WILEY | Hindawi

*Research Article*

# Classification of Fault Severity in Induction Machine Systems Based on Temporal Convolutions and Recurrent Networks

**V. Mashayekhi** (ID),[1] **S. Hasani Borzadaran** (ID),[2] **and M. Hoseintabar Marzebali** (ID)[1]

[1]*Department of Electrical Engineering, Shahrood University of Technology, Shahrood, Iran*
[2]*Faculty of Computer Engineering, Shahrood University of Technology, Shahrood, Iran*

Correspondence should be addressed to V. Mashayekhi; vmashayekhi@shahroodut.ac.ir

Detection and severity identification of mechanical and electrical faults by means of noninvasive methods such as electrical signatures of induction machine have attracted much attention in recent years. Since operating conditions of machines and severity of faults in incipient stages influence the amplitude of fault index in the fault detection process, diagnosing fault occurrence and severity can be more complicated. In this study, an efficient method for fault detection and classification in induction machine based on deep neural networks is introduced. The introduced method applies the long short-term memory (LSTM) and fully convolutional neural networks (FCNs) in a conjoined manner. The authors use the FCN architecture for feature extraction from the time-series signal and augment it with LSTM to improve classification performance. This structure has not been previously applied for fault severity detection in induction machine systems. The authors avoid manual feature engineering and, by eliminating the preprocessing phase, directly use time series of electrical signals for fault detection and classifications. The experimental results have been carried out in different fault severities and loads. The analysis of the results and comparison with other deep and classical methods show that the faulty cases can be separated based on severity and load levels with a high accuracy (98.92%), which shows that the adopted architecture is successful in automatically extracting discriminative features from the signal.

## 1. Introduction

Three-phase wound rotor induction machines (WRIMs) are widely used in industrial applications such as medium-power wind turbine and electrical motors [1, 2]. However, their performances are limited due to mechanical and electrical faults, which lead to catastrophic disaster in sensitive applications [3]. In this regard, different condition monitoring techniques have been presented to reduce maintenance cost and downtime of such systems [4, 5].

Two different approaches are considered for fault diagnosis in electrical machine, namely the model-based and signal-based methods. Due to some restrictions and uncertainty, the model-based approach cannot be an appropriate method for the fault detection process. Therefore, the data-driven methods, which consider measured signals such as vibration, acoustics, voltage, and current, are more likely

to be considered for the fault detection process [6, 7]. Since some of these signals such as vibration and acoustics have an invasive nature, motor current signature analysis (MCSA) is considered for this purpose. Recently, the MCSA methods are widely used for condition monitoring of electrical and mechanical faults in electrical machine. MCSA methods are well developed to show the effects of faults in electrical signatures properly. However, in the initial stages of electrical and mechanical faults and different load levels, the extracted features used for signal processing techniques such as time domain, frequency domain, and time scale cannot show the severity of fault properly [8]. Fault detection and severity identification based on machine learning methods have recently been introduced [9]. These data-driven approaches can be regarded as time-series classification (TSC) tasks. The major problem in the fault detection process is related to the interclass variability caused by the different

unknown load levels and severity of faults, which reduces the accuracy of the process [9].

Different approaches to TSC exist. The methods such as the K-nearest neighbors with different distance metrics such as dynamic time warping [10] can classify multivariate time series [11]. In addition to distance-based metrics, other traditional feature-based algorithms such as naive logistic (NL) model [12] are also used. These algorithms strongly rely on the extracted features [13]. The feature-based learning methods utilize signal processing algorithms to extract features from the input signals, and consequently, the severity in different load levels is diagnosed by the classification of the extracted features [9]. Different signal processing methods based on frequency-domain, time-domain, and time-scale methods are used for the signal processing step. Nevertheless, capturing the intrinsic features of time-series data is challenging. Conditional random fields (CRFs) are also high-level feature-based temporal classifiers, which make prediction at a time step as a function of the prediction at the previous time step. CRFs oversimplify the temporal dynamics of complex actions [14]. Other variants such as hidden-state CRFs [15] require large number of latent states, which may lead to data overfitting.

Recently, deep learning (DL) is increasingly used to automatically learn complex data representations from raw signals using a network of different abstraction levels [16, 17]. However, the capability of these algorithms in TSC is still understudied [18]. In contrast to the conventional machine learning methods, DL has higher performance in case of feature extraction, diagnostic performances, and transferability [19]. CNNs [20], auto-encoders [21], deep belief networks (DBNs) [22], recurrent neural networks (RNNs) [23], generative adversarial networks (GANs) [24], and other variants [25–27] are among most used DL architectures for machine health monitoring. Zhao et al. [28] present a comparison of different intelligent fault diagnostic systems including traditional ML algorithms and deep architectures. Convolutional neural networks (CNNs) are a popular deep architecture capable of extracting features at different abstraction levels [29].

Several studies use CNNs and their variations for the fault diagnosis, regularly using up to four layers of convolution and pooling. Typically, different preprocessing steps in the time and frequency domains are applied to the input signal of the deep network, to convert the input signal to a two-dimensional format. Lu et al. [30] adopt a four-layer CNN structure for fault classification. Chen et al. [20] preprocess the vibration signals using different statistical measures in the time domain. Moreover, by use of FFT, the multiband spectrum is obtained, further calculating the root-mean-square (RMS) value to maintain the energy shape at the spectrum peaks. The preprocessed signal is then classified using a CNN architecture. Zhang et al. [31] transform data into spectrograms and use a deep fully convolutional neural network with four convolution-pooling layer pairs. Wen et al. [32] convert signals into two-dimensional images and apply CNN based on LeNet-5 for fault diagnosis. Their architecture has two alternating convolutional-pooling layers and two fully connected layers, and padding is used to adapt the size of features.

Zhang et al. [33] use a very deep CNN of 14 layers to perform in noisy environments. Nevertheless, this architecture can increase the risk of overfitting.

Eliminating the preprocessing steps (conversion of data to a two-dimensional signal) can simplify the diagnostic process. Qian et al. use an adaptive overlapping CNN, which directly processes the raw vibration signal and avoids the shift variant property of the signals [34]. Similarly, Eren et al. use an adaptive one-dimensional CNN classifier for bearing fault diagnosis [35].

