Deep Variational Autoencoder Classifier for Intelligent Fault Diagnosis Adaptive to Unseen Fault Categories

Anqi He and Xiaoning Jin, *Member, IEEE*

***Abstract*—With the rapid development of artificial intelligence (AI) in recent years, fault diagnostics for industrial applications have leaped toward partially or fully automatic provided by the capability of analyzing massive condition monitoring data from sensors and actuators. Generally, AI-based fault diagnostics can achieve high accuracy when failure types appear in training dataset and testing dataset are the same. These diagnostic methods could be invalidated for applications dealing with unprecedented faults because the pretrained classifier for diagnostics tends to misclassify the novel instances into existing known classes. In order to address these limitations of conventional diagnostic approaches, we propose a unified diagnostics framework that can achieve novel fault detection and known fault classification tasks together. Through jointly training a variational autoencoder and a deep neural networks classifier, we convert the original entangled raw data into latent variables with Gaussian probabilistic distributions in the latent space and utilize the probabilistic latent variables to detect novel samples against known fault classes or classify them into one of the existing fault classes if they are not novel. The effectiveness of our proposed joint-training framework is validated through experimental studies on two different bearing datasets. Compared with the state-of-the-art methods in the literature, our unified framework is able to not only accurately detect the novel fault classes but also achieve high classification accuracy of known fault classes.**

***Index Terms*—Ball bearing, deep learning (DL), deep variational autoencoder (VAE), intelligent fault diagnostics, novelty detection.**

I. Introduction

TECHNOLOGICAL innovations in machine automation, integration, and precision have been driving higher requirements for reliability and operational safety in modern manufacturing systems [1], [2]. The ever-increasing complexity of machines and the processes causes various types of faults and failures, e.g., lubrication contamination, leakage, overloading, misalignment, etc. These faults lead to component degradation and malfunction and even dangerous failure of the entire machine, resulting in production loss and personnel injury [3], [4]. In order to improve the machinery reliability, it is crucial to detect and identify any types of incipient faults so that maintenance or control strategies actions can be performed timely.

The increasing implementation of new sensing and automation technologies has led to a tremendous growth in the volume of massive data in modern manufacturing systems, providing substantial opportunities to revolutionize process monitoring, fault detection and diagnostics (FDDs). Recent advances in machine learning (ML) based FDD methods take advantages of massive amounts of historical data collected from equipment and processes to provide accurate diagnosis results. These methods require extracting failure-sensitive features using statistics signal processing techniques, such as time domain, frequency domain, and time-frequency domain analysis on raw signals, and then training a classifier based on ML techniques, such as k-nearest neighbor, support vector machine (SVM), naive Bayes, and artificial neural networks, to differentiate health conditions [5]. However, ML-based methods require problem-specific manual feature engineering using domain knowledge, which is tedious and error-prone. Also, standard ML-based methods, such as SVM, are shallow learning models with no more than one nonlinear transformation, causing inefficiency and difficulty in learning complex nonlinear relationships of high-dimensional datasets [6], [7].

The promise of deep learning (DL) based methods for FDD problems is to surpass the limitations of ML by automatically transforming inputs of representations by stacking multiple simple neurons with nonlinear functions. Based on an end-to-end learning mechanism, DL networks have shown a significant capability of adapting to different types of data and discovering intrinsic structures from inputs [8]. Recently, a great number of researchers have proposed different novel DL models for engineering fault diagnosis. For example, Lei *et al.* [9] developed a stacked autoencoder (AE) to learn features from mechanical vibration signals first and then used the soft-max regression to classify the health conditions. Jia *et al.* [10] presented a deep neural network (DNN) model to diagnose the faults of rotating machinery. The DNN model is first pretrained by an unsupervised layer-by-layer learning from frequency spectra and then fine-tuned with a supervised algorithm. Abdeljaber *et al.* [11]

Manuscript received February 20, 2021; revised May 14, 2021; accepted June 13, 2021. Date of publication July 1, 2021; date of current version November 30, 2021. This work was supported by the National Science Foundation under Grant 1943801. Associate Editor: Q. Miao. *(Corresponding author: Xiaoning Jin.)*

The authors are with the Predictive Informatics Research Lab, Northeastern University, Boston, MA 02115 USA (e-mail: he.an@husky.neu.edu; xi.jin@northeastern.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TR.2021.3090310.

Digital Object Identifier 10.1109/TR.2021.3090310

proposed a novel approach for online condition monitoring and severity identification of ball bearings. They utilized two compact one-dimensional (1-D) convolutional neural networks to fuse the feature extraction and classification processes of a damage monitoring system into a single learning body. Gan *et al.* [12] proposed a new hierarchical diagnosis network automatic diagnosis system for rolling-element bearings, which is mainly composed of two layers of deep belief networks for fault identification and severity assessment.

Most DL-based FDD methods are based on the closed-set assumption such that the fault types from the training and testing data share the same class characterization. In the situations where the assumption is held, DL models have superior performances of fault detection by automatic constructing discriminative features from multidimensional condition monitoring data. However, in many industrial applications, it is expensive or impossible to identify and label all possible types of faults in the training stage because manual labeling using domain knowledge has two limitations: expensive and error-prone, and inaccurate for low-frequency fault types. When new fault types occur in testing samples, current diagnostics approaches may misclassify samples into the existing types of faults defined from the training data. To enhance robustness of FDD, it is therefore desirable to first determine whether a test sample is novel or significantly differs from the representative training data [13]. This preliminary process before fault diagnosis is novelty detection.

Novelty detection is the identification of testing data that differ in some respect from the data that is available during training. In general, a description of normality is learnt by constructing a model with numerous samples representing positive instances (e.g., data indicative of normal system behavior). Unprecedented or unknown patterns are then tested by comparing them with the model of normality, often resulting in some form of novelty score. The score is typically compared to a decision threshold, and the test data are then classified to be "novelty" if the threshold is exceeded [14]. The state-of-the-art novelty detection approaches of fault diagnostics for engineering assets can be roughly classified into four groups: 1) probabilistic-based (e.g., Gaussian mixture models), 2) domain-based (e.g., one-class SVMs), 3) distance-based (e.g., k-nearest neighbors), and 4) reconstruction-based (e.g., AE) [13], [15]–[17]. The approaches 1)–3) are based on standard ML methods, which are time-consuming in selecting and fine-tuning parameters. Their performances are not guaranteed for online learning with a large amount of data. In addition, for sensor data with complex structures and high noise level, it is difficult to use these methods to estimate probabilistic density functions or to apply distance-based classification due to intensive computation and difficult mathematical formulation. Differently, reconstruction-based approaches build a deep AE only based on normal samples and use the trained deep AE to reconstruct the testing samples. Those testing samples with large reconstruction errors are marked as "novelty." Such approaches present a strong learning ability to discover complex structures among the massive data without relying on much human knowledge and have emerged for novelty detection and anomaly detection.

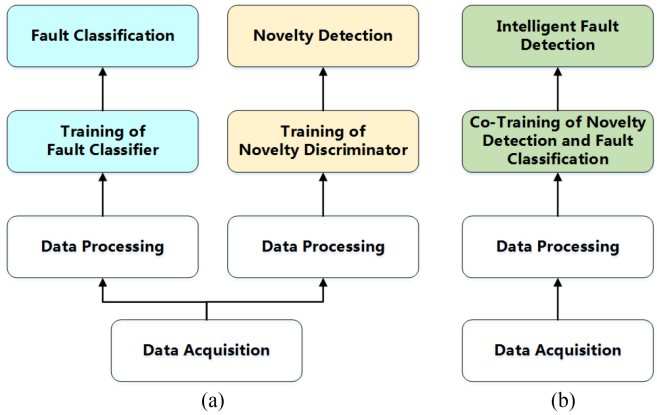


Fig. 1. (a) Classical scheme of fault detection and identification. (b) Proposed scheme of fault detection and identification.

Most deep AE-based methods for novelty detection calculate the novelty scores only based on reconstruction error of health state signals in the original space. These methods are applicable to discriminate between the known normal data and unknown faulty data because they share very distinguishable characteristics in the original space. But in most of time, it is difficult to use the reconstruction errors to differentiate the unknown fault types from known fault types when they share the similar characteristics in the original space. This implies that it is necessary to utilize the information of latent space to discriminate between known and unknown fault types. On the other hand, Novelty detection and fault classification are generally regarded as two independent research topics in existing body of the literature. Novelty detection and fault classification are commonly treated as two different problems and trained as two separate models (e.g., novelty discriminator and known fault classifier) in the classical FDD problems shown in Fig. 1(a). Such two-stage approaches have two common drawbacks: first, a false novelty detection result that is passed on to the fault classification model leads to further misclassification errors; and second, the overall approach is computationally expensive, where data processing, model selection, and fine tuning of model parameters are needed for two separate training processes. To the best knowledge of the authors, there are no existing intelligent fault diagnostic models that can handle both novelty detection and fault diagnosis tasks effectively in one joint DL model.

In this article, we address the aforementioned research gaps by developing a novel end-to-end DL-based fault diagnostic model named as deep variational AE classifier (DVAEC) capable of integrating novelty detection and fault classification into a single framework with high accuracy and computational efficiency. Specifically, we can detect unknown fault categories, whereas the accuracy of fault classification for known categories are maintained. The innovation of this approach is to take full advantage of multitask learning, which results in improved learning efficiency and prediction accuracy. During the training phase, the DVAEC learns a number of underlying latent variables with uncertainties that capture the underlying features of the input

