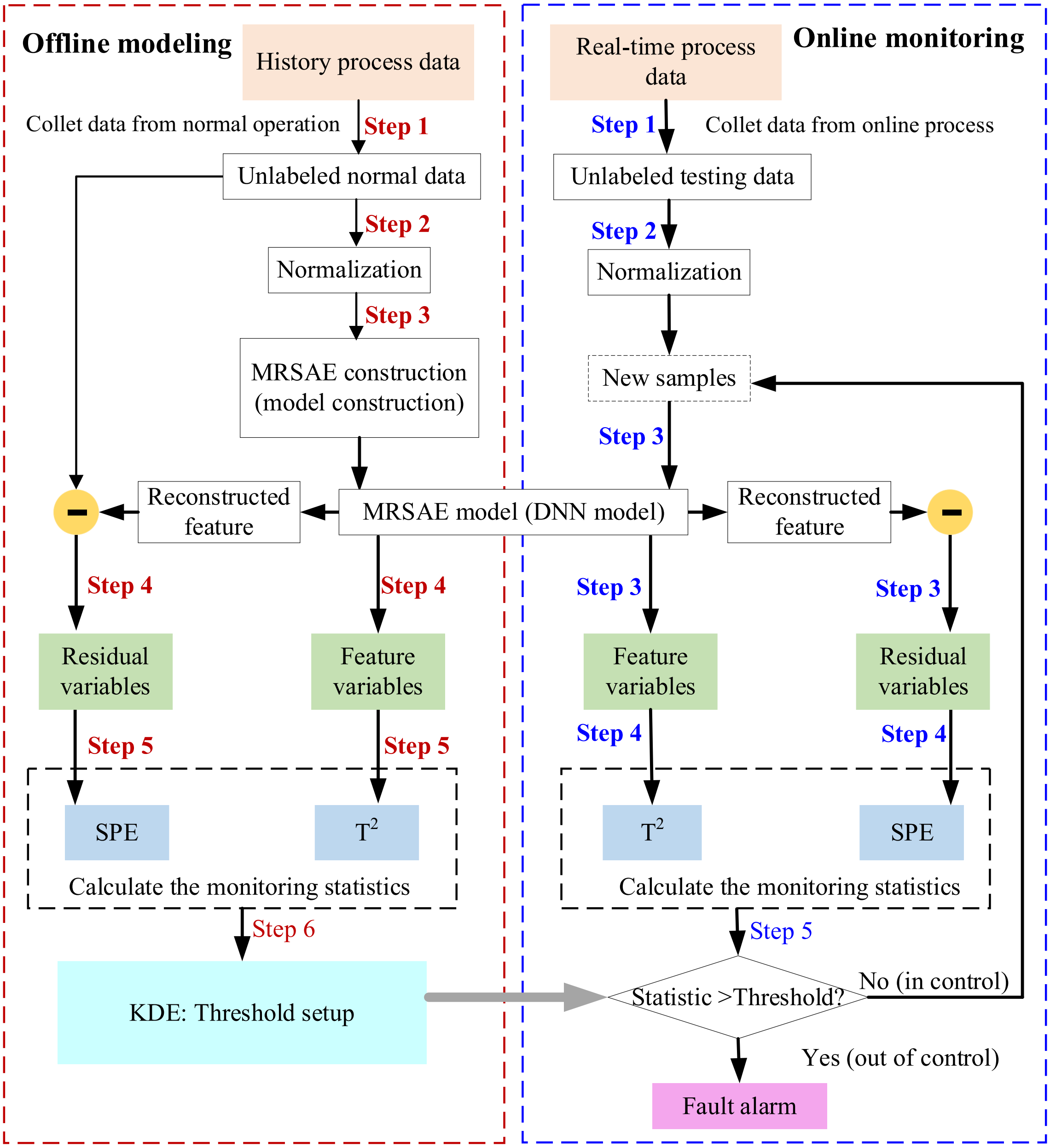


Fig. 9 Procedure of MRSAE-based process fault detection

The on-line modeling phase includes five steps. Step 1 is to collect a new observation x from the process online. Step 2 is to normalize the input within 0 and 1. Step 3 is to input x to the well-trained MRSAE model and project it into feature space and residual space. Step 4 is to compute the T^2 and SPE statistic based on the feature maps, respectively. Step 5 is to trigger an alarm once any of the two statistics exceed its corresponding threshold.

3.3.2 CNN-based approaches

3.3.2.1 A brief introduction to CNN CNN, as a deep feed-forward neural network, has commonly used in supervised learning problems in image recognition, computer vision, and target tracking [176]. A CNN model includes convolutional layers, pooling layers, and full-connected layers [39]. The fully connected layer has the same structure and operation mode as the conventional feedforward neural network. The structure diagram of CNN for processing two-dimensional data is given in Fig. 10.

A convolutional layer consists of multiple learnable kernels. Each kernel has a trainable weight and bias. The math operation in the l th layer between the specific j th and the input data x^{l-1} can be described by:

$$x_j^l = f\left(\sum_{i \in M_j} x_j^{l-1} * k_j^l + b_j^l\right) \tag{28}$$

where (*) represents the convolution operation and f is the activation function of rectified liner unit (ReLU). Assume that the input data x^{l-1} include m 2-D matrices. Every input matrix $x_i^{l-1}(i \in m)$ is convolved with the kernel k_j, and the sum of all convolution operation results will be added with the bias. Finally, the result will be fed into the activate function f to produce the final output of kernel j.

3.3.2.2 Application of CNN CNN can map low-dimensional shallow features to high-dimensional high-level features, which have been applied to many fields, e.g., image identification, text generation, machine translation, video classification. Zhang et al. [177] proposed an amplitude-frequency images-based CNN (ConvNet) for FDD in chemical processes. Each ConvNet works on a specific fault, so a flexible FDD framework can be trained and extended. Kim et al. [178] proposed a self-attentive CNN to detect and diagnose faults directly from the variable-length status variables identification (SVID). In order to extract effective features of complex multivariable process and improve fault diagnosis performance, Chen et al. [179] proposed a multivariable process fault diagnosis model based on CNN feature learning. Lee et al. [180] proposed a CNN model, which uses the receptive field of multivariable sensor signal to slide along the time axis to extract fault features of semiconductor manufacturing process. Wu et al. [181] developed a deep CNN (DCNN) model for chemical process fault diagnosis and verified the superiority of this method in TEP. Chen et al. [182] proposed a one-dimensional convolutional auto-encoder (1D-CAE) model for FDD of multivariable processes. Zheng et al. [183] proposed a hybrid system integrating SVM and CNN for pattern recognition of multivariable processes. Zheng et al. [184] presented a fault detection method based on convolutional gated recurrent unit auto-encoder for TEP. Hsu et al. [185] presented a multiple time-series CNN (MTS-CNN) model for FDD in semiconductor manufacturing process. In order to update the diagnosis model effectively to include new coming abnormal samples, Yu et al. [186] proposed a broad convolutional neural network (BCNN) with incremental learning capability for industrial processes.

Zhang et al. [187] proposed a new DNN, multichannel one-dimensional CNN (MC1-DCNN) to investigate feature learning from high-dimensional process signals. The application procedure of MC1-DCNN-based fault diagnosis comprises an off-line modeling phase and an on-line testing phase, which is presented in Fig. 11. The recognition rate of MC1-DCNN for 21 faults of TEP is shown in Fig. 12. It can be seen that MC1-DCNN has significant performance in feature extraction and process fault diagnosis.

3.3.3 RNN-based approaches

3.3.3.1 A brief introduction to RNN RNN [41] is a state-of-the-art neural network with recurrent hidden layers to perform sequence data processing and prediction. However, RNN has encountered the challenge of vanishing gradient and exploding gradient in the training procedure [188]. Thus, long short-term memory network (LSTM) is

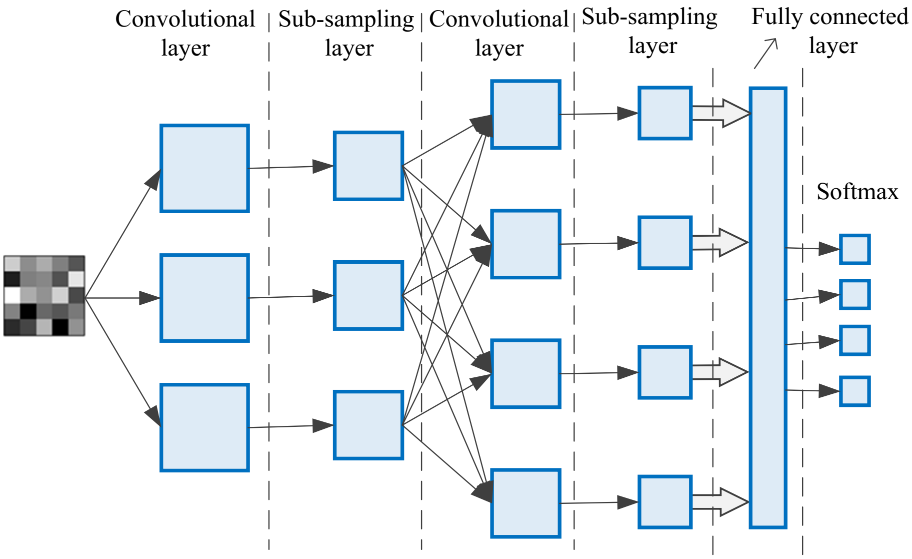


Fig. 10 Network structure of CNN

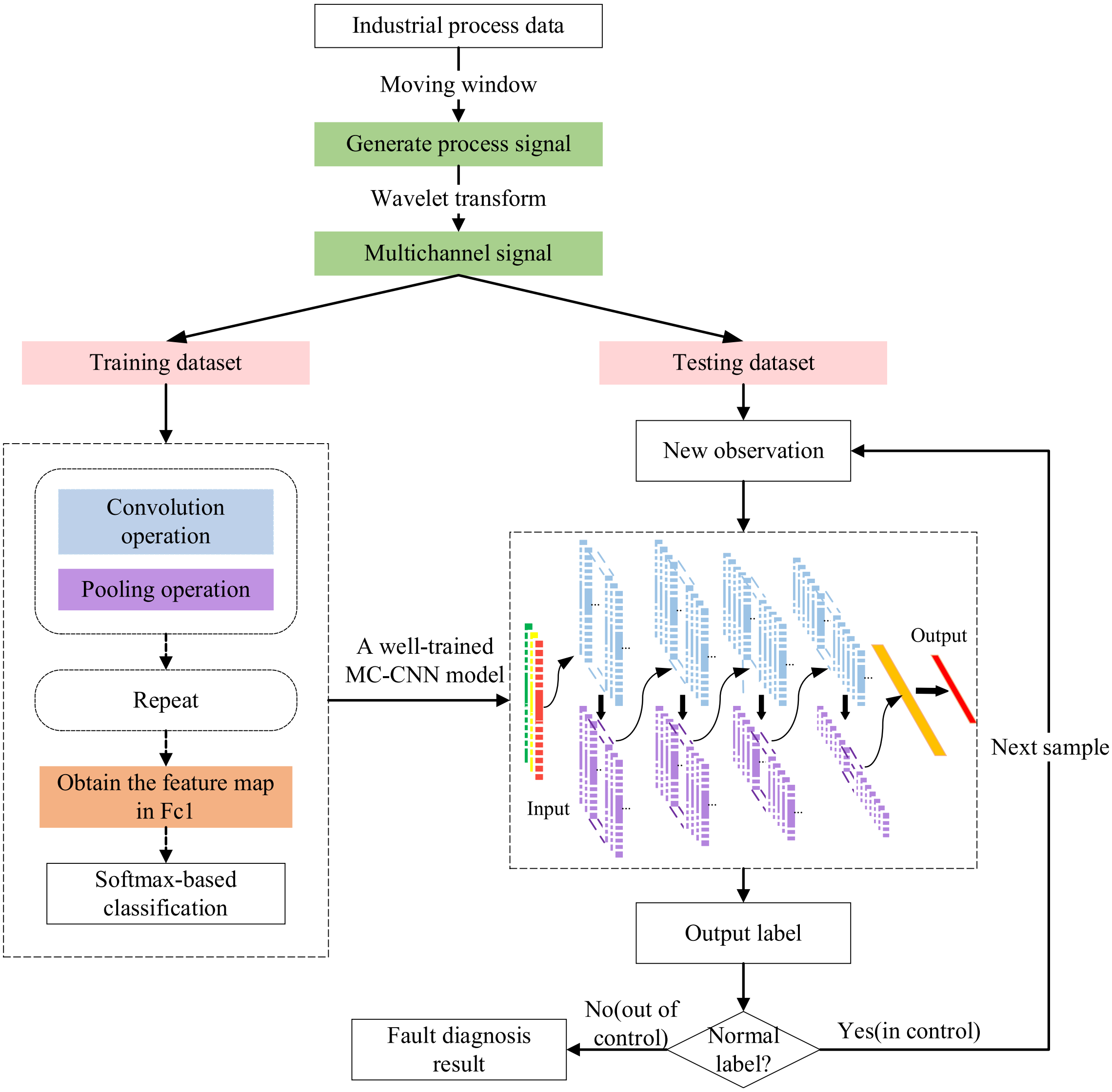


Fig. 11 Application flowchart of the MC1-DCNN-based fault diagnosis model

proposed to solve this problem [189]. The unit structure of RNN and LSTM is shown in Fig. 13.

The hidden layer information of RNN only comes from the current input and the previous hidden layer information, and the calculation formula is as follows:

$$h_{t-1} = tanh(W_h \bullet [h_{t-1}, x_t] + b_h) \tag{29}$$

where h_{t-1} represents the output of the previous cell, x_t represents the input of the current cell, and W_h and b_h represent the weight and bias of forget gate, respectively.

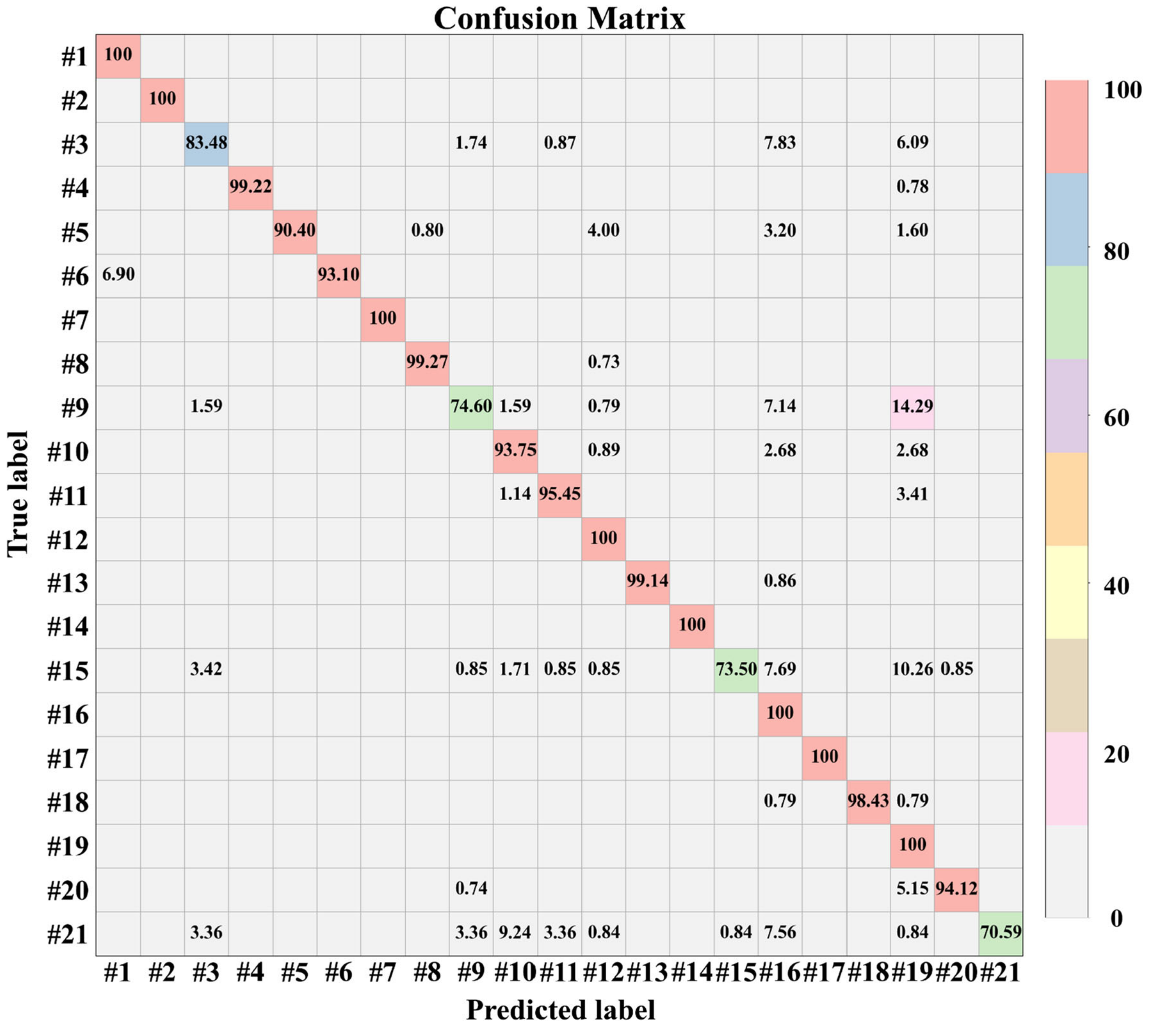
An LSTM layer is composed of recurrently connected memory blocks, each of which contains one or more memory cells, along with three multiplicative "gate" units: the input, output, and forget gate. LSTM overcomes the problems of vanishing gradient and exploding gradient.

