Computers & Industrial Engineering 203 (2025) 111064

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

A novel method of detection of noisy samples in high-dimensional sequential data considering both attribute and label noise

Liang Chang [a], Lu-Xin Guan [b], Enrico Zio [c,d], Yan-Hui Lin [a,*]

[a] School of Reliability and Systems Engineering, Beihang University, Beijing, China
[b] Zhejiang Industry & Trade Vocational College, Wenzhou, Zhejiang, China
[c] Energy Department, Politecnico di Milano, Milan, Italy
[d] Centre for Research on Risk and Crises (CRC), MINES Paris-PSL University, Paris, France

ARTICLE INFO

Keywords:
Data likelihood
High-dimensional sequential data
Noise detection
Variational recurrent neural network

ABSTRACT

The high-dimensional sequential data available across various industrial scenarios may be contaminated with both attribute and label noise, hindering the establishment of accurate deep learning-based prediction models. The existing noise detection methods can only detect one type of noise. Conversely, in this article, a novel noisy samples detection method is proposed to detect both types of noise simultaneously through generative learning. An enhanced variational recurrent prediction model (EVRPM) is proposed to model the log-likelihood of samples, which incorporates a label predictor and an auxiliary task into the variational recurrent neural network. Moreover, an iterative detection process is adopted to refine EVRPM training and enhance noisy sample detection, which is particularly beneficial for low-quality datasets. A prediction model with higher prediction accuracy can be obtained using the refined dataset. The effectiveness and superiority of the proposed method are verified using both public and real experimental datasets.

1. Introduction

With advances in sensor technologies and increased data storage capabilities, an abundance of high-dimensional sequential data is now available across various industrial settings, such as monitoring data from numerous sensors installed on machines or deployed within chemical production processes (Kara, 2022; Lin & Chang, 2022a; Xu et al., 2022). Deep learning techniques have risen to prominence for their ability to handle such data, achieving significant success in a range of applications, such as the remaining useful life (RUL) prediction of industrial machinery (Hong et al., 2023), quality control in manufacturing processes (Bai et al., 2021) and state-of-health assessment of consumer electronics (Tan & Zhao, 2020).

Nevertheless, the effectiveness of deep learning techniques relies on the high quality of the training data. In industrial settings, it is common for data to be contaminated with noise, which can significantly degrade the performance of prediction models (Guan et al., 2021). Two primary types of noise are recognized: attribute noise and label noise (Blanco et al., 2022; Guan et al., 2021). Attribute noise, also known as feature noise, arises from a range of issues such as sensor inaccuracies, transmission limitations, and noisy environment. Label noise arises when samples are annotated incorrectly, resulting in noisy labels (Cappozzo et al., 2020). Such labels may result from several factors, including delayed data acquisition, inaccurate sensor signals, human errors, and unknown impact events (Barrias et al., 2016). In this study, a sample contaminated with either or both types of noise is referred to as a "noisy sample". In practice, it is common for datasets to exhibit both attribute and label noise concurrently. Addressing both types of noise simultaneously is essential. However, the existing methods typically tackle them separately, under the assumption that only one type of noise is present in the dataset.

In the existing literature, methods to address noise are divided into two main categories. The first category focuses on developing algorithms that are robust to noise, with a predominant emphasis on label noise mitigation. It has been pointed out that an appropriate level of attribute noise during training can actually enhance the generalization capabilities of deep learning models (Graves, 2012). Conversely, label noise may cause overfitting (Feng et al., 2020) and has been found to be more harmful to the model performance than attribute noise (Guan et al., 2021). Consequently, much effort has been directed toward mitigating the negative effect of label noise on model training. Wang et al. (2019) proposed symmetric cross entropy, inspired by symmetric

* Corresponding author.
E-mail address: linyanhui@buaa.edu.cn (Y.-H. Lin).

https://doi.org/10.1016/j.cie.2025.111064
Received 12 May 2024; Received in revised form 23 January 2025; Accepted 17 March 2025
Available online 20 March 2025

KL-divergence, to prevent deep learning models from overfitting to noisy labels. Zhang and Sabuncu (2018) proposed a series of noise-robust loss functions, which generalize mean absolute error and categorical cross entropy. Ghosh et al. (2017) and Liu and Guo (2020) provided sufficient conditions for risk minimization under label noise and introduced peer loss functions designed to learn from noisy labels, respectively. However, these approaches primarily address classification problems rather than regression tasks. Several distribution alignment techniques (Qu et al., 2021; Zhang, Huang, Luo, & Zhao, 2022) have been proposed, but they too focus on classification rather than regression. Furthermore, such methods do not focus on processing noise, compromising their effectiveness, particularly in high-dimensional data contexts (Feng et al., 2020; Guan et al., 2021).

The second category, which tries to detect noise within datasets, has been increasingly popular in recent studies. The key benefit of these methods is that high-quality data can be obtained after eliminating the detected noise to train diverse deep learning-based prediction models which, in turn, can promote the accuracy of predictions on the test data (Teng, 2000). To detect attribute noise, reconstruction errors are commonly employed as indicators of anomalies. For non-sequential data, autoencoders (AEs) and variational autoencoders (VAEs) are widely used to learn reconstructions. It has been argued that VAEs can outperform AEs in noise detection, since they provide a probabilistic measure, rather than a mere reconstruction error, for anomaly scoring, thereby capturing data variability more effectively through latent random variables with greater expressiveness (An & Cho, 2015). For sequential data, AEs and VAEs are integrated with recurrent neural networks (RNNs) to learn reconstructions due to their powerful capability of modelling such data (Ergen & Kozat, 2020; Maleki et al., 2021; Zhou et al., 2021). Although these methods have shown promise in detecting attribute noise, their capacity for detecting label noise is less effective. Moreover, when dealing with sequential data, the independence of latent random variables across timesteps may compromise their representative ability (Girin et al., 2021).

To detect label noise, the core idea of the existing methods is that larger prediction errors during training indicate that the corresponding samples are more likely to be mislabeled. Guan et al. (2021) developed a sequential ensemble noise filter, where multiple classifiers were trained. A sample was detected as a mislabeled sample if the majority of these classifiers' predictions diverged from the given label of the sample. Chen et al. (2022) proposed to adjust the weight of each sample in the loss function according to the prediction error during training, assigning lower weights to samples with larger prediction errors due to their potential abnormality. Some works address the coexistence of both attribute and label noise (Zhang & Huang, 2023; Zhang, Huang, Bai, & Xu, 2022; Zhang, Huang, Xu, & Bai, 2024), but these methods also detect noisy labels when their losses exceed a certain threshold. However, various factors may lead to large prediction errors, thus limiting the reliability of these methods.