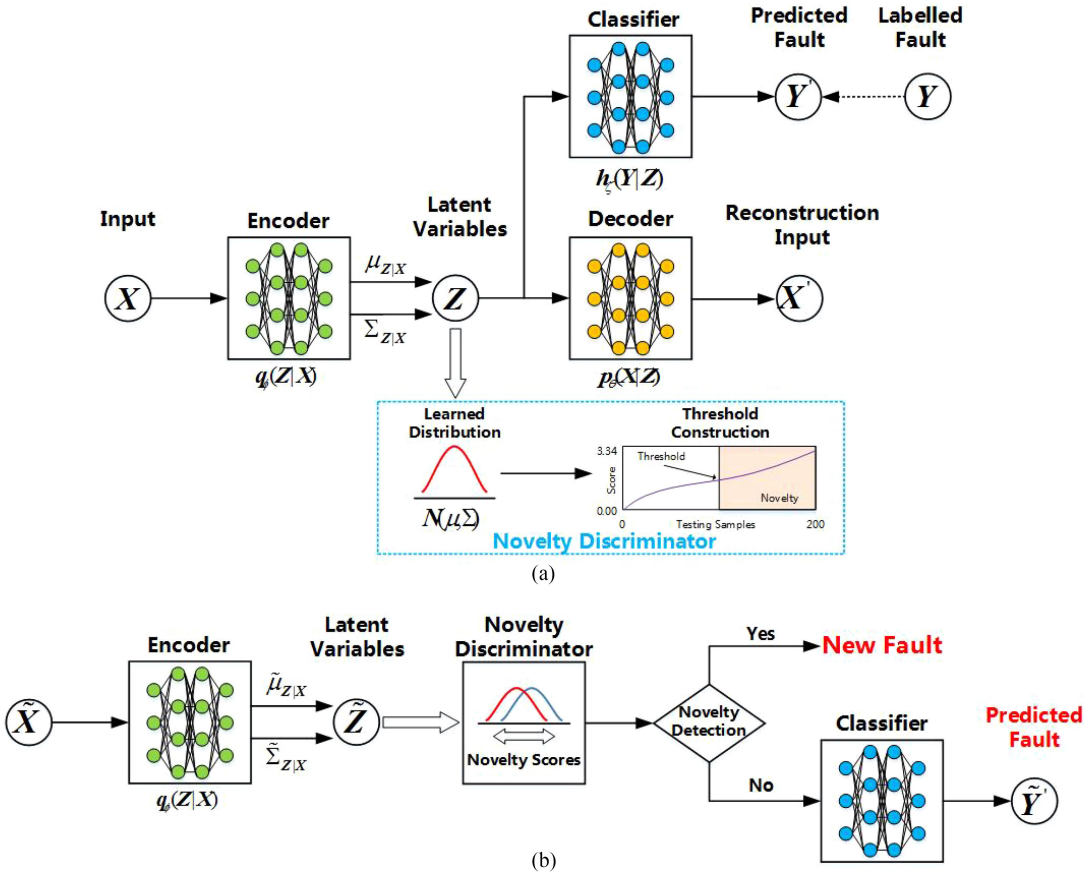


Fig. 2. Overall methodology framework. (a) Offline training process. (b) Online testing process.

data. These latent variables, as well as their associated uncertainties, are learned through self-supervision of a variational AE (VAE) network trained to reconstruct its own inputs. Since the latent variables with uncertainties can be considered as an effective representation of known classes in the training dataset, we make use of these information from the latent space for novelty detection. Meanwhile, the learned latent variables and their corresponding distribution are simultaneously supervised through a DNN for fault classification. In this way, we establish a unified DNN model that can cotrain the models of novelty detection and fault classification in one overarching DL architecture, as shown in Fig. 1(b). What is significantly different with the standard DL-based fault classifiers is that DVAEC can intelligently discriminate those new fault classes unseen in the training samples. We use two case studies of bearing health monitoring to validate the effectiveness of the proposed model.

The major scientific contributions of this work are as follows.

1) Establish a unified framework for simultaneous novelty detection and fault classification in one joint DL process, achieving significant learning efficiency in data-driven FDD.
2) Provide a new end-to-end data-driven approach to detect the unknown (unprecedented) fault classes by learning the distribution in a latent space that characterizes the training data without heavy manual feature extraction.
3) Verify and validate the accuracy and effectiveness of the proposed framework using two bearing condition monitoring datasets [18].

We show that the proposed framework and algorithms exhibit great potential for FDD applications for a wide variety of industrial engineering systems and components.

The remainder of this article is organized as follows. In Section II, the proposed DVAEC framework and models are described in detail. Section III demonstrates the proposed methods with two benchmarking datasets of bearings, followed by results and performance analysis. Finally, Section IV concludes this article.

II. Proposed Method

A. Method Overview

The overall architecture of the proposed DVAEC framework is composed of two phases: an offline training phase and an online monitoring phase shown as Fig. 2(a) and (b), respectively. In the offline training phase, a DNN with multibranch architecture—DVAEC—is designed to jointly train the fault classifier and VAE. Given the observed input data X and labels of known faulty samples Y, the VAE network is comprised of two parts, shown as an encoder and a decoder in Fig. 2(a). The encoder is responsible for learning a mapping $q_{Z|X}(X;\phi)$ from input X

directly to low-dimensional latent variables Z that maximally explain as much of the data as possible. We assume that the $q_{Z|X}(X;\phi)$ is normally distributed with mean and covariance matrix defined by a DNN with parameters ϕ. The decoder is responsible for learning the inverse mapping $p_{X|Z}(Z;\boldsymbol{\theta})$ with parameters $\boldsymbol{\theta}$, which takes the aforementioned latent variables Z as input and reconstructs the original input. As opposed to a standard AE model with deterministic latent variables, the latent space of a VAE learned from the training set X can be considered an effective representation of the distribution of input X. We make use of this latent representation of the distribution of input X and distance metrics in the latent space to train a novelty discriminator, which is capable of distinguishing the unseen fault classes from the known fault classes. During the online monitoring phase, the DVAEC provides a reliable and accurate fault identification of both seen and unseen (novel) fault categories. The details of DVAEC will be described in the following sections.

B. Variational AE

AE is one type of unsupervised DNNs capable of learning lower dimensional representations of the input data. Given an input signal x_i from a dataset $\{x^{(i)}\}_{i=1}^{N}$, the encoder network f_ϕ transforms the input data into codes in a low-dimensional space and then the decoder network $g_{\boldsymbol{\theta}}$ reconstructs the codes back into the original input space as closely as the input data by finding the parameter sets ϕ and $\boldsymbol{\theta}$ in order to minimize the reconstruction error $L(\boldsymbol{x},\boldsymbol{x'})$ over the N training examples, where $L(\boldsymbol{x},\boldsymbol{x'})$ is a loss function that measures the difference between $\boldsymbol{x}$ and $\boldsymbol{x'}$ such as $||\boldsymbol{x}-\boldsymbol{x'}||_2$

$$\mathcal{L}_{\mathrm{AE}}(\phi,\boldsymbol{\theta})=\frac{1}{N}\sum_{i=1}^{N}L(x^{(i)}-g_{\boldsymbol{\theta}}(f_\phi(x^{(i)}))). \tag{1}$$

VAE is a generative model that can learn the hidden representations of input by reconstructing the inputs. It has a similar encoder–decoder architecture with AE [19]. The major difference between VAE and AE is that the latent representation $\boldsymbol{z}$ learned by a VAE network is probabilistic instead of deterministic. Note that the prior over the latent representation $\boldsymbol{z}$ is defined as the isotropic unit Gaussian $p_{\boldsymbol{\theta}}(\boldsymbol{z})=N(\boldsymbol{z};0,\boldsymbol{I})$. The encoder and the decoder of VAE are probabilistic and given by $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$, an approximate posterior, and $p_\theta(\boldsymbol{x}|\boldsymbol{z})$, likelihood of the data $\boldsymbol{x}$ given the latent variable $\boldsymbol{z}$, respectively. Since the true posterior $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$ is intractable, it is assumed that the intractable posterior takes on an approximate multivariate Gaussian form with a diagonal covariance

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})=\mathcal{N}(\boldsymbol{z};\boldsymbol{\mu}_{\boldsymbol{z}|\boldsymbol{x}},{\boldsymbol{\sigma}_{\boldsymbol{z}|\boldsymbol{x}}}^{\boldsymbol{2}}\boldsymbol{I}) \tag{2}$$

where $\boldsymbol{\mu}_{\boldsymbol{z}|\boldsymbol{x}}$ and $\boldsymbol{\sigma}_{\boldsymbol{z}|\boldsymbol{x}}$ are computed from $\boldsymbol{x}$ through a neural network with variational parameter set $\boldsymbol{\phi}$.

The likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ is also given by a multivariate Gaussian whose probabilities are obtained from $\boldsymbol{z}$

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})=\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{z}},{\boldsymbol{\sigma}_{\boldsymbol{x}|\boldsymbol{z}}}^{\boldsymbol{2}}\boldsymbol{I}) \tag{3}$$

where $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{z}}$ and $\boldsymbol{\sigma}_{\boldsymbol{x}|\boldsymbol{z}}$ are computed from $\boldsymbol{x}$ through a neural network with variational parameter set $\boldsymbol{\theta}$.

The goal of the VAE is to minimize the difference between the approximated posterior $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ and the true intractable posterior $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$, which can be evaluated by the Kull back–Leibler (KL) divergence $D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}))$. Since this KL divergence cannot be computed directly, we can minimize it by maximizing the sum of variational lower bound on the marginal likelihood of individual data points $\boldsymbol{x}_i, i=1,\ldots,n$ [20]. Therefore, the cost function of VAE can be written as

$$\begin{aligned}\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\phi},\boldsymbol{\theta};\boldsymbol{x})=&\sum_{i=1}^{N}\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}^{(i)})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}|\boldsymbol{z})]\\&-\sum_{i=1}^{N}D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}^{(i)})||p_{\boldsymbol{\theta}}(\boldsymbol{z}))\end{aligned} \tag{4}$$

where the sum is computed over the training samples $\{\boldsymbol{x}_i\}_{i=1}^{N}$. The first term of the right-hand side of (4) can be regarded as a reconstruction error forcing the model to reconstruct the inputs and the second term acts as regularization term.

Specifically, the second term of the cost function is a KL term that is analytical and differentiable. A more specific form of this term can be expressed as

$$\begin{aligned}&\sum_{i=1}^{N}D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}^{(i)})||p_{\boldsymbol{\theta}}(\boldsymbol{z}))\\&=-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{J}(1+\log((\boldsymbol{\sigma}_j^{(i)})^2)-(\boldsymbol{\mu}_j^{(i)})^2)\end{aligned} \tag{5}$$

where J is the dimensionality of latent variables $\boldsymbol{z}$ and j denotes the jth element of $\boldsymbol{z}$.

Given the aforementioned cost function, the goal is to optimize the lower bound w.r.t. both ϕ and $\boldsymbol{\theta}$, which are as follows.