Forget gate The forget gate of LSTM can selectively discard part of the information from the memory unit, and the calculation is as follows:

$$f_t = \sigma(W_f \bullet [h_{t-1}, x_t] + b_f) \tag{30}$$

where h_{t-1} represents the output of the previous cell, x_t represents the input of the current cell, σ represents the sigmoid function, and W_f and b_f represent the weight and bias of forget gate, respectively.

Input gate The input gate decides to add new information to the memory unit. This process includes two steps: one is to determine the information to be updated through sigmoid layer; the other is to generate alternative information by the tanh function.



	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20	#21
#1	100																				
#2		100																			
#3			83.48						1.74		0.87					7.83			6.09		
#4				99.22															0.78		
#5					90.40			0.80				4.00				3.20			1.60		
#6	6.90					93.10															
#7							100														
#8								99.27				0.73									
#9			1.59						74.60	1.59		0.79				7.14			14.29		
#10										93.75		0.89				2.68			2.68		
#11										1.14	95.45								3.41		
#12												100									
#13													99.14			0.86					
#14														100							
#15			3.42						0.85	1.71	0.85	0.85			73.50	7.69			10.26	0.85	
#16																100					
#17																	100				
#18																0.79		98.43	0.79		
#19																			100		
#20									0.74										5.15	94.12	
#21			3.36						3.36	9.24	3.36	0.84			0.84	7.56			0.84		70.59

Fig. 12 Recognition rates (%) of MC1-DCNN in a confusion matrix

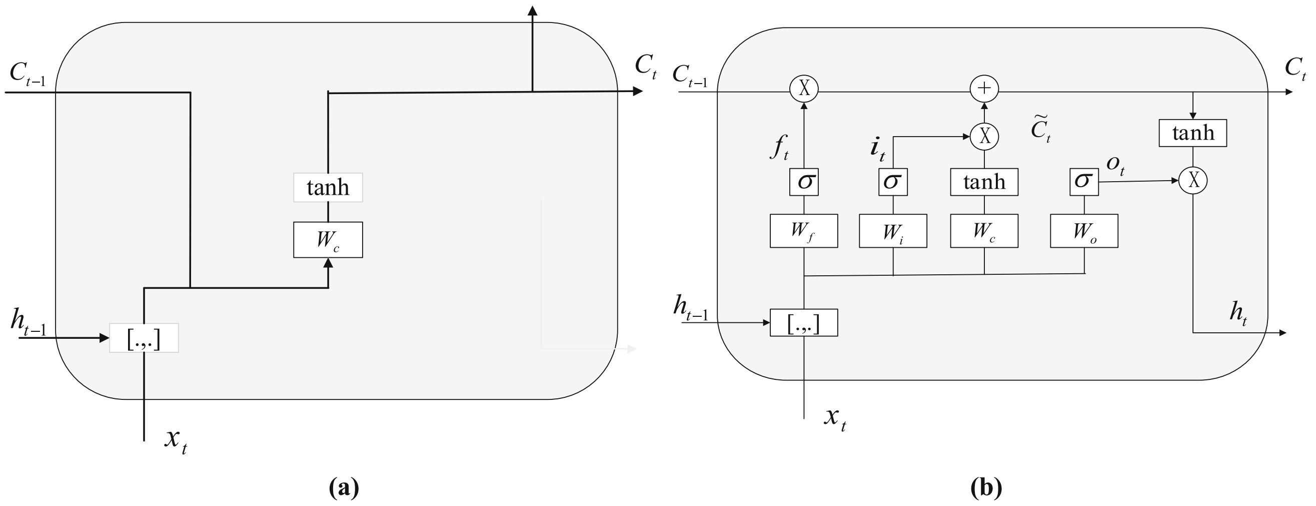


Fig. 13 Unit structure of RNN and LSTM, **a** RNN, **b** LSTM

$$i_t = \sigma(W_i \bullet [h_{t-1}, x_t] + b_i) \tag{31}$$

$$\widetilde{C}_t = tanh(W_c \bullet [h_{t-1}, x_t] + b_c) \tag{32}$$

where i_t and $\widetilde{C}_t$ are the output of the input gate and the alternative information of the memory unit, respectively, and tanh is the hyperbolic tangent function.

Memory unit The memory unit updates the memory information as follows:

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t \tag{33}$$

where C_t is the information saved by the memory unit.

Output gat Finally, the output gate will determine the final output based on the previous calculation. First, a sigmoid layer is used to determine the information, and then, the tanh function is used to calculate the final result.

$$o_t = \sigma(W_o \bullet [h_{t-1}, x_t] + b_o \tag{34}$$

$$h_t = o_t * \tanh(C_t) \tag{35}$$

where o_t and h_t are the output results of the output gate and LSTM unit at the current time, respectively.

3.3.3.2 Application of RNN

LSTM is a kind of specific RNN, which is mainly used to solve the problems of gradient disappearance and gradient explosion in the process of long-term training. Chadha et al. [190] presented a bidirectional RNN-based process condition monitoring and fault diagnosis method. Cheng et al. [191] proposed a new process monitoring method based on variational recurrent AE (VRAE). Ouyang et al. [192] proposed a fault detection and identification method based on the multidimensional gated recurrent unit (GRU) network for monitoring the blast furnace ironmaking process. Wang et al. [193] presented a high-level spatiotemporal feature extraction based on deep convolutional bidirectional encoder–decoder representation network with GRU cell for dynamic process fault diagnosis. Chen et al. [194] developed a deep RNN model of process variables with different time delays and established a residual graph to detect the mean shift of autocorrelated processes. Zhang et al. [195] employed a bidirectional RNN (BiRNN) to construct process FDD models with sophisticated RNN cells. Liu et al. [196] proposed a method based on the combination of LSTM and DBN. DBN-LSTM was used for feature extraction, time correlation analysis and fault diagnosis, and the method was applied to a class of semiconductor etching process. Yu et al. [197] proposed a convolutional LSTM-AE (CLSTM-AE) for feature learning from complex process signals.

3.3.4 DBN-based approaches

3.3.4.1 A brief introduction to DBN

Restricted Boltzmann machine (RBM) is a generative stochastic neural network that can learn probability distribution from input data set. DBN is a deep model constructed by stacking multiple RBMs, where the input of a layer is the output of the preceded layer.

RBM includes visible units $v = \{v_1, v_2, \ldots, v_m\}$ and hidden units $h = \{h_1, h_2, \ldots, h_n\}$ [198]. It is noted that all the units are binary, i.e., $v, h = \{0, 1\}$. The relationship of the visible layer and hidden layer is defined by energy function as follows:

$$E(v, h) = -\sum_{i=1}^{m}\sum_{j=1}^{n} \omega_{ij} v_i h_i - \sum_{i=1}^{m} b_i v_i - \sum_{j=1}^{n} a_j h_j \tag{36}$$

where $\theta = \{\omega, a, b\}$ represents the parameters of RBM. With the energy function, the probability distribution of the visual units can be assigned as:

$$p(v, h) = \frac{1}{Z}\exp[-E(v, h)] \tag{37}$$

where Z is the partition function, and is calculated by summing all possible visible-hidden node pairs.

$$Z = \sum_{v,h} \exp[-E(v, h)] \tag{38}$$

The conditional probability distributions of each unit are defined as follows:

$$p(v_i = 1|h) = \frac{1}{1 + \exp(-a_j - \sum_j \omega_{i,j} h_j)} \tag{39}$$

$$p(h_j = 1|v) = \frac{1}{1 + \exp(-b_i - \sum_i \omega_{i,j} v_i)} \tag{40}$$

Deep Boltzmann machine (DBM) is a deep model with many hidden layers stacked into a hierarchy structure. In order to obtain the DBN model, Bayes belief network is used at the part closer to the visible layer, while RBM is used at the part away from the visible layer, as shown in Fig. 14.

3.3.4.2 Application of DBN

Compared with the traditional fault diagnosis, DBN does not need too much signal technology and diagnosis experience to support, and has relatively strong adaptability, versatility, the ability to deal with high-dimensional and nonlinear data. The comparison between DBN-based fault diagnosis is shown in Fig. 15. The raw data in Fig. 15 include the training data and testing data, which are used for DBN model training and testing, respectively. Data preprocessing is a critical step for data-based process monitoring. It can transform the raw

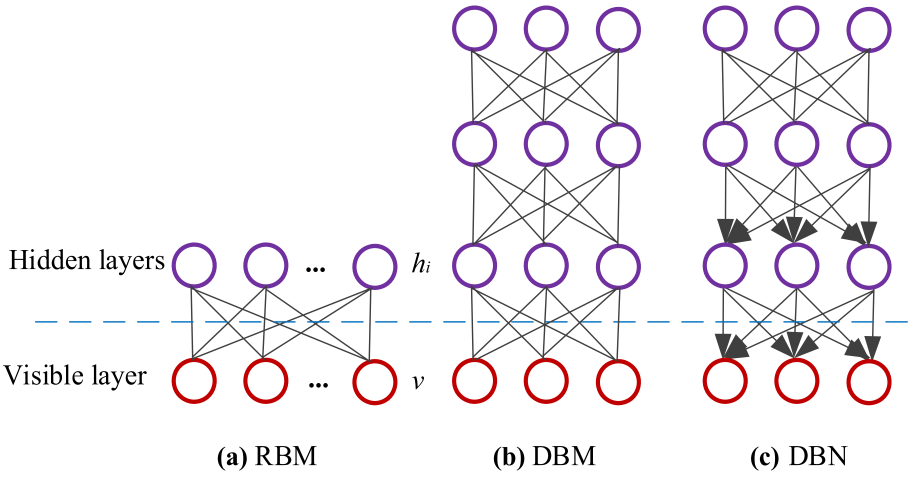


Fig. 14 Network structure of RBM, DBM and DBN, **a** RBM, **b** DBM, **c** DBN

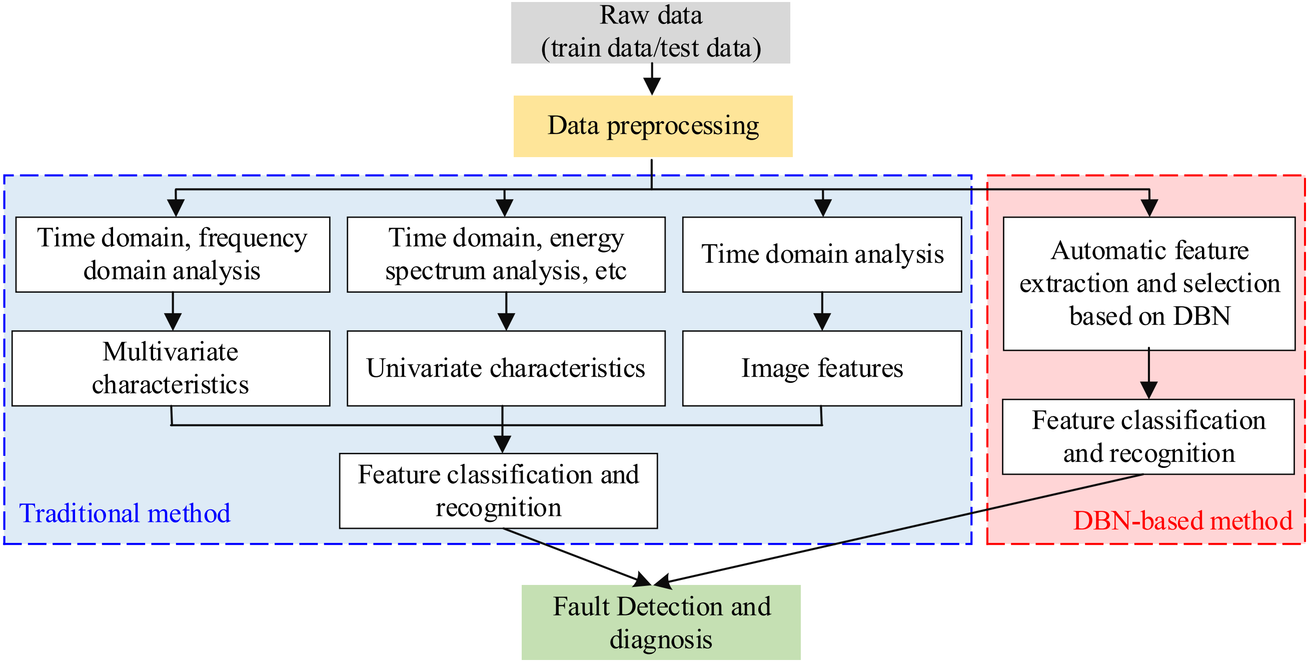

Raw data
(train data/test data)
Data preprocessing
Time domain, frequency domain analysis
Time domain, energy spectrum analysis, etc
Time domain analysis
Automatic feature extraction and selection based on DBN
Multivariate characteristics
Univariate characteristics
Image features
Feature classification and recognition
Feature classification and recognition
Traditional method
DBN-based method
Fault Detection and diagnosis

Fig. 15 Comparison of DBN-based process FDD with other process FDD methods

data into an appropriate manner (i.e., features) and can be effectively used for process modeling.

DBN is an effective method to solve the problem of FDD in industrial process. Zhang et al. [199] presented an extensible DBN-based fault diagnosis model for complex chemical processes. Tang et al. [200] proposed a fault detection model based on DBN for nonlinear processes and used the feature variables and residual variables generated by DBN to establish test statistics for abnormal monitoring of industrial processes. Kim et al. [201] proposed a DBN-based multi-classifier for fault detection prediction in the semiconductor manufacturing process. In order to select the features that are beneficial to process monitoring, Yu et al. [202] presented the concept of active feature, and applied the active features extracted by DBN (AF-DBN) to process monitoring. Based on the theoretical analysis and experimental study of unstable neurons, Yu et al. [203] proposed a novel method (UN-DBN) based on the unstable neurons in hidden layers for process monitoring. Wang et al. [204] proposed an extended deep belief network (EDBN) to make full use of the useful information in the original process data. Yu et al. [205] proposed multiple DBNs (M-DBN) to extract abstract and high-order information from each pattern in large-scale industrial processes. The training efficiency, accuracy of feature extraction, and monitoring performance of the model system are better than traditional DBN.

