A Novel Deep Learning Framework Based RNN-SAE for Fault Detection of Electrical Gas Generator

MOATH ALRIFAEY[ID]**, (Graduate Student Member, IEEE),**
WEI HONG LIM, (Senior Member, IEEE),
AND CHUN KIT ANG
Faculty of Engineering, Technology and Built Environment, UCSI University, Kuala Lumpur 56000, Malaysia
Corresponding author: Wei Hong Lim (limwh@ucsiuniversity.edu.my)

ABSTRACT The electrical generator is the key part of the electrical generation system for the oil and gas industry, and it is easy to fail, which disturbs the availability and reliability of the electrical generation in the power industry. Therefore, extracting and diagnosing the fault features from the process signals are useful to diagnose the status of the machine. Though, a common challenge in many applied applications is the practical knowledge about the risk of failure or historical records, which is totally unlabeled and difficult to be identified by traditional fault approaches. Hence, in the present study, a novel deep learning (DL) framework is proposed to fill the gap by balancing the three stages of fault feature extraction, fault detection, and parameter optimization based on the long short term memory- recurrent neural networks (RNN- LSTM), stacked autoencoders (SAE), and particle swarm optimization (PSO) techniques. The suggested framework focuses on failure detection through a sequence of numerous features for the unlabeled historical data and unknown anomaly. To validate the effectiveness of the proposed DL framework, an experiment for failure detection of the electrical generator was conducted for the data of risky environment at Yemen oil and gas plant. The experimental results compared with the earlier studies validate that, the DL framework can address the faults for vibration signals of the electrical generator in a well- diagnosis performance effectively.

INDEX TERMS Deep learning (DL), fault detection, long short-term memory (LSTM), oil and gas plant, recurrent neural networks (RNN), stacked autoencoders (SAE).

I. INTRODUCTION

Electrical generation plants are concern about the availability and reliability of mechanical equipment due to the rapid growth of energy demand. Currently, producing electrical energy without failures has drawn more attention to power production. Mechanical failures caused about 79.6% of machine downtime [1]. Thus, fault detection in an earlier stage will accelerate the PM (Preventive Maintenance) activities to prevent machine downtime and unscheduled maintenance activities [2]. Practically, faults will be combined during failures or abnormal operation, which makes the fault classification more complex, as a result, detecting and preventing the fault will be difficult [3]. Nowadays,

The associate editor coordinating the review of this manuscript and approving it for publication was Michele Magno[ID].

data-driven approaches are broadly used in fault diagnosis due to the availability of historical data for a fault diagnosis requirement [4], [5]. Feature extraction approaches can be achieved by analyzing the domains such as time, frequency, or frequency-time [6]–[8].

In general, deep learning (DL) is a multi-layer of neural networks involved in the processing of non-linear information in multiple hidden layers, which has the ability to extract and classify features. In fault analysis fields, DL can extract the potential fault features from multi-sensor data in a condition monitoring system, which includes process signals such as vibration, temperature, flux, flow, and pressure. The vibration signal has been broadly and revealed acceptable performance [9].

DL is used in fault diagnosis of mechanical equipment such as gear transmission [10], [11], bearing [12], and multi-joint

industrial robots [13]. In [14], an intelligent fault diagnosis was proposed based on a deep echo state network (ESN) and a hybrid evolutionary algorithm. Recently, deep learning models have been combined with different approaches such as SAE [15], deep belief network (DBN)[16], and LSTM [17], to improve the performance of fault detection and data modeling.

For industrial applications, early fault detection can enhance the identification of hazard patterns and achieve reliable maintenance for mechanical equipment [18]. The explosion of industrial data has emphasized the necessity for the DL technique in many applications, for instance, malware detection [19], multi-sensor fusion [20], anomaly detection [21], and soft sensor modeling [15].

The electrical generator mostly runs in tough working conditions with variant load in a hazardous environment, so the occurrence of faults will directly affect the production of power efficiency and may lead to ruinous accidents, especially electrical generation unit in a hazardous area like the oil and gas plant [22]. Thus, fault identification and diagnostic analysis are necessary to prevent future failures and to plan appropriate maintenance.

In the DL studies, it was previously shown that DL frameworks were enhanced to overcome shortcomings for feature identification and classification in online time series. Similarly, the work of [10] has demonstrated an improved framework of fault diagnosis by a novel DL technique based on the fused- Stacked autoencoders (SAEs) and quantum ant-colony algorithm into the methodology. However, unlike the study of [10], most of the improved DL frameworks lack a focus on the deep learning feature optimization for different faults without data labeling for fault detection purposes. According to [23], DL models can offer promising tools that can assist a decision-maker in planning and management decisions.

In this study, the motivations for designing the hybrid DL framework are: (1) Faults of electrical gas generator lead to plant shutdowns or cause hazardous event such as fire or blasting in electrical equipment. Therefore, failure detection can be taken in failure prevention and updating the maintenance planning; (2) Optimization of model parameters can increase the model accuracy for fault detection [10]; (3) Deep fault detection framework can be built by learning the failure patterns from important events in a time series signal of the electrical generation process. Thus, a novel hybrid LSTM – SAE learning algorithm is subsequently applied to overcome the drawbacks of RNN training and usage of a single technique separately and improve the fault detection rate.

To our knowledge, no earlier study in the DL framework has focused on filling the gap of two DL optimization phases in feature extraction and fault detection for proactive maintenance, particularly for electrical units in a risky region such as the oil and gas plant. Hence, a novel DL framework was proposed to achieve better results in deep feature learning and fault detection. This work aims to introduce a novel DL framework with the goal of extracting features

and fault detection for failure prevention and maintenance planning.

To this end, the main contributions of the present work can be concluded as follows:

(1) A novel DL fault detection with a simple and effective framework is designed to balance the three stages of fault feature extraction, fault detection, and parameter optimization based on the RNN-LSTM, SAE, and particle swarm optimization (PSO) techniques.

(2) A novel hybrid mathematical methodology can enhance the learning ability to solve the drawbacks of RNN training, such as error of decaying, deficient, gradient vanishing, and backflow.

(3) The DL framework offers a good unsupervised deep learning for unlabeled data, which allows the proposed DL methodology to not only offer the important features adaptively but also realize sequences without saving the previous sequence inputs.

(4) The proposed DL framework adds contributions to the area of the electrical gas generator for fault detection, which may be useful for other industrial deep learning applications, especially in hazardous areas.