1) The gradient of the cost function w.r.t. ϕ: Since $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the neural networks of input $\boldsymbol{x}$ with parameters ϕ, the second term of (4) can be easily differentiated w.r.t. ϕ in the form of: $\sum_{i=1}^{N}\nabla_{\boldsymbol{\phi}}D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}^{(i)})||p_{\boldsymbol{\theta}}(\boldsymbol{z}))$. However, the first term of (4) requires estimation sampling, which makes the gradient of this term w.r.t. $\boldsymbol{\phi}$ problematic. Here, this problem is solved by expressing the random variable $\boldsymbol{z}$ as a deterministic variable $\boldsymbol{z}=\boldsymbol{\mu}+\boldsymbol{\sigma}\otimes\epsilon=g(\boldsymbol{\sigma},\boldsymbol{x};\boldsymbol{\phi})$, with $\epsilon\sim p(\epsilon)=\mathcal{N}(0,I)$, which is also known as "reparameterization trick" [19]. Therefore, the gradient of the cost function w.r.t. $\boldsymbol{\phi}$ can be written as

$$\begin{aligned}\nabla_{\boldsymbol{\phi}}\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\phi},\boldsymbol{\theta};\boldsymbol{x})=&\sum_{i=1}^{N}\mathbb{E}_{p(\epsilon^{(i)})}\nabla_{\boldsymbol{\phi}}\log p_{\boldsymbol{\theta}}\\&(\boldsymbol{x}^{(i)}|g(\boldsymbol{\sigma}^{(i)},\boldsymbol{x}^{(i)};\boldsymbol{\phi}))\\&-\sum_{i=1}^{N}\nabla_{\boldsymbol{\phi}}D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}^{(i)})||p_{\boldsymbol{\theta}}(\boldsymbol{z})).\end{aligned} \tag{6}$$

2) The gradient of the cost function w.r.t. $\boldsymbol{\theta}$

$$\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\phi},\boldsymbol{\theta};\boldsymbol{x})=\sum_{i=1}^{N}\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}^{(i)})}\nabla_{\boldsymbol{\theta}}\log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}|\boldsymbol{z}). \tag{7}$$

With the gradients of the cost function $\nabla_{\phi,\theta}\mathcal{L}_{\text{VAE}}(\phi,\theta;\boldsymbol{x})$, the optimization methods, such as minibatch gradient descent, RMSprop, and ADAM, can be applied to train VAE as an unsupervised method to learn hidden representations of the inputs [19]. The hidden representations $\boldsymbol{z}$ are J dimensional variables sampled from a latent distribution $q_{\phi}(\boldsymbol{z}|\boldsymbol{x})$. In VAE, the projection of the input data to the latent space (via the encoder) captures the essential data characteristics, allowing execution of the regression task in a much lower dimensional space. On the other hand, the latent distribution can potentially be used to define novelty metrics for novelty detection.

C. Supervised Fault Classifier

In this section, we explain how the classic DL-based fault classifier works for a multiclass classification problem and discuss its overconfidence problem.

Supposing that we have a training dataset $\{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ (raw data or extracted features) with its corresponding fault label set $\{\boldsymbol{y}^{(i)}\}_{i=1}^{N}$, where $\boldsymbol{y}^{(i)} \in \{1,2,\ldots,K\}$. The DNN-based classifier usually consists of three or more layers and the softmax regression is often conducted in the final layer with multiclass classification problem [21]. The goal of the classifier is to learn the mappings between the training samples and their corresponding labels. Given a test sample $\boldsymbol{x}$, the model is able to estimate the probability of the label taking on each of the K classes. Thus, the output of the network is K estimated probabilities in the following form:

$$h_{\phi,\boldsymbol{\omega}}(\boldsymbol{x}^{(i)}) = \begin{bmatrix} p(\boldsymbol{y}(i)=1|\boldsymbol{x}^{(i)};\phi,\boldsymbol{\omega}) \\ p(\boldsymbol{y}(i)=2|\boldsymbol{x}^{(i)};\phi,\boldsymbol{\omega}) \\ \vdots \\ p(\boldsymbol{y}(i)=K|\boldsymbol{x}^{(i)};\phi,\boldsymbol{\omega}) \end{bmatrix} = \frac{1}{\sum_{j=1}^{K} e^{\boldsymbol{\omega}_j^T g_{\phi}(\boldsymbol{x}^{(i)})}} \begin{bmatrix} e^{\boldsymbol{\omega}_1^T g_{\phi}(\boldsymbol{x}^{(i)})} \\ e^{\boldsymbol{\omega}_2^T g_{\phi}(\boldsymbol{x}^{(i)})} \\ \vdots \\ e^{\omega_K^T g_{\phi}(\boldsymbol{x}^{(i)})} \end{bmatrix} \tag{8}$$

where $\boldsymbol{\omega} = [\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \ldots, \boldsymbol{\omega}_K]$ are the parameters of the final softmax layer. $g_{\phi}(.)$ is the DNN with parameter ϕ before the final softmax layer. The cost function of softmax regression model can be expressed as

$$\mathcal{L}_C(\phi,\boldsymbol{\omega};\boldsymbol{x},\boldsymbol{y}) = -\left[\sum_{i=1}^{N}\sum_{k=1}^{K} l\{\boldsymbol{y}^{(i)=k}\}\log\frac{e^{\boldsymbol{\omega}_k^T g_{\phi}(\boldsymbol{x}^{(i)})}}{\sum_{j=1}^{K} e^{\boldsymbol{\omega}_j^T g_{\phi}(\boldsymbol{x}^{(i)})}}\right] \tag{9}$$

where $l(.)$ represents the indicator function returning 1 if the condition is true, and 0 otherwise.

In order to train such models for multiclass classification, we can obtain the gradient of the cost function $\nabla_{\zeta}\mathcal{L}_C(\zeta;\boldsymbol{x},\boldsymbol{y})$, where for convenience, let $\zeta = \{\phi,\boldsymbol{\omega}\}$. Then, we can minimize the cost function over the training samples using an optimization method, such as gradient descent. By stacking functional layers (such as convolutional layers, recurrent neuron layers, or multilayer perceptron) on the softmax regression layer, the DL-based classifiers show strong capability of representation learning and classification. These DL-based classifiers can provide high performance in a variety of applications, so long as the data seen at test time is similar to the training data. However, when the testing samples are unseen in the training phase, DNN classifiers tend to classify these novel input as one of the existing classes and also results in overconfidence predictions on new test examples [22], [23]. Therefore, ideally, we would like that the classifier, in addition to its generalization ability, be able to detect novel inputs [24]. In the next section, we will introduce the proposed method, which can properly solve the aforementioned issue.

D. Deep VAE Classification

We propose a new DVAEC framework, which can learn a shared encoding representation for two tasks simultaneously: first, capture complex distributions of classes in the training dataset in a latent space and use the learned distribution to determine novelty score metrics to detect novel (unknown) samples; second, learn the deep mappings between the input data and labels to classify known faults. All these networks counteract each other, enabling to learn a more shared and generalized latent distribution $\boldsymbol{z}$ that jointly optimizes VAE and classifier. We will first introduce how we train these two tasks simultaneously. We show that DVAEC can be optimized by using backpropagation similarly as the standard neural networks. Next, we define a new novelty metric defined by the learned distribution from VAE instead of reconstruction error from the standard AE. The novelty metric can be used to construct the novelty threshold, which helps to identify those testing data from unseen or unknown classes.

1) Joint Learning Algorithm: The DVAEC should satisfy two criteria: first, reconstruct well the training data, which can be viewed as an ideal approximate of the distributions of the training set; second, classify well the training data. Therefore, DVAEC is based on a DNN architecture that has two pipelines with a shared encoding part. The first pipeline is a deep VAE network for training data reconstruction with parameters $\{\phi,\boldsymbol{\theta}\}$, whereas the second one is a fault classifier for in-distribution classification with parameters $\{\phi,\boldsymbol{\zeta}\}$ with parameters. Given a training dataset $\boldsymbol{x} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ with its corresponding label set $\boldsymbol{y} = \{\boldsymbol{y}^{(i)}\}_{i=1}^{N}$, where $\boldsymbol{y}^{(i)} \in \{1,2,\ldots,K\}$. By combining the cost functions of the VAE (4) and supervised classifier (9) together into one, the objective of the DVAEC can be expressed as

$$\mathcal{L}_{\text{total}} = c_1\mathcal{L}_{\text{VAE}}(\phi,\boldsymbol{\theta};\boldsymbol{x}) + c_2\mathcal{L}_C(\phi,\boldsymbol{\zeta};\boldsymbol{x},\boldsymbol{y}) \tag{10}$$

where c_1 and c_2 are used to weight the importance of each cost function in case that one individual component overpowers the other during training.

All these networks counteract each other, which leads to learn a more shared and generalized latent distribution z that jointly optimizes the VAE and the classifier. As shown in Sections II-B and II-C, we can easily get the gradients of each parameter. Then, we can use stochastic gradient descent to optimize the objective function. The parameters of the DVAEC can be updated as

followings:

$$\begin{aligned}\boldsymbol{\phi} &= \boldsymbol{\phi} - \alpha_1(c_1\nabla_{\boldsymbol{\phi}}\mathcal{L}_{\text{VAE}}(\boldsymbol{\phi},\boldsymbol{\theta};\boldsymbol{x}) + c_1\nabla_{\boldsymbol{\phi}}\mathcal{L}_C(\boldsymbol{\phi},\boldsymbol{\zeta};\boldsymbol{x},\boldsymbol{y}))\\ \boldsymbol{\theta} &= \boldsymbol{\theta} - \alpha_2 c_1\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{VAE}}(\boldsymbol{\phi},\boldsymbol{\theta};\boldsymbol{x})\\ \boldsymbol{\zeta} &= \boldsymbol{\zeta} - \alpha_3 c_2\nabla_{\boldsymbol{\zeta}}\mathcal{L}_C(\boldsymbol{\phi},\boldsymbol{\zeta};\boldsymbol{x},\boldsymbol{y})\end{aligned} \tag{11}$$

where α_1, α_1, and α_1 are the learning rates.

Once the training is completed, the optimal parameters are used to form the DVAEC to learn the distribution of the training dataset and also perform well on those fault categories similar to the training samples.

2) Novelty Threshold Construction in Latent Space: We treat the DVAEC encoder network as a posterior model inference of the parameters $\boldsymbol{\phi}$ and the latent variables $\boldsymbol{z}$ are sampled from this posterior model. The latent space of the DVAEC serves as an effective representation of the distribution of training data. The idea here is to construct a reference dataset by transforming the training samples to a latent space while the latent distribution of each test sample should also be inferred. Then, we use a distance measure to find the closest sample from the reference dataset to the test sample in the latent space. Such a distance measure in latent space is used as a novelty score. A novel class is detected if the novelty score exceeds the novelty threshold. In this section, we construct a Bhattacharyya distance based novelty metric for novelty detection.

The Bhattacharyya distance is defined as $D_B(p,q) = -\ln(\text{BC}(p,q))$, where $\text{BC}(p,q) = \int\sqrt{p(z)q(z)}dz$ is the Bhattacharyya coefficient of distributions p and q. This approach utilizes information about the fully learned distributions, computing the amount of the overlap between them. The proposed novelty score is defined as the Bhattacharyya distance between the latent-space distribution of a test sample $\overline{\boldsymbol{x}}$ and the most similar latent-space distribution of a training sample $\boldsymbol{x}$ from training set $\boldsymbol{X}$

$$D(\overline{\boldsymbol{x}}) = \min_{\boldsymbol{x}\in\boldsymbol{X}} D_B(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\overline{\boldsymbol{x}}), q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})) \tag{12}$$

where $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\overline{\boldsymbol{x}}) = \mathcal{N}(\boldsymbol{z};\overline{\boldsymbol{\mu}},\overline{\boldsymbol{\Sigma}})$ and $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z};\boldsymbol{\mu},\boldsymbol{\Sigma})$, in which $\overline{\boldsymbol{\mu}}, \boldsymbol{\mu}$ and $\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Sigma}$ are the means and the covariances of distributions of the testing sample and the training sample, respectively. In this case, $D_B(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\overline{\boldsymbol{x}}), q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}))$ can be expressed as

$$\begin{aligned}D_B(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\overline{\boldsymbol{x}}), q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})) = \frac{1}{8}(\overline{\boldsymbol{\mu}}-\boldsymbol{\mu})^T{\boldsymbol{\Sigma}'}^{-1}(\overline{\boldsymbol{\mu}}-\boldsymbol{\mu})\\ + \frac{1}{2}\ln\frac{\det\boldsymbol{\Sigma}'}{\sqrt{\det\overline{\boldsymbol{\Sigma}}\det\boldsymbol{\Sigma}}}\end{aligned} \tag{13}$$

where $\boldsymbol{\Sigma}' = \frac{\overline{\boldsymbol{\Sigma}}+\boldsymbol{\Sigma}}{2}$ and $\det$ is the determinant of the matrix.