3.4 Step 3: process monitoring

After construction of the deep learning-based models, they are used for the process FDD. Control chart is a functional chart that can detect the out-of-control signals in processes. It is an important statistical tool in on-site quality management. In process monitoring, control charts are often used to analyze and judge whether the process is in a stable state. For example, two monitoring statistics $T.^2$ and SPE are typically constructed in the multivariate statistical approaches-based process monitoring. When a possible fault is detected in the process, the next step is to diagnose it to find the root cause of the fault, reconstruct its direction and size, and identify the type of fault. Based on the above analysis, the operation process can be restored under normal conditions through process control strategies. Finally, process recovery can be completed through fault isolation. [2]

3.4.1 Fault detection

Fault detection refers to the real-time monitoring of process data by various sensors and equipment to determine whether a fault occur in the process. The process stage is usually judged according to the selected monitoring statistics (e.g., T^2, SPE). If the monitoring statistical value exceeds its corresponding threshold, a fault alarm shall be triggered. The steps of fault detection based on deep learning are shown in Fig. 16, which includes two stages: offline modeling and online monitoring.

3.4.2 Fault diagnosis

Fault diagnosis is to determine what kind of fault occurs in the process, specifically to determine the type of fault, fault magnitude, fault location and time. The basic framework of process fault diagnosis based on deep learning is shown in Fig. 17, which includes five steps: state definition, data preprocessing, training, testing and diagnosis result evaluation. The fault diagnosis method based on deep learning has the following advantages: (1) unlike those shallow networks, deep learning does not need fault feature links manually, but integrates fault feature learning and classification model; (2) deep learning is a multi-hidden layer network, which can avoid the limitation of dimension disaster and insufficient diagnostic ability of shallow networks.

3.4.3 Fault reconstruction

Fault reconstruction is to estimate the nominal normal measurement value according to the measured value of the process variables that have been affected by the fault when the fault occurs. The purpose is to explore the direction and size of the fault and minimize the influence of fault factors on the normal part of the data. Through fault reconstruction, the severity of the fault can be estimated, and

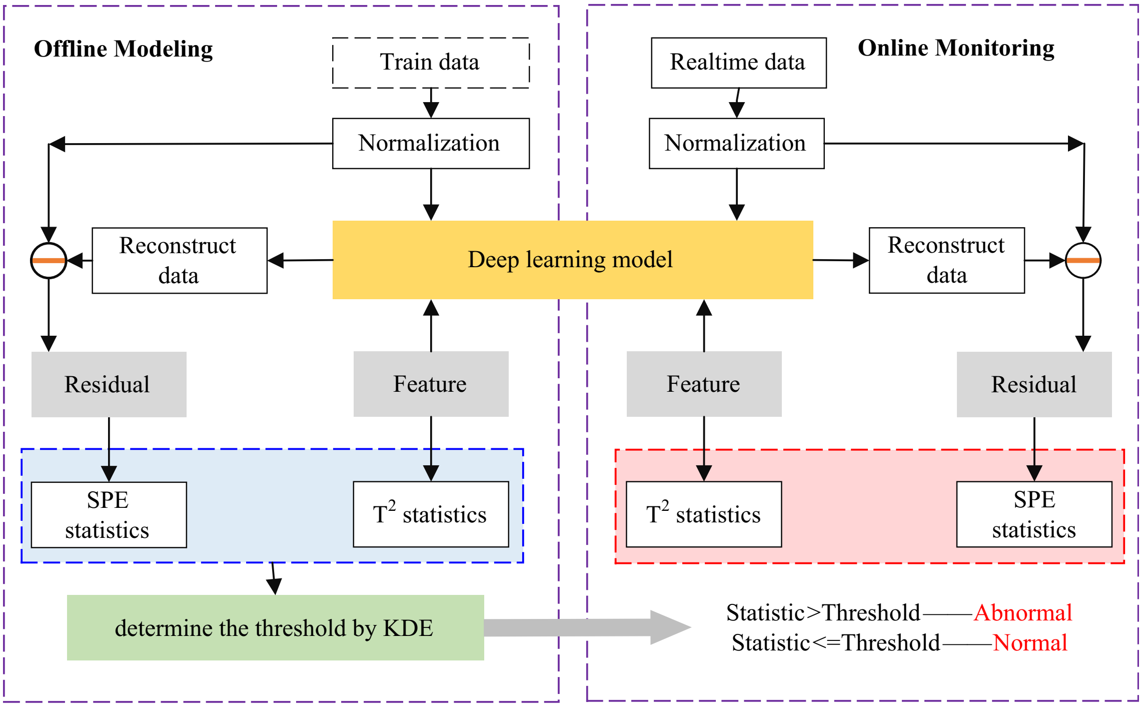


Fig. 16 Procedure of deep learning-based process fault detection

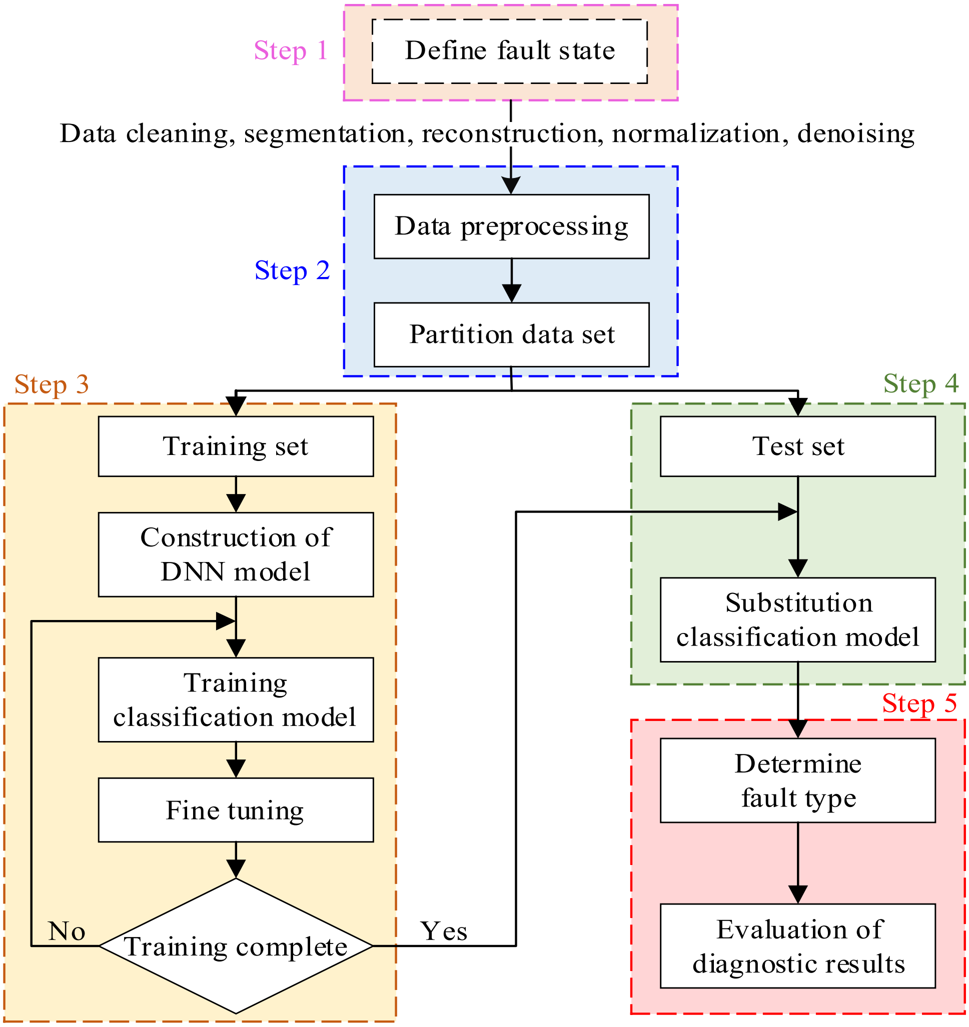

Step 1
Define fault state
Data cleaning, segmentation, reconstruction, normalization, denoising
Step 2
Data preprocessing
Partition data set
Step 3
Training set
Construction of DNN model
Training classification model
Fine tuning
Training complete
No
Yes
Step 4
Test set
Substitution classification model
Step 5
Determine fault type
Evaluation of diagnostic results

Fig. 17 A basic framework of process fault diagnosis based on deep learning

corresponding measurements can be taken to eliminate the impact of the fault on the process.

3.4.4 Fault identification

Fault identification is to find out the most relevant observation variables after a fault occurs. The information of fault type can help the operator quickly understand the fault and find the appropriate maintenance strategy to make the process return to normal as soon as possible. Fault recognition can be regarded as a pattern matching and recognition problem. Different data-based pattern analysis methods can be used for fault identification.

3.4.5 Fault isolation and process recovery

The process needs to be adjusted according to the analysis results. Fault isolation is to determine the location of the fault in the process and isolate the fault from other parts of the process after detecting the detailed information of the fault. The industrial process after fault isolation will not be significantly affected. Then, based on the process control strategies, the operation process can be restored under normal conditions through process maintenance and repair (e.g., parameters selection, process optimization, equipment maintenance). Control strategies were set according to different production objectives.

4 Process fault detection and diagnosis using transfer learning and deep learning-based approaches in the future

Under the premise of sufficient labeled data and balanced data, traditional deep learning methods presented in Sect. 3 can effectively mine and extract the relationship between the input process variables and industrial process faults. However, in actual industrial process, the fault data and normal data are often unbalanced, and the data are often incomplete and noisy. According to the above characteristics, it is necessary to train a reliable FDD model for industrial process.

This section presents the most potential research prospects of process FDD from four parts. The first part presents the novel deep learning models evolved from traditional deep learning models, i.e., deep residual shrinkage network (DRSN), independent RNN (IndRNN), and transformer. The second part introduces the applications of transfer learning and GAN in the process FDD. The third part presents the advantages of capsule network (CapsNet), GNN, DGP, and deep Gaussian mixture model (DGMM) in solving process FDD problem in the future. Finally, the application prospects of large-scale neural networks in process monitoring are presented.

Generally, it requires a large number of process data in the training of deep learning model. The possible data problems (e.g., inaccurate data, missing data, unbalanced data) will have a great negative impact on process monitoring. Moreover, it is very difficult for traditional machine learning methods to solve these problems. In recent years, DNNs have been applied to the process FDD widely. For example, CNN can process high-dimensional data effectively by sharing convolution kernel. However, it needs to be trained with sufficient labeled samples. On the basis of CNN, CapsNet considers the spatial relationship of data and can achieve good performance in small sample data. RNN can mine the sequence information in the data to solve time-varying process, but it takes a large time cost to train the model. As a variant of RNN, transformer can effectively solve the problem of high time cost of RNN training.

The deep learning-based process FDD model automatically learns features from the input data, and identifies the state of industrial process [206]. The DNNs mainly include feature extraction layer and classification layer. The feature extraction layer usually includes multiple hidden layers, which can be superimposed by the networks (e.g., AE, CNN, RNN, DBN) to learn abstract features layer by layer. The last layer of the DNN is the classification layer, which is used to classify the extracted features. In the training process, the back propagation (BP) algorithm is often used to update the training parameters of the DNNs to minimize the error between the actual output and the target. In recent years, various novel deep network architectures have been proposed (e.g., DRSN, transfer learning, GAN, CapsNets, GNN). Transfer learning and GAN can solve the problem of small sample problems in process FDD through knowledge transfer. However, the performance of the model is affected by the source domain and network parameters. CapsNets considers the spatial relationship of data, which makes it obtain higher classification performance. However, the time cost of the model training is high, which limits its further applications in process FDD. In addition, some emerging deep models, e.g., GNN, DGP, DGMM, are different from traditional DNNs in network architecture. It will be interesting to apply these specific deep learning models in process FDD in the future. According to the characteristics of these models, the advantages and disadvantages of them and their applications in process FDD are summarized in Table 2.

4.1 Deep learning

These traditional DNNs, e.g., CNN, RNN, AE, DBN, have been widely applied to industrial process FDD. However, these DNNs still have these defects: (1) the feature resolution of CNN will decrease with the growth of layers, resulting in the loss of information. In addition, due to the fixed size of the core, CNN is not suitable for fault diagnosis under non-stationary conditions; (2) RNN has the problem of gradient disappearance and explosion and cannot be stacked into deeper networks; (3) AE does not capture the correlation of information. In recent years, some novel deep learning models have been developed on the basis of traditional DNNs to improve the performance of the model. The subsection presents several new network models of CNN and RNN, which will effectively solve these problems, e.g., data imbalance, multi fault categories, time correlation, attention mechanism in industrial process control.

4.1.1 Deep residual shrinkage network

Deep residual shrinkage network (DRSN) is an improved method of ResNet. Its characteristic is “shrinkage,” which refers to soft thresholding that is almost the necessary step of signal denoising algorithm [215]. As shown in Fig. 18, the cross-layer identical path has the following improvements compared with CNN: (1) soft thresholding is used to eliminate redundant information; (2) SENet type subnet structure is adopted to automatically set the threshold. Under the effect of soft thresholding, compared with ordinary CNN, deep residual shrinkage network is more suitable for noisy data classification task.

4.1.2 Independent RNN

RNNs are usually difficult to train and learn long-term patterns due to gradient vanishing and exploding problems. The gated cycle unit (GRU) is used to solve the above problems, but the use of hyperbolic tangent and S-type action function will lead to gradient attenuation on the layer. Thus, it is challenging to construct an effective and trainable deep network. To solve these problems, Li et al., [216] proposed a new neural network called IndRNN. The neurons in the same layer are independent of each other

Table 2 Summary of applications of deep learning in process FDD

Methods	Advantages	Disadvantages	Applications
AE	(1) It can handle both linear and nonlinear data (2) It can be used for complex large data sets	It needs to be trained with sufficient samples	TEP[166, 167, 169–175], multivariate process [169], FBFP [170, 173], CSTR [175]
CNN	(1) It is very effective to learn features from high dimensional data (2) It is able to directly learn features from the data without preprocessing	(1) It spends high time cost to train the model on a large dataset (2) It needs to be trained with sufficient labeled samples	TEP [170, 172, 181–184, 186, 187], FBFP [170, 182], CSTR [172], chemical processes [177], multivariable process [179], semiconductor manufacturing process [178, 180, 185]
LSTM	(1) It can make full use of the received information to obtain good performance (2) It can deal with autocorrelated process data	(1) It takes a high time cost to train the model (2) Its ability to process high-dimensional data is limited	TEP [190, 191, 193, 195, 197], blast furnace ironmaking process [192], autocorrelated process [194], semiconductor etching process [196], CSTR [197]
DBN	(1) It can deal with high-dimensional nonlinear data (2) It is able to obtain high-level information in process data	It needs to be trained with sufficient labeled samples	Semiconductor etching process [196, 201], TEP [199, 200, 202–205]
Transfer learning	It can solve the problem of small data in process monitoring	It is difficult to guarantee the effectiveness of knowledge transfer and the performance of the model	FBFP [207], multimode chemical processes [208], industrial multiphase flow process [209]
GAN	It can solve the problems of small data and uneven data distribution	It is difficult for model training and its performance is unstable	TEP [210, 211], smoke detection process [212], chiller process [213]
CapsNets	It can achieve good performance on small sample data sets	(1) It takes a high time cost to train the model (2) Its performance in large-scale data sets is unknown	
GNN	It can construct the relationship in the data from the perspective of graph, and can mine the high-order information of the data	It is difficult to construct one-dimensional data into graphic data	Converter steelmaking process [214]
DGP	It is a highly flexible multi-layer prediction model, which can accurately model the uncertainty	It takes a high time cost to train the model	
DGMM	(1) It can deal with nonlinear process (2) It can be used for unsupervised fault detection	It needs to be trained with sufficient labeled samples	

and connected across layers. IndRNN is easy to adjust, which can prevent the problem of gradient explosion and disappearance, and allows the network to learn long-term dependence.

The basic architecture of IndRNN is shown in Fig. 19, where "weight" and "Recursion + ReLU" represent the input process and the recursive process of each step with ReLU as the activation function. Compared with LSTM, sigmoid and hyperbolic tangent functions are used to attenuate the gradient on the layer, and unsaturated activation function such as ReLU reduces the gradient vanishing problem on the layer. In addition, before or after the activation function shown in Fig. 19, the batch standardization represented as "BN" can also be used in the IndRNN network.