To improve the quality of high-dimensional sequential data and consequently promote the prediction accuracy for test data, we propose to simultaneously detect both attribute and label noise by leveraging the log-likelihood of each sample, denoted by $logp(\boldsymbol{X},y)$, where $\boldsymbol{X}$ represents the input sequence and y denotes the corresponding label. In this study, $\boldsymbol{X}$ is a high-dimensional sequence and y is a scalar. The log-likelihood of each sample serves as a key measure of its alignment with the model's learned data distribution. It is an effective tool for detecting noisy samples, including both attribute noise and label noise, as both types disrupt the learned distribution and yield lower log-likelihood values. To this end, an enhanced variational recurrent prediction model (EVRPM) is proposed based on the variational RNN (VRNN) (Chung et al., 2016) to approximate $logp(\boldsymbol{X},y)$. The VRNN, which extends the VAE into a recurrent framework, effectively models high-dimensional sequential data by introducing temporal structure into the prior distribution of the latent random variable, thereby improving the representative ability of the model. In the EVRPM, we incorporate a label predictor and an auxiliary task—inspired by Goyal et al. (2017)—into the VRNN structure to more precisely model the joint probability distribution of the inputs and labels. Moreover, we adopt an iterative learning approach in our detection method, wherein outcomes from each iteration inform and refine the subsequent one. Specifically, the initially trained EVRPM might be less optimal due to the presence of numerous noisy samples. However, by iterative learning, its reliability is expected to be improved as more noisy samples are removed. Additionally, the robustness of the proposed detection method is enhanced by reducing sensitivity to the detection threshold, as it is not required to detect all noisy samples in a single iteration. The iterative process is terminated when the prediction error of the validation set no longer decreases. After removing the detected noisy samples, a long short-term memory (LSTM)-based prediction model is built using the remaining samples, which can achieve more accurate prediction results for the test data. A public aircraft turbofan engine degradation dataset (Saxena et al., 2008) and a real experimental dataset related to the diesel hydrofining process collected from a petrochemical workshop are used to verify the effectiveness of the proposed detection method.

The main contributions are summarized as follows:

1) The EVRPM is introduced to model the log-likelihood of samples by incorporating a label predictor and an auxiliary task into the VRNN. It is specifically designed for high-dimensional sequential data, accounting for temporal dependencies in the input, which are common in many industrial applications where noise detection remains a crucial yet underexplored challenge. The log-likelihood quantifies how well a sample aligns with the learned data distribution, making it able to detect both attribute and label noise more effectively than using prediction errors.

2) The proposed detection method incorporates an iterative process to refine the accuracy of the $logp(\boldsymbol{X},y)$ modelling and mitigate the influence of the detection threshold on overall detection performance. More noisy samples can be detected and the EVRPM can be trained to be more and more accurate with iteration going on, leading to accurate and stable detection results.

The remaining of this article is organized as follows. In Section 2, the proposed method is detailed, including the proposed EVRPM and the proposed iterative detection process. The effectiveness and superiority of the proposed method are verified on a public turbofan engine degradation dataset and a real experimental dataset regarding the diesel hydrofining process collected from a petrochemical workshop in Sections 3 and 4, respectively. Finally, Section 5 concludes this work.

2. The proposed detection method

In this section, the proposed detection method is detailed. Firstly, the proposed EVRPM used for $logp(\boldsymbol{X},y)$ modelling is introduced and the approximation of $logp(\boldsymbol{X},y)$ is derived, which is used as an indicator for identifying normal samples. Secondly, the implementation of the proposed iterative detection process is detailed, in which the detected samples in each iteration are removed and the EVRPM is retrained in the next iteration using the remaining samples. The iteration process is terminated when the prediction error of the validation set no longer increases. Finally, a LSTM-based prediction model is built using the remaining samples to predict the test data.

2.1. The proposed EVRPM for $logp(\boldsymbol{X},y)$ modelling

For high-dimensional sequential data, the VRNN is an effective modelling paradigm combining the advantages of VAE and RNN. It consists of an inference network for encoding $\boldsymbol{X}$ into the latent random variables $\boldsymbol{Z}$ to obtain the variational approximation of the intractable posterior distribution, and a generation network for decoding $\boldsymbol{Z}$ back into reconstructions of $\boldsymbol{X}$. The latent random variables can model the variability observed in the data and the temporal structure is further introduced into the prior distribution of $\boldsymbol{Z}$ in the VRNN to improve the

representative power of the model.

The VRNN contains a VAE at each timestep t and these VAEs are conditioned on the state variable $\boldsymbol{h}_{t-1}$ of an RNN. It will help the VAE to take into account the temporal structure of the sequential data (Chung et al., 2016). Specifically, let $\boldsymbol{x}^t$ and $\boldsymbol{z}^t$ denote the input and the corresponding latent random variable at timestep t, respectively. Let T denote the number of timesteps in $\boldsymbol{X}$, then $\boldsymbol{X} = [\boldsymbol{x}^t]_{t=1}^T, \boldsymbol{x}^t \in \mathbb{R}^{d_x}$ and $\boldsymbol{Z} = [\boldsymbol{z}^t]_{t=1}^T$, $\boldsymbol{z}^t \in \mathbb{R}^{d_z}$. Instead of setting the prior distribution of $\boldsymbol{z}^t$ to be a standard Gaussian distribution like in the original VAE, the prior distribution $\nu(\boldsymbol{z}^t)$ is derived from $\boldsymbol{h}_{t-1}$ as follows:

$$\nu(\boldsymbol{z}^t) = N\left(\boldsymbol{z}^t; \boldsymbol{\mu}_0^t, diag\left(\boldsymbol{\sigma}_0^{t\,2}\right)\right), \tag{1}$$

$$\left(\boldsymbol{\mu}_0^t, \boldsymbol{\sigma}_0^t\right) = \varphi^{prior}(\boldsymbol{h}_{t-1})$$

where φ^{prior} is a neural network (e.g., fully-connected layers) to compute the parameters of $\nu(\boldsymbol{z}^t)$ from $\boldsymbol{h}_{t-1}$. In this way, $\nu(\boldsymbol{z}^t)$ depends on all the preceding inputs via the RNN hidden state $\boldsymbol{h}_{t-1}$, which introduces temporal structure into $\nu(\boldsymbol{z}^t)$. In a similar fashion, the approximate posterior distribution of $\boldsymbol{z}^t$ will not only be a function of $\boldsymbol{x}^t$ but also of $\boldsymbol{h}_{t-1}$ as follows:

$$\varphi(\boldsymbol{z}^t|\boldsymbol{x}^t) = N\left(\boldsymbol{z}^t; \boldsymbol{\mu}_z^t, diag\left(\boldsymbol{\sigma}_z^{t\,2}\right)\right), \tag{2}$$

$$\left(\boldsymbol{\mu}_z^t, \boldsymbol{\sigma}_z^t\right) = \varphi^{enc}(\varphi^x(\boldsymbol{x}^t), \boldsymbol{h}_{t-1})$$

where φ^{enc} and φ^x are neural networks to compute the parameters of $\varphi(\boldsymbol{z}^t|\boldsymbol{x}^t)$ and extract features from $\boldsymbol{x}^t$, respectively. The hidden state is updated using the recurrence equation:

$$\boldsymbol{h}_t = f_\theta(\varphi^x(\boldsymbol{x}^t), \varphi^z(\boldsymbol{z}^t), \boldsymbol{h}_{t-1}) \tag{3}$$

where f_θ is a deterministic non-linear transition function, which is implemented with LSTM in this work, and φ^z is a neural network to extract features from $\boldsymbol{z}^t$.

The generating distribution for reconstructions is conditioned on $\boldsymbol{z}^t$ and $\boldsymbol{h}_{t-1}$ as follows:

$$\nu(\boldsymbol{x}^t|\boldsymbol{z}^t) = N\left(\boldsymbol{x}^t; \boldsymbol{\mu}_x^t, diag\left(\boldsymbol{\sigma}_x^{t\,2}\right)\right), \tag{4}$$

$$\left(\boldsymbol{\mu}_x^t, \boldsymbol{\sigma}_x^t\right) = \varphi^{dec}(\varphi^z(\boldsymbol{z}^t), \boldsymbol{h}_{t-1})$$

where φ^{dec} is a neural network to compute the parameters of $\nu(\boldsymbol{x}^t|\boldsymbol{z}^t)$. To model the $logp(\boldsymbol{X}, y)$, a label predictor is further introduced in the EVRPM. For a scalar y, the distribution for y is conditioned on $\boldsymbol{z}^T$ and $\boldsymbol{h}_T$ as follows:

$$\nu(y|\boldsymbol{z}^T) = N\left(y; \mu_y, diag(\sigma_y{}^2)\right), \tag{5}$$

$$\left(\mu_y, \sigma_y\right) = \varphi^{pred}\left(\varphi^z\left(\boldsymbol{z}^T\right), \boldsymbol{h}_T\right)$$

where φ^{pred} is a neural network to compute the parameters of $\nu(y|\boldsymbol{z}^T)$. Note that $\boldsymbol{h}_T$ is a function of $\boldsymbol{z}^{\leq T}$ and $\boldsymbol{x}^{\leq T}$. Therefore, $\nu(y|\boldsymbol{z}^T)$ also defines the distribution $\nu(y|\boldsymbol{z}^{\leq T})$.