In order to determine the novelty threshold, we use 15% of training samples as a validation set and guarantee that the validation set is a class balance dataset similar with the training set. Then, the novelty scores of the validation set are computed using (13) and the novelty score set $\mathcal{D}(\boldsymbol{x})$ is obtained. We set a novelty threshold γ at the δth percentile of $\mathcal{D}(\boldsymbol{x})$, determined by the validation set. Thus, for any new testing data $\boldsymbol{x}_{\text{test}}$, novelty is detected if

$$D(\boldsymbol{x}_{\text{test}}) > \gamma. \tag{14}$$

Algorithm 1: Fault Diagnosis Algorithm of the DVAEC.

// **Joint Learning:**
Input: $\{\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}\}_{i=1}^{N} \in \mathcal{D}_{train}$.
$\boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{\zeta} \leftarrow$ Initialize the parameter of the DVAEC.
repeat
  $\mathcal{L}_{total} \leftarrow$ Forward pass according to (10);
  $g \leftarrow \nabla_{\boldsymbol{\phi},\boldsymbol{\theta},\boldsymbol{\zeta}}\mathcal{L}_{total}$
  $\boldsymbol{\phi}, \boldsymbol{\theta}, \boldsymbol{\zeta} \leftarrow$ Update parameters using gradient;
until convergence
Output: The DVAEC parameters $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ and $h_{\boldsymbol{\zeta}}(\boldsymbol{y}|\boldsymbol{z})$
// **Novelty Threshold Construction:**
Input: $\boldsymbol{\phi}$ and $\{\boldsymbol{x}^{(j)}\}_{j=1}^{M} \in \mathcal{D}_{val}$,
for $j = 1 \rightarrow M$ **do**
  for $i = 1 \rightarrow N$ **do**
    $D_B^{(i)} \leftarrow D_B(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}_{(j)}), q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}_{(i)}))$
  end for
  $D^{(j)} \leftarrow$ Find the minimum value in $\{D_B(i)\}_{i=1}^{N}$.
end for
$\gamma \leftarrow \delta th$ percentile of $\{D_{(j)}\}_{j=1}^{M}$
Output: Novelty threshold γ
// **Hierarchical Diagnosis Strategy:**
Input: $\{\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)}\}_{i=1}^{K} \in \mathcal{D}_{test}, \boldsymbol{\phi}, \boldsymbol{\zeta}$ and Threshold γ
for $k = 1 \rightarrow K$ **do**
  for $i = 1 \rightarrow N$ **do**
    $D_B^{(i)} \leftarrow D_B(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}_{(k)}), q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}_{(i)}))$
  end for
  $D^{(k)} \leftarrow$ Find the minimum value in $\{D_B(i)\}_{i=1}^{N}$.
  if $D^{(k)} > \gamma$ **then**
    $\widehat{\boldsymbol{y}}^{(k)} \leftarrow$ Novel Class
  else
    $\widehat{\boldsymbol{y}}^{(k)} \leftarrow h_{\boldsymbol{\zeta}}(\boldsymbol{y}|\boldsymbol{z})$; Classified as known classes
  end if
end for

If the sample is identified as nonnovel, the DVAEC classifies it as one of the pre-existing classes.

Algorithm 1 summarizes the DVAEC algorithm.

III. Application to Bearing Health State Diagnosis

In this section, two case studies are conducted to validate the effectiveness of the DVAEC for bearing health state diagnosis. The test of the proposed methodology is performed using a sequential diagnosis scheme, which implies that each testing sample is first evaluated whether it is a novel fault class and then, if it is labeled as known class, the specific fault class is identified. In this case, the detection error of novelty detection will propagate into the classification accuracy and influence the performance of fault classification of known classes. Therefore, to evaluate the performance of each stage, the novelty detection and fault classification are evaluated and compared with other methods separately.

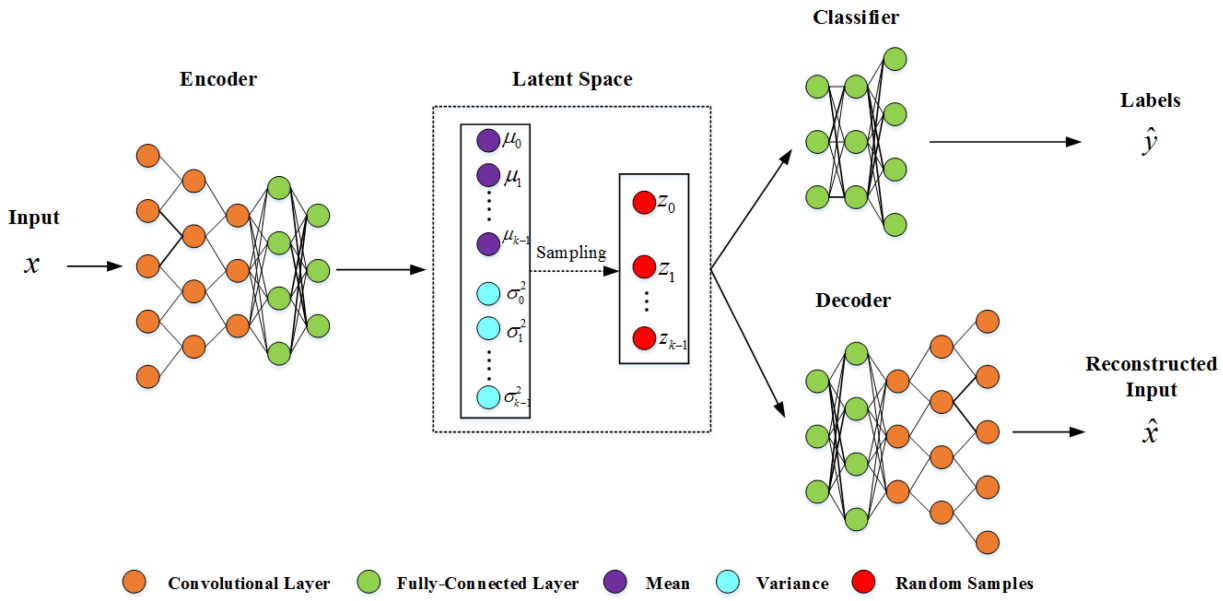


Fig. 3. Schematic of the DVAEC model.

A. Experimental Settings

In Section II, the proposed DVAEC method is in a general form requiring no specifications for the input data. Hence, the DVAEC can be applied on a wide range of data type, such as time domain, frequency domain, or time-frequency domain signals. In this study, the short-time Fourier transform (STFT) is used to obtain a 2-D time-frequency matrix from sensor data and the raw signals refer to the signals in the time-frequency domain. The main reason is that the time-frequency matrix can represent the nonstationary vibration signals in both time and frequency domain, and those characteristics may provide clear information about the health conditions of rotating machinery and enhance the quality of inputs. Spectrograms are a visual representation of the STFT, where the horizontal and vertical axes are time and frequency, respectively, and the color scale of the image indicates the amplitude of the frequency.

In the VAE pipeline, a network composed of convolutional layers and fully connected layers is built. For the classifier pipeline, dense layers and softmax layer are used. The reason why we use the convolutional layers is that the input of the model is the spectrogram, which can be regarded as color scale image, and the convolutional layers are useful for extracting the hidden features from these images. The schematic of the DVAEC architecture is shown as Fig. 3. We perform a hyperparameter search by varying the dimensionality of the latent space, the depth of the architecture, and the number of hidden layers to find the model that achieves the highest AUC score and classification accuracy. The detailed parameters of the resulting architecture are listed as Table I.

All the models in this article were trained on a NVIDIA Geforce GTX 1080Ti GPU and the Adam optimizer with the following hyperparameters: learning rate $\alpha = 0.001$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. The number of latent variables is $J = 10$ for the rest of the analysis. The weights of c_1 and c_2 in (10) are selected as $c_1 = 1$ and $c_2 = 0.5$, which provide a reasonable tradeoff in importance between the classifier and novelty detection functions, wherein no individual loss component overpowers the others during training.

TABLE I
PARAMETERS OF THE DVAEC MODEL

Name	Layer	Output	Activation	Filter	Stride
Input	-	$96 \times 96 \times 4$	Identity	-	-
Encoder	Conv1	$96 \times 96 \times 4$	Relu	3×3	1×1
	Conv2	$48 \times 48 \times 32$	Relu	3×3	2×2
	Maxpool1	$24 \times 24 \times 32$	-	2×2	1×1
	Conv3	$24 \times 24 \times 64$	Relu	3×3	1×1
	Maxpool2	$12 \times 12 \times 64$	-	2×2	1×1
	Conv4	$12 \times 12 \times 128$	Relu+Flatten	3×3	1×1
	Dense1	18432	Relu	-	-
	Dense2	128	Relu	-	-
Latent Space	Dense3	dim z	Identity	-	-
Decoder		Mirror of Encoder Part			
Classifier	Dense4	256	Relu	-	-
	Dense5	128	Relu	-	-
	Dense6	64	Relu	-	-
	Dense7	4	Softmax	-	-

B. Methods Comparison

The DVAEC is used to detect each testing sample in a sequential scheme such that if a testing sample is identified as nonnovel, a known class is then identified; otherwise, it is labeled as a novel class. This implies that the performance of the first step (novelty detection) will influence the prediction accuracy of the second step (fault classification of known set). The progressive validation results of the proposed DVAEC model are compared with other DL-based methods from two aspects: accuracy of novelty detection of novel samples and accuracy of fault classification. Two sets of performance metrics are considered as follows.