4.1.3 Transformer

Transformer improves the most criticized shortcoming of RNN, which is slow training, and uses self-attention mechanism to achieve fast parallel. The structure of transformer is composed of encoder and decoder [217]. The overall architecture of transformer is shown in Fig. 20a, which uses stacked self-focus and point type. Figure 20a shows the full connection layer of encoder and decoder, respectively. Transformer uses multi attentions to linearly project queries, keys, and values to d_k, $d_{k,}$ and d_v

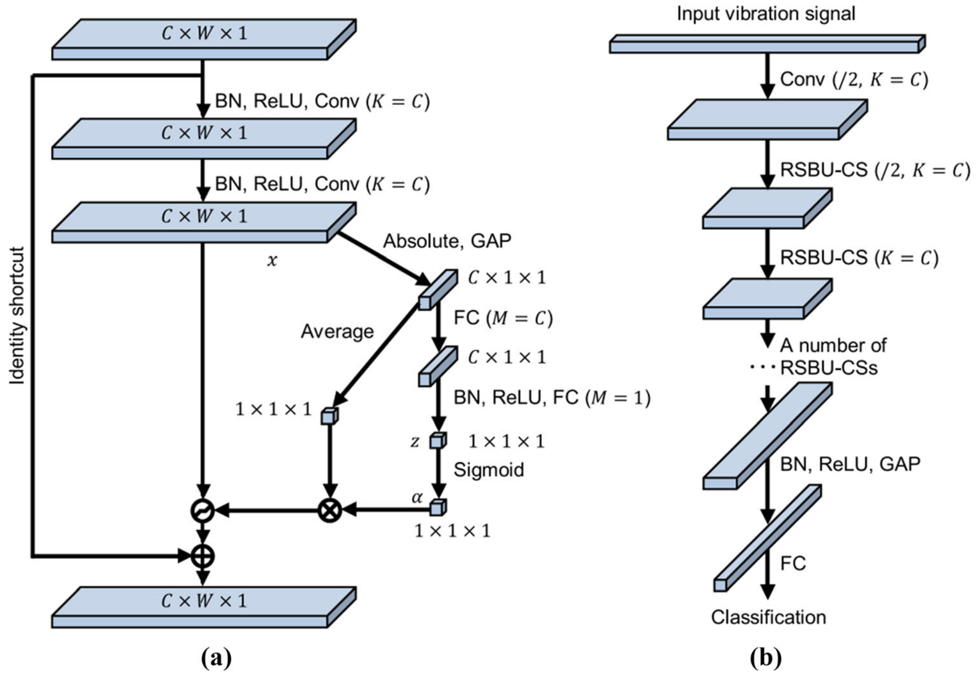


Fig. 18 Deep residual shrinkage network, **a** basic residual shrinkage module, **b** overall architecture of DRSN [215]

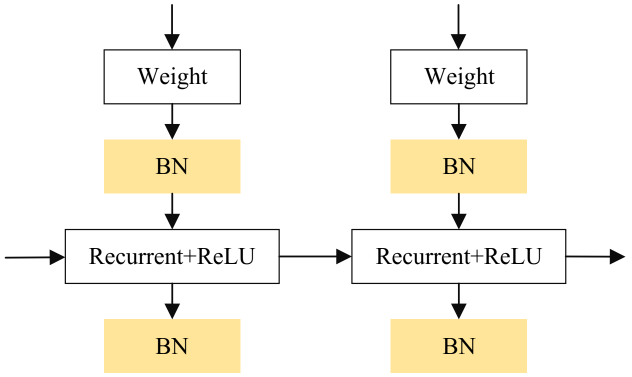


Fig. 19 Basic network structure of IndRNN

dimensions and uses different linear projections for h times. On each projection version of queries, keys and values, it performs the attention function in parallel to generate d_v-dimension output value. These values are concatenated and projected again to output the final value, as depicted in Fig. 20b.

4.2 Applications of transfer learning

4.2.1 Transfer learning

In the typical intelligent fault diagnosis model, it is generally assumed that the training dataset and the testing dataset follow the same distribution. Thus, these models directly use the pretrained fault diagnosis model on the training set to diagnose the testing set. However, in the real industry, due to the influence of various factors, there are inevitably some differences between the source domain and the target domain. The intelligent fault diagnosis based on transfer learning considers that there are differences between the target domain and the source domain. The transfer learning algorithm learns the difference between the source domain and the target domain on the basis of learning the source domain, so as to further improve the generalization ability of the model. The comparison between the intelligent algorithm based on transfer learning and the traditional algorithm is shown in Fig. 21.

Transfer learning aims at the defect of traditional machine learning, which is based on the assumption of the same distribution and needs a large number of labeled data. It solves the problem of data distribution difference and labeled data overdue in practical work and ensures the model accuracy in new tasks by making full use of labeled data (existing knowledge). Xu et al. [218] proposed an online fault diagnosis method based on deep transfer CNN. By transferring the shallow layer of trained offline CNN to online CNN, the real-time performance of process FDD can be significantly improved. Zhu et al. [207] developed a transfer learning framework to improve effectiveness of process monitoring in similar batch scenarios and verified the feasibility and effectiveness of the framework. Wu et al. [208] proposed a FDD method based on transfer learning for multimode chemical processes. Chai et al. [209] proposed a multisource-refined transfer network (MRTN) to solve the problem of fault diagnosis when the source and target fault category sets are inconsistent, and the superiority of this method is proved in a case of industrial multiphase flow process.

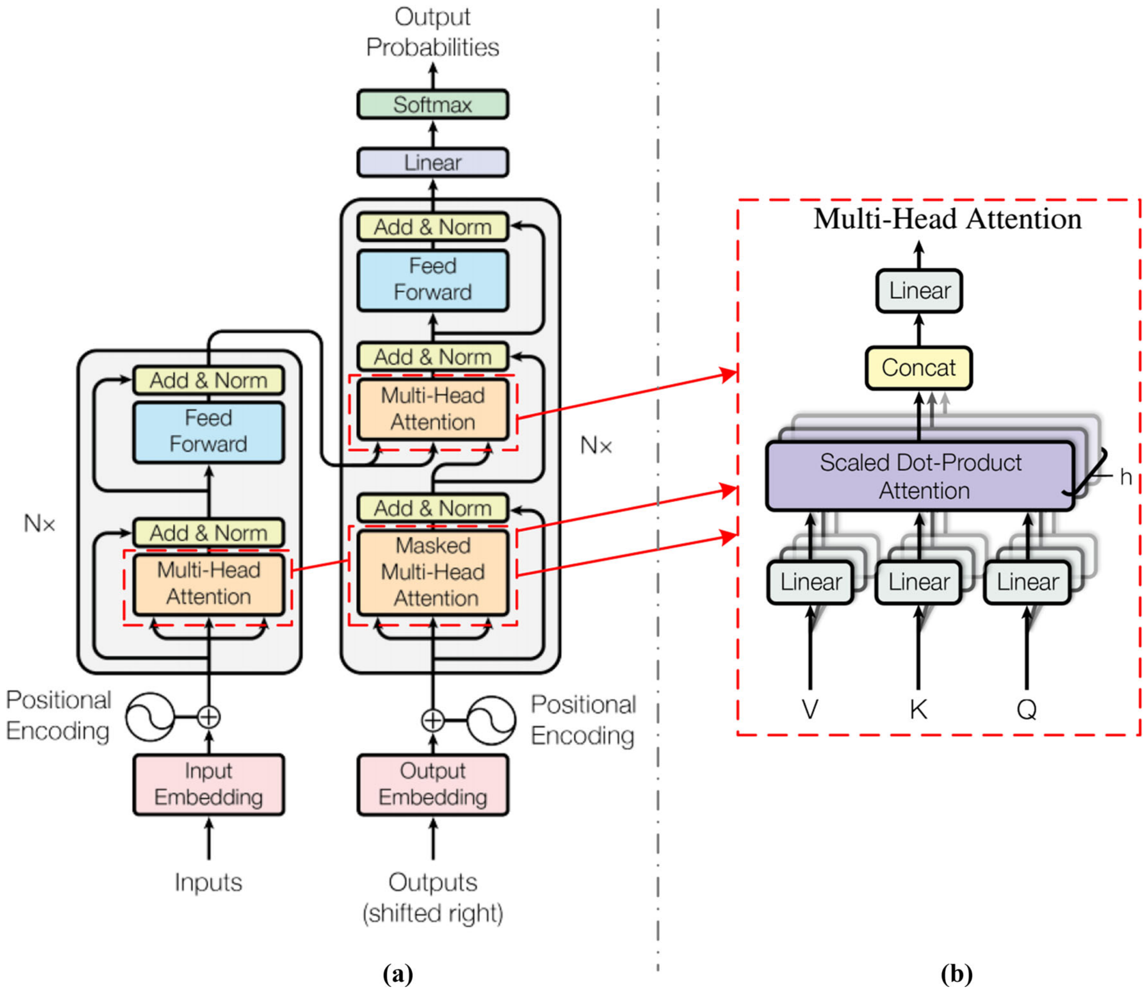


Fig. 20 Transformer, **a** network architecture, **b** multi-head attention consists of several attention layers running in parallel

4.2.2 GAN

The idea of adversarial learning in GAN is inspired by the Nash equilibrium in game theory, and the structure of GAN is shown in Fig. 22. In the model, both sides of the game are realized by generator (G) and discriminator (D). The generator G is mainly responsible for generating samples subject to real distribution through random hidden variables z, and the discriminator D is responsible for identifying whether the input samples are false samples or real samples. Through the iterative adversarial learning between the two generators, the potential distribution of real samples is obtained and new samples are generated.

In real-world scenarios, it is usually very difficult to collect enough faulty training samples to generate a balanced training dataset. Thus, some data generation methods are proposed to identify those process faults with unbalanced data distribution in industrial process. Gao et al. [210] proposed a data augmentation method based on Wasserstein GAN with gradient penalty (WGAN-GP) to improve the fault diagnosis accuracies. Yang et al. [211] proposed a bidirectional GAN (BiGAN) and applied it to process fault detection. Inspired by the GAN, Huang et al. [212] proposed an end-to-end attentive DesmokeGAN, which realized the visual attention in the generation network and effectively applied to the smoke detection process. Yan [213] proposed a new GAN to generate high-quality synthetic fault samples for data-driven fault detection and diagnosis of chiller.

4.3 Other deep learning-based methods

4.3.1 CapsNet-based methods

The traditional DNNs, e.g., CNN, ignore the location relationship of features in classifying fault time–frequency graphs. In order to overcome the limitations, CapsNet takes into account the size and location of the image. The architecture of CapsNet is shown in Fig. 23. CapsNet can solve the problem that the traditional DNNs cannot reflect

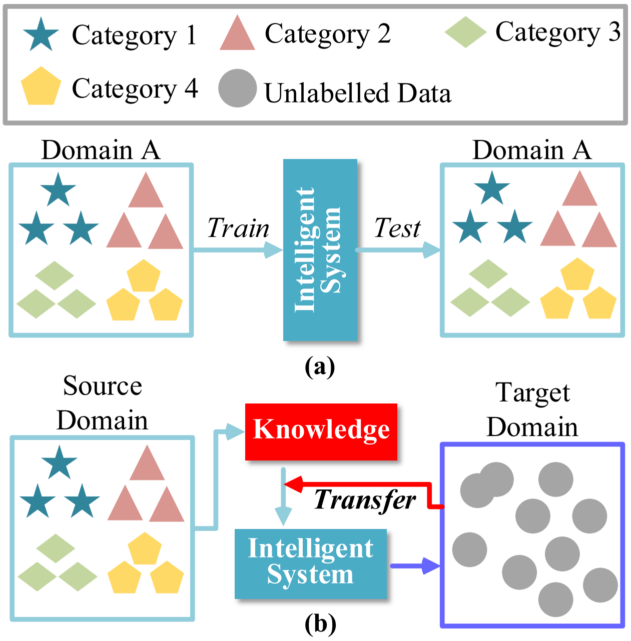


Fig. 21 Fault diagnosis model based on traditional methods and transfer learning-based methods, **a** traditional methods, **b** transfer learning-based methods

the internal relationship. Although the application of CapsNet in industrial process is still blank, there are some successful applications in machinery fault diagnosis. For example, Wang et al. [219] proposed a capsules network combined with the Xception module (XCN) to improve the classification accuracy of intelligent fault diagnosis. Chen et al. [220] proposed a novel method called deep CapsNet with stochastic delta rule (DCN-SDR) for rolling bearing fault diagnosis. Chen et al. [221] used a contemporary novel neural network architecture called CapsNet to accomplish the recognition and classification of seven working conditions of a high-speed train (HST) bogie. These applications demonstrate the advantages of CapsNet over traditional DNNs in machinery fault diagnosis. Applications of CapsNet and the expansion to industrial process FDD will be an interesting issue in the future.

4.3.2 GNN-based approaches

Graph is a data structure that models a group of objects (nodes) and their relationships (edges). GNN [222] is a deep learning method based on graph domain. Although deep learning has achieved great success in Euclidean space data, the data in many practical application scenarios are generated from non-Euclidean space and need to be analyzed effectively. The complexity of graph data poses a great challenge to existing machine learning algorithms. Graph analysis is a data structure that models nodes and their relationships. Graph analysis can be used for node classification, link prediction and clustering.

The core assumption of existing deep learning algorithms is that data samples are independent of each other. However, this is not the case for a graph. Each data sample (node) in the graph has edges related to other real data samples (nodes) in the graph, which can be used to capture the interdependence between instances. GNN can well construct the influence relationship among the key components in process control problems and provides a new idea for process FDD from the perspective of node association.

In order to apply GNN to process control, it is necessary to construct one-dimensional process data into graph structure data. There are two common methods of graph construction. One is unsupervised composition by measuring joint feature statistics. The other is supervised

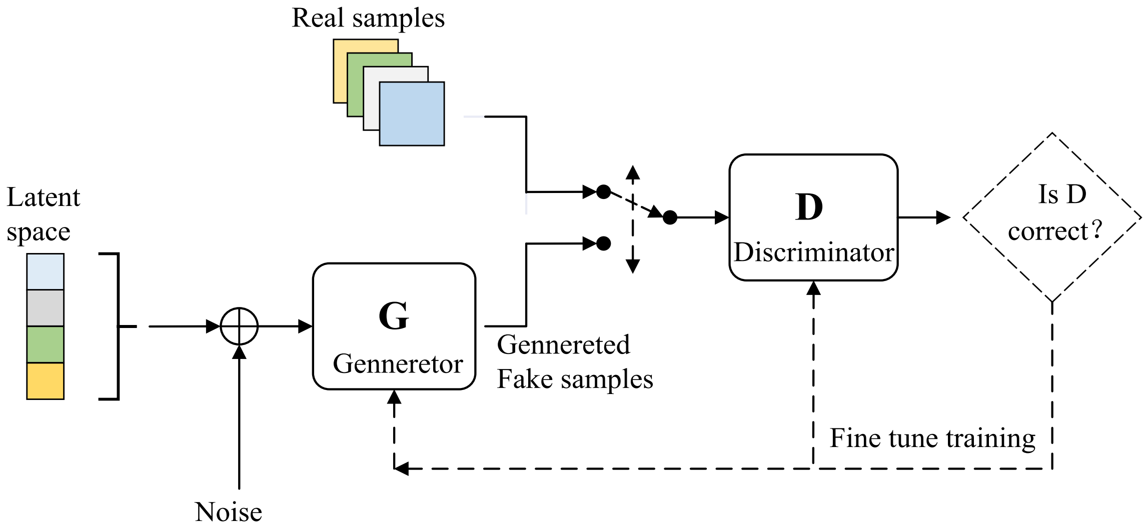


Fig. 22 Architecture of GAN

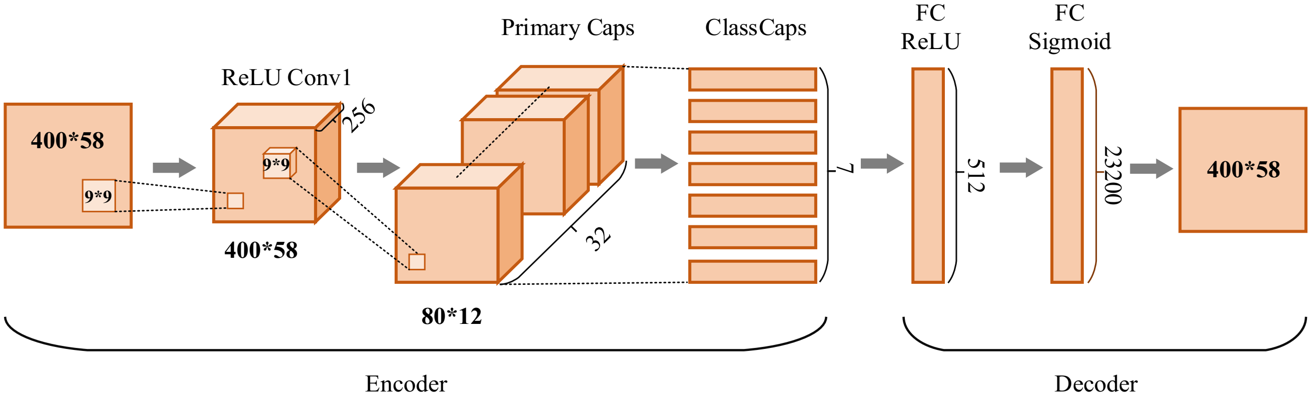


Fig. 23 Architecture of CapsNet

composition by using initial network as a proxy for the estimation [223]. Feng et al. [214] proposed a multichannel diffusion graph convolutional network (MCDGCN) to solve the problem of composition prediction in realistic converter steelmaking process. At present, there are still less researches on GNN-based process FDD. However, it is very interesting to apply GNN to process FDD in the future.