The notations used through the technical part are explained as follows:

x_M^d	Input samples d of a dataset with M-dimension
$\hat{x}^d$	Output sample vector
$S(t)$	Input signal
$c(t)$	Intrinsic mode function(IMF)
I	The I_th dimension of high to low-frequency components
P	Optimal autoregressive order for IMF
r_I	The residual value of the IMF component
a_k	Weighted coefficients of component k
$w(t)$	The white noise of the IMF
$c_\varepsilon(t)$	The IMF components of signal ε
z_P	The P-reflection vectors of the input sample vector x^d
x_t	Input simple x at time t
M	Number of training samples
α_t	Forget gate layer
β_t	Input gate layer
γ_t	Recurrent gate layer
o_t	Output gate layer to hold information for the future step
$W_\alpha, W_\beta, W_\gamma, W_o$	The weights of four layers of the neural network $\alpha, \beta, \gamma,$ o in the memory-cell operations LSTM
$b_\alpha, b_\beta, b_\gamma, b_o$	The biases of four layers of the neural network $\alpha, \beta, \gamma,$ o in the memory-cell operations LSTM
h^d	The hidden vector
f	Active sigmoid function
$W^{(1)}, W^{(2)}$	The optimal parameter sets of weights

$b^{(1)}, b^{(2)}$	The optimal parameter sets of bias
$L(x^d, \hat{x}^d)$	The reconstruction error between $\hat{x}^d$ and x^d
λ	Decay weight parameter
$W_{ji}^{(l)}$	The weight of linking neuron i to neuron j within layers $(l, l+1)$
n_l	The number of network layer within layer l
s_l	The number of neurons within layer l
$\hat{\rho}_g$	The average value of activating for hidden unit e
ρ	Sparsity parameter
X_{id}	Position of the i_th particle with d dimensions
V_{id}	The velocity of the i_th particle with d dimensions
N	The population size of particles
D	The dimension of searching space
r_1, r_3	Random values in period [0,1]
c_1, c_2	learning factors
$pBest_{id}^t$	The best local of particle i in time t
$gBest^t$	The global best of the population in time t
ω	The inertia weight
θ	Sparsity penalty parameter
σ	Sigmoid function
C_{t-1}, C_t	The last state and the present state
$h_{t-1}.$	The input state of the present and earlier frames
h_t	The present state of the RNN hidden layer
N_T	The total number of instances
h_{final}	The final decision output from the RNN-LSTM network

The rest of this paper is structured as follows. Sections 2 and 3 describe data and the proposed methodology of the new hybrid DL framework based on the SAE, RNN-LSTM, and PSO algorithm. Section 4 shows the discussion and results of the experimental application of the framework via historical data of Yemen electrical generators in the oil and gas plant. Lastly, the conclusions and future work are presented in Section 5.

II. DATA AND METHODS

Recently, secondary data for deep learning methods have been used in different research areas due to their outstanding performance. Secondary data were collected from DCS (Distributed Control System) of the electrical generator in Yemen oil and gas plant.

The training of the proposed framework was carried out with the simulation computer, which has environment configurations as described in Table 1.

For fault diagnosis, the ensemble empirical mode decomposition (EEMD) was implemented in this work to pre-process the recorded signals of non-stationary fault.

TABLE 1. Experimental environment.

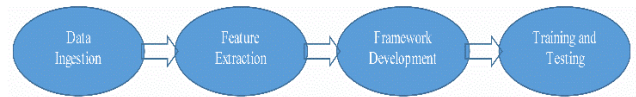| Hardware properties		Software properties	
Memory	16GB	Operating system	Windows 10-64bit
CPU	Intel Core i7+8700/4.7 GHz	Code software	TensorFlow-GPU Python3.8.4
Graphics card	NVIDIA-GeForce-RTX2060		

FIGURE 1. Deep Learning pipeline for physics data.

EEMD was used to provide the fault representative frequency to be effectively distinguished as normal or fault. Traditional machine learning methods greatly depend on extracted features of artificial discriminative as the inputs. Besides, data augmentation and the SAE-LSTM framework were applied for feature extraction and fault detection. The Deep Learning pipeline for physics data is illustrated, as shown in Fig. 1.

A. DATA INGESTION

In this paper, all the vibration signals are obtained from the DCS of the electrical generator. In this study, about 2000 samples were divided into a training data set (80%) and a testing dataset (20%) before fed to the proposed DL framework. These simples are categorized into two groups of training and testing dataset, which may be faulty or non-faulty.

For data ingestion, ensemble empirical mode decomposition (EEMD) was used in this framework to pre-process the recorded signals of non-stationary faults due to important information of fault events, which is mostly revealed in high-frequency Intrinsic mode functions (IMFs). EEMD was used to provide the frequencies of vibration signals, which are effectively distinguished the information of normal or fault. Equations (1) and (2) provide further elaborations on the mathematical computations, which have been extensively discussed in an earlier study by [12].

EEMD converts signal $S(t)$ into numerous IMFs, as follow:

$$S(t) = \sum_{\varepsilon=1}^{I} c_\varepsilon(t) + r_I \tag{1}$$

where r_I is the remaining value, and $c_\varepsilon(t)$ $(\varepsilon = 1, 2, \cdots I)$ are represented components of high to low frequency from different intrinsic mode functions.

For an intrinsic function with zero-mean of $c(t)$, the model is represented as:

$$c(t) = \sum_{k=1}^{P} a_k c(t-k) + w(t) \tag{2}$$

where P is the order, a_k denotes for weighted coefficients $(k = 1, 2 \cdots P)$, and white noise $w(t)$.
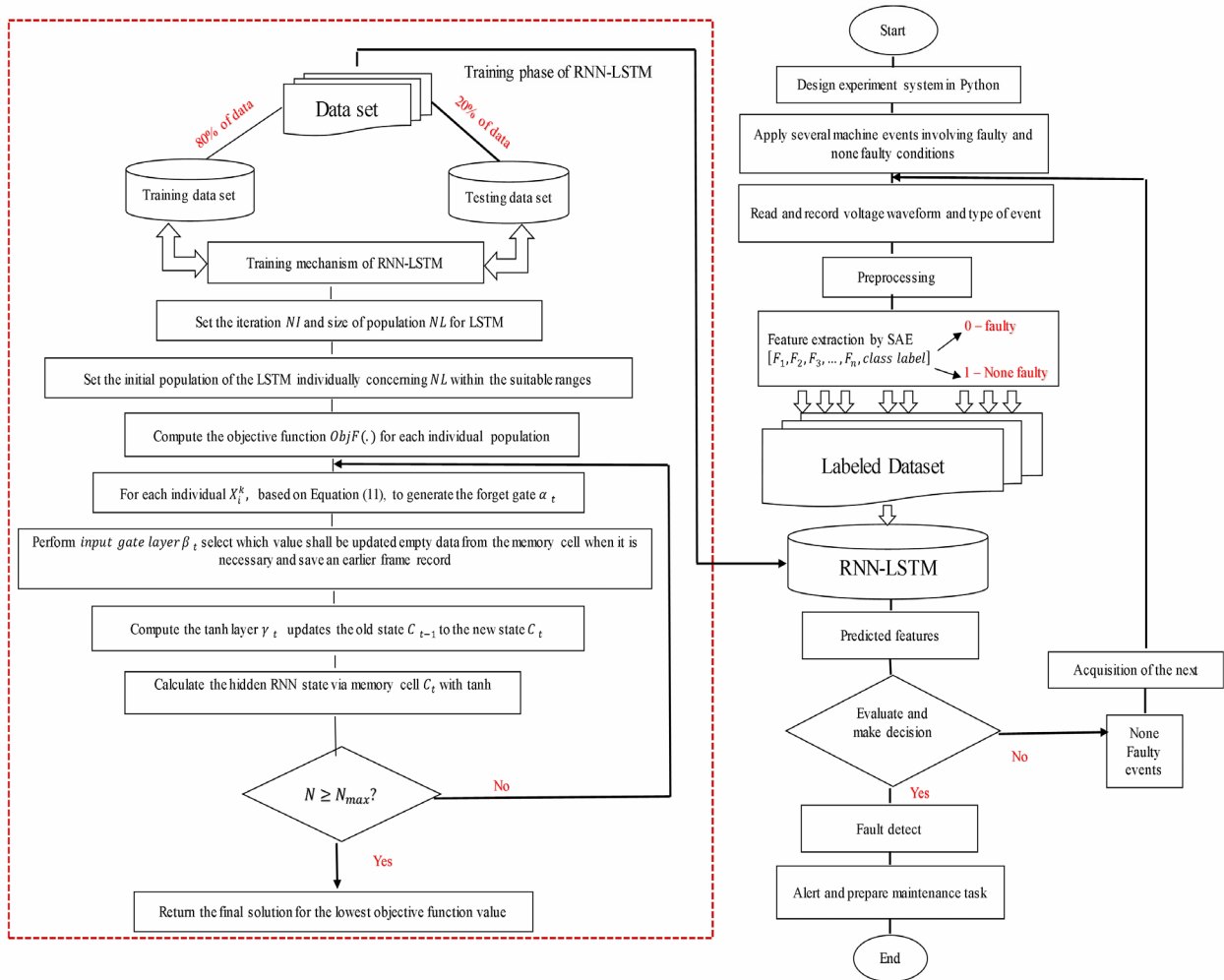
FIGURE 2. The proposed DL framework.

III. THE PROPOSED DL FRAMEWORK

In this section, the proposed DL framework was established based on the RNN-LSTM, SAE, and PSO techniques to achieve better results in deep feature learning and fault detection. The secondary data from the industrial plant was considered to validate a deep learning framework and fault detection during the executing of the proposed methodology. Fig. 2 illustrates the proposed framework for fault detection in a risky oil and gas industry.

A. FAULT FEATURE EXTRACTION BY SAE

Stacked autoencoder (SAE) is a type of deep neural network, which is firstly offered in 2007 [9]. SAE is developed from a basic AE model and has multiple layers of autoencoder with individual pre-training via layer by layer of the autoencoder. The data input of SAE is taken from the output itself, which reduce data dimension and leverage for the effective encoding of learning. Specifically, SAE can be used for feature extraction without class labeling.

Deep autoencoder has more than three layers of the neural network [24]. For big and row data of industrial working conditions, extracting features can improve the performance of predictive maintenance in several states via fault classification and detection. SAE can identify the features and successfully determine the distinguished information of the machine conditioning signals [24].

Several practical studies have also revealed the mining capability of SAE to offer important information from the frequency domain regarding the analysis problems [25]. SAE has advantages in dimensionality reduction by using a hidden layer as a feature extractor of any desired size, which can predict the same input data at the output without requiring labels. Fig. 3 illustrates the structure of an SAE three layers for the unsupervised learning and feature extracting. SAE layers consist of input, output, and hidden layers, which provide extracted features. SAE layers are divided into two parts, the encoder network (input-hidden) and the decoder network (hidden-output).

Supposing that unlabeled training samples $x^d = \{x_1^d, x_2^d, \cdots, x_M^d\}$, every sample x^d has P reflection vectors $x^d = \{z_1, z_2, \cdots, z_P\}$. The encoder process and decoder process for every input sample x^d and datasets
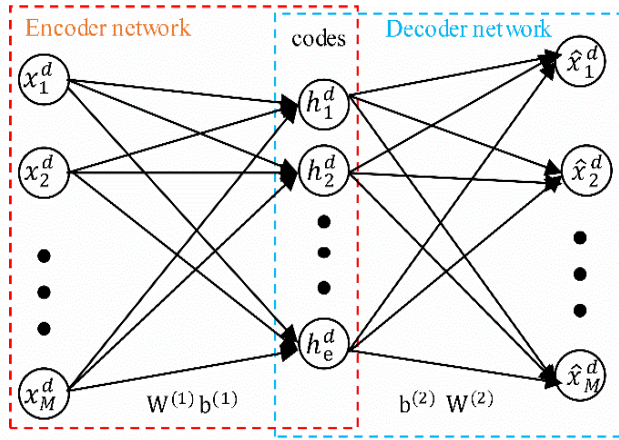
FIGURE 3. Structure of feature extracting SAE neural network.

$\{x^d\}_{d=1}^{M}$ are expressed as follows:

$$h^d = f(W^{(1)}x^d + b^{(1)}) \tag{3}$$

$$\hat{x}^d = f(W^{(2)}h^d + b^{(2)}) \tag{4}$$

where h^d is the hidden vector, $\hat{x}^d$ is the output vector, f denotes the active sigmoid function , $W^{(1)}$, $W^{(2)}$ denote for the weight, and $b^{(1)}$, $b^{(2)}$ denote for the bias.

The error of reconstruction $L(x^d, \hat{x}^d)$ between $\hat{x}^d$ and x^d is defined as the following:

$$L\left(x^d, \hat{x}^d\right) = \frac{1}{2}\left\| x^d - \hat{x}^d \right\|^2 \tag{5}$$

The general cost function of samples is expressed as:

$$J(W, b) = \left[\frac{1}{M}\sum_{d=1}^{M}L\left(x^d, \hat{x}^d\right)\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(W_{ji}^{(l)})^2 \tag{6}$$

where λ denotes the parameter of decay weight, $W_{ji}^{(l)}$ is the weight of linking neuron i to neuron j within layers $(l, l+1)$, n_l and s_l denote the number of the network layer and the number of neurons within layer l.

Equation (6) consists of two terms, which are the total dataset error of reconstruction and the penalty weight. The second term is used to avoid over-fitting by limiting the value of weights.