TABLE II
STATISTICAL FORMULA OF THE EXTRACTED FEATURES FOR COMPARATIVE METHODS

Name	Formula	Name	Formula
Square-mean-root:	$p_1 = (\frac{1}{N}\sum\sqrt{\lvert x\rvert})^2$	Crest Factor:	$p_6 = \frac{\max(\lvert x\rvert)}{p_3}$
Mean-absolute:	$p_2 = \frac{1}{N}\sum\lvert x\rvert$	Shape Factor:	$p_7 = \frac{p_3}{p_2}$
Root-mean-Square:	$p_3 = \sqrt{\frac{1}{N}\sum x^2}$	Clearance Factor:	$p_8 = \frac{\max(\lvert x\rvert)}{p_1}$
Kurtosis:	$p_4 = \frac{\frac{1}{N}\sum x^4}{p_3^4}$	Impulse Indicator:	$p_9 = \frac{\max(\lvert x\rvert)}{p_2}$
Skewness:	$p_5 = \frac{\frac{1}{N}\sum x^3}{p_3^3}$		

1) Novelty detection (unknown faults): We used the area under the curve of the receiver operating characteristic (AUC) and F1 score as the quality metric, which are the most common accuracy metric for novelty detection in the literature.
2) Fault classification (known faults): We use classification accuracy as the quality metric to validate the performance of the fault classification regarding to the known set.

To evaluate the diagnostic effectiveness of the proposed method, we compare the novelty detection and fault classification results of our proposed model with other standard methods from the related literature according to the aforementioned evaluation metrics.

1) As for the baseline of novelty detection, we consider the standard novelty detection methods: One-class SVMs [13], [15], [25] and deep AE models with deterministic latent variables. The reconstruction error is employed as a measure of normality versus novelty [17].
2) Fault classification: We consider two conventional ML-based fault classification methods: SVMs and DNNs. The five-layer DNNs model [9] for fault classification was used for comparison.

Standard bearing fault diagnosis methods mainly rely on some hand-crafted feature extraction methods. Statistical analysis is one of the effective methods to extract features because these common standard statistical features can represent the characteristics of bearings operating in different health conditions. In this method demonstration, nine widely used feature indexes are used for the comparative analysis, as shown in Table II [13], [26].

C. Case 1: SEU Bearing Dataset

In this case study, we test the DVAEC model using a bearing dataset collected from a drivetrain dynamic simulator (DDS) [27]. The test stand for DDS consists of four main components: motor, planetary gearbox, parallel gearbox, and break system. The test rolling element bearing supports the shaft for the planetary gearbox. Vibration signals are collected from the planetary gearbox at a sampling rate of 12 000 Hz. The speed load configuration of this experiment was set to 30 Hz-2 V. Three fault types occurs in the different locations of the test bearing: crack in the inner ring, crack in the outer ring, and cracks in both inner and outer rings [28], [29].

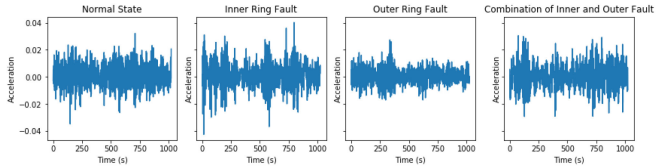


Fig. 4. Time-domain illustration of samples created from bearing data.

TABLE III
TRAINING AND TESTING DATA SPLIT OF BEARING HEALTH STATE SAMPLES FOR DIFFERENT SCENARIOS IN CASE 1

scenario Index	Training Set	Testing Dataset	
		Known Set	Novel Set
1	(N)	(N)	(IR)
2	(N, IR)	(N, IR)	(OR)
3	(N, IR, OR)	(N, IR, OR)	(C)
4	(N, IR, OR, C)	(N, IR, OR, C)	\

TABLE IV
NOVELTY DETECTION PERFORMANCE IN CASE 1

	OCSVM	CAE	DVAEC
scenario 1			
AUC	0.54	0.91	0.97
F1	0.35	0.86	0.99
scenario 2			
AUC	0.49	0.81	0.83
F1	0.41	0.77	0.80
scenario 3			
AUC	0.54	0.70	0.91
F1	0.43	0.65	0.87

We prepare the dataset for the model testing as the following: First, four health conditions are obtained and labeled as normal condition (N), inner race fault (IR), outer race fault (OR), and combination of inner and outer ring faults (C). Second, we segmented the vibration signals of different health conditions into data chunks using tumbling time windowing. In this case, we can obtain 2048 data samples with a window of 1024 points for each health condition. In Fig. 4, we visualize four raw data chunks randomly selected from different health states. We set four scenarios to evaluate the capability of the proposed method to detect and classify the testing samples, which may include novel samples that do not belong to any of the existing known classes. Each scenario is composed of a training set and a testing set, where the testing set consists of samples from both known classes and novel classes, as shown in Table III. The settings of different scenarios are inspired by a progressing stage of the fault detection approach in the bearing FDD case study, where only samples of normal condition (N) are initially available, and new classes are progressively detected and incorporated into the original training set through DVAEC retraining.

Table IV shows the comparison between our method and other standard methods. Their novelty detection results of first three scenarios based on two evaluation metrics—AUC and F1—are displayed. Generally, our methods in scenarios 1 to 3 all outperform standard novelty detection methods, such as OCSVM and reconstruction-based CAE method. More specifically, the DL-based methods—DVAEC and CAE methods—in all three scenarios perform better than the OCSVM method. In this case

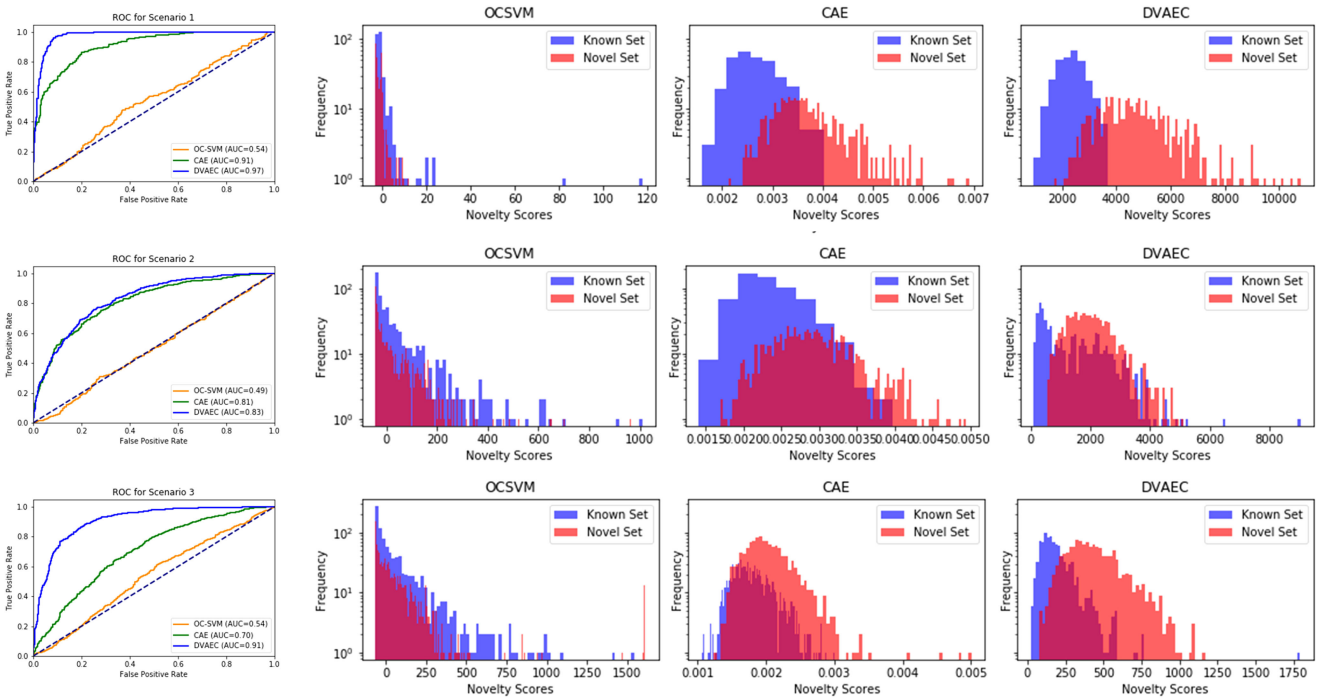


Fig. 5. Novelty detection performance for SEU dataset. First column: ROC curves of comparison methods in three scenarios; Second-fourth column: histogram of the novelty scores for normal and novel samples of comparison methods in three scenarios (testing samples in novel set have considerably higher novelty scores than those in known set).

TABLE V
FAULT CLASSIFICATION PERFORMANCE

Methods	Classification Accuracy (%)		
	scenario 2	scenario 3	scenario 4
SVM	63.00	58.75	68.55
DNN	94.00	90.40	92.15
DVAEC	100.00	99.75	100.00

study, it seems that OCSVM cannot directly define a decision boundary in the original space, which can be used to detect the novel fault category outside the frontier in the testing dataset. With the increase in the number of fault classes in the training samples (scenarios 2 and 3), which makes the composition of the training dataset more complex, our method can still provide a high performance of novelty detection, whereas the other methods cannot guarantee an accurate and robust novelty detection. This indicates that the latent distribution learned by DVAEC can successfully learn the hidden representation of the complex structured inputs. This latent distribution is beneficial to novelty detection. In Fig. 5, we further demonstrate the effectiveness of DVAEC by comparing the ROC curves and the histogram of the novelty score for known normal or fault categories and novel fault category. We can see that novel fault category (red part) has considerably higher novelty scores than known fault categories (blue part). The larger separation between the histograms of novel set and known set represents a better novelty detection ability based on the novelty scores. We can see that the DVAEC outperforms the baselines in scenarios 1–3.

The fault classification results are shown in Table V. We can see that our proposed method can produce a classification accuracy as almost 100% in three scenarios. To give more information about the classification performance of our proposed method, the confusion matrices of fault classification are shown in Fig. 6.

Compared with other standard fault classification methods, from Table V, we see that the proposed DVAEC outperforms standard methods and achieves higher classification accuracy for the known sets in scenarios 2 to 4. The standard DL methods, such as DNNs, can also achieve similarly high accuracy as our proposed method. This implies that DL model shows a superior performance over standard ML method relying on the handcrafted statistical features on discovering hidden characteristics of normal and fault condition .

To visualize the feature extraction ability of our proposed method, t-SNE [30] is used to graphically present features extracted by DVAEC under scenario 4, including four health states. At the same time, the raw data and statistic features by Table II are also plotted for comparison. Our method shows great feature extraction ability. As shown in Fig. 7(a), we can see that the normal vibration data are clustered separately far away from the mixture cluster of three fault categories. This implies that the normal category is naturally easy to be discriminated from other fault categories, whereas it is hard to make a classification among fault classes because they share similar data pattern and characteristics. The similar phenomenon can be observed from the visualization of statistic features in Fig. 7(b). However, in Fig. 7(c), the features extracted by our proposed method form clusters with distinguishable deviations with each other among different health states.