4.3.3 DGP-based approaches

DGP is multilayer predictive model that is highly flexible and can accurately model uncertainty [224]. As depicted in Fig. 24, the input to the hidden layer is the input data x and the output of the hidden layer f_1 serves as the input data to the output layer, which itself is formed by GPs. The output of the layer is probabilistic rather than exact, so uncertainty propagates through the network [225]. Santiago et al. [226] presented a hybrid deep learning-Gaussian process method for Diabetic Retinopathy diagnosis and uncertainty quantification. Tagade et al. [227] proposed a deep Gaussian process algorithm for lithium-ion battery health monitoring. The application of this method in industrial process is still blank.

4.3.4 DGMM-based approaches

DGMM is a network of multiple layers of latent variables [228]. In each layer, the variables follow the mixture of Gaussian distribution. The deep hybrid model consists of a set of nested linear hybrid models, which provides a nonlinear model that can describe the data in a very flexible way.

As depicted in Fig. 25, structure of a DGMM model with $h = 3$ and the number of layer components be $k_1 = 3$, $k_2 = 3$ and $k_3 = 2$. In the first layer, the conditional distribution of the observed data for a given $Z^{(1)}$ is a mixture of three components. If $y = Z^{(0)}$ is regarded as zero level,

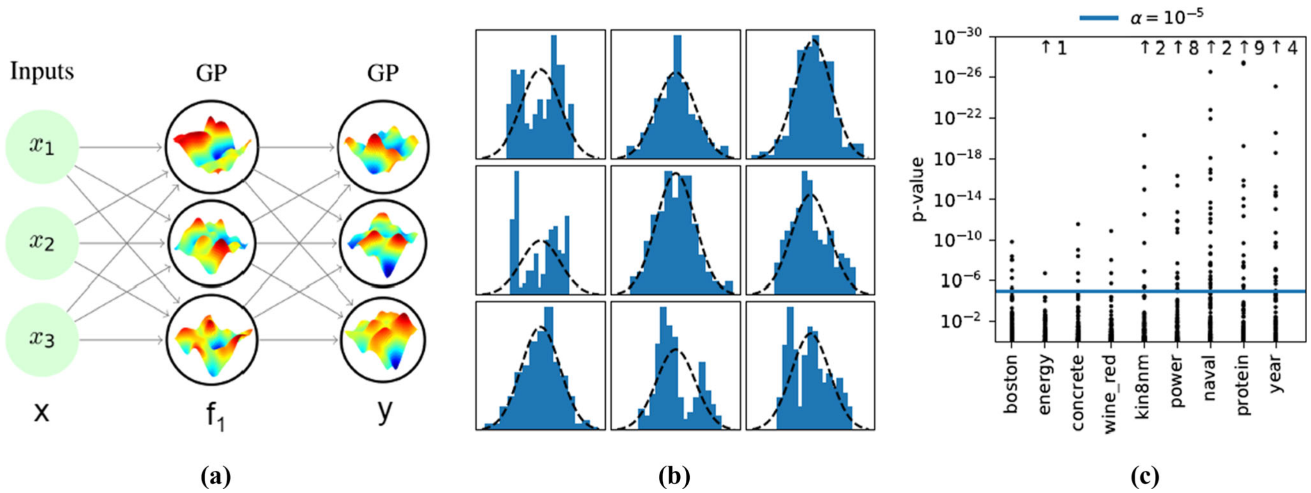


Fig. 24 Deep Gaussian process with a single hidden layer. **a** Deep Gaussian process illustration. **b** Histograms of a random selection of inducing outputs. The best-fit Gaussian distribution is denoted with a dashed line. Some of them exhibit a clear multimodal behavior. **c** P-values for 100 randomly selected inducing outputs per dataset. The null hypotheses are that their distributions are Gaussian [225]

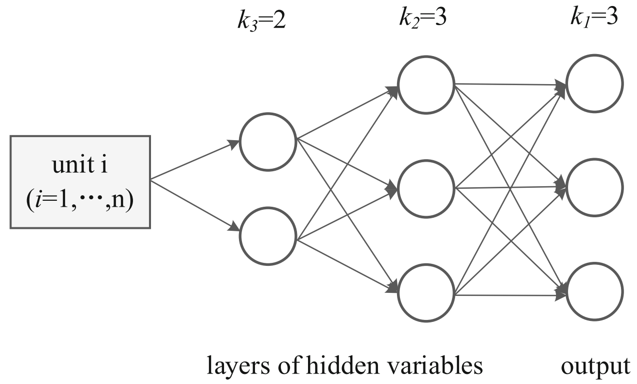


Fig. 25 Structure of a DGMM model with $h = 3$ and the number of layer components $k_1 = 3$,$k_2 = 3$ and $k_3 = 2$

all conditional distributions follow the first order Markov property, that is,

$$f(z^{(l)}|(z^{(l+1)}, (z^{(l+2)}, \ldots, (z^{(h)}; \theta) = f(z^{(l)}|(z^{(l+1)}; \theta)$$

for $l = 0, \ldots, h-1$. At each layer:

$$f(z^{(l)}|\left(z^{(l+1)};\theta\right) = \sum_{i=1}^{k_{l+1}} \pi_i{}^{(l+1)} N(\varphi_i^{(l+1)} + \gamma_i^{(l+1)} z^{(l+1)}, \rho_i^{(l+1)}) \tag{41}$$

where $z_i^{(n)} \sim N\left(0, \boldsymbol{I}_p\right)(i = 1, \ldots, n)$ and $u_i^{(1)}, \ldots, u_i^{(n)}$ are specific random errors that follow a Gaussian distribution with zero expectation and covariance matrices $\rho_{s1}^{(1)}, \ldots, \rho_{sh}^{(h)}$. $\varphi_{s1}^{(1)}, \ldots, \varphi_{sh}^{(h)}$ are vectors of length p, and $\gamma_{s1}^{(1)}, \ldots, \gamma_{sh}^{(h)}$ are square matrices of dimension p. Purohit et al. [229] suggested a deep autoencoding GMM with hyper-parameter optimization (DAGMM-HO) to solve the problem of unsupervised anomaly detection.

4.4 Large-scale neural network

DNN is usually over parameterized, which leads to a huge waste of calculation and storage. In view of the large number of parameters of large-scale DNNs and the need to rely on the hardware platform with large storage space and excellent computing performance, it is necessary for the structure compression and optimization acceleration of large-scale neural network for process FDD. The goal is to shorten the training of DNNs and broaden the application scope of DNNs.

With the development of artificial intelligence technology, the depth of neural network is getting deeper and deeper, followed by the disadvantages of high storage and high-power consumption. This restricts the application of DNNs in the resource limited application environment. A large amount of redundant information is stored in a DNN model with more than one million levels. Thus, the compression and acceleration of DNNs are a feasible and effective solution.

Process FDD in modern industry involves a large number of equipment and data. Small scale DNNs cannot describe and model such process control system well. As a result, large-scale neural network will gradually replace small-scale neural network in the application in modern industrial process control. Although large-scale neural network can effectively improve the accuracy of process FDD, "large-scale" also brings many problems, such as complex neural network structure, network parameter explosion and model operation difficulty. There are five main methods to compress and accelerate large-scale neural networks: parameter pruning, parameter sharing, low-rank decomposition, designing compact convolutional filters, and knowledge distillation [230]. Parameter pruning can remove redundant parameters by a criterion that can judge whether the parameters are important or not. Low-rank decomposition uses matrix or tensor decomposition to estimate and decompose the original convolution kernel in the depth model. The compact convolutional filters mainly reduce the storage and computational complexity of the model by special structured convolution kernel or tight convolution computing unit. This study mainly uses the knowledge of large-scale network and transfers it to the model of compact distillation. Knowledge distillation mainly uses the knowledge of large-scale network and transfers its knowledge to the compact distillation model.

5 Discussion about future challenges and opportunities in process fault detection and diagnosis

With the continuous development of technology and production, the process of modern industrial system has become extremely complex. It is necessary to provide effective guidance for future industrial process FDD. Relevant technologies for process FDD are essential to improve the safety and efficiency of the production process. A developing route for process FDD is shown in Fig. 26. Figure 26 presents three main problems of machine learning-based process monitoring, i.e., data acquisition and preprocessing, machine learning model training and validation, results visualization, and process recovery. The problems, characteristics, and corresponding methods of each stage are listed in the developing route. Future challenges and research directions related to process monitoring are discussed in detail below.

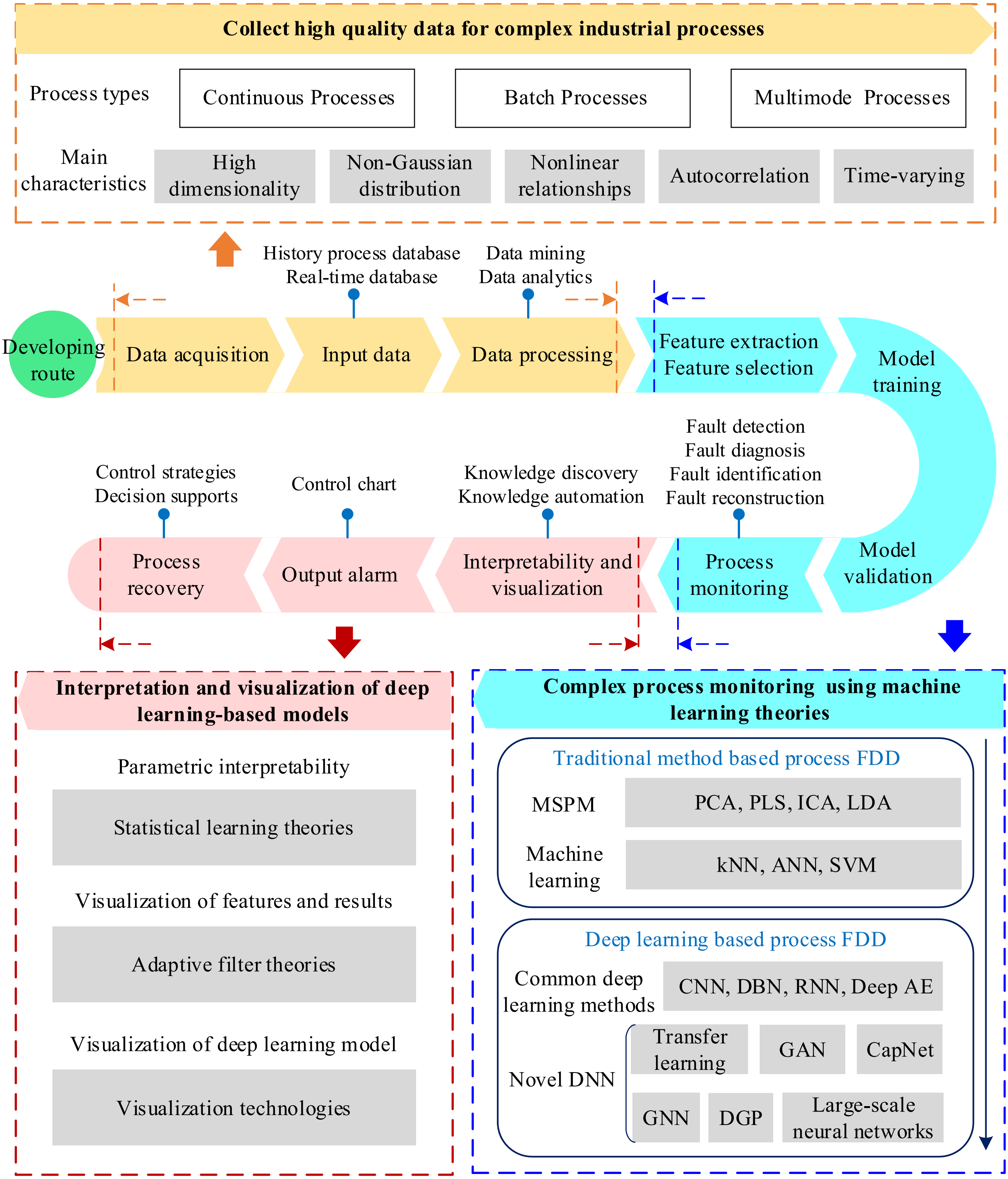


Fig. 26 Development route of applications of deep learning to process FDD

5.1 Collection of high-quality data in complex industrial processes

In modern industrial production, the process data are more numerous and complex than ever before. However, due to the interference of working environment or the abnormality of data acquisition equipment, the collected data are often uncertain and incomplete. Generally, deep learning requires a large amount of process data to be trained, and these inaccurate data will have a large negative impact on the related research of process monitoring. Continuous processes, batch processes, and multimode processes are three common types in industrial processes. Modern industrial processes always consist of various parts, and each of these parts may have a significant impact on process variables. As a result, these main characteristics are often involved in industrial process data: high dimensionality, non-Gaussian distribution, nonlinear relationships, autocorrelations, time varying, data autocorrelation, and multimode behaviors. These complex characteristics of the process also pose a great challenge to process monitoring. In order to ensure the quality of process data and model with the most appropriate data, the inspection and selection of data are a necessary step. Data preprocessing is also a critical step for data-based process monitoring. It can transform raw data into a more appropriate manner and can be effectively used for system modeling. Finally, a complete database should be constructed to manage these data and improve data quality for future process monitoring based on deep learning.

In many application scenarios, the uncertainty is very important, but the accuracy is not so important. DGP is a hierarchical generalization of Gaussian process. The model combines the uncertainty estimation of multi-layer model and has high scalability. It can model uncertainty and deal with missing data and abnormal data. Besides, GAN can be used to solve the problem of data missing and data imbalance in industrial processes.

5.2 Traditional methods-based process detection and diagnosis in complex environments

Over the past decades, the data projection-based and machine learning-based methods have been widely used in process monitoring. (1) These data projection-based methods, e.g., PCA, PLS, ICA, FDA, solve the problem of process monitoring (especially for fault detection) to some extent by dimension reduction or data projection. However, in the face of complex industrial environment, e.g., nonlinearity, autocorrelation, multimode, the fault detection results of these methods are often unsatisfactory. A common method is to modify or extend these projection-based methods to adapt to different process environments. In addition, based on the advantages of feature extraction of deep learning, these methods can be further combined with deep learning and applied to fault detection in complex environments. This strategy can also obtain better results of process fault detection. (2) Traditional machine learning methods, e.g., kNN, ANN, SVM, adopt a simpler network structure and are suitable for various classification tasks, and they have been widely used for process fault diagnosis. To further improve the performance of machine learning for process fault diagnosis in complex environments, these methods can be combined with MSPM methods. Firstly, dimension reduction and data projection are used to reduce the complexity of datasets. Secondly, machine learning techniques are used to model and perform process FDD tasks.