The total cost function of SAE is depicted as:

$$J_{sparse}(W, b) = J(W, b)$$
$$+ \theta\sum_{g=1}^{e}(\rho\log\frac{\rho}{\hat{\rho}_g} + (1-\rho)\log\frac{1-\rho}{1-\hat{\rho}_g}) \tag{7}$$

where $\hat{\rho}_g$ is the average value of activating for $(g = 1, 2, \cdots, e)$ hidden unit, ρ is a parameter of sparsity, θ denotes a parameter of sparsity penalty, which is utilized in controlling the relative significance for the terms of penalty and reconstruction.

The above-mentioned hyperparameters are optimized by applying the Particle Swarm Optimization (PSO) algorithm, which is an evolutionary procedure that executes a computation process to catch optimum solutions on search space presented earlier [26]. PSO is modeling the cooperative behavior for the swarming of birds or fish. Besides, the PSO procedure has been effectively implemented across a wide area, such as forecasting of traffic flow [27], optimization power losses in the distribution system [28], the self-adaptive mechanism [29], and fault diagnosis in power transformers [30].

In PSO, every ith particle X_{id} has positions $X = \{X_{i1}, X_{i2}, \ldots, X_{iD}\}$, and the velocity vectors $V = \{V_{i1}, V_{i2}, \ldots, V_{iD}\}$. Furthermore, positions and velocity will be determined after the computation of the fitness function.

For velocity V_{id}^{t+1} and position X_{id}^{t+1} are determined, as follows:

$$V_{id}^{t+1} = \omega.V_{id}^t + c_1.r_1.\left(pBest_{id}^t - X_{id}^t\right)$$
$$+ c_2.r_2.\left(gBest^t - X_{id}^t\right) \tag{8}$$

$$X_{id}^{t+1} = X_{id}^t + X_{id}^{t+1} \tag{9}$$

where $i = 1, 2, \ldots, N; d = 1, 2, \ldots, D; N$ and D are population size and dimension of searching space respectively; r_1 and r_3 are random values in period [0,1]; c_1 and c_2 are learning factors; $(pBest_{id}^t, gBest^t)$ are the best local of particle i in time t and the global best of the population in time t, respectively; ω is the inertia weight.

The PSO algorithm can be described in the following steps:

(a) Initialization, initialize the particle position X_{id}, and velocity V_{id} in d dimensions of the searching space randomly.

(b) Calculate the fitness for every particle in the swarm.

(c) Compare each fitness in each iteration with its earlier best fitness $pBest_{id}^t$. Then, select the value with better fitness to be $pBest_{id}^t$ of the current location in the d-dimensional space.

(d) Compare the best local for all particles with each other in time t and set the global best location of population in time t to the greatest fitness $gBest^t$.

(e) Update the particle velocity and position according to (8) and (9), respectively.

(f) Perform steps (b)–(e) until iteration reached the maximum criteria.

The objective of training the autoencoder is to learn the features of spare and representative through minimization of cost function and (W, b) parameter sets.

The average activating value is expressed as follows:

$$\hat{\rho}_g = \frac{1}{M}\sum_{g=1}^{e}h_g^d \tag{10}$$

The optimal parameter sets $W^{(1)}$, $W^{(2)}$, $b^{(1)}$, and $b^{(2)}$, which can be learned simultaneously using the minimizing process, which is called the training process of the SAE, as shown in Fig. 4.
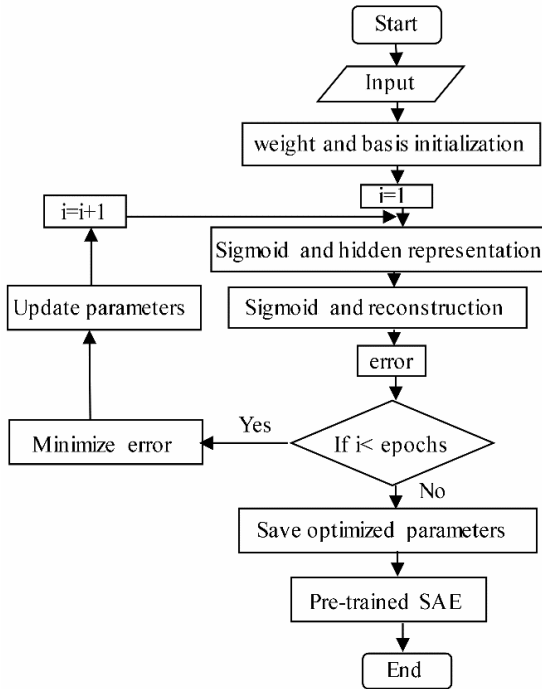
FIGURE 4. The SAE training algorithm.

FIGURE 5. LSTM-Memory cell.

B. FAULT DETECTION USING (RNN-LSTM) WITH TIME SERIES

RNNs are presented to analyze ambiguous in sequential patterns for the spatial and temporal sequential data [31]. Vibration signals have continuous amplitude values that are sequential data denote in samples for varied amplitude. Sampling converts data from continuous to discrete, making the vibration context easier to understand and analyze in the rotary machine conditioning.

RNNs can realize sequences without saving the previous sequence inputs, so sequences of long term, which is called in deep learning as the problem of gradient vanishing. To solve the problem of gradient vanishing, a special architecture of RNN named LSTM was offered in 1997 to solve the problem in RNN training, such as error of decaying, deficient, and backflow [32]. LSTM has the advantage of leverages memory cells with forget gates over standard RNN [33].

The essential idea of the LSTM structural design is a memory cell, which has the capability to keep its information using non-linear gate units and control the flow of information from or into the cell. Moreover, forget gates add enhancements in the learning ability to forget (eliminate) information stored in the memory cell. Furthermore, connections of peepholes allow the LSTMs to realize accurately timed patterns and calculate the internal states within the matrices of cost and weight [34].

In Fig. 5, LSTM contains four unique layers of the neural network (α, β, γ, o), which are different from one layer of traditional RNN.

To understand the memory-cell operations of LSTM, assuming at time t and input x_t in (11), the forget gate of the first layer can be represented as α_t that removes information
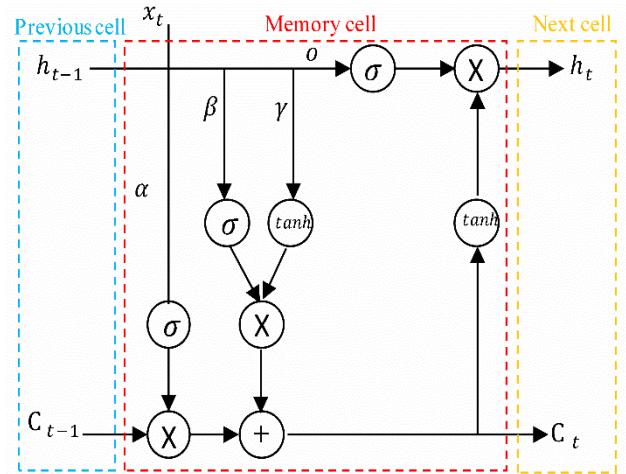
of the memory cell once required and saves the previous state of information before emptying the memory.

$$\alpha_t = \sigma(W_\alpha \cdot [h_{t-1}, x_t] + b_\alpha) \tag{11}$$

where σ denotes the sigmoid function, [] is the operation of concatenate, and W_α denotes the α layer weight.