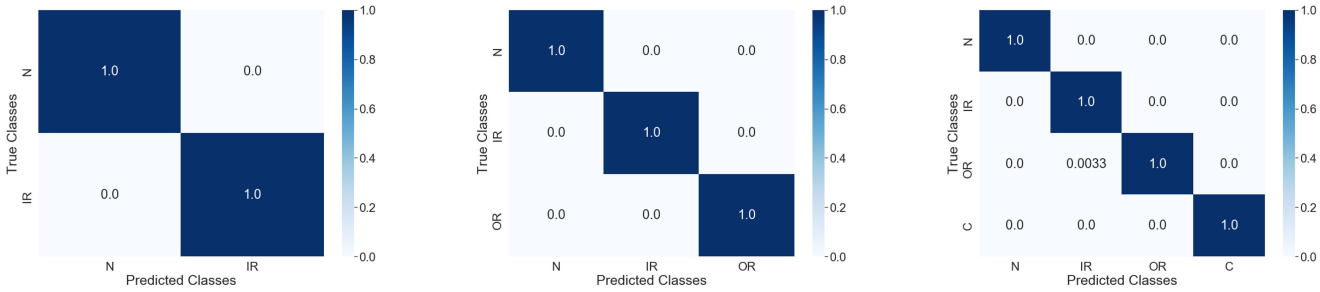


Fig. 6. Confusion matrices for known set fault classification of case 1 using our proposed method. (a) Scenario 2. (b) Scenario 3. (c) Scenario 4.

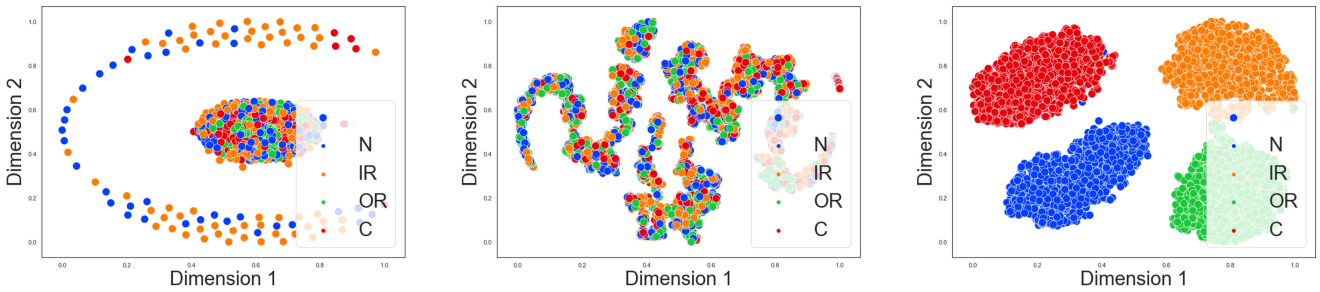


Fig. 7. Feature visualization using t-SNE for scenario 4. The color intensity of data points signifies the corresponding health states, including N, IR, OR, and C. We visualize the raw fault data and the extracted statistic features (see Table II) in (a) and (b), respectively. In (c), we plot the hidden variables sampled from the hidden distribution using our proposed method. (a) Raw data. (b) Statistic features. (c) DVAEC.

Another major concern for online fault detection is computational efficiency. In this case study, we obtain the average time for novelty detection and average time for known fault classification. The average time for novelty detection is approximately 0.05 s, whereas the average time for fault classification is 0.0018 s, which are acceptable for most industry applications.

D. Case 2: FEMTO Bearing Dataset

The proposed DVAEC can also be used to identify the different health states of a bearing during its whole life cycle. In case 2, FEMTO bearing dataset is composed of 17 run-to-failure data of rolling element bearings acquired from a PRONOSTIA platform. The PRONOSTIA platform conducts accelerated degradation tests in a few hours with a high-level radial force larger than the bearings' maximum dynamic load. During the tests, the rotating speed and the load of the bearing were kept stable under three different operating conditions. Two accelerometers and a thermocouple were used to capture the vibration signals and the temperatures of the bearings. The bearing useful life is considered to end when the amplitude of the vibration signal exceeds 20 g. It is worth noting that no assumption on the type of failure was given (no or little knowledge about the nature and the origin of the degradation: e.g., balls, inner or outer races, cage).

For the purpose of this article, we adapt the FEMTO bearing dataset based on the following rules.

1) We choose the vibration signals of one run-to-failure bearing dataset. The sampling frequency is 25.6 kHz. The vibration signals are sampled every 10 s and lasts 0.1 s every time. Fig. 8 shows the run-to-failure vibration signals ending with a roller defect of the bearing. We can observe that vibration amplitude of the bearing has an increasing trend over the time as a whole, which demonstrates the nonlinear process of bearing degradation over time. The nonlinearity between vibration signals and health status makes it difficult to assess the bearing health condition. The vibration amplitude increases gradually, which means that health status is getting worse over time [31].
2) The health states in the whole lifetime are divided according to the root mean square (rms) of the vibration signals, as illustrated in Fig. 9. The degradation severity of a bearing during its life cycle can be regarded as four

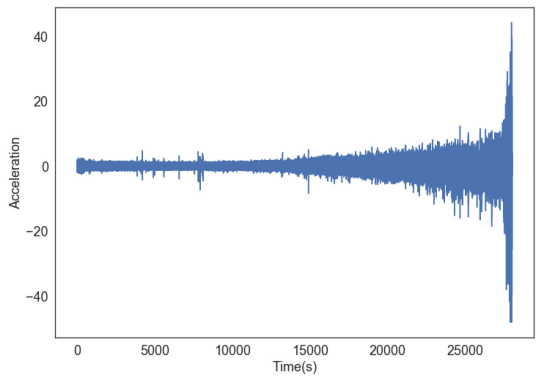


Fig. 8. Vibration signals of the selected bearing.

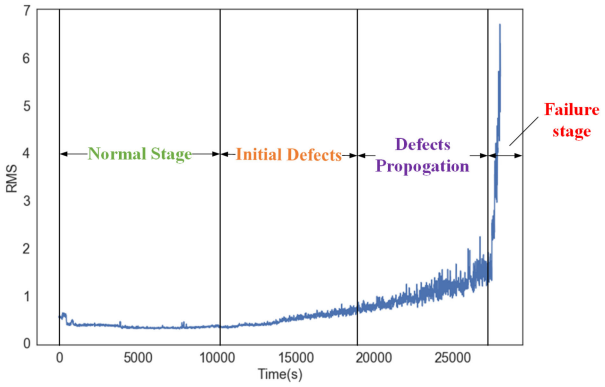


Fig. 9. RMS of vibration signals.

TABLE VI
TRAINING AND TESTING DATA SPLIT OF BEARING HEALTH STATE SAMPLES FOR DIFFERENT SCENARIOS IN CASE 2

scenario Index	Training Set	Testing Dataset	
		Known Set	Novel Set
1	(h_0)	(h_0)	(h_1)
2	(h_0, h_1)	(h_0, h_1)	(h_2)
3	(h_0, h_1, h_2)	(h_0, h_1, h_2)	(h_3)
4	(h_0, h_1, h_2, h_3)	(h_0, h_1, h_2, h_3)	\

fault classes denoted as h_0 to h_3: normal stage (h_0), initial defect (h_1), fault propagation stage (h_2), and failure stage (h_3) [32]. We segmented the vibration signals of different health states into data chunks using tumbling time windowing with 2048 data points such that we can obtain 1500 data samples for each health conditions of h_0–h_2 and 755 data samples for h_3. Then, we use 75% of data samples of each health state as training samples, 10% as validation samples, and 15% as testing samples.

3) We set four scenarios to evaluate the capability of the proposed method to detect and classify the testing samples. Each scenario is composed of a training set and a testing set, where the testing set consists of samples from both known classes and novel classes, as shown in Table VI. The settings of different scenarios are designed in the way that new state(s) emerge in a progressive manner during different stages of health condition assessment. For example, in the initial stage, assuming the bearing is in its initial stage of life cycle, only samples of normal condition (N) are available, and new classes (discrete degradation states) emerge gradually. During each stage, we use the training data of the degradation trajectory in the proposal DVAEC model for new state detection. Moving along with the degradation process, new data are incorporated into the training set and start the detection/classification process for the next stage. This process evolves over the entire bearing life cycle.

Table VII shows the comparison between our method and other standard methods. Their novelty detection results of first three scenarios based on two evaluation metrics—AUC and F1—are displayed. Generally, our methods in scenarios 1 and 2 outperform standard novelty detection methods, such as OCSVM and reconstruction-based CAE method, whereas

TABLE VII
NOVELTY DETECTION PERFORMANCE IN CASE 2

	OCSVM	CAE	DVAEC
scenario 1			
AUC	0.74	0.58	1.00
F1	0.70	0.36	1.00
scenario 2			
AUC	0.99	0.82	1.00
F1	0.97	0.65	1.00
scenario 3			
AUC	0.98	0.75	0.96
F1	0.94	0.44	0.90

TABLE VIII
FAULT CLASSIFICATION PERFORMANCE

Methods	Classification Accuracy(%)		
	scenario 2	scenario 3	scenario 4
SVM	91.78	84.30	80.53
DNN	93.11	90.22	98.43
DVAEC	99.55	99.11	99.33

our proposed method performs slightly worse than OCSVM, whereas still better than CAE method in scenario 3. The possible reason why our method performs slightly worse than OCSVM in scenario 3 is the discrepancy between failure signals and signals from other stages is relatively apparent, which makes the scenario 3 as the easiest novelty detection task among all scenarios. On the other hand, OCSVM method is relying on some manual time domain indexes, such as rms and shape factor. These time domain indexes may help OCSVM distinguish the different degradation patterns between failure stage and other stages. However, these time domain indexes may not be so effective in scenario 1, in which the discrepancy between normal signals and initial faulty signals is not so apparent. Compared with OCSVM and CAE, with the increase in the number of fault classes in the training samples, which makes the composition of the training dataset more complex, our proposed method can always guarantee a superior and robust novelty performance. The possible reason why OCSVM and CAE both have bad performance in scenario 1 is the discrepancy between normal signals and initial faulty signals is not so apparent, which makes the scenario 1 as the most difficult novelty detection task among all scenarios. In Fig. 10, we further demonstrate the effectiveness of DVAEC by comparing the ROC curves and the histogram of the novelty score for known set and novel set in testing dataset. We can see that the larger separation between the histograms of novel set (red part) and known set (blue part) represents a better novelty detection ability based on the novelty scores. We observe that the DVAEC outperforms the baselines in scenarios 1 and 2, whereas the DVAEC works slightly worse than OCSVM but still has a great performance in scenario 3.

The fault classification results are shown in Table VIII. We can see that our proposed method can produce a classification accuracy as over 99% in three scenarios. To give more information about classification performance of our proposed method, the confusion matrices of fault classification are shown in Fig. 11.