5.3 Deep learning-based process monitoring in complex environments

The traditional MSPM methods (e.g., PCA, PLS, LDA, LCA) directly use the raw data as model input. There is no effective way to deal with high-dimensional and noisy data, resulting in inefficient identification of process faults. In addition, it is difficult for traditional methods to model auto-correlation processes accurately, resulting in a large number of false alarms triggered by control charts. Due to the limited parameters and calculation units, the regular ANNs have limited representation learning. Thus, it is unable to extract effective features from process data.

Compared to the MSPM method, deep learning simplifies the preprocessing of raw data and directly inputs the raw data into the model. In the pre-training process, the model filters the input data by a deep network structure and multi-layer nonlinear transformation, reduces the dimension, and uses the extracted features for regression prediction or classification model construction. Compared with the shallow model, it has better convergence and optimization mechanism, and has a broad application prospect in the FDD of complex process. For example, CNN uses convolution layer to extract features automatically and processes high-dimensional data effectively by sharing convolution kernel. DBN does not need too much signal technology and diagnosis experience to support, and has relatively strong adaptability, versatility, the ability to deal with high-dimensional and nonlinear data. RNN is a neural network with recurrent hidden layers, which can effectively mine the sequencing information in the data.

5.3.1 Different process control tasks

Due to the powerful feature extraction ability, deep learning has been widely applied in process FDD. (1) The typical deep learning methods, e.g., AE, DBN, CNN, have achieved great success in process monitoring, which is discussed in Sect. 3. A great advantage of deep learning is the flexibility of its network, and users can change their network structure according to their demands to adapt to different process environments. For example, combining CNN with RNN can effectively process autocorrelation data. The attention mechanism can improve its ability of adaptive learning. (2) In recent years, transfer learning, GAN, and CapsNet have been applied to process FDD. Transfer learning imposes constraints on the parameters by minimizing the distance metric to distribution discrepancy, which can correct the serious cross-domain discrepancy. GAN uses an adversarial game mechanism to generate better samples from generators and discriminators. Considering the size and location of images, CapsNet can effectively locate and identify process faults. These three deep learning models can be further developed for future research in process FDD.

5.3.2 Optimization of deep network structure

The fault diagnosis method based on deep learning has the characteristics of large amount of training data and large amount of calculation, which is one of the bottlenecks for deep learning in online process FDD. At present, the main optimization methods of network model are as follows: (1) Combination of deep learning models and traditional shallow models to construct a hybrid model shows good performance in feature learning and classification [231, 232]. Through the organic combination of various methods to learn from each other's strong points and complement each other's strong points, the fault diagnosis and prediction of complex industrial system with high efficiency and high accuracy can be realized; (2) During and after training, different calculation strategies are adopted to adjust the network parameters [233–235]; (3) Network structure optimization. The computational complexity of various deep learning architectures is reduced by imposing structural constraints. On the premise of ensuring the accuracy, the computing speed of the model is improved, and the parallel operation, GPU acceleration, data parallel, and model parallel are realized.

5.3.3 Edge computing

The model of deep learning has a multi-layer structure, and the amount of calculation parameters increases exponentially with the increase in the number of network layers. The most outstanding performance is the number of parameters of convolution layer and full connection layer of DNN. In order to improve the training speed of DNN and realize online process monitoring, edge computing emerges as the times require. Edge computing is an open platform that integrates network, computing, storage, and other core capabilities on the side close to the object or data source. IOT devices are not suitable for running deep learning model, but edge computing-based DNN model can effectively solve this problem. In terms of timeliness, edge computing is faster than the traditional centralized cloud computing method, and it can also reduce the burden of cloud computing. Deep learning for edge computing can automatically extract high-level features of data and can also mine accurate information from sensor data. To sum up, edge computing and deep learning provide solutions for mining big data of IOT devices and can effectively improve process FDD.

5.3.4 The prospect of transfer learning and GAN in FDD

The existing large amount of data is used to provide solutions to the problem of insufficient data in industrial process monitoring. Because transfer learning does not need to be trained from the beginning, it can effectively reduce the training time. In addition, transfer learning only needs a small amount of target domain data, which can effectively solve the problem of less data in the process industry.

In the process monitoring model training, it is usually assumed that the data are labeled and balanced. However, data are often unlabeled and unbalanced in real process industry. To solve these problems, GAN can generate fault data according to specific probability distribution, which balances process dataset. Moreover, a GAN with unsupervised training can realize the training of unlabeled data and improve the accuracy of the FDD model.

5.3.5 Application prospect of GNN in FDD

In the face of massive monitoring data and huge distributed system, maintenance personnel need to locate faults and make rapid and accurate maintenance decisions, which requires intelligent FDD technology to improve the availability of the system. GNN can effectively describe and store all kinds of information of the process control system, including the attributes of each sensor node, the connection relationship between different fault nodes and the node importance. The intelligent process FDD system based on GNN can realize fault location and fault correlation analysis and improve the reliability and availability of process control system. At present, GNN has been widely used in community network, social network, recommendation system, and so on by virtue of its powerful information description and relationship expression. GNN is also suitable for process control systems to describe the interaction between faults, which will be a valuable exploration direction in the future.

5.4 Interpretability and visualization of deep learning models

In general, these DNN models are considered as a black box, and researchers cannot understand their internal operations. Although deep learning has obtained tremendous success in process FDD, it is not known how it learns from process data about FDD. (1) Statistical learning theory can better explain the operation mechanism of network through strict theoretical knowledge. For example, the composition of the network, related parameters, and monitoring results can be described in a statistical way. In addition, adaptive filter theory is useful for analyzing the physical significance of deep learning-based models. (2) At present, the methods commonly used in deep learning models are visualization techniques. For the features extracted from the network, t-SNE and other techniques can be used to project the features to two-dimensional or three-dimensional planes, which can help readers understand the distribution of features. In addition, the deep network model can also be visualized to understand its internal structure and operation mechanism. (3) Knowledge extraction of neural network. By combining symbolic rules with deep neuron model, knowledge is extracted from deep network, and the dynamic operation mechanism of deep network is described [236, 237]. Yu et al. [236] proposed the knowledge-based artificial neural network (KBANN) model and a genetic algorithm-based rule extraction approach (GARule) to discover the causal relationship between manufacturing parameters and product quality. These rules are applied to the diagnosis of manufacturing process, to provide guidance for improving product quality, and to construct knowledge-based neural network. Based on this study, Liu et al. [238, 239] explained the principle of neural network fault diagnosis by symbolic language, realized accurate prediction, and effectively extracted the key knowledge of manufacturing process.

6 Conclusions

In this study, the technology of process FDD in recent decades is reviewed. The development of process FDD is divided into three stages. In the past, the prevalent techniques of FDD are data projection-based methods and machine learning methods. Although these methods have achieved some successes in process FDD in the past few decades, they are difficult to extract effective features of complex processes. In recent years, deep learning has attracted extensive attentions and brought many constructive ideas and methods to process FDD. They learn and extract deep features of the process signals through multi-layer networks, which is conducive to detect and diagnose those faults in the process. The sample imbalance of training data sets in industrial processes poses a certain challenge to the development of deep learning in process FDD. At present, transfer learning and GAN overcome these problems and have been gradually applied to process FDD. Transfer learning and GAN will attract extensive attention and be widely used in process FDD in the near future. In addition, CapsNet, GNN, DGP, and DGMM also provide new ideas for solving process control problems. Finally, a developing route is provided and the future development and research plan of process FDD is discussed, which provides some valuable ideas for future research in process FDD.

Acknowledgements This research was supported by National Key Research and Development Program (No. 2022YFF0605702, 2022YFF0605704), the National Natural Science Foundation of China (92167107, 71771173), and Fundamental Research Funds for the Central Universities.

Data availability Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Conflict of interest None declared.

References

1. Wang J, Ma Y, Zhang L, Gao RX, Wu D (2018) Deep learning for smart manufacturing: methods and applications. J Manuf Syst 48:144–156
2. Ge Z, Song Z, Gao F (2013) Review of recent research on data-based process monitoring. Ind Eng Chem Res 52:3543–3562
3. Rathinasabapathy R, Elsass MJ, Josephson JR, Davis JF (2016) A smart manufacturing methodology for real time chemical process diagnosis using causal link assessment. Aiche J 62:3420–3431
4. Montmain J, Labreuche C, Imoussaten A, Trousset F (2015) Multi-criteria improvement of complex systems. Inform Sci 291:61–84
5. Butte VK, Tang LC (2010) Multivariate charting techniques: a review and a line-column approach. Qual Reliab Eng Int 26:443–451
6. Kang JH, Kim SB (2015) False alarm classification for multivariate manufacturing processes of thin film transistor-liquid crystal displays. J Process Contr 35:21–29
7. Peres FAP, Fogliatto FS (2018) Variable selection methods in multivariate statistical process control: a systematic literature review. Comput Ind Eng 115:603–619
8. Jiang QC, Huang B (2016) Distributed monitoring for large-scale processes based on multivariate statistical analysis and Bayesian method. J Process Contr 46:75–83
9. Bakshi BR (1998) Multiscale PCA with application to multivariate statistical process monitoring. Aiche J 44:1596–1610

10. Lee JM, Yoo CK, Lee IB (2004) Statistical process monitoring with independent component analysis. J Process Contr 14:467–485
11. Jiang QC, Yan XF, Lv ZM, Guo MJ (2014) Independent component analysis-based non-Gaussian process monitoring with preselecting optimal components and support vector data description. Int J Prod Res 52:3273–3286
12. Nomikos P, MacGregor JF (1995) Multi-way partial least squares in monitoring batch processes. Chemometr Intell Lab 30:97–108
13. He QP, Qin SJ, Wang J (2005) A new fault diagnosis method using fault directions in fisher discriminant analysis. Aiche J 51:555–571
14. Ding SX, Zhang P, Naik A, Ding EL, Huang B (2009) Subspace method aided data-driven design of fault detection and isolation systems. J Process Contr 19:1496–1510
15. Dong D, McAvoy TJ (1996) Nonlinear principal component analysis - Based on principal curves and neural networks. Comput Chem Eng 20:65–78
16. Lee JM, Yoo CK, Choi SW, Vanrolleghem PA, Lee IB (2004) Nonlinear process monitoring using kernel principal component analysis. Chem Eng Sci 59:223–234
17. Xi Z, Weiwu Y, Xu Z, Huihe S (2007) Nonlinear biological batch process monitoring and fault identification based on kernel fisher discriminant analysis. Process Biochem 42:1200–1210
18. Deng XG, Tian XM, Chen S, Harris CJ (2017) Fault discriminant enhanced kernel principal component analysis incorporating prior fault information for monitoring nonlinear processes. Chemometr Intell Lab 162:21–34
19. Chiang LH, Russell EL, Braatz RD (2000) Fault diagnosis in chemical processes using Fisher discriminant analysis, discriminant partial least squares, and principal component analysis. Chemometr Intell Lab 50:243–252
20. Ding SX, Zhang P, Jeinsch T, Ding EL, Gui W (2011) A Survey of the Application of Basic Data-Driven and Model-Based Methods in Process Monitoring and Fault Diagnosis. IFAC Proc 44:12380–12388
21. Yin S, Ding SX, Haghani A, Hao HY, Zhang P (2012) A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process. J Process Contr 22:1567–1581
22. Yin S, Ding SX, Xie XC, Luo H (2014) A review on basic data-driven approaches for industrial process monitoring. Ieee T Ind Electron 61:6418–6428
23. He XF, Cai D, Shao YL, Bao HJ, Han JW (2011) Laplacian regularized Gaussian mixture model for data clustering. IEEE Trans Knowl Data En 23:1406–1418
24. Tenenbaum JB, de Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290:2319–2323
25. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290:2323–2326
26. Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput 15:1373–1396
27. Zhang ZY, Zha HY (2004) Principal manifolds and nonlinear dimensionality reduction via tangent space alignment, Siam. J Sci Comput 26:313–338
28. Peng X, Tang Y, Du WL, Qian F (2017) Multimode Process Monitoring and Fault Detection: A Sparse Modeling and Dictionary Learning Method. IEEE Trans Ind Electron 64:4866–4875
29. Zhou YJ, Xu K, He F, He D (2020) Nonlinear fault detection for batch processes via improved chordal kernel tensor locality preserving projections. Control Eng Pract 101:1–14
30. Yu JB (2010) Hidden Markov models combining local and global information for nonlinear and multimodal process monitoring. J Process Contr 20:344–359
31. Yu JB (2012) Local and global principal component analysis for process monitoring. J Process Contr 22:1358–1373
32. Yu JB (2016) Process monitoring through manifold regularization-based GMM with global/local information. J Process Contr 45:84–99
33. Li N, Yan WW, Yang YP (2015) Spatial-statistical local approach for improved manifold-based process monitoring. Ind Eng Chem Res 54:8509–8519
34. Lunga D, Prasad S, Crawford MM, Ersoy O (2014) Manifold-learning-based feature extraction for classification of hyperspectral data. Ieee Signal Proc Mag 31:55–66
35. Tong CD, Shi XH, Lan T (2016) Statistical process monitoring based on orthogonal multi-manifold projections and a novel variable contribution analysis. ISA Trans 65:407–417
36. Yang J, Zhang MS, Shi HB, Tan S (2017) Dynamic learning on the manifold with constrained time information and its application for dynamic process monitoring. Chemometr Intell Lab 167:179–189
37. Sharifian S, Sotudeh-Gharebagh R, Zarghami R, Tanguy P, Mostoufi N (2021) Uncertainty in chemical process systems engineering: a critical review. Rev Chem Eng 37:687–714
38. Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA (2010) Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 11:3371–3408
39. Krizhevsky A, Sutskever I, Hinton GE (2017) ImageNet classification with deep convolutional neural networks. Commun Acm 60:84–90
40. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313:504–507
41. Lipton ZC, Berkowitz J, Elkan C (2015) A critical review of recurrent neural networks for sequence learning. Comput Sci 1:1–38
42. Boroumand M, Chen M, Fridrich J (2019) Deep residual network for steganalysis of digital images. IEEE Trans Inf Foren Sec 14:1181–1193
43. Das A, Maiti J, Banerjee RN (2012) Process monitoring and fault detection strategies: a review. Int J Qual Reliabil Manag 29:720–752
44. Qin SJ (2012) Survey on data-driven industrial process monitoring and diagnosis. Annu Rev Control 36:220–234
45. Ge ZQ, Song ZH, Deng SX, Huang B (2017) Data mining and analytics in the process industry: the role of machine learning. IEEE Access 5:20590–20616
46. Nor NM, Hassan CRC, Hussain MA (2020) A review of data-driven fault detection and diagnosis methods: applications in chemical process systems. Rev Chem Eng 36:513–553
47. Taqvi SAA, Zabiri H, Tufa LD, Uddin F, Fatima SA, Maulud AS (2021) A review on data-driven learning approaches for fault detection and diagnosis in chemical processes. Chembioeng Rev 8:239–259
48. Lei YG, Yang B, Jiang XW, Jia F, Li NP, Nandi AK (2020) Applications of machine learning to machine fault diagnosis: a review and roadmap. Mech Syst Signal Pr 138:1–39
49. Onel M, Kieslich CA, Guzman YA, Pistikopoulos EN (2018) Simultaneous fault detection and identification in continuous processes via nonlinear support vector machine based feature selection. Comput Aided Chem Eng 44:2077–2082
50. Yao Y, Gao FR (2009) A survey on multistage/multiphase statistical modeling methods for batch processes. Annu Rev Control 33:172–183