Among the deep structures, recurrent neural networks (RNNs) are most popular for TSC [36]. RNN variations, including long short-term memory (LSTM) and gated recurrent units (GRUs), model hidden temporal states via internal gating mechanisms. In such networks, the predictions are a function of a set of latent states at each time step. A recent deep structure used for TSC is fully convolutional networks (FCNs) [14–16]. FCN does not require heavy data preprocessing or feature engineering. It has shown superior performance in classifying time-series data [36].

In this study, a new method for fault detection of induction machine by means of electrical signatures is presented using classification of time-series signals. The proposed method is novel from different aspects. First, the authors use the recent FCN architecture for feature extraction from time-series signal. To the best of our knowledge, the use of temporal convolutions as feature extractors in a FCN is not previously studied for diagnosing fault severity in induction machine systems. Second, the authors augment the FCN architecture using the LSTM network, which is popular for TSC. The combined architecture is adopted for unbalanced winding fault (UWF) detection and severity classification in the rotor windings of WRIG in different load levels by means of stator current signature. In comparison with traditional fault detection approaches, the proposed method minimizes the need for expert supervision. Third, in contrast to some previous studies [37–41], our proposed method eliminates the preprocessing phase and directly uses time series of electrical signals for fault detection and classifications. As a result, the proposed method has less complexity and can automatically detect the discriminative features for fault detection and classification. Two WRIMs with the same rating values are used for the validation of proposed method. The authors train the models using only the data of the first machine, while the data of the second machine are used for the testing process. The results show that the faulty cases can be separated based on severity and load levels with a high accuracy (98.92%), and the proposed method can outperform other compared approaches.

The rest of this study is organized as follows: the description of test rig is described in Section 2. Background works and proposed time-series methods are explained in Sections 3 and 4, respectively. Finally, the conclusion section is given in Section 5.

## 2. Test-Rig Description

To testify the impacts of faults in the rotor windings of electrical machine, 250 W, 50 Hz, 400 V, 4 poles, and 1360 rpm WRIM has been used for this purpose. The stator

windings are connected to the three-phase supply system through an autotransformer. The windings of rotor are connected as star structure, and one phase of winding is unbalanced by means of additional resistance. The severity of fault can be changed through variable resistance. The schematic of experimental setup and the test bench is shown in Figure 1. Since the impacts of UWF in the rotor windings of WRIG can be tracked through stator current signatures, the stator current of one phase is measured through the current sensor. The effects of UWF in the rotor windings of WRIM in the frequency domain of stator current are named as fault index ($f_i$) and can be observed as additive frequencies around the supply frequency ($f_s$) (equation (1), Figure 2(a) and 2(b)) [1].

$$(1 \pm 2ks)f_s, \quad k = 1, 2, 3, \ldots, (s: \text{slip}). \tag{1}$$

To show the effects of load levels on stator current signature, prime mover is linked to WRIG through coupling. The prime mover is responsible to rotate WRIG at different rotational speeds. In this regard, in different load levels and fault severities in healthy and faulty conditions, stator current signature has been observed. In the incipient stage of faulty case and low load levels, diagnosing fault can be more difficult in comparison with other cases and does not have specific rhythm in comparison with variation of slips (Figure 2(c)). Therefore, 12 different classes have been considered for this purpose. Each class can be identified based on fault severity (resistance unbalance ($R_{\text{unb}}$)) and load level ($s$: slip). Table 1 shows these 12 classes ($A_1, \ldots, A_{12}$) and are considered for the fault detection process. The data have been collected at three different speeds and three external resistances with different amplitudes.

The amplitudes of external resistance are normalized to the rotor resistance. It can be easily found out that this amplitude covers an extensive range of fault severity from very low unbalance fault ($R_{\text{unb}} = 0.029$ p.u.) to higher values. It is necessary to note that the collected data in healthy conditions are shown in Table 1 as $R_{\text{unb}} = 0$ ($A_1, A_2, A_3$). The plots for healthy and faulty cases ($R_{\text{unb}} = 0$) and their spectrums are now included in Figure 2, and comments are added to the manuscript text. It is evident that the amplitude of faults in the case of healthy conditions has lower amplitude in comparison with faulty ones ($-54$ dB) (Figure 2(d)). It is necessary to note that the fault characteristic frequency in the case of healthy condition cannot be detected clearly in the spectrum of stator current and the amplitude modulation can be found out in the stator current of machine (Figures 2(e) and 2(f)).

All collected data are derived with $2k$ sampling frequency. Each class, $A_i$ ($i = 1, \ldots, 12$), has 45 saved data with 12.5 s time duration with the sampling frequency of $2k$. Two WRIMs with the same rating values are used for the validation of proposed method. In this regard, the authors train the models using only the data of the first machine, while the data of the second machine are used for the testing process. Therefore, $22500 \times 45$ data are considered for the training process and the same number of data is also used for testing process.

## 3. Computational Model

*3.1. Temporal Convolutional Networks.* In this study, the authors apply the fully convolutional neural network and the long short-term memory (LSTM) network in a conjoined manner for fault detection and severity classification of an induction machine. This combined model is shown to be effective in time-series classification [16]. In this section, the authors describe the required background and the employed structure.

Based on the neurobiology of the visual cortex, convolutional neural network (CNN) [17] is a neural network model that is generally composed of multiple convolutional layers along with fully connected layers. It may also contain subsampling steps. The convolution filters along with an appropriate pooling function can reasonably reduce the data dimensionality delivered to the fully connected classifier network.

In this study, the authors extract features using a temporal convolutional network (TCN) in a fully convolutional network (FCN). A TCN is a variation of CNN for the sequence modelling tasks [14]. The distinguishing characteristics of TCNs are twofold. First, it can map an input sequence to an output sequence of identical length. Second, the convolutions in TCN are causal, such that no information is exposed from future to past.