By comparison, from Table VIII, we see that the proposed DVAEC outperforms standard methods and achieves higher

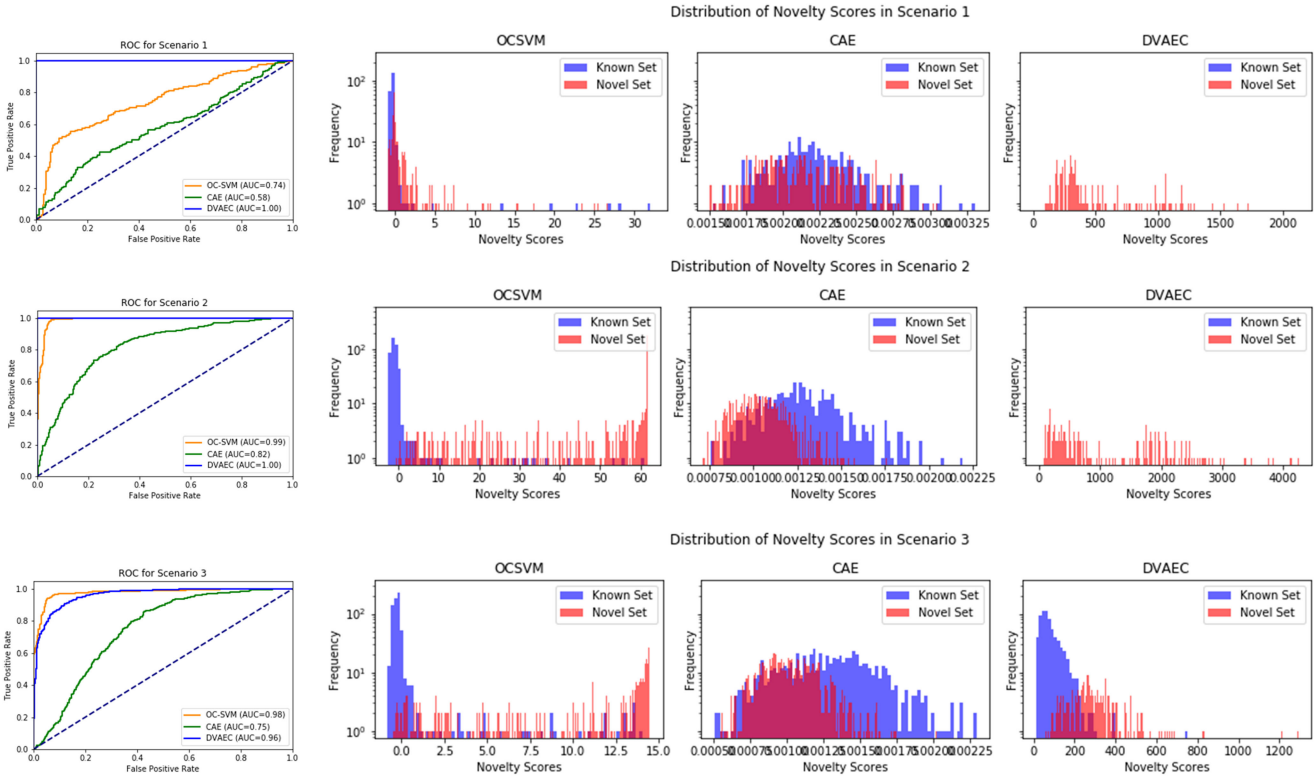
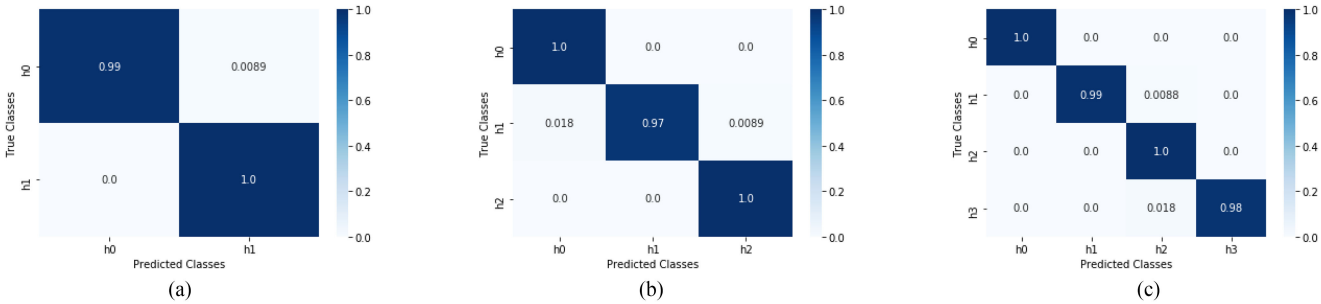


Fig. 10. Novelty detection performance for FEMTO dataset. First column: ROC curves of comparison methods in three scenarios; Second-fourth column: histogram of the novelty scores for normal and novel samples of comparison methods in three scenarios (testing samples in novel set have considerably higher novelty scores than those in known set).



(a) (b) (c)

Fig. 11. Confusion matrices for known set fault classification of case 2 using our proposed method. (a) Scenario 2. (b) Scenario 3. (c) Scenario 4.

classification accuracy for the known sets in scenarios 2 to 4. The standard DL methods, such as DNNs, can also achieve relatively high accuracy (over 90%) in three scenarios. This implies that DL model shows a superior performance over standard ML method relying on the hand-crafted statistical features for supervised classification problem.

To visualize the feature extraction ability of our proposed method, the t-SNE is used to graphically present features extracted by DVAEC under scenario 4, including four health states. At the same time, the raw data and statistic features by Table II are also plotted for comparison. Our method shows great feature extraction ability. As shown in Fig. 12(a), we can see that most of the failure stage (h_3) data is clustered all around the outside of the mixture cluster of other three health states. This implies that the failure stage data are naturally easy to be discriminated from other fault categories, whereas it is hard to make a classification among fault classes because they share similar data pattern and characteristics. The similar phenomenon can be observed from the visualization of statistic features in Fig. 12(b). However, in Fig. 12(c), the features extracted by our proposed method form clusters with distinguishable deviations with each other among different health states, including normal and fault data.

We further test the model performance by applying the trained DVAEC models to assess the health states of a different bearing. The new bearing for testing works under the same operating conditions with the bearing used for model training. The run-to-failure vibration signals of the new bearing is shown in Fig. 13. As a result shown in Fig. 14, the health states of the new bearing in different scenarios can be identified in real time according to the proposed Algorithm 1 in Section II-D. The predicted labels

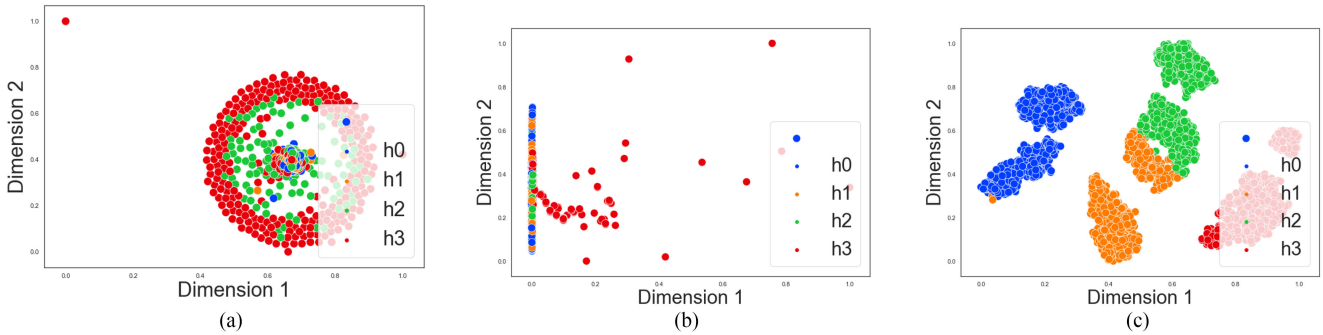


Fig. 12. Feature visualization using t-SNE for scenario 4. The color intensity of data points signifies the corresponding health states including h_0, h_1, h_2, and h_3. We visualize the raw fault data and the extracted statistic features (see Table II) in (a) and (b), respectively. In (c), we plot the hidden variables sampled from the hidden distribution using our proposed method. (a) Raw data. (b) Statistic features. (c) DVAEC.

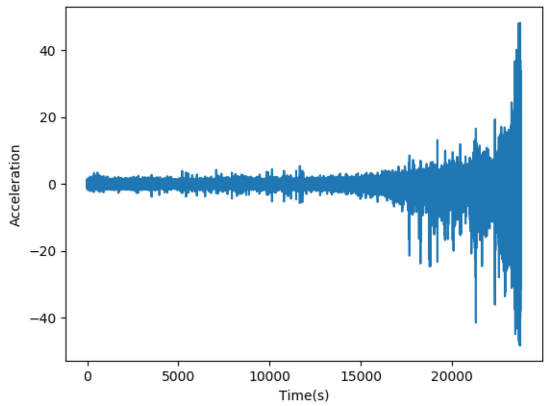


Fig. 13. Vibration signals of the new bearing.

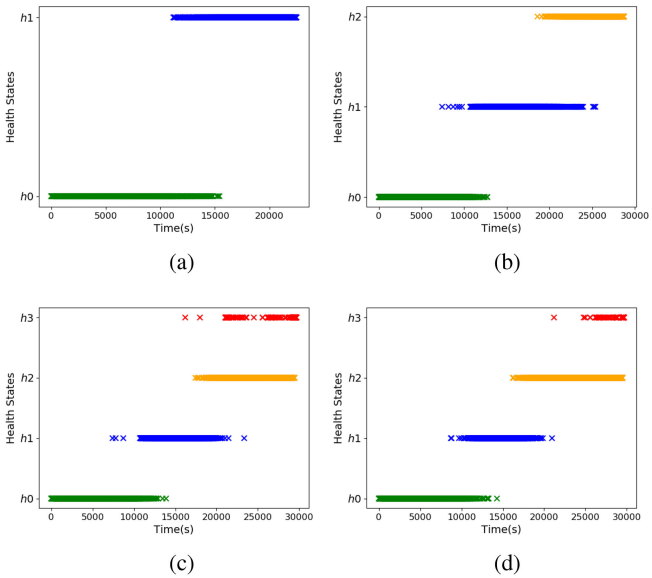


Fig. 14. Health state assessment performance of the new bearing in different scenarios. (a) Scenario 1. (b) Scenario 2. (c) Scenario 3. (d) Scenario 4.