51. Tong CD, Palazoglu A, Yan XF (2013) An adaptive multimode process monitoring strategy based on mode clustering and mode unfolding. J Process Contr 23:1497–1507
52. Wang F, Zhu HL, Tan S, Shi HB (2016) Orthogonal nonnegative matrix factorization based local hidden Markov model for multimode process monitoring. Chin J Chem Eng 24:856–860
53. Afzal MS, Tan W, Chen TW (2017) Process monitoring for multimodal processes with mode-reachability constraints. IEEE Trans Ind Electron 64:4325–4335
54. Lou Z, Wang Y (2017) Hidden semi-Markov model based monitoring algorithm for multimode processes. In: 2017 6th Data driven control and learning systems (DDCLS), pp 66–71.
55. Yang CM, Hou J (2016) Fed-batch fermentation penicillin process fault diagnosis and detection based on support vector machine. Neurocomputing 190:117–123
56. Yu J, Qin SJ (2008) Multimode process monitoring with Bayesian inference-based finite Gaussian mixture models. Aiche J 54:1811–1829
57. He QP, Wang J (2007) Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes. IEEE Trans Semicond Manuf 20:345–354
58. Sun K, Liu JL, Kang JL, Jang SS, Wong DSH, Chen DS (2014) Development of a variable selection method for soft sensor using artificial neural network and nonnegative garrote. J Process Contr 24:1068–1075
59. Pani AK, Mohanta HK (2015) Online monitoring and control of particle size in the grinding process using least square support vector regression and resilient back propagation neural network. Isa T 56:206–221
60. Liu TI, Jolley B (2015) Tool condition monitoring (TCM) using neural networks. Int J Adv Manuf Tech 78:1999–2007
61. Pian JX, Zhu YL (2015) A hybrid soft sensor for measuring hot-rolled strip temperature in the laminar cooling process. Neurocomputing 169:457–465
62. Du SC, Xi LF (2011) Fault diagnosis in assembly processes based on engineering-driven rules and PSOSAEN algorithm. Comput Ind Eng 60:77–88
63. Majid N, Young BR, Taylor MP, Chen J (2012) K-means clustering pre-analysis for fault diagnosis in an aluminium smelting process. In: 2012 4th Conference on Data Mining and Optimization (DMO), 2012, pp 43–46
64. Ku WF, Storer RH, Georgakis C (1995) Disturbance detection and isolation by dynamic principal component analysis. Chemometr Intell Lab 30:179–196
65. Luo RF, Misra M, Himmelblau DM (1999) Sensor fault detection via multiscale analysis and dynamic PCA. Ind Eng Chem Res 38:1489–1495
66. Russell EL, Chiang LH, Braatz RD (2000) Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis. Chemometr Intell Lab 51:81–93
67. Mina J, Verde C (2005) Fault detection using dynamic principal component analysis by average estimation. In: 2005 2nd International conference on electrical and electronics engineering, 2005, pp 374–377
68. Choi SW, Lee IB (2005) Multiblock PLS-based localized process diagnosis. J Process Contr 15:295–306
69. Ding SX, Yin S, Peng KX, Hao HY, Shen B (2013) A novel scheme for key performance indicator prediction and diagnosis with application to an industrial hot strip mill. IEEE Trans Ind Inform 9:2239–2247
70. Shen Y, Wei Z, Gao H, Peng K (2012) Data-driven quality related prediction and monitoring. Conference of the IEEE Industrial Electronics Society 2012, pp 3874–3879
71. Peng K, Zhang K, You B, Dong JJCT, Iet A (2015) Quality-relevant fault monitoring based on efficient projection to latent structures with application to hot strip mill process 9:1135–1145
72. He QP, Wang J (2010) Large-scale semiconductor process fault detection using a fast pattern recognition-based method. IEEE Trans Semicond Manuf 23:194–200
73. Guo XP, Jie Y, Yuan L (2013) KPCS-kNN based fault detection for batch processes. In: International conference on machine learning and cybernetics (ICMLC), pp 698–703
74. Li Y, Zhang XM (2014) Diffusion maps based k-nearest-neighbor rule technique for semiconductor manufacturing process fault detection. Chemometr Intell Lab 136:47–57
75. Zhang C, Gao XW, Li Y, Feng LW (2019) Fault detection strategy based on weighted distance of k nearest neighbors for semiconductor manufacturing processes. IEEE Trans Semiconduct Manuf 32:75–81
76. Chen Y, Du R (1998) An improved artificial neural network method for monitoring and diagnosis of engineering processes with applications. J Vib Control 4:635–650
77. Iliyas SA, Elshafei M, Habib MA, Adeniran AA (2013) RBF neural network inferential sensor for process emission monitoring. Control Eng Pract 21:962–970
78. Yu JB, Xi LF, Zhou XJ (2009) Identifying source(s) of out-of-control signals in multivariate manufacturing processes using selective neural network ensemble. Eng Appl Artif Intel 22:141–152
79. Yu JB, Xi LF (2007) A Neural Network ensemble for classifying source(s) in multivariate manufacturing processes. In: IEEE international conference on industrial engineering and engineering management, pp 1246–1250
80. Maleki MR, Sahraeian R (2015) Online monitoring and fault diagnosis of multivariate-attribute process mean using neural networks and discriminant analysis technique. Int J Eng 28:1634–1643
81. Liu X, Kang L, Li S, Fei M (2012) Input selection for dynamic RBF models in process monitoring. In: Proceedings of the 10th world congress on intelligent control and automation, pp 3037–3042
82. Bo C, Li J, Lu A, Zhuang G (2007) Nonlinear process monitors method based on kernel function and PNN. Chinese control conference, pp 511–515
83. Wu SX (2012) Simultaneous process mean and variance monitoring using wavelet transform and probabilistic neural network. Appl Mech Mater 157–158:11–15
84. Simon L, Karim MN (2001) Probabilistic neural networks using Bayesian decision strategies and a modified Gompertz model for growth phase classification in the batch culture of Bacillus subtilis. Biochem Eng J 7:41–48
85. Peng Y, Chen XG, Ye QX, Jiao JB (2014) Fault detection and classification in chemical processes using NMFSC and structural SVMs. Can J Chem Eng 92:1016–1023
86. Yin Z, Hou J (2016) Recent advances on SVM based fault diagnosis and process monitoring in complicated industrial processes. Neurocomputing 174:643–650
87. Onel M, Kieslich CA, Guzman YA, Floudas CA, Pistikopoulos EN (2018) Big data approach to batch process monitoring: simultaneous fault detection and diagnosis using nonlinear support vector machine-based feature selection(Reprinted from Computers and Chemical Engineering, vol 115, pg 46–63. Comput Chem Eng 116(2018):503–520
88. Yu J (2011) Localized fisher discriminant analysis based complex chemical process monitoring. Aiche J 57:1817–1828
89. Yu J (2011) Nonlinear bioprocess monitoring using multiway kernel localized Fisher discriminant analysis. Ind Eng Chem Res 50:3390–3402

90. Zhao Z, Zhang J, Sun YG, Tian HX (2016) Fault detection and diagnosis method for batch process based on ELM-based fault feature phase identification. Neural Comput Appl 27:167–173
91. Nor NM, Hussain MA, Hassan CRC (2017) Fault diagnosis and classification framework using multiscale classification based on kernel fisher discriminant analysis for chemical process system. Appl Soft Comput 61:959–972
92. Tang Q, Chai Y, Qu JF, Fang XY (2019) Industrial process monitoring based on Fisher discriminant global-local preserving projection. J Process Contr 81:76–86
93. Yang XH, Rui SH, Zhang XL, Xu SP, Yang CS, Liu PX (2019) Fault diagnosis in chemical processes based on class-incremental FDA and PCA. IEEE Access 7:18164–18171
94. Nor NM, Hussain MA, Hassan CRC (2020) Multi-scale kernel Fisher discriminant analysis with adaptive neuro-fuzzy inference system (ANFIS) in fault detection and diagnosis framework for chemical process systems. Neural Comput Appl 32:9283–9297
95. Ren SJ, Song ZH, Yang MY, Ren JG (2015) A novel multimode process monitoring method integrating LCGMM with modified LFDA. Chin J Chem Eng 23:1970–1980
96. Nomikos P, Macgregor JF (1994) Monitoring batch processes using multiway principal component analysis. Aiche J 40:1361–1375
97. Dong D, McAvoy TJ (1996) Batch tracking via nonlinear principal component analysis. Aiche J 42:2199–2208
98. Macgregor JF, Kourti T (1995) Statistical process-control of multivariate processes. Control Eng Pract 3:403–414
99. Dunia R, Qin SJ (1998) Joint diagnosis of process and sensor faults using principal component analysis. Control Eng Pract 6:457–469
100. Raich A, Cinar A (1996) Statistical process monitoring and disturbance diagnosis in multivariable continuous processes. Aiche J 42:995–1009
101. Mujica LE, Vehi J, Ruiz M, Verleysen M, Staszewski W, Worden K (2008) Multivariate statistics process control for dimensionality reduction in structural assessment. Mech Syst Signal Pr 22:155–171
102. Ivosev G, Burton L, Bonner R (2008) Dimensionality reduction and visualization in principal component analysis. Anal Chem 80:4933–4944
103. Dunia R, Edgar TF, Nixon M (2013) Process monitoring using principal components in parallel coordinates. Aiche J 59:445–456
104. Wang RC, Edgar TF, Baldea M, Nixon M, Wojsznis W, Dunia R (2015) Process fault detection using time-explicit Kiviat diagrams. Aiche J 61:4277–4293
105. Jiang QC, Yan XF (2019) Quality-driven kernel projection to latent structure model for nonlinear process monitoring. IEEE Access 7:74450–74458
106. Li Z, Yan Z, Jiang L (2009) Fault detection and diagnosis based on KPCA-LSSVM Model. Int Conf Measur Technol Mechat Autom 2009:634–638
107. Xu J, Hu S, Shen Z (2010) Fault detection and diagnosis of chemical process based on an improved multi-scale KPCA. Chin J Sci Instrum 31:51–55
108. Lee JM, Qin SJ, Lee IB (2006) Fault detection and diagnosis based on modified independent component analysis. Aiche J 52:3501–3514
109. Zhang YW, Zhang Y (2010) Fault detection of non-Gaussian processes based on modified independent component analysis. Chem Eng Sci 65:4630–4639
110. Hsu CC, Chen MC, Chen LS (2010) Integrating independent component analysis and support vector machine for multivariate process monitoring. Comput Ind Eng 59:145–156
111. Zhang X, Wang X, Fan Y, Wu J (2014) Fault diagnosis of industrial process based on KICA and LSSVM. In: The 26th Chinese control and decision conference, pp 3802–3807
112. Li S, Zhou XF, Pan FC, Shi HB, Li KT, Wang ZW (2017) Correlated and weakly correlated fault detection based on variable division and ICA. Comput Ind Eng 112:320–335
113. Wang B, Yan XF, Jiang QC (2016) Independent component analysis model utilizing de-mixing information for improved non-Gaussian process monitoring. Comput Ind Eng 94:188–200
114. Yu JB (2011) Fault detection using principal components-based gaussian mixture model for semiconductor manufacturing processes. IEEE Trans Semicond Manuf 24:432–444
115. Yu JB (2012) Semiconductor manufacturing process monitoring using Gaussian Mixture model and Bayesian method with local and nonlocal information. Ieee T Semiconduct M 25:480–493
116. Yu JB (2011) Pattern recognition of manufacturing process signals using Gaussian mixture models-based recognition systems. Comput Ind Eng 61:881–890
117. Jie Y (2012) A nonlinear kernel Gaussian mixture model based inferential monitoring approach for fault detection and diagnosis of chemical processes. Chem Eng Sci 68:506–519
118. Peng P, Zhao J, Zhang Y, Zhang H (2019) Hidden Markov model combined with kernel principal component analysis for nonlinear multimode process fault detection. In: 2019 IEEE 15th International conference on automation science and engineering (CASE), pp 1586–1591
119. Ge ZQ, Xie L, Kruger U, Lamont L, Song ZH, Wang SQ (2009) Sensor fault identification and isolation for multivariate non-Gaussian processes. J Process Contr 19:1707–1715
120. Yao M, Wang HG, Xu WL (2014) Batch process monitoring based on functional data analysis and support vector data description. J Process Contr 24:1085–1097
121. Chen MC, Hsu CC, Malhotra B, Tiwari MK (2016) An efficient ICA-DW-SVDD fault detection and diagnosis method for non-Gaussian processes. Int J Prod Res 54:5208–5218
122. Lv ZM, Yan XF, Jiang QC (2014) Batch process monitoring based on just-in-time learning and multiple-subspace principal component analysis. Chemometr Intell Lab 137:128–139
123. Zhou J, Guo AH, Celler B, Su S (2014) Fault detection and identification spanning multiple processes by integrating PCA with neural network. Appl Soft Comput 14:4–11
124. Yu JB, Xi LF (2008) Using an MQE chart based on a self-organizing map NN to monitor out-of-control signals in manufacturing processes. Int J Prod Res 46:5907–5933
125. Corona F, Mulas M, Baratti R, Romagnoli JA (2010) On the topological modeling and analysis of industrial process data using the SOM. Comput Chem Eng 34:2022–2032
126. Yu HY, Khan F, Garaniya V, Ahmad A (2014) Self-organizing map based fault diagnosis technique for non-Gaussian processes. Ind Eng Chem Res 53:8831–8843
127. Song Y, Jiang QC, Yan XF, Guo MJ (2014) A multi-SOM with canonical variate analysis for chemical process monitoring and fault diagnosis. J Chem Eng Jpn 47:40–51
128. Chen XY, Yan XF (2012) Using improved self-organizing map for fault diagnosis in chemical industry process. Chem Eng Res Des 90:2262–2277
129. Bolon-Canedo V, Sanchez-Marono N, Alonso-Betanzos A (2013) A review of feature selection methods on synthetic data. Knowl Inf Syst 34:483–519
130. Geladi P, Kowalski BR (1986) Partial least-squares regression—a tutorial. Anal Chim Acta 185:1–17
131. Höskuldsson A (1988) PLS regression methods. J Chemometr 2:211–228
132. Dejong S (1993) Simpls—an alternative approach to partial least-squares regression. Chemometr Intell Lab 18:251–263