As stated by Lea et al. [14], the input to a TCN is a time-series signal. Let $X_t \in \mathbb{R}^{F_0}$ be the input feature vector of length $F_0$ in time step $t$ where $1 < t \leq T$. Each sequence may have a specific $T$ value. The number of time steps in each layer $l$ is denoted as $T_l$. A set of 1D filters are applied in each convolution layer to capture the dynamics of input signals. The filters for each layer $l$ are parameterized by tensor $W^{(l)} \in \mathbb{R}^{F_l \times d \times F_{l-1}}$ and biases $b^{(l)} \in \mathbb{R}^{F_l}$, where $d$ is the filter duration. In the same layer, the $i^{\text{th}}$ entry of the unnormalized activation $\widehat{E}_t^{(l)} \in \mathbb{R}^{F_l}$ is a function of the incoming normalized activation matrix $E^{(l-1)} \in \mathbb{R}^{F_{l-1} \times T_{l-1}}$ from the previous layer [14]:

$$\widehat{E}_{i,t}^{(l)} = f\left( b_i^{(l)} + \sum_{t'=1}^{d} \left\langle W_{i,t',\cdot}^{(l)}, E_{\cdot, t+d-t'}^{(l-1)} \right\rangle \right), \tag{2}$$

where $f$ is a rectified linear unit (ReLU).

A basic convolution block consists of a convolution layer, followed by batch normalization, followed by a rectified linear unit (ReLU) activation function. For each layer, 1D convolutions capture the dynamics of lower-level features, and pooling can aid in computing long-range temporal patterns.

*3.2. Long Short-Term Memory.* A recurrent neural network (RNN) [36] is derived from the feedforward neural network and has an internal state enabling it to deal with an input sequence with variable length. It is a deep structure capable of detecting structures in streams of data. The connections between nodes are designed to consider temporal relations such that in each state, the output is dependent on the outputs of the previous states. Therefore, the network
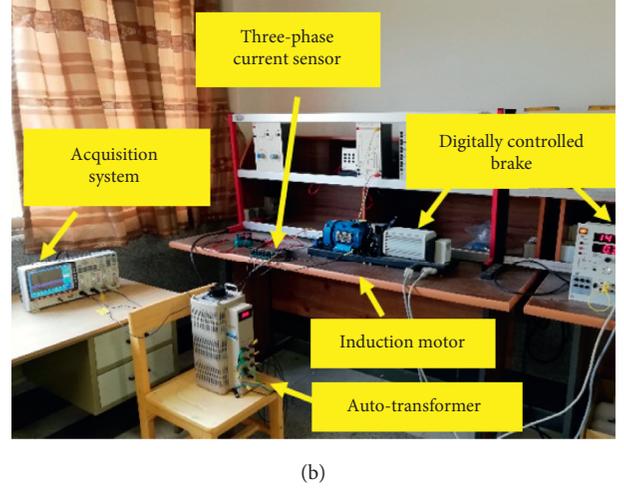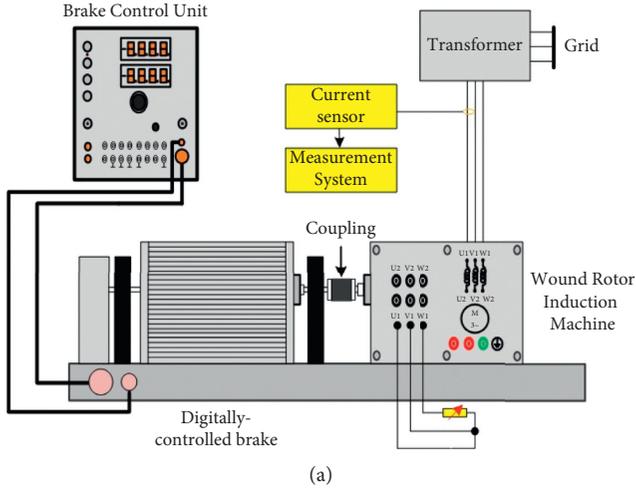
FIGURE 1: (a) Schematic of experimental setup for unbalanced winding fault and (b) experimental test bench.

outcome is influenced by what it has learnt from the past. In recent years, a main variant of RNN named long short-term memory (LSTM) [39] has become popular in different applications.

In a RNN, the input and the hidden states are simply passed through a single tanh layer. Long short-term memory (LSTM) networks [40] improve on this simple transformation and introduce additional gates and a cell state as follows: the forget gate ($f$) controls the persistence of a value in the cell, the input gate restrains entrance of a new value into the cell, and the output gate ($o$) determines how much effect the cell value has on the cell output.

The hidden layer of LSTM consists of a set of recurrently connected units. At time $t$, the input vector $x^t$ is fed into the network. The elements of each block in a layer of memory cells are defined by the set of equation (3) [41], in which $h_{l-1}^t$ shows the previous layer (or network input) at the same step $t$, and $h_l^{t-1}$ is the same layer at the previous step $t-1$. In addition, $w$ is the weight parameter, and $b$ is the bias. $\sigma$ and tanh are pointwise sigmoid (logistic) and hyperbolic tangent activations, respectively. $\odot$ operator symbolizes pointwise multiplication.

$$
\begin{aligned}
g_l^t &= \tanh\left(w_l^{gx} h_{l-1}^t + w_l^{gh} h_l^{t-1} + b_l^g\right), \\
i_l^t &= \sigma\left(w_l^{ix} h_{l-1}^t + w_l^{ih} h_l^{t-1} + b_l^i\right), \\
f_l^t &= \sigma\left(w_l^{fx} h_{l-1}^t + w_l^{fh} h_l^{t-1} + b_l^f\right), \\
o_l^t &= \sigma\left(w_l^{ox} h_{l-1}^t + w_l^{oh} h_l^{t-1} + b_l^o\right), \\
s_l^t &= g_l^t \odot i_l^t + s_l^{t-1} \odot f_l^t, \\
h_l^t &= \tanh\left(s_l^t\right) \odot o_l^t.
\end{aligned}
\tag{3}
$$

The attentional mechanism can also be applied to the LSTM architecture (Figure 3), such that the output is selectively related to elements in the input sequence [18]. This mechanism, which is now widely used in different contexts, was initially designed for the sequence-to-sequence models. It can lift the limitation of fixed-length internal representation [42] and help improve the network performance for sequences of longer lengths.

*3.3. Network Architecture.* As shown in Figure 3, the system has two subblocks, namely, the fully convolutional block and the LSTM block. The time series is inserted in both blocks, with the former receiving the input as univariate with multiple time steps, while the latter seeing it as a multivariate time series in one time step. Therefore, the length of the input sequence determines the number of time steps for the FCN block input and number of variables for the LSTM block input. The dimension shuffle element has the responsibility of preparing the multivariate input for the LSTM block. It is worth to note that this transformation increases the performance of the LSTM.