(h_0, h_1, h_2, and h_3) by DVAEC are visualized over time for different scenarios, which also imply the health state transition during the bearing's life cycle. Note that the novelty threshold is set at the 90% percentile of the training set in scenario 1 and the threshold is set at the 95% percentile in scenarios 1 and 2. From

TABLE IX
HEALTH STATE ASSESSMENT PERFORMANCE FOR NEW BEARING

scenario Index	AUC Novel Set	Accuracy(%) Known Set
1	0.93	\
2	0.80	90.03
3	0.82	84.25
4	\	79.30

Fig. 14(a)–(d) and Table IX, we can see that the predicted health state labels can clearly show the degradation trend of the new bearing and misclassified health states are mainly distributed around the change time of the adjacent health states associated with a certain level of classification uncertainty. We can also see that in scenarios 3 and 4, there is no health state h_3 close to the failure time misclassified as normal health state h_0. This shows the great safety and reliability of the proposed model.

In terms of case 2, the average time for novelty detection is approximately 0.07 s, whereas the average time for known set fault classification is 0.0005 s, which are acceptable for most industry applications.

E. Noise Influence

In order to explore the influence of noise on the bearing health state diagnosis results, additional white noise is added into testing data for robustness analysis. Each testing sample is added with zero-mean Gaussian white noise and then tested by the previous trained network. By changing the standard deviation of the Gaussian noise, signals with different signal-to-noise ratios can be obtained. By adding white noise, information in the original signals would be disrupted and features would be covered by noise. The trained network would misjudge samples due to noise influence. It could be expected that accuracy would drop when noise is added. In scenario 3 of both case studies, we compare the novelty detection and known set classification performance of our proposed method with other methods separately.

Fig. 15 displays the AUC with respect to different standard deviation of additional noise for novelty detection performance in both two cases. In all scenarios, AUC scores decrease as higher level of white noise is introduced. However, the AUC of the DVAEC drops relatively slower than OCSVM and CAE

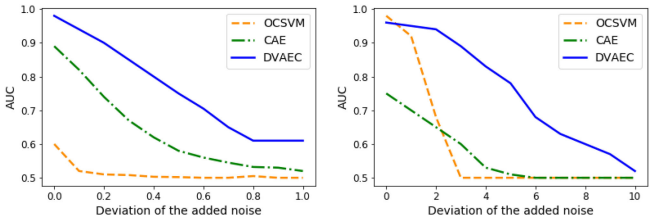


Fig. 15. Robustness analysis for unknown set novelty detection in scenario 3. (a) Case 1. (b) Case 2.

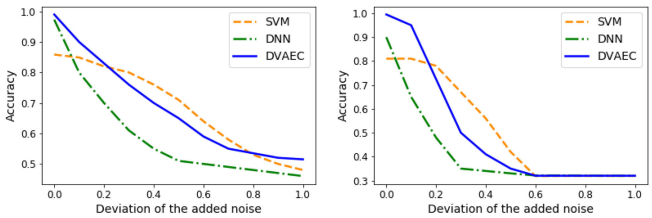


Fig. 16. Robustness analysis for known set fault classification in scenario 3. (a) Case 1. (b) Case 2.

methods. For case 2, OCSVM plummets in a short time and is least robust against noise. The result shows that the proposed DVAEC network is robust to noise without prior knowledge.

In terms of fault classification performance, Fig. 16 displays the classification accuracy with respect to different standard deviation of additional noise in both case studies. All three classification accuracy curves decrease as higher level of white noise is introduced. Although the classification of DVAEC drops slightly faster than SVM but slower than DNN, our proposed method can also maintain a high classification accuracy when the noise level increases.

IV. Conclusion

This article presented a novel DL-based fault diagnostics framework (DVAEC) to address the key challenges in dealing with unprecedented faults—enabling robust performance of both new fault identification and known fault diagnostics. The proposed framework unified novelty detection and fault classification into a single framework through jointly training a deep VAE model and a classifier to provide robust and accurate FDD results when unknown fault samples emerge unexpectedly without prior knowledge. The DVAEC model transformed fault signals with similar characteristics entangled in their original space into latent variables with uncertainty information in a latent space so that the underlying structure of the inner distribution can be captured and characterized probabilistically. These probabilistic latent space information empowered the model to establish a distance-based novelty threshold to identify whether the testing sample is novel and then automatically classify nonnovel samples into the existing known labeled fault classes. Two roller bearing related cases were carried out to validate the proposed method. Specifically, the effectiveness of the proposed method was validated by a comparison with existing DL-based benchmarking methods. The model performance was further validated by accurately detect and identify degradation states on a new bearing different from the one for training. The significance of this study was to provide a generalizable DL-based approach for health state diagnosis of mechanical systems with limited labeled monitoring data, which has shown a great potential for incremental fault diagnosis starting from only health normal data.

The methods developed in this article can be further improved from the following aspects in the future. First, it is reasonable to preprocess the data by generating spectrogram images, and this method is commonly used for vibration data since this type of data is nonstationary and periodic. It is meaningful to take other types of data into consideration and develop more generalizable methods for fault diagnostics, second, it is interesting to further study how to retrain the model when the new fault types are incrementally detected and incorporated into the original dataset. The problem of class imbalance is worth further investigation to improve incremental fault diagnostics.

References

[1] A. Kusiak, "Smart manufacturing," *Int. J. Prod. Res.*, vol. 56, no. 1/2, pp. 508–517, 2018.
[2] Z. Li, Y. Wang, and K.-S. Wang, "Intelligent predictive maintenance for fault diagnosis and prognosis in machine centers: Industry 4.0 scenario," *Adv. Manuf.*, vol. 5, no. 4, pp. 377–387, 2017.
[3] L. Guo, N. Li, F. Jia, Y. Lei, and J. Lin, "A recurrent neural network based health indicator for remaining useful life prediction of bearings," *Neurocomputing*, vol. 240, pp. 98–109, 2017.
[4] M. Kordestani, M. Saif, M. E. Orchard, R. Razavi-Far, and K. Khorasani, "Failure prognosis and applications—A survey of recent literature," *IEEE Trans. Rel.*, vol. 70, no. 2, pp. 728–748, Jun. 2021.
[5] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mech. Syst. Signal Process.*, vol. 115, pp. 213–237, 2019.
[6] R. Liu, B. Yang, E. Zio, and X. Chen, "Artificial intelligence for fault diagnosis of rotating machinery: A review," *Mech. Syst. Signal Process.*, vol. 108, pp. 33–47, 2018.
[7] H. Shao, H. Jiang, H. Zhang, W. Duan, T. Liang, and S. Wu, "Rolling bearing fault feature learning using improved convolutional deep belief network with compressed sensing," *Mech. Syst. Signal Process.*, vol. 100, pp. 743–765, 2018.
[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, pp. 436–444, 2015.
[9] Y. Lei, F. Jia, J. Lin, S. Xing, and S. X. Ding, "An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data," *IEEE Trans. Ind. Electron.*, vol. 63, no. 5, pp. 3137–3147, May 2016.
[10] F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mech. Syst. Signal Process.*, vol. 72, pp. 303–315, 2016.
[11] O. Abdeljaber, S. Sassi, O. Avci, S. Kiranyaz, A. A. Ibrahim, and M. Gabbouj, "Fault detection and severity identification of ball bearings by online condition monitoring," *IEEE Trans. Ind. Electron.*, vol. 66, no. 10, pp. 8136–8147, Oct. 2019.
[12] M. Gan *et al.*, "Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings," *Mech. Syst. Signal Process.*, vol. 72, pp. 92–104, 2016.
[13] J. A. Carino *et al.*, "Fault detection and identification methodology under an incremental learning framework applied to industrial machinery," *IEEE Access*, vol. 6, pp. 49 755–49766, 2018.
[14] M. A. Pimentel, D. A. Clifton, L. Clifton, and L. Tarassenko, "A review of novelty detection," *Signal Process.*, vol. 99, pp. 215–249, 2014.
[15] M. D. Prieto, J. A. C. Corrales, D. Z. Millán, M. M. Gonzalvez, J. A. O. Redondo, and R. d. J. R. Troncoso, "Evaluation of novelty detection methods for condition monitoring applied to an electromechanical system," in *Fault Diagnosis and Detection*. London, U.K.: InTech, 2017, p. 131.
[16] H.-C. Yan, J.-H. Zhou, and C. K. Pang, "Gaussian mixture model using semisupervised learning for probabilistic fault diagnosis under new data

categories," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 4, pp. 723–733, Apr. 2017.

[17] E. Principi, D. Rossetti, S. Squartini, and F. Piazza, "Unsupervised electric motor fault detection by using deep autoencoders," *IEEE/CAA J. Automatica Sinica*, vol. 6, no. 2, pp. 441–451, Mar. 2019.

[18] K. Loparo, "Bearings vibration data set," *Case Western Reserve University Bearing Data Center*, Cleveland, OH, USA, 2011.

[19] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv preprint arXiv:1312.6114*.

[20] C. Doersch, "Tutorial on variational autoencoders," 2021, *arXiv preprint arXiv:1606.05908*.

[21] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[22] D. Hendrycks, M. Mazeika, and T. G. Dietterich, "Deep anomaly detection with outlier exposure," 2019, *arXiv preprint arXiv:1812.04606*.

[23] B. Jin, D. Li, S. Srinivasan, S.-K. Ng, K. Poolla, and A. Sangiovanni-Vincentelli, "Detecting and diagnosing incipient building faults using uncertainty information from deep neural networks," in *Proc. IEEE Int. Conf. Prognostics Health Manage.*, 2019, pp. 1–8.

[24] M. Kliger and S. Fleishman, "Novelty detection with GAN," 2018, *arXiv preprint arXiv:1802.10560*.

[25] M. S. Sadooghi and S. E. Khadem, "Improving one class support vector machine novelty detection scheme using nonlinear features," *Pattern Recognit.*, vol. 83, pp. 14–33, 2018.

[26] J. Pan, Y. Zi, J. Chen, Z. Zhou, and B. Wang, "LiftingNet: A novel deep learning network with layerwise feature learning from noisy mechanical data for fault classification," *IEEE Trans. Ind. Electron.*, vol. 65, no. 6, pp. 4973–4982, Jun. 2018.

[27] S. Shao, S. McAleer, R. Yan, and P. Baldi, "Highly accurate machine fault diagnosis using deep transfer learning," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2446–2455, Apr. 2019.

[28] H. Fang, J. Deng, B. Zhao, Y. Shi, J. Zhou, and S. Shao, "LEFE-Net: A lightweight efficient feature extraction network with strong robustness for bearing fault diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 70, 2021, Art. no. 3513311.

[29] Z. Zhao *et al.*, "Unsupervised deep transfer learning for intelligent fault diagnosis: An open source and comparative study," 2019, *arXiv preprint arXiv:1912.12528*.

[30] L. v. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

[31] M. Ma, C. Sun, and X. Chen, "Discriminative deep belief networks with ant colony optimization for health status assessment of machine," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 12, pp. 3115–3125, Dec. 2017.

[32] L. L. H. Wang and J. Lee, "Bearing health assessment and fault diagnosis using the methods of fast Fourier transform and self-organizing map," in *Proc. 61st Meeting Soc. Mach. Failure Prevention Technol.*, 2007.