133. Zhou DH, Li G, Qin SJ (2010) Total projection to latent structures for process monitoring. Aiche J 56:168–178
134. Qin SJ, Zheng YY (2013) Quality-relevant and process-relevant fault monitoring with concurrent projection to latent structures. Aiche J 59:496–504
135. Das A, Maiti J, Banerjee RN (2010) A hierarchical process monitoring strategy for a serial multi-stage manufacturing system. Int J Prod Res 48:2459–2479
136. Cover TM, Hart PE (1953) Nearest neighbor pattern classification. IEEE Trans Inf Theory 13:21–27
137. Ghosh A, Wang GN, Lee J (2020) A novel automata and neural network based fault diagnosis system for PLC controlled manufacturing systems. Comput Ind Eng 139:1
138. Taqvi SA, Tufa LD, Zabiri H, Maulud AS, Uddin F (2020) Fault detection in distillation column using NARX neural network. Neural Comput Appl 32:3503–3519
139. Bienkowski M, Gasieniec L, Klonowski M, Korzeniowski M, Mans B, Schmid S, Wattenhofer R (2005) Handbook of neural network signal processing. Ieee T Neural Networ 16:780–780
140. Meireles MRG, Almeida PEM, Simoes MG (2003) A comprehensive review for industrial applicability of artificial neural networks. Ieee T Ind Electron 50:585–601
141. Specht DF (1990) Probabilistic neural networks. Neural Netw 3:109–118
142. Shin HJ, Eom DH, Kim SS (2005) One-class support vector machines - an application in machine fault detection and classification. Comput Ind Eng 48:395–408
143. Ma MD, Wong DSH, Jang SS, Tseng ST (2010) Fault detection based on statistical multivariate analysis and microarray visualization. IEEE Trans Ind Inform 6:18–24
144. MacGregor JF, Marlin TE, Kresta J, Skagerberg B (1991) Multivariate statistical methods in process analysis and control. Chemical process control 1:79–100
145. Qin SJ, Mcavoy TJ (1992) Nonlinear Pls modeling using neural networks. Comput Chem Eng 16:379–391
146. Chaouch H, Charfedine S, Ouni K, Jerbi H, Nabli L (2019) Intelligent supervision approach based on multilayer neural PCA and nonlinear gain scheduling. Neural Comput Appl 31:1153–1163
147. Qiang G, Wang G, Hao X (2012) A hybrid fault detection and diagnosis system based on KPCA and DDAG. Adv Intell Soft Comput 125:549–555
148. Hsu CC, Chen LS, Liu CH (2010) A process monitoring scheme based on independent component analysis and adjusted outliers. Int J Prod Res 48:1727–1743
149. Cho HW (2009) Data description and noise filtering based detection with its application and performance comparison. Expert Syst Appl 36:434–441
150. Schmidhuber J (2015) Deep learning in neural networks: An overview. Neural Netw 61:85–117
151. Abid A, Khan MT, Iqbal J (2021) A review on fault detection and diagnosis techniques: basics and beyond. Artif Intell Rev 54:3639–3664
152. Issa NT, Byers SW, Dakshanamurthy S (2014) Big data: the next frontier for innovation in therapeutics and healthcare. Expert Rev Clin Phar 7:293–298
153. Hoyt D (2015) Big data: a revolution that will transform how we live, work, and think. Res Technol Manag 58:66–67
154. Lavasani MS, Ardali NR, Sotudeh-Gharebagh R, Zarghami R, Abonyi J, Mostoufi N (2021) Big data analytics opportunities for applications in process engineering. Rev Chem Eng 1:1
155. Shu YD, Ming L, Cheng FF, Zhang ZP, Zhao JS (2016) Abnormal situation management: challenges and opportunities in the big data era. Comput Chem Eng 91:104–113
156. Yao L, Ge ZQ (2021) Industrial big data modeling and monitoring framework for plant-wide processes. IEEE T Ind Inform 17:6399–6408
157. Jiang QC, Yan SF, Cheng H, Yan XF (2021) Local-global modeling and distributed computing framework for nonlinear plant-wide process monitoring with industrial big data. IEEE Trans Neur Net Lear 32:3355–3365
158. Guo JY, Wang X, Li Y (2019) Fault detection based on improved local entropy locality preserving projections in multimodal processes. J Chemometr 33:1–18
159. Geng ZQ, Duan XY, Han YM, Liu FF, Xu W (2021) Novel variation mode decomposition integrated adaptive sparse principal component analysis and it application in fault diagnosis. ISA Trans 1:1–11
160. Alrifaey M, Lim WH, Ang CK, Natarajan E, Solihin MI, Juhari MRM, Tiang SS (2022) Hybrid deep learning model for fault detection and classification of grid-connected photovoltaic system. IEEE Access 10:13852–13869
161. Xu S, Lu B, Baldea M, Edgar TF, Wojsznis W, Blevins T, Nixon M (2015) Data cleaning in the process industries. Rev Chem Eng 31:453–490
162. Allison PD (2012) Handling missing data by maximum likelihood. SAS Glob Forum Stat Data Anal 1:1–21
163. Ding XO, Wang HZ, Su JX, Li ZJ, Li JZ, Gao H (2019) Cleanits: a data cleaning system for industrial time series. Proc Vldb Endow 12:1786–1789
164. Wang X, Wang C (2020) Time series data cleaning: a survey. IEEE Access 8:1866–1881
165. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323:533–536
166. Chadha GS, Schwung A (2017) Comparison of deep neural network architectures for fault detection in tennessee eastman process. In: IEEE international conference on emerging technologies & factory automation, 2017, pp 1–8
167. Yan SF, Yan XF (2019) Design teacher and supervised dual stacked auto-encoders for quality-relevant fault detection in industrial process. Appl Soft Comput 81:1
168. Yu JB, Zheng XY, Wang SJ (2019) A deep autoencoder feature learning method for process pattern recognition. J Process Contr 79:1–15
169. Yu JB, Zheng XY, Wang SJ (2019) Stacked denoising autoencoder-based feature learning for out-of-control source recognition in multivariate manufacturing process. Qual Reliab Eng Int 35:204–223
170. Zhang CY, Yu JB, Wang SJ (2021) Fault detection and recognition of multivariate process based on feature learning of one-dimensional convolutional neural network and stacked denoised autoencoder. Int J Prod Res 59:2426–2449
171. Li ZC, Tian L, Jiang QC, Yan XF (2021) Distributed-ensemble stacked autoencoder model for nonlinear process monitoring. Inform Sciences 542:302–316
172. Liu X, Yu J, Ye L (2021) Residual attention convolutional autoencoder for feature learning and fault detection in nonlinear industrial processes. Neural Comput Appl 33:3085–3104
173. Yu JB, Liu X (2021) One-dimensional residual convolutional auto-encoder for fault detection in complex industrial processes. Int J Prod Res 1:1–20
174. Li SP, Luo JX, Hu YM (2022) Toward interpretable process monitoring: slow feature analysis-aided autoencoder for spatiotemporal process feature learning. IEEE Trans Instrum Meas 71:1–11
175. Yu JB, Zhang CY (2020) Manifold regularized stacked autoencoders-based feature learning for fault detection in industrial processes. J Process Contr 92:119–136

176. Gu JX, Wang ZH, Kuen J, Ma LY, Shahroudy A, Shuai B, Liu T, Wang XX, Wang G, Cai JF, Chen T (2018) Recent advances in convolutional neural networks. Pattern Recogn 77:354–377
177. Zhang HL, Wang P, Gao XJ, Qi YS, Gao HH (2019) Amplitude-frequency images-based ConvNet: applications of fault detection and diagnosis in chemical processes. J Chemometr 33:1–18
178. Kim E, Cho S, Lee B, Cho M (2019) Fault detection and diagnosis using self-attentive convolutional neural networks for variable-length sensor data in semiconductor manufacturing. IEEE Trans Semiconduct Manuf 32:302–309
179. Chen S, Yu J (2020) Convolution neural network multivariable process feature learning and fault diagnosis. J Harb Inst Technol 52:1
180. Lee KB, Cheon S, Kim CO (2017) A convolutional neural network for fault classification and diagnosis in semiconductor manufacturing processes. IEEE Trans Semicond Manuf 30:135–142
181. Wu H, Zhao JS (2018) Deep convolutional neural network model based chemical process fault diagnosis. Comput Chem Eng 115:185–197
182. Chen S, Yu J, Wang S (2020) One-dimensional convolutional auto-encoder-based feature learning for fault diagnosis of multivariate processes. J Process Contr 87:54–67
183. Zheng X, Yu J (2019) Multivariate process monitoring and fault identification using convolutional neural networks. In: 24th International conference on industrial engineering and engineering management, proceeding of the 24th international conference on industrial engineering and engineering management 2018, 2019, pp 197–208
184. Yu J, Liu X (2020) A fault detection method based on convolutional gated recurrent unit auto-encoder for tennessee eastman process. Chin Autom Cong (CAC) 2020:1234–1238
185. Hsu CY, Liu WC (2021) Multiple time-series convolutional neural network for fault detection and diagnosis and empirical study in semiconductor manufacturing. J Intell Manuf 32:823–836
186. Yu WK, Zhao CH (2020) Broad convolutional neural network based industrial process fault diagnosis with incremental learning capability. IEEE Trans Ind Electron 67:5081–5091
187. Yu JB, Zhang CY, Wang SJ (2021) Multichannel one-dimensional convolutional neural network-based feature learning for fault diagnosis of industrial processes. Neural Comput Appl 33:3085–3104
188. Bengio Y, Simard P, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Networ 5:157–166
189. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9:1735–1780
190. Chadha GS, Panambilly A, Schwung A, Ding SX (2020) Bidirectional deep recurrent neural networks for process fault classification. Isa T 106:330–342
191. Cheng FF, He EE, Zhao JS (2019) A novel process monitoring approach based on variational recurrent autoencoder. Comput Chem Eng 129:1–14
192. Ouyang H, Zeng JS, Li YF, Luo SH (2020) Fault Detection and Identification of Blast Furnace Ironmaking Process Using the Gated Recurrent Unit Network. Processes 8:1–15
193. Wang Q, Luo L (2021) Sequential fault detection and classification in chemical process using a deep convolutional encoder–decoder architecture based on system dynamics. IOP Conf Ser Earth Environ Sci 696:1–7
194. Chen SM, Yu JB (2019) Deep recurrent neural network-based residual control chart for autocorrelated processes. Qual Reliab Eng Int 35:2687–2708
195. Zhang SY, Bi KX, Qiu T (2020) Bidirectional recurrent neural network-based chemical process fault diagnosis. Ind Eng Chem Res 59:824–834
196. Liu F, Wang P, Cai Z, Zhou Z, Yang Z (2019) Batch process fault diagnosis based on the combination of deep belief network and long short-term memory network. In: 2019 CAA symposium on fault detection, supervision and safety for technical processes (SAFEPROCESS), pp 208-214
197. Yu J, Liu X, Ye L (2020) Convolutional long short-term memory autoencoder-based feature learning for fault detection in industrial processes. IEEE Trans Instrum Measur 70:3505615
198. Hinton GE (2002) Training products of experts by minimizing contrastive divergence. Neural Comput 14:1771–1800
199. Zhang ZP, Zhao JS (2017) A deep belief network based fault diagnosis model for complex chemical processes. Comput Chem Eng 107:395–407
200. Peng T, Peng K, Kai Z, Chen Z, Yang X, Li L (2018) A deep belief network-based fault detection method for nonlinear processes. IFAC-PapersOnLine 51:9–14
201. Kim JK, Lee JS, Han YS (2019) Fault detection prediction using a deep belief network-based multi-classifier in the semiconductor manufacturing process. Int J Softw Eng Know 29:1125–1139
202. Yu JB, Yan XF (2019) Active features extracted by deep belief network for process monitoring. Isa T 84:247–261
203. Yu JB, Yan XF (2020) Whole process monitoring based on unstable neuron output information in hidden layers of deep belief network. IEEE Trans Cybern 50:3998–4007
204. Wang YL, Pan ZF, Yuan XF, Yang CH, Gui WH (2020) A novel deep learning based fault diagnosis approach for chemical process with extended deep belief network. ISA Trans 96:457–467
205. Yu JB, Yan XF (2020) Modeling large-scale industrial processes by multiple deep belief networks with lower-pressure and higher-precision for status monitoring. IEEE Access 8:20439–20448
206. Zhang LW, Lin J, Liu B, Zhang ZC, Yan XH, Wei MH (2019) A Review on deep learning applications in prognostics and health management. IEEE Access 7:162415–162438
207. Zhu JL, Yao Y, Gao FR (2018) Transfer of qualitative and quantitative knowledge for similar batch process monitoring. Ieee Access 6:73856–73870
208. Wu H, Zhao JS (2020) Fault detection and diagnosis based on transfer learning for multimode chemical processes. Comput Chem Eng 135:1–13
209. Chai Z, Zhao CH, Huang B (2021) Multisource-refined transfer network for industrial fault diagnosis under domain and category inconsistencies. IEEE Trans Cybern 1–13
210. Gao X, Deng F, Yue XH (2020) Data augmentation in fault diagnosis based on the Wasserstein generative adversarial network with gradient penalty. Neurocomputing 396:487–494
211. Yang X, Feng D (2019) Generative adversarial network based anomaly detection on the benchmark tennessee eastman process. In: 2019 5th International conference on control, automation and robotics (ICCAR), pp 644–648
212. Huang YF, Chen X, Xu L, Li KY (2021) Single image desmoking via attentive generative adversarial network for smoke detection process. Fire Technol 57:3021–3040
213. Yan K (2021) Chiller fault detection and diagnosis with anomaly detective generative adversarial network. Build Environ 201:1–9
214. Feng L, Zhao C, Li YL, Zhou M, Fu C (2020) Multichannel diffusion graph convolutional network for the prediction of endpoint composition in the converter steelmaking process. IEEE Trans Instrum Measur 1:1

215. Zhao MH, Zhong SS, Fu XY, Tang BP, Pecht M (2020) Deep Residual Shrinkage Networks for Fault Diagnosis, Ieee T Ind. Inform 16:4681–4690
216. Li S, Li W, Cook C, Zhu C, Gao Y (2018) Independently recurrent neural network (IndRNN): building a longer and deeper rnN. IEEE/CVF Conference on Computer Vision and Pattern Recognition 2018:5457–5467
217. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L (2017) Attention is All You Need. Comput Lang 1:1–15
218. Xu GW, Liu M, Jiang ZF, Shen WM, Huang CX (2020) Online Fault Diagnosis Method Based on Transfer Convolutional Neural Networks. Ieee T Instrum Meas 69:509–520
219. Wang ZJ, Zheng LK, Du WH, Cai WN, Zhou J, Wang JT, Han XF, He GF (2019) A novel method for intelligent fault diagnosis of bearing based on capsule neural network. Complexity 2019:1–17
220. Chen TY, Wang ZH, Yang X, Jiang K (2019) A deep capsule neural network with stochastic delta rule for bearing fault diagnosis on raw vibration signals. Measurement 148:1–15
221. Chen LL, Qin N, Dai X, Huang DQ (2020) Fault diagnosis of high-speed train bogie based on capsule network. IEEE Trans Instrum Meas 69:6203–6211
222. Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: International conference on learning representations, pp 1–14
223. Henaff M, Bruna J, Lecun Y (2015) Deep convolutional networks on graph-structured data. Comput Sci 1–10
224. Damianou AC, Lawrence ND (2012) Deep Gaussian processes. Comput Sci 207–215
225. Havasi M, Hernández-Lobato J, Murillo-Fuentes JJ (2018) Inference in deep Gaussian processes using stochastic gradient hamiltonian Monte Carlo. arXiv e-prints 7517–7527
226. Santiago TC, Melissa DL, Oscar P, Fabio GA (2020) Hybrid deep learning gaussian process for diabetic retinopathy diagnosis and uncertainty quantification. Ophthalmic Med Image Anal 12069:206–215
227. Tagade P, Hariharan KS, Ramachandran S, Khandelwal A, Naha A, Kolake SM, Han SH (2020) Deep Gaussian process regression for lithium-ion battery health prognosis and degradation mode diagnosis. J Power Sources 445:1–14
228. Viroli C, McLachlan GJ (2019) Deep Gaussian mixture models. Stat Comput 29:43–51
229. Purohit H, Tanabe R, Endo T, Suefusa K, Nikaido Y, Kawaguchi Y (2020) Deep autoencoding GMM-based unsupervised anomaly detection in acoustic signals and its hyper-parameter optimization. arXiv e-prints arXiv:1806.01551.
230. Ji R, Lin S, Chao F, Wu Y, Huang F (2018) Deep neural network compression and acceleration: a review. J Comput Res Dev 55:1871–1888
231. Deng XG, Tian XM, Chen S, Harris CJ (2017) Deep learning based nonlinear principal component analysis for industrial process fault detection. IEEE Int Jt Conf Neural Netw (IJCNN) 1237–1243
232. Deng XG, Zhang Z (2020) Nonlinear chemical process fault diagnosis using ensemble deep support vector data description. Sensors 20:1–19
233. He K, Zhang X, Ren S, Sun J (2015) Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: Proceedings of the international conference on computer vision, pp 1026–1034
234. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv e-prints (2015) arXiv:1502.03167v03163.
235. Sun C, Ma M, Zhao ZB, Tian SH, Yan RD, Chen XF (2019) Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing. IEEE T Ind Inform 15:2416–2425
236. Yu J, Xi L (2008) Intelligent monitoring and diagnosis of manufacturing process using an integrated approach of neural network ensemble and genetic algorithm. Int J Comput Appl Technol 33:109–119
237. Yu JB, Xi LF, Zhou XJ (2008) Intelligent monitoring and diagnosis of manufacturing processes using an integrated approach of KBANN and GA. Comput Ind 59:489–501
238. Liu G, Yu J (2019) Machining roughness prediction based on knowledge-based deep belief network. J Mech Eng 55:94–106
239. Liu G, Yu J (2020) Application of neural-symbol model based on stacked denoising auto-encoders in wafer map defect recognition. J Autom 1–18

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.