Input gate layer of the second layer β_t is applied to select the suitable value that should be updated and obtained as:

$$\beta_t = \sigma(W_\beta \cdot [h_{t-1}, x_t] + b_\beta) \tag{12}$$

Then, updating and storing values via the layer of tanh γ_t, as follows:

$$\gamma_t = \tanh(W_\gamma \cdot [h_{t-1}, x_t] + b_\gamma) \tag{13}$$

Next, the last state C_{t-1} is updated by the present state C_t, as following:

$$C_t = \alpha_t \cdot C_{t-1} + \beta_t \cdot \gamma_t \tag{14}$$

where γ_t denotes the recurrent component, which is calculated via "tanh" as the activating function for the input state of the present and earlier frames h_{t-1}.

The last layer will be a layer of sigmoid function for output gate o_t to hold information for the future step. The output gate o_t can be obtained as:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{15}$$

The present state of the RNN hidden layer is determined as follows:

$$h_t = o_t \tanh(C_t) \tag{16}$$

where W_α, W_β, W_γ, and W_o are the layer weights; and b_α, b_β, b_γ, and b_o are the layer biases.

Training sequence patterns of data (such as time series) are not able to be recognized by one cell of LSTM. Hence, the structure of stacking multiple memory cells was proposed for learning the long term sequence and dependency in time series data.
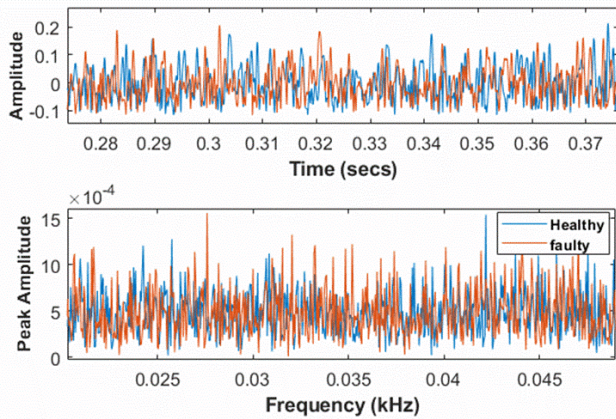
FIGURE 6. Vibration signal in time and spectrum domains for the shaft of the electrical generator.

Lastly, Softmax classifier was applied to get the final decision output from the RNN-LSTM network, which can be expressed as:

$$h_{\text{final}} = softmax(h_{\text{t}}) \qquad (17)$$

C. TESTING THE PERFORMANCE OF PROPOSED DL FRAMEWORK

The proposed DL framework will be trained and tested with a dataset of 2000 samples. The performance of the proposed DL framework and other models will be evaluated to compare their precision and recall values. The fault detection performance of the DL framework is evaluated with respect to features to validate the effect of features on fault detection.

The accuracy, precision, and recall can be determined as:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{N_T} \qquad (18)$$

where N_T denotes the total number of instances.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \qquad (19)$$

$$\text{Recall} = \frac{\text{TrueP ositives}}{\text{True Positives} + \text{False Negatives}} \qquad (20)$$

IV. RESULT AND DISCUSSION
A. DATA PREPROCESSING
In this step, the EEMD method was conducted to pre-process the collected vibration signals to get several IMFs. Fault features are typically discovered in the high frequency of IMFs and mostly could not reveal major differences in the vibration signals within the time dominion, as can be seen from Fig. 6. The components of IMFs were obtained by (1)-(2) with the selected component number and optimal order ($I = 4, P = 25$). Then, the component of IMFs was converted into the vectors as inputs to the SAE network to obtain the features of fault signals in the next step.

B. DEEP FEATURE LEARNING
In this step, features of vibration signals were automatically extracted by The SAE approach, which is an unsupervised

TABLE 2. The detailed structure of the proposed SAE network.

SAE Structure	Setting value
The dimension of original signals	2000
The dimension of pre-processed signals	200
Numbers of hidden layer	5
Number of neurons for each layer	200-100-30-10-5
Learning rate of each layer	0.3-0.3-0.3-0.3-0.2
Number of input layer neurons	200
Number of output layer neurons	5

TABLE 3. Optimized SAE parameters using PSO.

	Hidden Layer number				
	1	2	3	4	5
Sparsity penalty	$\theta_1 = 0.001$	$\theta_2 = 0.154$	$\theta_3 = 0.131$	$\theta_4 = 0.05$	$\theta_5 = 0.202$

learning network with the dimension reduction of original features.

The SAE network is constructed to extract the fault features from unlabeled time series data, which can be used for further usage and investigation of fault detection by the LSTM approach. The deep SAE network that was used during the experiment has five hidden layers with dimensional data 2000-1000-300-100-5. For example, depth and breadth of dimensional data of (2000-200) denote that original signals have dimension 2000 to get the pre-processed signals with dimension 200 as inputs, essentially help in reducing the computation of feature extraction. SAE Depth shows an essential role due to defining the extracted feature qualities. Table 2 shows the total numbers of hidden layers, neurons for each hidden layer, and the learning rate of each layer.

In this experiment, the SAE network is set to five hidden layers as recommended in [35] with the learning rate (0.2-0.3) based on the procedure in [12]. In Table 2, the setting value (200-100-30-10-5) means that five hidden layers, with 200 neurons in the first hidden layer, 100 neurons in the second layer, and (30-10-5) neurons for other hidden layers. Selecting or designing a proper network structure depends on the dimension of original signals, pre-processed signals, and the size of hidden layers to be set by trial and error to improve accuracy [10].

The configuration structure of SAE hidden layers and their optimized parameters (such as a sparsity parameter ρ, and sparsity penalty θ) were optimized by the PSO before using the SAE network in the training process of the DL framework. The number of particles in the PSO swarm was set as 20, and the maximum number of iterations was set as 100.

Table 3 illustrates the optimized sparsity penalty parameters for SAE hidden layers with optimal sparsity parameter ($\rho = 0.11$).