It has been empirically observed that it may be difficult to learn meaningful $\boldsymbol{Z}$ in the VRNN when coupled with a strong autoregressive decoder (Goyal et al., 2017). The approximate posterior of $\boldsymbol{Z}$ tends to provide a too weak or noisy signal, due to the variance induced by the stochastic gradient approximation. As a result, the strong decoder may learn to ignore $\boldsymbol{Z}$ and instead to rely solely on $\boldsymbol{X}$, causing $\boldsymbol{X}$ and $\boldsymbol{Z}$ to be independent. To solve this problem, it was proposed in Goyal et al. (2017) to force $\boldsymbol{Z}$ to contain useful information by predicting the hidden state of the backward encoder using $\boldsymbol{Z}$, and bi-directional LSTM was adopted for the backward encoder. Similarly, we propose to add an auxiliary task in the EVRPM, which is to predict $\varphi^x(\boldsymbol{x}^t)$ using $\boldsymbol{z}^t$, to force $\boldsymbol{z}^t$ to encode useful information. Specifically, a neural network φ^{auxi} takes $\boldsymbol{z}^t$ as input and outputs the predicted $\varphi^x(\boldsymbol{x}^t)$, denoted by $\varphi^{auxi}(\boldsymbol{z}^t)$, at each timestep. Different from the method proposed in Goyal et al. (2017), the proposed auxiliary task does not require a backward encoder which, in turn, eases the training of the model.

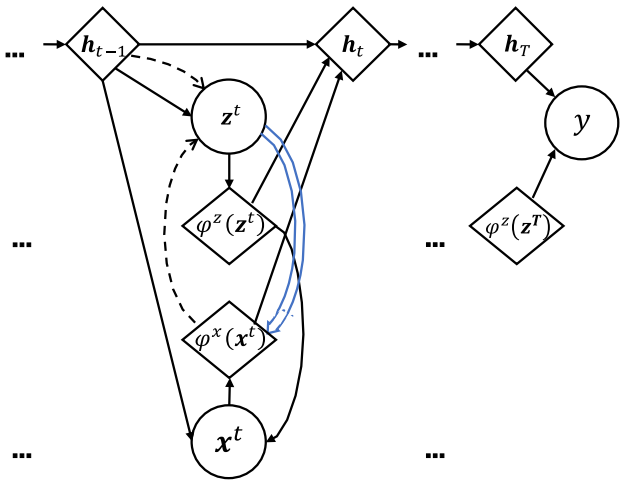


Fig. 1. The graphical illustration of the EVRPM. In the inference network, $\varphi(\boldsymbol{z}^t|\boldsymbol{x}^t)$ is derived from $\varphi^x(\boldsymbol{x}^t)$ and $\boldsymbol{h}_{t-1}$. While in the generation network, $\nu(\boldsymbol{z}^t)$ is derived from $\boldsymbol{h}_{t-1}$. Reconstructions of $\boldsymbol{x}^t$ is conditioned on $\boldsymbol{z}^t$ and $\boldsymbol{h}_{t-1}$, and y is conditioned on $\boldsymbol{z}^T$ and $\boldsymbol{h}_T$. The double line indicates predicting $\varphi^x(\boldsymbol{x}^t)$ using $\boldsymbol{z}^t$.

The graphical illustration of the EVRPM is shown in Fig. 1, where solid lines and dashed lines represent respectively the computation of generation network and inference network, and diamonds and circles represent respectively deterministic and random variables. A double line indicates the proposed auxiliary task.

The training objective of the EVRPM is a regularized version of the variational evidence lower bound (ELBO) of the $log\nu(\boldsymbol{X}, y)$ based on the variational free-energy, where the regularization is imposed by the prediction accuracy of the auxiliary task. The ELBO of $log\nu(\boldsymbol{X}, y)$ is derived as follows:

$$\begin{aligned} log\nu(\boldsymbol{X}, y) &= E_{\varphi(\boldsymbol{Z}|\boldsymbol{X},y)}[log\nu(\boldsymbol{X}, y)] = E_{\varphi(\boldsymbol{Z}|\boldsymbol{X},y)}\left[log\left[\frac{\nu(\boldsymbol{X}, y, \boldsymbol{Z})}{\varphi(\boldsymbol{Z}|\boldsymbol{X}, y)}\frac{\varphi(\boldsymbol{Z}|\boldsymbol{X}, y)}{\nu(z|\boldsymbol{X}, y)}\right]\right] \\ &= E_{\varphi(\boldsymbol{Z}|\boldsymbol{X},y)}\left[log\left[\frac{\nu(\boldsymbol{X}, y, \boldsymbol{Z})}{\varphi(\boldsymbol{Z}|\boldsymbol{X}, y)}\right]\right] + D_{KL}(\varphi(\boldsymbol{Z}|\boldsymbol{X}, y)|\nu(z|\boldsymbol{X}, y)) \\ &\geq E_{\varphi(\boldsymbol{Z}|\boldsymbol{X},y)}\left[log\left[\frac{\nu(\boldsymbol{X}, y, \boldsymbol{Z})}{\varphi(\boldsymbol{Z}|\boldsymbol{X}, y)}\right]\right] \end{aligned} \tag{6}$$

where D_{KL} denotes the KL-divergence between two distributions, and $E_{\varphi(\boldsymbol{Z}|\boldsymbol{X},y)}\left[log\left[\frac{\nu(\boldsymbol{X},y,\boldsymbol{Z})}{\varphi(\boldsymbol{Z}|\boldsymbol{X},y)}\right]\right]$ is the ELBO of $log\nu(\boldsymbol{X}, y)$. The expectation is approximated with one sample from the $\varphi(\boldsymbol{Z}|\boldsymbol{X}, y)$ by using re-parameterization. To maximize the ELBO of $log\nu(\boldsymbol{X}, y)$ and minimize the prediction error of the auxiliary task, the final loss function for EVRPM training is as follows:

$$\begin{aligned}\mathscr{L}(\boldsymbol{X},y) &= -E_{q(\boldsymbol{Z}|\boldsymbol{X},y)}\left[log\left[\frac{p(\boldsymbol{X},y,\boldsymbol{Z})}{q(\boldsymbol{Z}|\boldsymbol{X},y)}\right]\right] + \lambda\left[\varphi^{auxi}(\boldsymbol{Z}) - \varphi_\tau^x(\boldsymbol{X})\right]^2 \\ &= E_{\mathscr{g}(\boldsymbol{Z}|\boldsymbol{X},y)}[log_{\mathscr{h}}(\boldsymbol{X},y|\boldsymbol{Z})] - D_{KL}(\mathscr{g}(\boldsymbol{Z}|\boldsymbol{X},y)|_{\mathscr{h}}(\boldsymbol{Z}))\Bigg) \\ &\quad + \lambda\left[\varphi^{auxi}(\boldsymbol{Z}) - \varphi_\tau^x(\boldsymbol{X})\right]^2 \\ &= \frac{1}{T}\sum_{t=1}^{T}\left\{\frac{1}{2}log(\sigma_x^t) + \frac{(\boldsymbol{x}^t - \boldsymbol{\mu}_x^t)^2}{2\times\sigma_x^t} + D_{KL}(q(\boldsymbol{z}^t|\boldsymbol{x}^t)|p(\boldsymbol{z}^t))\right. \\ &\quad \left. + \lambda\left[\varphi^{auxi}(\boldsymbol{z}^t) - \varphi_\tau^x(\boldsymbol{x}^t)\right]^2\right\} + \frac{1}{2}log(\sigma_y) + \frac{(y-\mu_y)^2}{2\times\sigma_y}\end{aligned} \tag{7}$$

where λ is a weight to rebalance the values of the ELBO of $log_{\mathscr{h}}(\boldsymbol{X},y)$ and the prediction error of the auxiliary task to a same scale. Using this loss function, the EVRPM can be trained via stochastic gradient descent with the re-parameterization trick (Goyal et al., 2017).

To detect noisy samples, the ELBO of $logp(\boldsymbol{X},y)$ is used to approximate $logp(\boldsymbol{X},y)$ as an indicator S for normal samples. For a sample $(\boldsymbol{X}_i,y_i)$, $\boldsymbol{X_i} = [\boldsymbol{x}_i^t]_{t=1}^T$, its S_i is calculated as follows:

$$\begin{aligned}S_i &= \frac{1}{T}\sum_{t=1}^{T}\left\{\frac{1}{2}log(\sigma_x^t) + \frac{(\boldsymbol{x}_i^t - \boldsymbol{\mu}_x^t)^2}{2\times\sigma_x^t} + D_{KL}(q(\boldsymbol{z}^t|\boldsymbol{x}^t)|p(\boldsymbol{z}^t))\right\} \\ &\quad + \frac{1}{2}log(\sigma_y) + \frac{(y_i-\mu_y)^2}{2\times\sigma_y}\end{aligned} \tag{8}$$

A small value of S suggests a high probability of noise in the sample.

2.2. The implementation of the proposed detection method

To build the EVRPM, N available labeled samples $D = \{\boldsymbol{X}_i,y_i\}_{i=1}^N$ are divided into a training set D_{train} and a validation set D_{val} at a 9:1 ratio. The training set is used to train the model, whereas the validation set is used to determine the number of training epochs in each iteration and when to terminate the iterative process. The algorithm of the proposed detection method is given in Algorithm 1.

Algorithm 1. The proposed detection method

Require: dataset: $D = \{\boldsymbol{X}_i,y_i\}_{i=1}^N$
Output: the refined dataset: D
1: **initialize** EVRPM, $j = 0$
2: $D_{train}^0, D_{val}^0 = \{\boldsymbol{X}_i,y_i\}_{i=1}^{\lfloor 0.9N\rfloor}, \{\boldsymbol{X}_i,y_i\}_{i=\lfloor 0.9N\rfloor+1}^{N}$
3: **repeat**
4: $j \leftarrow j+1$
5: train the EVRPM using D_{train}^{j-1}
6: calculate S_i for each sample in D by Eq. (8).
7: calculate μ_S and σ_S of all S_i by Eq. (11) and (12), respectively
8: $D_{train}^j \leftarrow \{\boldsymbol{X}_i,y_i | S_i > \mu_S - 3\sigma_S, 1 \le i \le \lfloor 0.9N\rfloor\}$
9: $D_{val}^j \leftarrow \{\boldsymbol{X}_i,y_i | S_i > \mu_S - 3\sigma_S, \lfloor 0.9N\rfloor < i \le N\}$
10: calculate mse_{val}^j using the trained EVRPM by Eq. (13).
11: **until** $mse_{val}^j > mse_{val}^{j-1}$
12: **return** $D = D_{train}^{j-1} \cup D_{val}^{j-1}$

The iterative process is presented in Steps 3–11 of Algorithm 1. In Step 5, the EVRPM is trained in the j-th iteration to minimize the loss of D_{train}^{j-1} with mini-batch technique. The loss of D_{val}^{j-1} is monitored and the training is terminated when the loss of D_{val}^{j-1} no longer decreases. The loss of a batch of samples $\{\boldsymbol{X}_i,y_i\}_{i=1}^{N_b}$ is calculated based on (7) as follows:

$$\mathscr{L}(\{\boldsymbol{X}_i,y_i\}_{i=1}^{N_b}) = \frac{1}{N_b}\sum_{i=1}^{N_b}\mathscr{L}(\boldsymbol{X}_i,y_i) \tag{9}$$

where N_b is the number of samples in the batch.

Afterwards, the trained EVRPM is used to calculate the S_i for each sample in D for detection. It is assumed that the values of S_i for normal samples follow a Gaussian distribution, which is also observed in the experimental results. Considering more noisy samples can be detected as iterations go on, a relatively small detection threshold is preferred to avoid detecting normal samples as noisy samples. Therefore, the detection threshold is set according to the 3σ principle as follows in each iteration:

$$threshold = \mu_S - 3\sigma_S \tag{10}$$

where μ_S and σ_S are the mean value and the standard deviation of all S_i in each iteration, respectively.

$$\mu_S = \frac{1}{N^j}\sum_{i=1}^{N^j} S_i \tag{11}$$

$$\sigma_S = \sqrt{\frac{1}{N^j-1}\sum_{i=1}^{N^j}(S_i-\mu_S)^2} \tag{12}$$

The samples whose $S_i < \mu_S - 3\sigma_S$ are identified as noisy samples and removed. The remaining samples are used for the subsequent iterations. Note that it is unnecessary to detect all noisy samples in one iteration, since then noisy samples can be detected in the subsequent iterations. Therefore, a relatively small detection threshold value can also achieve satisfactory detection performance. That is to say, the iterative detection process reduces the sensitivity to the detection threshold. Besides, the EVRPM can be trained to be more accurate as iterations go on, since more noisy samples are removed. Consequently, more accurate detection results can be achieved by this iterative process.

Considering that the purpose of the noisy samples detection is to improve the data quality and further build a more accurate prediction model for test data prediction, the prediction error of D_{val}^j is monitored in Step 11 of the j-th iteration, which is measured by the mean square error (MSE) as follows:

$$mse_{val}^j = \frac{1}{N_{val}^j}\sum_{y_i\in D_{val}^j}\left(\mu_{y_i} - y_i\right)^2 \tag{13}$$

where N_{val}^j is the number of samples in D_{val}^j and μ_{y_i} is calculated in (5) by the EVRPM. If mse_{val}^j is larger than mse_{val}^{j-1}, then the iteration process is terminated and the detection is completed.

Finally, using the samples in the refined D, a LSTM-based prediction model is built for test data prediction. It is expected to achieve more accurate prediction results after removing the detected samples.

3. C-MAPSS case study

In this section, the effectiveness of the proposed method is verified using a publicly available dataset, related to the degradation process of turbofan engines (Saxena et al., 2008). This dataset is denoted by C-MAPSS.