As shown in Figure 4, the FCN block is composed of temporal convolutions, which have been shown to be practical in time-series analysis [36]. Four stacked units, each comprised of a temporal convolutional layer, followed by batch normalization, followed by the ReLU activation, construct the FCN block. Using this setting, discriminative features can be extracted from the input [43]. Based on 1D convolutions, pooling, and channel-wise normalization, this structure can hierarchically capture low- to high-level temporal information. Global average pooling decreases the number of output parameters before producing the block output.

The combined architecture with LSTM can boost the performance of a sole FCN. In LSTM, intermediate activations are a function of the low-level features at the current time step and the state at the previous time step. The temporal convolutional filters, on the other hand, are a function of raw data across a longer time period.

The LSTM is followed by a dropout to combat overfitting. As described earlier, the attention mechanism may be introduced to the network by substituting the LSTM cells resulting in the LSTM-FCN architecture. The output of the two blocks is concatenated and fed into the softmax classifier.

## 4. Experiments

In this section, the authors first discuss the parameter setting of the network, the dataset, and evaluation metrics. Then, the actual experimental results are presented. The implementation
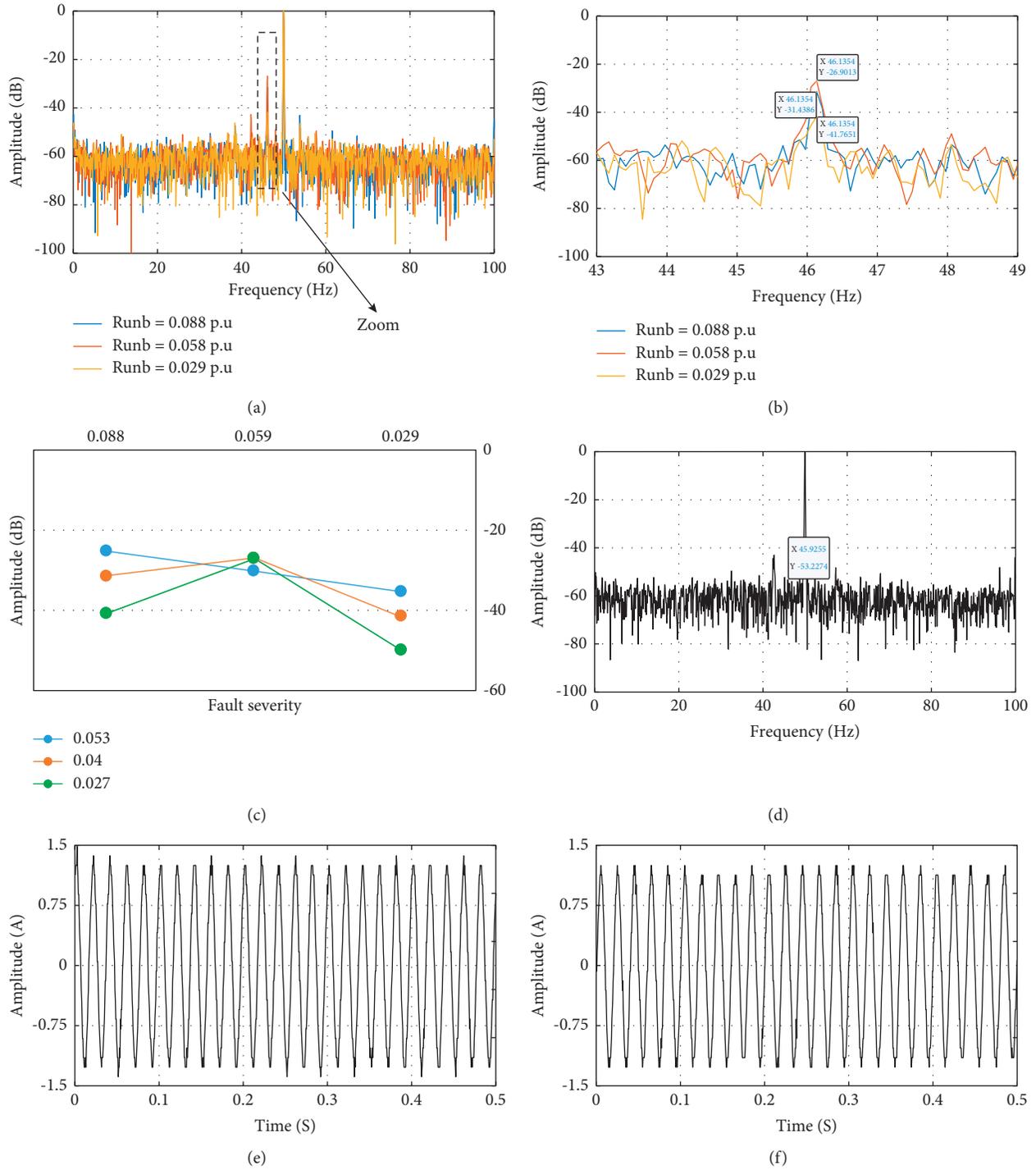
FIGURE 2: Comparison between healthy and faulty stator current spectrums at different fault severities and slips, ((a), (b)) spectral analysis at $s = 0.04$ and different fault severities, (c) comparison between fault index $((1-2s)f_s)$ in different slips and fault severities, (d) spectrum of stator current in healthy condition at $s = 0.04$, (e) stator current at healthy condition at $s = 0.04$, and (f) stator current at faulty condition at $s = 0.04$.