Fig. 7 shows that the optimization process at the iteration number 72 inclined to be the steady-state, and the recognition
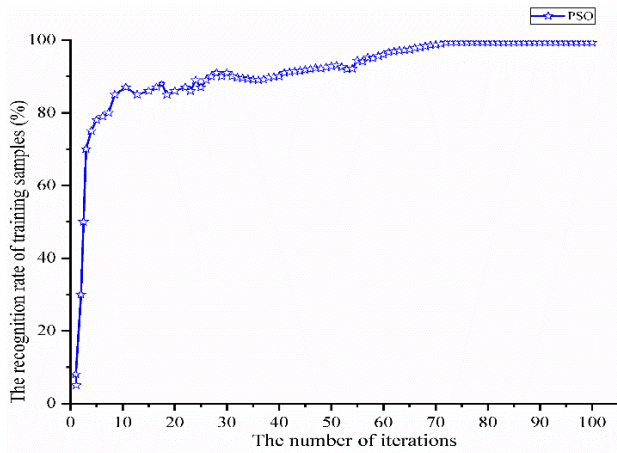
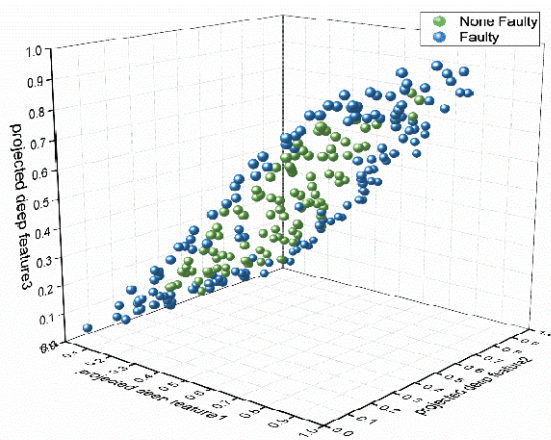FIGURE 7. The optimization process using the PSO algorithm.

FIGURE 8. 3D visualization of dimensional features extracted by SAE.

FIGURE 9. Graph of Accuracy versus Number of Epochs.

TABLE 4. Overall accuracy comparison between the proposed framework and other models.

Models	Accuracy (%)	Detection Time (Sec)	Precision value		Recall value		F1 Score	
			fault	none	fault	none	fault	none
ANN	49.5	3.62	0.00	0.50	0.00	0.99	0.00	0.66
RNN	69.0	1.28	0.65	0.66	0.66	0.65	0.66	0.65
Proposed RNN-LSTM	93.5	**0.17**	0.94	0.93	0.93	0.94	0.93	0.94

rate of training samples extended 99.32%. Hence, a well-trained deep feature learning network can be achieved after the completion of the SAE optimization process.

Though, there was some of the group overlapping between different vibration events (faulty, non-faulty) on the single deep feature, so for more clearance, faulty events need to be projected with more than two features. It was difficult to present the features completely, so the first three projected features were chosen for 3D visualization of dimensional features extracted by SAE, as displayed in Fig. 8.

However, it is noted that the extracted features are dominant in deep learning approaches, which were stored in the feature database to be used in the DL framework testing phase. After training the SAE, fault detection in time series data can deeply learn the multiple features sequence. Therefore, the final step of the proposed DL framework was held in the fault detection phase using RNN-LSTM.

C. FAULT DETECTION USING DL FRAMEWORK

The RNN-LSTM approach was applied in this stage to learn long term dependencies in data of time series to assist in detecting the faults efficiently. In this respect, 2000 samples
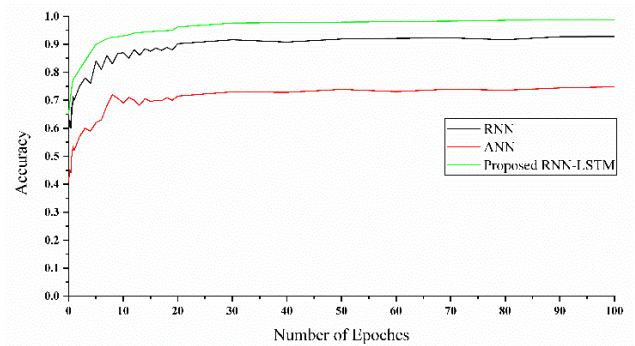
were selected and applied to train and validate RNN-LSTM to be used in the fault detection phase of the proposed DL framework, as was illustrated in Fig. 2. Subsequently, the collected dataset (2000 samples) was divided into two parts, the training dataset (80% (including 800 samples for the normal state and 800 samples for every fault state) and the remaining 20 for the testing dataset.

For verifying effectiveness, the proposed DL framework, models of ANN and RNN were also evaluated for the data set. In Fig. 9, the proposed DL framework has a better accuracy rate than ANN and RNN because of the stage of feature extraction and optimization in its methodology.

The overall accuracy, the running time of fault detection, F1-score, and precision and recall values of models are summarized in Table 4. The accuracy of the proposed framework is the highest when compares to other approaches, which are ANN (49.5%), RNN (69.0%). It is noticeable that the time efficiency for the proposed DL framework is higher with rapid detection time (0.17 sec) compared with other methods, as indicated in Table 4. The running time of fault detection was executed in this paper on an experimental environment with configurations, as described in Table 1.

In order to validate the fault detection of the proposed DL framework, a testing dataset of 400 samples (including 300 samples for the normal state and 100 samples for every fault state) was leveraged for validating. Likewise, the proposed DL framework was evaluated with respect to features to conclude the effect of features on fault detection and which feature is more effective. As can be seen from Table 5, the detection accuracy was calculated based on the extracted features and four data partitions. The predominant feature (F1, F3, and F4), which were provided by SAE, increased the

TABLE 5. Performance of DL framework for fault detection.

Features	Data partition(%)			
	60/40	70/30	80/20	90/10
F1	88.79	90.77	91.48	94.41
F2	86.33	89.49	91.18	92.07
F3	92.66	95.20	95.00	96.14
F4	84.62	87.31	91.6	93.59
Multiple features	96.53	97.39	98.27	**99.67**

accuracy of fault detection. The highest accuracy of detection was for feature 3 (F3) and the combination of features (multi-features) at data partition of (90/10).

The increase in detection accuracy with the changing of data partitions is obvious. Moreover, detection at multiple features has better performance than other features.

The next step after fault detection is alerting and preparing maintenance tasks. According to the hybrid RCM model [18], the CBM task is to be assigned for high-risk failure, while PSM for medium risk, and CM for low risk.

From Tables 4-5, the proposed framework has a better performance than other methods due to advantages in feature extraction and fault detection phases with a detection accuracy of 99.67%. Additionally, the results reveal that the proposed deep learning framework is capable of detecting faults from industrial data without labels. This can professionally help data engineers to extract feature automatically and overcome relying on human experience based on the unsurprised fault detection approach.