3.1. Dataset description and preprocessing

The C-MAPSS dataset contains four subsets differing in the number of fault types and experienced operational conditions. One of them is used in this case study, denoted by FD001, which has only one fault type and the experienced operation condition remains unchanged. Two data subsets are included in FD001 and denoted by FD001_train and FD001_test, respectively. The former is used to generate D to build the prediction model and the latter is used for its test. FD001 is composed of high-dimensional sequential data collected from 21 sensors and 3 operational condition measurements. The data from 14 sensors that are time-varying are chosen as inputs, i.e., $d_x = 14$ in this case study. They are normalized to be within the range of [0,1] using the min–max

normalization method. The target is the number of remaining operational cycles before failure, i.e., RUL. According to related researches (Li et al., 2018; Lin & Chang, 2022b), a piecewise linear function is applied to obtain the labels, where it is assumed that the RUL labels in the early period are constant and set to 125. Sequential samples are generated via a sliding window, which slides from the first timepoint to the last timepoint of the monitoring data collected from one engine. For each sample, the input sequence contains all the monitoring data within the window, and the label is the RUL corresponding to the last timestep. The size of sliding window is set to 40, i.e., $T = 40$.

The noisy samples may arise in such a situation. An unknown impact event during the engine's degradation process triggers rapid degradation, leading to premature failure. As a result, labels of samples generated prior to the impact are underestimated, and certain samples exhibit abnormal attributes, as illustrated in Fig. 2. To simulate this situation, we remove monitoring data spanning T_{im} consecutive timepoints, starting at random points in the degradation process, per h engines in the FD001_train. Then, samples are generated from the remaining data using a sliding window. In Fig. 2, the colors red, green and blue represent the samples with label noise, attribute noise and no noise, respectively. In this case study, we set $T_{im} = 50$ and $h = 3$. Fig. 3 shows the original monitoring data from a sensor of one engine and the remaining data used to generate the noisy samples. Note that noise is only added to the data of the FD001_train, whereas the FD001_test is assumed to be a normal dataset with no noise for evaluating prediction accuracy.

The values of the hyper-parameters for the EVRPM training in this case study are provided in Table 1, as determined according to the loss of the validation set.

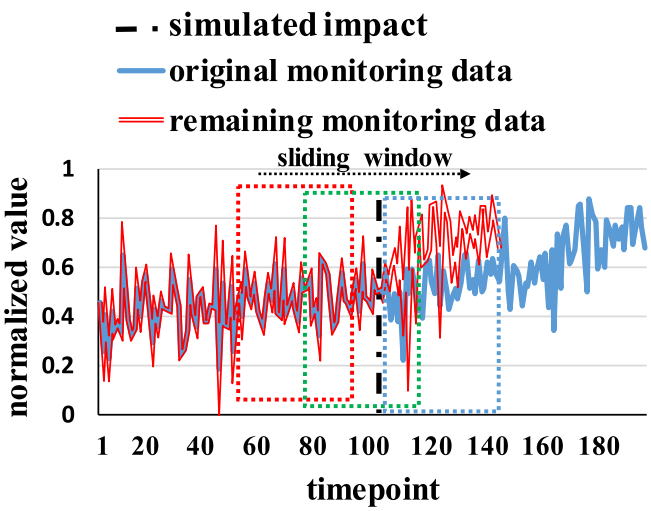


Fig. 3. Illustration of the original monitoring data and the remaining monitoring data of one sensor for noisy samples generation. The red, green and blue dashed boxes represent samples with label noise, attribute noise and no noise, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
The hyper-parameters for the EVRPM training for the C-MAPSS dataset.

Hyper-parameters	Values
training batch size	1024
d_z	8
learning rate	0.003
λ	1

3.2. Experimental results and comparison

The effectiveness of the proposed method is verified through an ablation study by comparison with three baseline methods. Method I trains a LSTM-based prediction model using the noisy data in the FD001_train directly. Compared with the proposed detection method, Method II omits the auxiliary task from the EVRPM training, and Method III does not employ the iterative detection process. To further verify the performance of the proposed method, it is compared against two widely used noise detection methods for high-dimensional sequential data: a prediction error-based method and a reconstruction error-based method. In the former, a LSTM-based prediction model is built and its prediction error of each sample is used as an indicator for noisy samples. In the latter, a LSTM-based AE is built to simultaneously reconstruct attributes and labels of samples. The noisy samples are identified as the

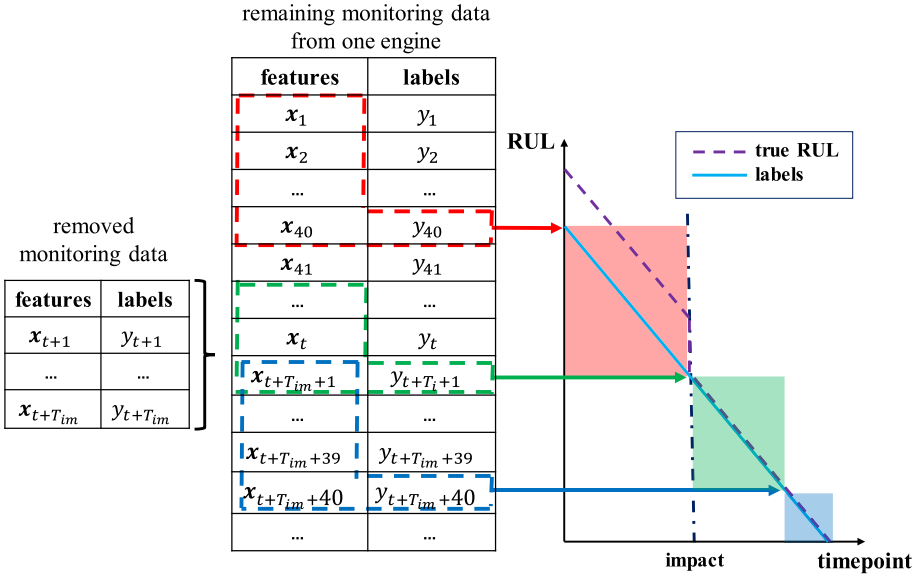


Fig. 2. Illustration of the noisy samples generation procedure in the C-MAPSS dataset. At each timepoint, $\boldsymbol{x}_t \in \mathbb{R}^{14}$ is collected and the corresponding label y_t is annotated. Monitoring data from $\boldsymbol{x}_{t+1}$ to $\boldsymbol{x}_{t+T_{im}}$ are removed to simulate an unknown impact. Consequently, the samples with attribute noise and label noise are generated and highlighted by the red and green dashed box, respectively. While the blue dashed box denotes a normal sample. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

samples with reconstruction errors larger than the threshold. For fair comparison, these two methods also adopt the same iterative process as the proposed method.

In this paper, the root mean squared error (RMSE) is used to measure the prediction error of test set, calculated as follows:

$$RMSE(D_{test}) = \sqrt{\frac{1}{N_{test}}\sum_{i=1}^{N_{test}} d_i^2} \tag{14}$$

where N_{test} is the number of samples in the test set and $d_i = y_{i,test} - \hat{y}_{i,test}$ denotes the error between the label and the corresponding predicted value of the i-th sample in the test set. The prediction error of the test set is expected to be reduced by training the prediction model using the refined D. Additionally, two types of detection errors are also presented for each detection method. The first type is called E_1, in which normal samples are incorrectly identified as noisy samples. The second type is called E_2, in which noisy samples are incorrectly identified as normal data. We adopt three indicators (Brodley & Friedl, 1999; Guan et al., 2021) to measure the detection errors as follows:

$$p(E_1) = \frac{K - M \cap K}{N - M} \tag{15}$$

$$p(E_2) = \frac{M - M \cap K}{M} \tag{16}$$

$$p(E) = \frac{1}{2}(p(E_1) + p(E_2)) \tag{17}$$

where K denotes the number of detected samples, M denotes the number of real noisy samples and N denotes the number of samples in D.