TABLE 1: Classification of collected data from stator current sensor.

| Slip/severity ($R_{\text{unb}}$) (p.u.) | 0 | 0.029 | 0.059 | 0.088 |
|---|---|---|---|---|
| 0.053 (1420 rpm) | $A_1$ | $A_4$ | $A_7$ | $A_{10}$ |
| 0.04 (1440 rpm) | $A_2$ | $A_5$ | $A_8$ | $A_{11}$ |
| 0.027 (1460 rpm) | $A_3$ | $A_6$ | $A_9$ | $A_{12}$ |

was performed using the Keras [44] library with the TensorFlow [45] backend. The authors used the Adam optimizer [46], with initial and final learning rates of $1e-3$ and $1e-4$, respectively. The number of training epochs and initial batch size was set to 1000 and 128, respectively. In the LSTM block, the authors used 64 units and the dropout rate was set to 80%.
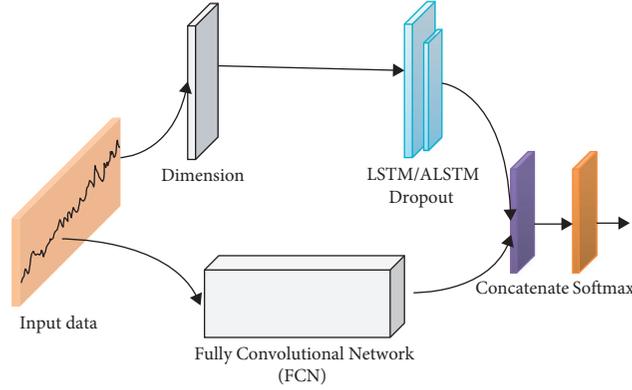
Figure 3: LSTM-FCN architecture.



128 Conv1D          128 Conv1D          128 Conv1D          64 Conv1D          Global Pooling
BN + ReLU           BN + ReLU           BN + ReLU           BN + ReLU
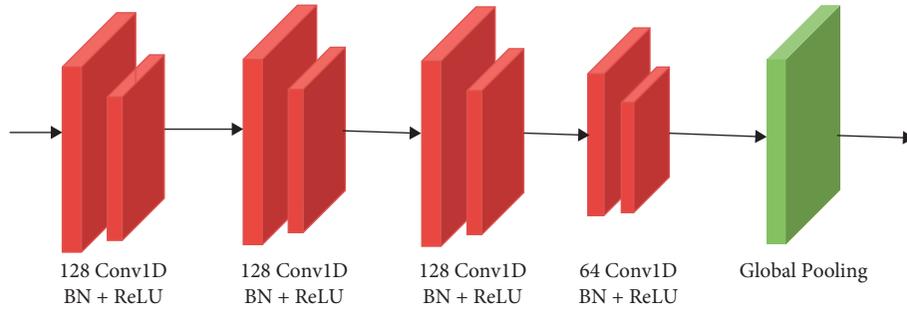
Figure 4: FCN structure of the LSTM-FCN architecture.

*4.1. Evaluation Metrics.* The metrics used for evaluating the performance of the proposed approach are described in this section. The accuracy measure indicates the percent of correctly classified instances. The precision and recall measures are simultaneously used to measure classification performance [47]. Precision denotes the percent of relevant cases among the retrieved ones, while recall refers to the fraction of all relevant cases retrieved by the algorithm. Higher values of both measures (ideally equal to one) demonstrate the superior performance of the classifier. Therefore, if the precision and recall values of one class are both higher high, the classifier has a desirable efficiency in detecting that class [47].

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$
$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{4}$$

In a two-class scenario (positive and negative), the number of cases where the algorithm outcome and the actual class are identical and positive is called true positive (TP). If the actual class is negative, it is false positive (FP). Conversely, true negative (TN) is the number of cases where the algorithm outcome is correctly negative. Finally, false negative (FN) refers to the actual positive cases predicted to be negative.

Another popular performance measurement is $F1$ score. The value of $F1$ score ranges from zero to one [48] and offers a combined measure based on the individual precision and recall values.

$$F1 - \text{score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{5}$$

A $F1$ score value of 1 indicates that both precision and recall are perfectly equal to 1. A high $F1$ score indicates simultaneous efficiency in terms of both precision and recall.

The AUC-ROC curve provides another means of evaluating the performance of a classifier. The AUC-ROC curve demonstrates the capability of the model in discriminating the classes. In the ROC, true-positive rate (TPR) or sensitivity, which is similar to recall, is plotted against the false-positive rate (FPR) or 1-specificity. Sensitivity and specificity, which are inversely proportional, are two measures used together to measure the predictive performance of a classification model [49].

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$
$$\text{specificity} = \frac{\text{FP}}{\text{FP} + \text{TN}}, \tag{6}$$
$$\text{FPR} = 1 - \text{specificity} = \frac{\text{FP}}{\text{FP} + \text{TN}}.$$

To plot the ROC curve, the discrimination threshold of the classifier is varied. With smaller threshold values, more

positive predictions are produced and sensitivity increases. Larger threshold values increase the number of negative predictions, which yields larger specificity values. As mentioned, FPR is 1-specificity. So, when TPR increases, FPR also increases and vice versa.

*4.2. Results.* To assess the performance of the detection model, the authors trained the two LSTM-FCN and ALSTM-FCN using the data of engine 1 and computed the evaluation metrics per class, based on the test data from machine 2. The precision and recall of LSTM-FCN and ALSTM-FCN are presented in Tables 2 and 3, respectively. As the problem is a multiclass task, the authors have used the one vs all approach to compute the precision and recall values. As observed, the classification performance is high for nearly all classes involving healthy and faulty operations. The average accuracy is above 98% and 97% for the two architectures, respectively.

In Figure 5, the precision, recall, and F1 scores are shown for the healthy and fault categories averaged over different slip values. Both networks are competing in detecting the healthy cases according to different metrics. The ALSTM-FCN is superior or competing in the faulty cases from the precision perspective, meaning that it has relatively higher true positives and/or less false positives in these classes. However, according to recall, LSTM-ALSTM majorly outperforms the ALSTM-FCN for lower fault severities (i.e., $R_{unb}$ (p.u.) equal to 0.003 and 0.031), meaning that it has relatively higher true positives and/or less false negatives in these classes. The ALSTM network has an increasing performance when fault severity increases. Combining the two measures, ALSTM-FCN and LSTM-FCN are competing in healthy conditions and higher fault severities, but LSTM-FCN is superior in lower fault severities.

In this study, the authors have a multiclass (12 classes) problem. To extend the ROC curve to multiclass classification, the authors use the one vs all method. Also, as the number of classes is large, instead of plotting ROC for each class, the authors plot the micro- and macroaverages to check the performance of LSTM-FCN and ALSTM-FCN (Figures 6 and 7).