V. CONCLUSION AND FUTURE WORK
This study presents a new DL framework for fault detection of the electrical generator from process signals. There are a lot of challenges in fault detection for process signals of traditional methods due to a large number of different types of process signals and hazards in the oil and gas. Variations of failure modes in the hazardous area cause difficulty to predict or diagnose the hidden faults; furthermore, the presence of hazardous gas and electricity, which increases the chance of fault misdetection. Though, the deep learning method has been performed greatly in a hazardous condition for identifying faults from the huge database of machine process signals.

The experimental results demonstrate that the proposed DL framework is capable of detecting faults and extracting deep features by deep processing methodology for vibration signals, which can reduce the fault risks of the electrical generator in risky oil and gas areas. Also, the framework offers good unsupervised deep learning for unlabeled data. Moreover, industrial plants could add to this DL framework to detect and categorize their risky faults, which avoid future shutdowns, specifically in hazardous zones such as petrochemical and nuclear-powered plants.

The main contribution of the research is developing the DL framework for fault detection systems in hazardous oil and gas plan. For limitations and future directions of the present work, the proposed DL framework was applied for fault detection and did not cover fault prediction. Also, looking for multi-objective optimization of hyperparameters and analysis

of the extracted fault features are additional challenging tasks. All of these could be improved by future researches. In future research, we will extend our DL framework in online fault diagnosis and ranking by means of different artificial techniques that can help operators and engineers to analyze and study the root causes of machine failures for an effective maintenance plan and failure prevention.

REFERENCES
[1] R. Teti, K. Jemielniak, G. O'Donnell, and D. Dornfeld, "Advanced monitoring of machining operations," *CIRP Ann.*, vol. 59, no. 2, pp. 717–739, 2010, doi: 10.1016/j.cirp.2010.05.010.
[2] M. Canizo, E. Onieva, A. Conde, S. Charramendieta, and S. Trujillo, "Real-time predictive maintenance for wind turbines using big data frameworks," in *Proc. IEEE Int. Conf. Prognostics Health Manage. (ICPHM)*, Jun. 2017, pp. 70–77, doi: 10.1109/ICPHM.2017.7998308.
[3] M. Hernandez-Vargas, E. Cabal-Yepez, and A. Garcia-Perez, "Real-time SVD-based detection of multiple combined faults in induction motors," *Comput. Electr. Eng.*, vol. 40, no. 7, pp. 2193–2203, Oct. 2014, doi: 10.1016/j.compeleceng.2013.12.020.
[4] F. Liu, C. Shen, Q. He, A. Zhang, Y. Liu, and F. Kong, "Wayside bearing fault diagnosis based on a data-driven Doppler effect eliminator and transient model analysis," *Sensors*, vol. 14, no. 5, pp. 125–8096, May 2014, doi: 10.3390/s140508096.
[5] S. S. Ng, P. W. Tse, and K. L. Tsui, "A one-versus-all class binarization strategy for bearing diagnostics of concurrent defects," *Sensors*, vol. 14, no. 1, pp. 321–1295, Jan. 2014, doi: 10.3390/s140101295.
[6] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: A review and roadmap," *Mech. Syst. Signal Process.*, vol. 138, Apr. 2020, Art. no. 106587, doi: 10.1016/j.ymssp.2019.106587.
[7] X. Jin, Y. Sun, Z. Que, Y. Wang, and T. W. S. Chow, "Anomaly detection and fault prognosis for bearings," *IEEE Trans. Instrum. Meas.*, vol. 65, no. 9, pp. 2046–2054, Sep. 2016, doi: 10.1109/TIM.2016.2570398.
[8] Z. Feng, M. Liang, and F. Chu, "Recent advances in time–frequency analysis methods for machinery fault diagnosis: A review with application examples," *Mech. Syst. Signal Process.*, vol. 38, no. 1, pp. 165–205, Jul. 2013, doi: 10.1016/j.ymssp.2013.01.017.
[9] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, Jun. 2015, doi: 10.1109/TIE.2015.2417501.
[10] X. Chen, A. Ji, and G. Cheng, "A novel deep feature learning method based on the fused-stacked AEs for planetary gear fault diagnosis," *Energies*, vol. 12, no. 23, p. 4522, Nov. 2019, doi: 10.3390/en12234522.
[11] D. Chen, S. Yang, and F. Zhou, "Transfer learning based fault diagnosis with missing data due to multi-rate sampling," *Sensors*, vol. 19, no. 8, p. 1826, Apr. 2019, doi: 10.3390/s19081826.
[12] Y. Qi, C. Shen, D. Wang, J. Shi, X. Jiang, and Z. Zhu, "Stacked sparse autoencoder-based deep network for fault diagnosis of rotating machinery," *IEEE Access*, vol. 5, pp. 15066–15079, 2017, doi: 10.1109/ACCESS.2017.2728010.
[13] J. Long, J. Mou, L. Zhang, S. Zhang, and C. Li, "Attitude data-based deep hybrid learning architecture for intelligent fault diagnosis of multi-joint industrial robots," *J. Manuf. Syst.*, Sep. 2020, doi: 10.1016/j.jmsy.2020.08.010.
[14] J. Long, S. Zhang, and C. Li, "Evolving deep echo state networks for intelligent fault diagnosis," *IEEE Trans. Ind. Informat.*, vol. 16, no. 7, pp. 4928–4937, Jul. 2020, doi: 10.1109/TII.2019.2938884.
[15] X. Yuan, J. Zhou, B. Huang, Y. Wang, C. Yang, and W. Gui, "Hierarchical quality-relevant feature representation for soft sensor modeling: A novel deep learning strategy," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 3721–3730, Jun. 2020, doi: 10.1109/TII.2019.2938890.
[16] Y. Wang, Z. Pan, X. Yuan, C. Yang, and W. Gui, "A novel deep learning based fault diagnosis approach for chemical process with extended deep belief network," *ISA Trans.*, vol. 96, pp. 457–467, Jan. 2020, doi: 10.1016/j.isatra.2019.07.001.
[17] X. Yuan, L. Li, Y. Shardt, Y. Wang, and C. Yang, "Deep learning with spatiotemporal attention-based LSTM for industrial soft sensor model development," *IEEE Trans. Ind. Electron.*, early access, Apr. 9, 2020, doi: 10.1109/TIE.2020.2984443.

[18] M. Alrifaey, T. S. Hong, A. As'arry, E. E. Supeni, and C. K. Ang, "Optimization and selection of maintenance policies in an electrical gas turbine generator based on the hybrid reliability-centered maintenance (RCM) model," *Processes*, vol. 8, no. 6, p. 670, Jun. 2020, doi: 10.3390/pr8060670.

[19] M. K. Alzaylaee, S. Y. Yerima, and S. Sezer, "DL-droid: Deep learning based Android malware detection using real devices," *Comput. Secur.*, vol. 89, Feb. 2020, Art. no. 101663, doi: 10.1016/j.cose.2019.101663.