The distributions of the values of S_i in two iterations are shown in Fig. 4. The distribution of the values of S_i for normal samples is close to a Gaussian distribution in each iteration. Besides, as iterations go on, more noisy samples are removed and the distribution of the values of S_i for all samples becomes closer to a Gaussian distribution. Therefore, it is reasonable to calculate the detection threshold according to the 3σ principle. Whereas S_i of certain noisy samples are larger than the threshold in the first iteration, their values diminish in the second iteration since the EVRPM is trained to be more accurate. Therefore, employing an iterative process is both logical and essential for noise detection.

Table 2 presents the experimental results of all methods. Firstly, the results of the ablation study show that each component of the proposed

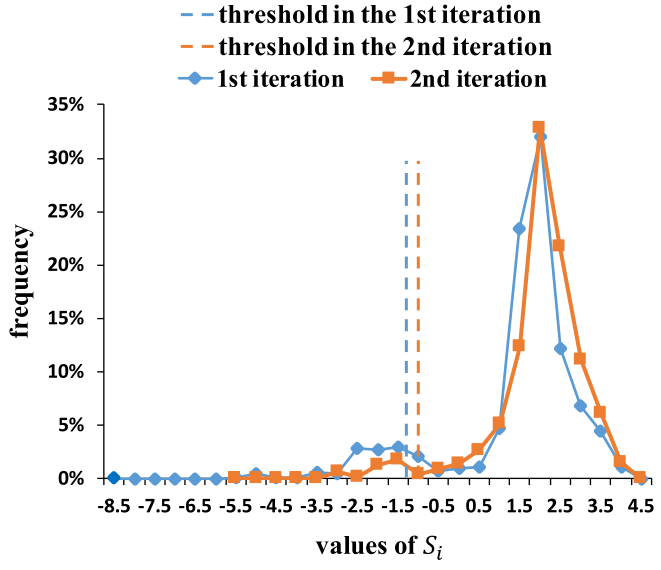


Fig. 4. Distributions of the values of S_i in two iterations for C-MAPSS dataset.

Table 2
Experimental results of all methods on the C-MAPSS dataset.

Methods	RMSE	$p(E_1)$	$p(E_2)$	$p(E)$
Method I	18.12	–	–	–
Method II	15.09	0.12	0.45	0.29
Method III	14.80	**0.00**	0.33	0.17
prediction error-based method (Guan et al., 2021)	15.40	0.04	0.39	0.22
reconstruction error-based method (Maleki et al., 2021)	16.03	0.06	0.63	0.35
proposed method	**13.98**	0.06	**0.18**	**0.12**

method plays a significant role in improving prediction accuracy and robustness. Method I obtains the largest prediction error, indicating that it is necessary to detect and remove the noisy samples before building the prediction model for test data prediction. Method II achieves unsatisfactory results since the independence of latent random variables across timesteps hinders accurate modelling of the log-likelihood of samples. In Method III, the noisy samples are detected in a single iteration. All detected samples are real noisy but only 67 % of real noisy samples are detected, leading to a larger prediction error of the test set compared to the proposed method. Both the prediction error-based and the reconstruction error-based methods deliver unsatisfactory results since the prediction errors and the reconstruction errors are influenced by several factors, such as parameters initialization and model capability. The proposed method achieves the best performance.

To evaluate the performance of the proposed method with larger ratios of noisy samples, partial monitoring data are removed from additional engines to increase the proportion of noisy samples. Specifically, removing monitoring data from every 3 engines (h = 3) resulted in a noisy sample ratio of approximately 1/3. Furthermore, two additional experiments were conducted in which half and two-thirds of the engines had their partial monitoring data removed, leading to noisy sample ratios of approximately 1/2 and 2/3, respectively. The experimental results for these 3 different noisy sample ratios are presented in Table 3. Although the prediction accuracy decreases as the ratio of noisy samples increases, the proposed method improved the prediction accuracy by more than 18 % in each case, owing to the proposed iterative detection process.

4. The diesel hydrofining process case study

In this section, the proposed method is applied on a real experimental dataset related to the diesel hydrofining process of a petrochemical workshop.

4.1. Dataset description and preprocessing

This dataset was collected from a diesel hydrofining process. A simplified process flowchart with a part of sensors is illustrated in Fig. 5. The diesel hydrofining process involves feed oil and hydrogen passing through a catalyst bed within reactors under the action of the

Table 3
Experimental results of different ratios of noisy samples.

Ratios of noisy samples	Methods	RMSE	$p(E_1)$	$p(E_2)$	$p(E)$
1/3	Method I (no detection)	18.12	–	–	–
	proposed method	13.98	0.06	0.18	0.12
1/2	Method I (no detection)	19.88	–	–	–
	proposed method	15.75	0.11	0.21	0.16
2/3	Method I (no detection)	22.12	–	–	–
	proposed method	18.01	0.17	0.25	0.21

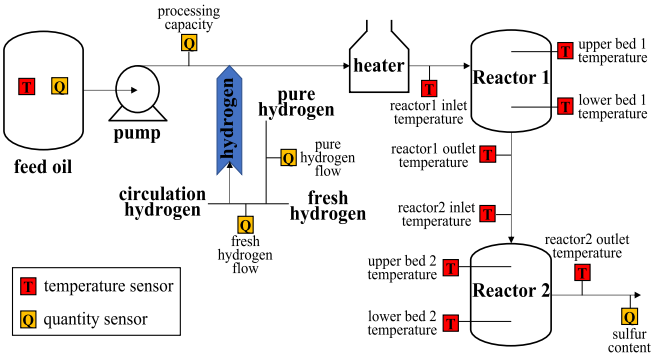


Fig. 5. Illustration of the diesel hydrofining process and some sensors.

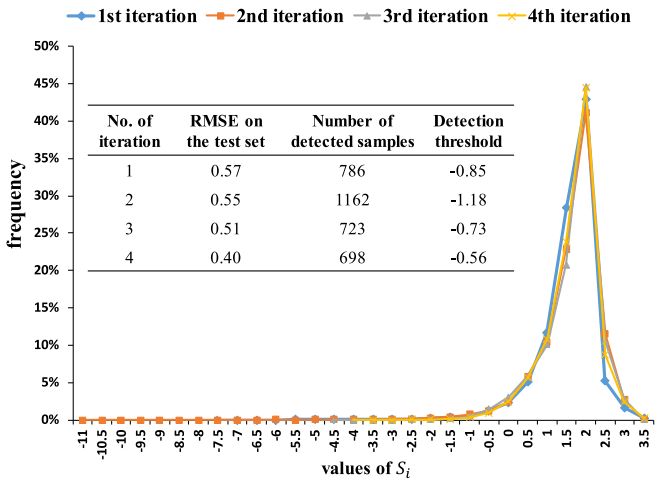
hydrofining catalyst. In this way, non-hydrocarbon compounds such as sulfur, nitrogen, and oxygen are transformed into their respective hydrocarbons, hydrogen sulfide, ammonia, and water, thereby reducing the sulfur content in diesel. The sulfur content is determined by the properties of the feed oil and the values of process parameters such as reaction temperature and flow. The target is to predict the sulfur content using the monitoring data from multiple sensors, which record the properties of the feed oil and process parameters. At each timepoint, monitoring data from 19 sensors are collected. The monitoring data of the recent six timepoints are used as input to predict the label of the last timepoint, i.e., $T = 6$ and $d_x = 19$ in this case study. The labels are obtained from a sulfur content monitoring device. The min–max normalization method is applied also to this dataset. All samples are divided into two subsets at a 10:1 ratio. The first subset is used as D to build the prediction model and the second subset is used as the test set D_{test}. The partial monitoring data from three randomly selected sensors and the partial labels are shown in Fig. 6.