Macro-averaging computes the TPR and FPR metrics independently for each class and takes the average by giving equal weight to all classes [50]. In the microaverage method, the individual true-positive, false-positive, and false-negative values are computed for each class and aggregated to get the statistics [50]. In a multiclass classification, microaverage is preferable if class imbalance is suspected.

As shown in Figures 6 and 7, the ROC curve is plotted for both LSTM-FCN and ALSTM-FCN. As observed, the ROC curve is close to an ideal showing that false positives and false negatives of the classifier are very low. Table 4 shows the area of micro-average ROC curve for both networks. As it can be seen, the AUC in Figure 6 is slightly greater than that of Figure 7, which indicates that LSTM-FCN has better performance than the ALSTM-FCN.

The authors used *t*-distributed stochastic neighbor embedding (*t*-SNE) to visually show the behavior of the LSTM-FCN for classification. The *t*-SNE algorithm

Table 2: Accuracy, precision, recall, and F1 score measurement of 12 classes for the LSTM-FCN.

| | $R_{unb}$ (p.u.) | Class | Precision | Recall | F1 score |
|---|---|---|---|---|---|
| Healthy | 0 | $A_1$ | 1.00 | 1.00 | 1.00 |
| | | $A_2$ | 0.98 | 1.00 | 0.99 |
| | | $A_3$ | 098 | 1.00 | 0.99 |
| Faulty | 0.003 | $A_4$ | 1.00 | 1.00 | 1.00 |
| | | $A_5$ | 0.99 | 0.98 | 0.98 |
| | | $A_6$ | 0.97 | 0.99 | 0.98 |
| | 0.031 | $A_7$ | 1.00 | 1.00 | 1.00 |
| | | $A_8$ | 1.00 | 0.98 | 0.99 |
| | | $A_9$ | 0.96 | 0.98 | 0.99 |
| | 0.093 | $A_{10}$ | 1.00 | 1.00 | 1.00 |
| | | $A_{11}$ | 1.00 | 1.00 | 1.00 |
| | | $A_{12}$ | 1.00 | 0.95 | 0.97 |
| Accuracy | | | | | 98.92% |

Table 3: Accuracy, precision, recall, and F1 score measurement of 12 classes for the ALSTM-FCN.

| | $R_{unb}$ (p.u.) | Class | Precision | Recall | F1 score |
|---|---|---|---|---|---|
| Healthy | 0 | $A_1$ | 1.00 | 1.00 | 1.00 |
| | | $A_2$ | 0.97 | 1.00 | 0.99 |
| | | $A_3$ | 0.97 | 1.00 | 0.99 |
| Faulty | 0.003 | $A_4$ | 1.00 | 1.00 | 1.00 |
| | | $A_5$ | 1.00 | 1.00 | 1.00 |
| | | $A_6$ | 1.00 | 0.59 | 0.74 |
| | 0.031 | $A_7$ | 1.00 | 0.85 | 0.92 |
| | | $A_8$ | 1.00 | 1.00 | 0.94 |
| | | $A_9$ | 0.99 | 1.00 | 1.00 |
| | 0.093 | $A_{10}$ | 1.00 | 1.00 | 1.00 |
| | | $A_{11}$ | 1.00 | 1.00 | 1.00 |
| | | $A_{12}$ | 0.99 | 0.96 | 0.98 |
| Accuracy | | | | | 97.83% |

provides an effective method to visualize a complex high-dimensional dataset [50]. It can successfully uncover hidden structures in the data. Furthermore, transposing similarities between data points to joint probabilities, it attempts to minimize the Kullback–Leibler divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional data. In other words, the *t*-SNE provides a simplified image of the layout and structure of high-dimensional data in two- or three-dimensional frames. Figure 8 shows the *t*-SNE for the LTSM-FCN. As observed, the network is successful in discriminating data of different classes into distinguished groups.

Also, the authors used confusion matrix to visually show the behavior of the LSTM-FCN for classification (Figure 9). This confusion matrix summarizes the classification performance of LSTM-FCN classifier with respect to some test data, which are prepared from machine 2.