[20] D. Verstraete, E. Droguett, and M. Modarres, "A deep adversarial approach based on multi-sensor fusion for semi-supervised remaining useful life prognostics," *Sensors*, vol. 20, no. 1, p. 176, Dec. 2019, doi: 10.3390/s20010176.

[21] M. Munir, S. A. Siddiqui, M. A. Chattha, A. Dengel, and S. Ahmed, "FuseAD: Unsupervised anomaly detection in streaming sensors data by fusing statistical and deep learning models," *Sensors*, vol. 19, no. 11, p. 2451, May 2019, doi: 10.3390/s19112451.

[22] M. Alrifaey, T. S. Hong, E. Supeni, A. As'arry, and C. Ang, "Identification and prioritization of risk factors in an electrical generator based on the hybrid FMEA framework," *Energies*, vol. 12, no. 4, p. 649, Feb. 2019, doi: 10.3390/en12040649.

[23] X. Li, J. Fang, W. Cheng, H. Duan, Z. Chen, and H. Li, "Intelligent power control for spectrum sharing in cognitive radios: A deep reinforcement learning approach," *IEEE Access*, vol. 6, pp. 25463–25473, 2018, doi: 10.1109/ACCESS.2018.2831240.

[24] S. Jiang, K.-S. Chin, L. Wang, G. Qu, and K. L. Tsui, "Modified genetic algorithm-based feature selection combined with pre-trained deep neural network for demand forecasting in outpatient department," *Expert Syst. Appl.*, vol. 82, pp. 216–230, Oct. 2017, doi: 10.1016/j.eswa.2017.04.017.

[25] C. Lu, Z.-Y. Wang, W.-L. Qin, and J. Ma, "Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification," *Signal Process.*, vol. 130, pp. 377–388, Jan. 2017, doi: 10.1016/j.sigpro.2016.07.028.

[26] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci. (MHS)*, 1995, pp. 39–43, doi: 10.1109/MHS.1995.494215.

[27] W. Cai, J. Yang, Y. Yu, Y. Song, T. Zhou, and J. Qin, "PSO-ELM: A hybrid learning model for short-term traffic flow forecasting," *IEEE Access*, vol. 8, pp. 6505–6514, 2020, doi: 10.1109/ACCESS.2019.2963784.

[28] M. Abd El-salam, E. Beshr, and M. Eteiba, "A new hybrid technique for minimizing power losses in a distribution system by optimal sizing and siting of distributed generators with network reconfiguration," *Energies*, vol. 11, no. 12, p. 3351, Nov. 2018, doi: 10.3390/en11123351.

[29] W. H. Lim, N. A. M. Isa, S. S. Tiang, T. H. Tan, E. Natarajan, C. H. Wong, and J. R. Tang, "A self-adaptive topologically connected-based particle swarm optimization," *IEEE Access*, vol. 6, pp. 65347–65366, 2018, doi: 10.1109/ACCESS.2018.2878805.

[30] H. Dong, X. Yang, A. Li, Z. Xie, and Y. Zuo, "Bio-inspired PHM model for diagnostics of faults in power transformers using dissolved Gas-in-Oil data," *Sensors*, vol. 19, no. 4, p. 845, Feb. 2019, doi: 10.3390/s19040845.

[31] K.-I. Funahashi and Y. Nakamura, "Approximation of dynamical systems by continuous time recurrent neural networks," *Neural Netw.*, vol. 6, no. 6, pp. 801–806, Jan. 1993, doi: 10.1016/S0893-6080(05)80125-X.

[32] M. A. Ranzato, C. Poultney, S. Chopra, and Y. L. Cun, "Efficient learning of sparse representations with an energy-based model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1137–1144.

[33] Z. Li, Y. Wang, and K. Wang, "A deep learning driven method for fault classification and degradation assessment in mechanical equipment," *Comput. Ind.*, vol. 104, pp. 1–10, Jan. 2019, doi: 10.1016/j.compind.2018.07.002.

[34] F. D. S. Lima, G. M. R. Amaral, L. G. D. M. Leite, J. P. P. Gomes, and J. D. C. Machado, "Predicting failures in hard drives with LSTM networks," in *Proc. Brazilian Conf. Intell. Syst. (BRACIS)*, Oct. 2017, pp. 222–227, doi: 10.1109/BRACIS.2017.72.

[35] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014, doi: 10.1109/jstars.2014.2329330.

MOATH ALRIFAEY (Graduate Student Member, IEEE) received the B.S. degree in electronic engineering from IBB University, Yemen, in 2005, the M.S. degree in engineering management from the Delft University of Technology and Taiz University, in 2014, and the Ph.D. degree in industrial engineering from Universiti Putra Malaysia, Malaysia, in 2020.

From 2008 to 2016, he worked as a Control System and Instrumentation Engineer with the Oil and Gas Plant, Yemen. He is currently a Postdoctoral Research Fellow with the Faculty of Engineering, Technology and Built Environment, UCSI University. His current research interests include intelligent manufacturing, reliable maintenance, and computational intelligence, focusing on problems of classification, optimization, and machine learning, deeply in industrial applications, and big data engineering.

WEI HONG LIM (Senior Member, IEEE) received the B.Eng. degree (Hons.) in mechatronic engineering and the Ph.D. degree in computational intelligence from Universiti Sains Malaysia, Penang, Malaysia, in 2011 and 2014, respectively. From 2015 to 2017, he was with the Intelligent Control Laboratory, National Taipei University of Technology, Taiwan, as a Postdoctoral Researcher, where he was a Visiting Researcher, in 2019. He is currently an Assistant Professor and a Researcher with the Faculty of Engineering, Technology and Built Environment, UCSI University. He is working with two national research grants awarded by the Ministry of Education, Malaysia, and five internal grant projects supported by UCSI University. He has published more than 40 research articles in research areas related to computational intelligence, optimization algorithms, energy management, and digital image processing. To date, he has been awarded with the Chartered Engineer (CEng) Qualification from the U.K. Engineering Council and the Professional Technologist (PTech) Qualification from the Malaysia Board of Technologist (MBOT). He is an active reviewer for various reputable journals, such as IEEE Access, *Complexity*, *Mathematical Problem in Engineering*, *Computational Intelligence and Neuroscience*, and so on. He is also involved in various professional bodies.

CHUN KIT ANG received the Bachelor of Engineering degree (Hons.) in mechatronic engineering from the University of UCSI, Malaysia, in 2010, and the Ph.D. degree in mechanical and manufacturing engineering from Universiti Putra Malaysia, in 2014. He is currently the Dean of the Faculty of Engineering, Technology and Built Environment, UCSI University. He has published many articles which related to the application of artificial intelligence in recent years. His research interests include artificial intelligence, soft computing, robotics, and mechatronics.

• • •