Both attribute and label noise exist in this dataset. Monitoring data from each sensor may be contaminated with noise, arising attribute noise. Meanwhile, inaccuracies in the sulfur content sensor introduce label noise.

The values of the hyper-parameters for the EVRPM training in this case study are the same as those in the previous case study, except for $d_z = 2$ in this case study.

4.2. Experimental results and comparison

The detection results and the distributions of the values of S_i in a total of four iterations in this case study are shown in Fig. 7. In this case study, a real dataset collected from a petrochemical workshop is used, which has low data quality and complex mechanisms. As a result, a larger number of iterations are required, compared to the previous case study, to obtain optimal detection performance. It can be observed that the



No. of iteration	RMSE on the test set	Number of detected samples	Detection threshold
1	0.57	786	-0.85
2	0.55	1162	-1.18
3	0.51	723	-0.73
4	0.40	698	-0.56

Fig. 7. Detection results and distributions of the values of S_i for the diesel hydrofining process dataset.

RMSE of the test set decreases with iterations going on. This confirms that an iterative process is essential especially for low-quality data.

The experimental results of the proposed method and 5 comparative methods are listed in Table 4. These results support conclusions similar to those drawn for the previous case study in Section 3.2. Note that the real noisy samples are unknown in this dataset. Therefore, the detection errors cannot be quantified. To illustrate the detection results, the projections of $\boldsymbol{z}^T$ of randomly selected partial samples are shown in Fig. 8 (a), where each point represents a sample and different colors represent different normalized label values. Points encircled in red signify samples detected as noisy samples. It can be observed that two types of noisy samples are detected. One type is the samples far away from others, which are contaminated with attribute noise. The other type is the samples whose labels diverge from neighboring ones, indicating label

Table 4
Experimental results of all methods on the diesel hydrofining process dataset.

Methods	RMSE	Number of detected samples
Method I	0.64	0
Method II	0.55	586
Method III	0.57	786
prediction error-based method (Guan et al., 2021)	0.46	7203
reconstruction error-based method (Maleki et al., 2021)	0.62	754
proposed method	**0.40**	3369

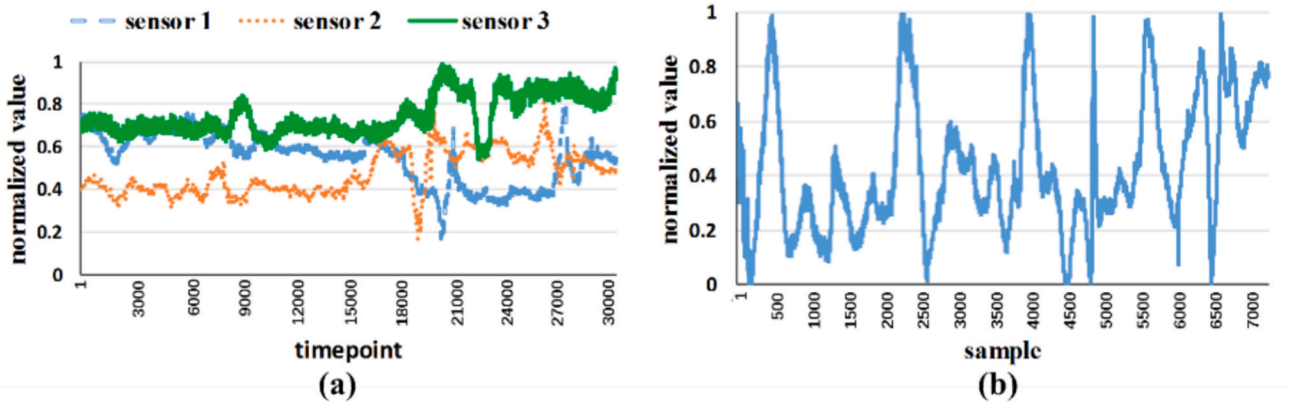


Fig. 6. (a) Partial monitoring data and (b) partial labels in the diesel hydrofining process dataset.

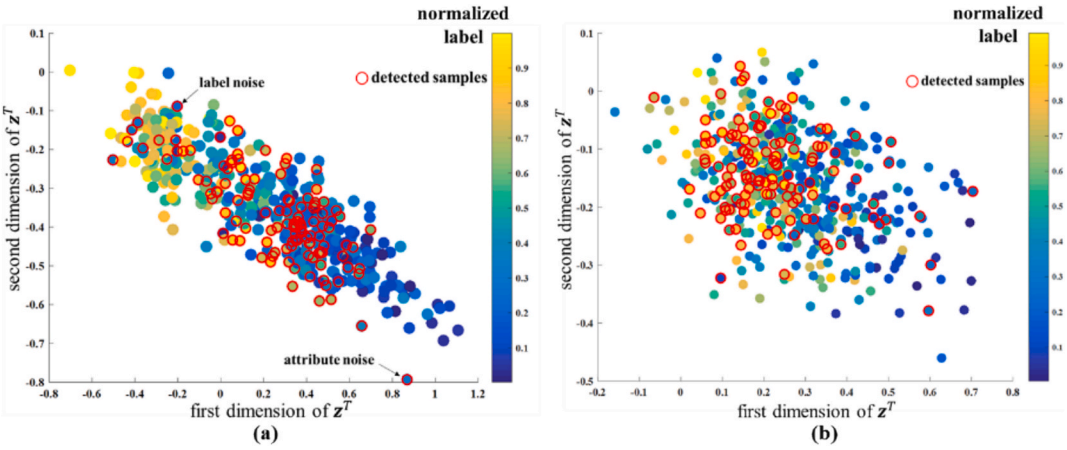


Fig. 8. Projections of z^T learned in (a) the EVRPM and (b) Method II.

noise. Similarly, the projections of z^T of randomly selected partial samples learned in Method II are also shown in Fig. 8(b). Compared with Fig. 8(b), the z^T corresponding to different labels can be disentangled in the two dimensions in Fig. 8(a). It verifies that the representative ability of the latent random variables associated with the labels can be improved by the proposed auxiliary task.

5. Conclusions

In this article, a novel detection method for noisy samples in high-dimensional sequential data is proposed. Different from existing methods, the proposed method can detect both attribute and label noise simultaneously by considering the log-likelihood of samples. It can be very useful for deep learning-based predictions in industrial applications, where data quality directly affects model accuracy and reliability. The EVRPM is proposed to model the log-likelihood of samples by introducing a label predictor and an auxiliary task into the VRNN. The following main conclusions can be obtained according to the experimental results from two case studies: 1) The proposed detection method can refine the noisy dataset, building a more accurate prediction model than the baseline methods and two widely used methods. 2) With the integration of the auxiliary task, the proposed EVRPM can model the log-likelihood of samples more accurately, which in turn facilitates both attribute and label noise detection 3) Through an iterative detection process, noisy samples can be detected more thoroughly, and the accuracy of EVRPM can be further improved.

One limitation of the proposed method is that the attribute noise and label noise are not distinguished. In the future, we will investigate to distinguish between them by modelling $logp(\boldsymbol{X})$ and $logp(y|\boldsymbol{X})$. Besides, mislabeled samples can be corrected based on similarities in the feature space.

CRediT authorship contribution statement

Liang Chang: Writing – original draft, Visualization, Software, Methodology, Data curation, Conceptualization. **Lu-Xin Guan:** Conceptualization, Data curation, Methodology, Software. **Enrico Zio:** Writing – review & editing, Supervision, Investigation, Formal analysis. **Yan-Hui Lin:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported by the National Natural Science Foundation of China under Grant 72471011.