Figure 10 demonstrated the performance of the proposed detection model against other architectures. The authors compare five models (LSTM-FCN, ALSTM-FCN, 1-layer LSTM, support vector machine (SVM), and hierarchical LSTM [51]) with first training using engine 1 data and then report the metrics on engine 2 data. Similar to our model, the recent hierarchical LSTM [51] does not use
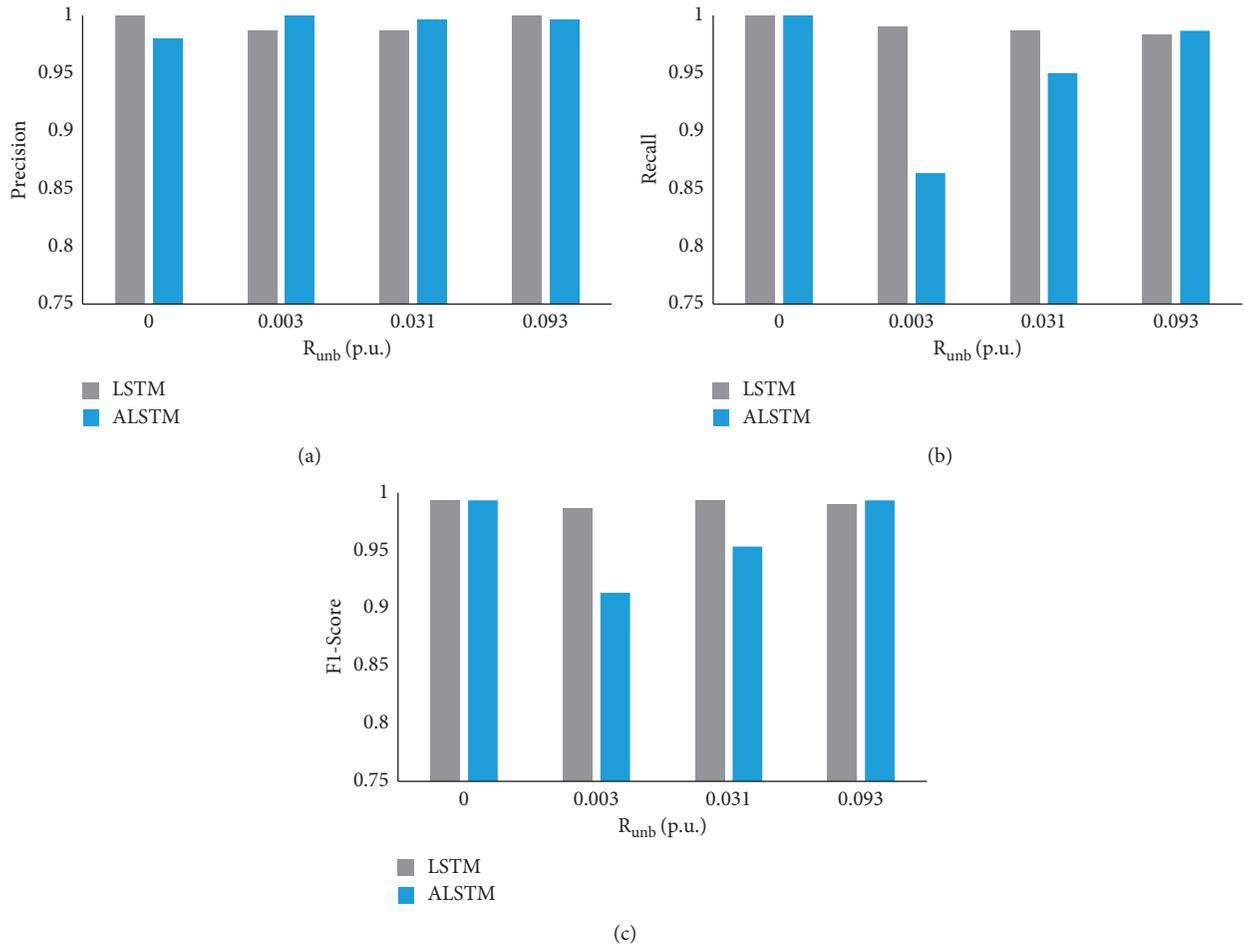
FIGURE 5: Average (a) precision, (b) recall, and (c) F1 score measurements of healthy and different fault categories for LSTM/ALSTM-FCN.
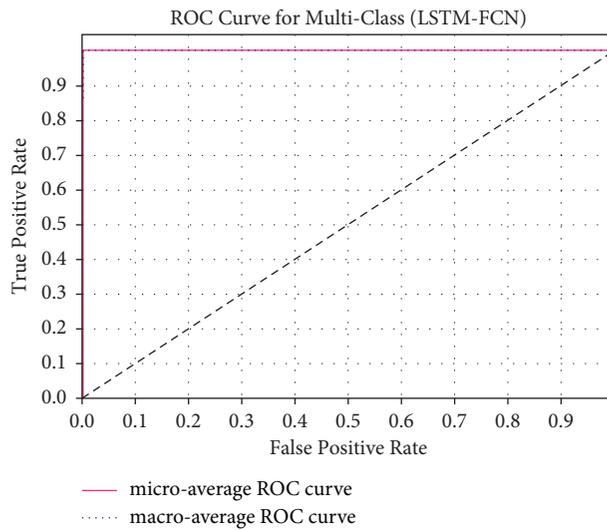


FIGURE 6: ROC curve for multiclass (LSTM-FCN).

preprocessing operations or manual feature extraction. It relies on the memorize forget mechanism of LSTM to extract features inherent in raw temporal signals hierarchically by stacking LSTMs. SVM is a prevalent pattern recognition algorithm used in rotating machinery fault diagnostic issues [52, 53].

As observed, both LSTM-FCN and ALSTM-FCN achieve higher accuracy values compared with the sole
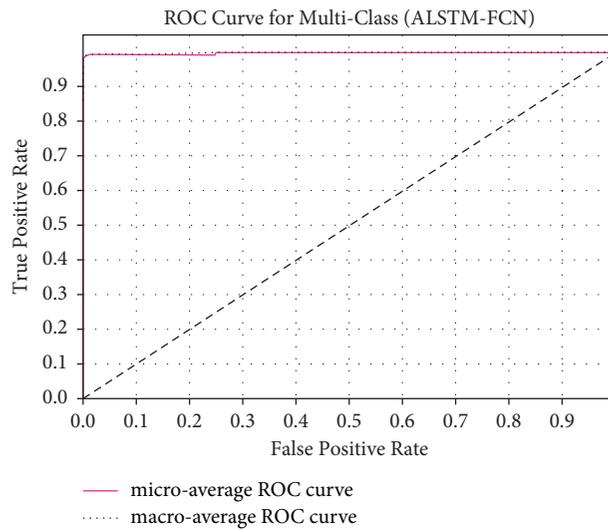
FIGURE 7: ROC curve for multiclass (ALSTM-FCN).

TABLE 4: Area under curve (AUC) of microaverage method.

| LSTM-FCN | ALSTM-FCN |
| --- | --- |
| 0.9995 | 0.9993 |



(a)

(b)