Data availability

The data that has been used is confidential.

References

An, J. & Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. Retrieved from http://dm.snu.ac.kr/static/docs/TR/SNUDM-TR-2015-03.pdf.

Bai, Y., Xie, J., Wang, D., Zhang, W., & Li, C. (2021). A manufacturing quality prediction model based on AdaBoost-LSTM with rough knowledge. *Computers & Industrial Engineering, 155*, Article 107227. https://doi.org/10.1016/j.cie.2021.107227

Barrias, A., Casas, J. R., & Villalba, S. (2016). A review of distributed optical fiber sensors for civil engineering applications. *Sensors, 16*(5), Article 748. https://doi.org/10.3390/s160507482016

Blanco, V., Japón, A., & Puerto, J. (2022). A mathematical programming approach to SVM-based classification with label noise. *Computers & Industrial Engineering, 172*, Article 108611. https://doi.org/10.1016/j.cie.2022.108611

Brodley, C. E., & Friedl, M. A. (1999). Identifying mislabeled training data. *Journal of Artificial Intelligence Research, 11*, 131–167. https://doi.org/10.1613/jair.606

Cappozzo, A., Greselin, F., & Murphy, T. B. (2020). Anomaly and novelty detection for robust semi-supervised learning. *Statistics and Computing, 30*, 1545–1571. https://doi.org/10.1007/s11222-020-09959-1

Chen, F., He, S., Li, Y., & Chen, H. (2022). Data-driven monitoring for distributed sensor networks: An end-to-end strategy based on collaborative learning. *IEEE Sensors Journal, 22*(22), 21795–21805. https://doi.org/10.1109/JSEN.2022.3197443

Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A. & Bengio, Y. (2016). A recurrent latent variable model for sequential data. arXiv preprint arXiv:1506.02216v6.

Ergen, T., & Kozat, S. S. (2020). Unsupervised anomaly detection with LSTM neural networks. *IEEE Transactions on Neural Networks and Learning Systems, 31*(8), 3127–3141. https://doi.org/10.1109/TNNLS.2019.2935975

Feng, W., Quan, Y., & Dauphin, G. (2020). Label noise cleaning with an adaptive ensemble method based on noise detection metric. *Sensors, 20*(23), 6718. https://doi.org/10.3390/s20236718

Ghosh, A., Kumar, H., & Sastry, P. S. (2017). Robust loss functions under label noise for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence* (pp. 1919–1925).

Girin, L., Leglaive, S., Bie, X., Diard, J., Hueber, T., Alameda-Pineda, X., et al. (2021). Dynamical variational autoencoders: A comprehensive review. *Now Foundations and Trends*.

Goyal, A., Sordoni, A., Côté, M. A., Ke, N. R. & Bengio, Y. (2017). Z-forcing: training stochastic recurrent networks. arXiv preprint arXiv:1711.05411v2.

Graves, A. (2012). *Supervised sequence labelling with recurrent neural networks*. Springer.

Guan, D., Chen, K., Han, J., Huang, S., Yuan, W., Guizani, M., et al. (2021). A novel class noise detection method for high-dimensional data in industrial informatics. *IEEE Transactions on Industrial Informatics, 17*(3), 2181–2190. https://doi.org/10.1109/TII.2020.3012658

Hong, S., Kang, M., Kim, J., & Baek, J. (2023). Sequential application of denoising autoencoder and long-short recurrent convolutional network for noise-robust remaining-useful-life prediction framework of lithium-ion batteries. *Computers & Industrial Engineering, 179*, Article 109231. https://doi.org/10.1016/j.cie.2023.109231

Kara, A. (2022). Multi-scale deep neural network approach with attention mechanism for remaining useful life estimation. *Computers & Industrial Engineering, 169*, Article 108211. https://doi.org/10.1016/j.cie.2022.108211

Li, X., Ding, Q., & Sun, J. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety, 172*, 1–11. https://doi.org/10.1016/j.ress.2017.11.021

Lin, Y., & Chang, L. (2022a). An online transfer learning framework for time-varying distribution data prediction. *IEEE Transactions on Industrial Electronics, 69*(6), 6278–6287. https://doi.org/10.1109/TIE.2021.3090701

Lin, Y., & Chang, L. (2022b). An unsupervised noisy sample detection method for deep learning-based health status prediction. *IEEE Transactions on Instrumentation and Measurement, 71*, Article 2502211. https://doi.org/10.1109/TIM.2021.3132374

Liu, Y., & Guo, H. (2020). Peer loss functions: Learning from noisy labels without knowing noise rates. In *Proceedings of the International Conference on Machine Learning* (pp. 6226–6236).

Maleki, S., Maleki, S., & Jennings, N. R. (2021). Unsupervised anomaly detection with LSTM autoencoders using statistical data-filtering. *Applied Soft Computing, 108*(2), Article 107443. https://doi.org/10.1016/j.asoc.2021.107443

Qu, Y., Mo, S., & Niu, J. (2021). DAT: Training deep networks robust to label-noise by matching the feature distributions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 6821–6829).

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. In *2008 International Conference on Prognostics and Health Management* (pp. 1–9). https://doi.org/10.1109/PHM.2008.4711414

Tan, Y., & Zhao, G. (2020). Transfer learning with long short-term memory network for state-of-health prediction of lithium-ion batteries. *IEEE Transactions on Industrial Electronics, 67*(10), 8723–8731. https://doi.org/10.1109/TIE.2019.2946551

Teng, C. M. (2000). Evaluating noise correction. In *Proceedings of the 6th Pacific Rim international conference on Artificial intelligence* (pp. 188–198).

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., Bailey, J., et al. (2019). Symmetric cross entropy for robust learning with noisy labels. *IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, South Korea*.

Xu, X., Li, X., Ming, W., & Chen, M. (2022). A novel multi-scale CNN and attention mechanism method with multi-sensor signal for remaining useful life prediction. *Computers & Industrial Engineering, 169*, Article 108204. https://doi.org/10.1016/j.cie.2022.108204

Zhang, P., & Huang, Z. (2023). Multi-head siamese prototype learning against both data and label corruption. In *Proceedings of the 5th ACM International Conference on Multimedia in Asia* (pp. 1–7). https://doi.org/10.1145/3595916.3626435

Zhang, P., Huang, Z., Bai, G., & Xu, X. (2022). IDEAL: High-order-ensemble adaptation network for learning with noisy labels. In *Proceedings of the 30th ACM International Conference on Multimedia* (pp. 325–333). https://doi.org/10.1145/3503161.3548053

Zhang, P., Huang, Z., Luo, X., & Zhao, P. (2022). Robust learning with adversarial perturbations and label noise: A two-pronged defense approach. In *Proceedings of the 4th ACM International Conference on Multimedia in Asia* (pp. 1–7). https://doi.org/10.1145/3551626.3564934

Zhang, P., Huang, Z., Xu, X., & Bai, G. (2024). Effective and robust adversarial training against data and label corruptions. *IEEE Transactions on Multimedia, 26*, 9477–9488. https://doi.org/10.1109/TMM.2024.3394677

Zhang, Z., & Sabuncu, M. R. (2018). Generalized cross entropy loss for training deep neural networks with noisy labels. *32nd Conference on Neural Information Processing Systems (NIPS), Montreal, CANADA*.

Zhou, X., Hu, Y., Liang, W., Ma, J., & Jin, Q. (2021). Variational LSTM enhanced anomaly detection for industrial big data. *IEEE Transactions on Industrial Informatics, 17*(5), 3469–3477. https://doi.org/10.1109/TII.2020.3022432