(c)
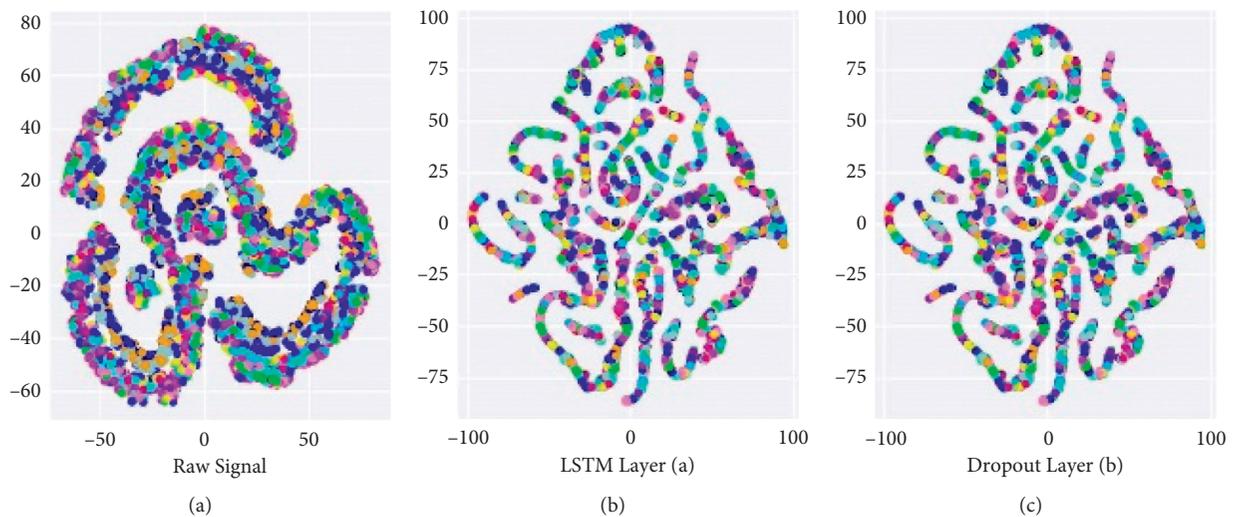
FIGURE 8: Continued.
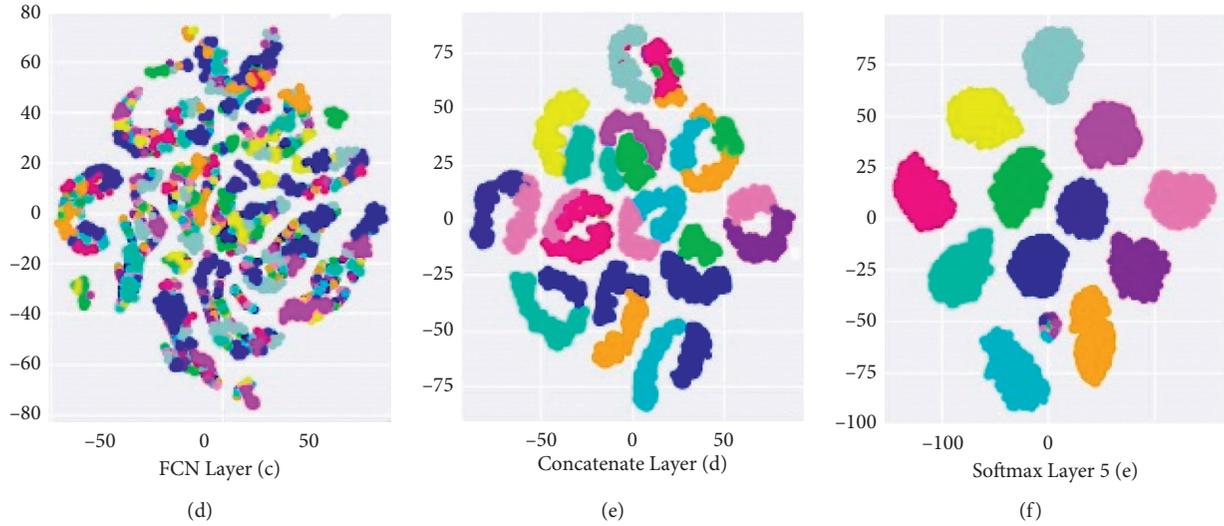
FIGURE 8: Feature visualization via *t*-SNE for the test dataset: *t*-SNE for ((b), (c)) the LSTM structure of the ALSTM-FCN architecture, ((d), (e)) the FCN structure, and (f) the softmax layer. (a) Raw signal. (b) LSTM layer. (c) Dropout layer. (d) FCN layer. (e) Concatenate layer. (f) Softmax layer.



|  | A1 | A2 | A3 | A4 | A5 | A6 | A7 | A8 | A9 | A10 | A11 | A12 |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **A1** | 1273 8.32% | 0 0.0% | 0 0.0% | 6 0.04% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 1279 99.53% |
| **A2** | 0 0.0% | 1271 8.31% | 1 0.01% | 0 0.0% | 22 0.14% | 5 0.03% | 0 0.0% | 4 0.03% | 0 0.0% | 0 0.0% | 0 0.0% | 1 0.01% | 1304 97.47% |
| **A3** | 0 0.0% | 0 0.0% | 1272 8.31% | 0 0.0% | 0 0.0% | 31 0.20% | 0 0.0% | 1 0.01% | 2 0.01% | 0 0.0% | 0 0.0% | 0 0.0% | 1306 97.40% |
| **A4** | 2 0.01% | 0 0.0% | 0 0.0% | 1265 8.27% | 0 0.0% | 0 0.0% | 0 0.0% | 1 0.01% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 1268 99.76% |
| **A5** | 0 0.0% | 3 0.02% | 0 0.0% | 0 0.0% | 1242 8.12% | 0 0.0% | 0 0.0% | 2 0.01% | 0 0.0% | 0 0.0% | 0 0.0% | 1 0.01% | 1248 99.52% |
| **A6** | 0 0.0% | 0 0.0% | 2 0.01% | 0 0.0% | 5 0.03% | 1216 7.95% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 3 0.02% | 1226 99.18% |
| **A7** | 0 0.0% | 0 0.0% | 0 0.0% | 4 0.03% | 0 0.0% | 0 0.0% | 1269 8.29 | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 1273 99.69% |
| **A8** | 0 0.0% | 1 0.01% | 0 0.0% | 0 0.0% | 0 0.0% | 2 0.01% | 0 0.0% | 1263 8.25% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 1269 99.53% |
| **A9** | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 4 0.03% | 16 0.10% | 0 0.0% | 0 0.0% | 1272 8.31% | 0 0.0% | 0 0.0% | 45 0.29% | 1337 95.14% |
| **A10** | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 6 0.04% | 0 0.0% | 0 0.0% | 1275 8.33% | 2 0.01% | 0 0.0% | 1283 99.38% |
| **A11** | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 1 0.01% | 0 0.0% | 0 0.0% | 4 0.03% | 0 0.0% | 0 0.0% | 1271 8.31% | 0 0.0% | 1276 99.61% |
| **A12** | 0 0.0% | 0 0.0% | 0 0.0% | 0 0.0% | 1 0.01% | 5 0.03% | 0 0.0% | 0 0.0% | 1 0.01% | 0 0.0% | 0 0.0% | 1224 8.00% | 1231 99.43% |
|  | 1275 99.84% | 1275 99.69% | 1275 99.76% | 1275 99.22% | 1275 97.41% | 1275 95.37% | 1275 99.53% | 1275 99.06% | 1275 99.76% | 1275 100% | 1275 99.69% | 1275 96.00% | 15300 98.78% |

Predicted (rows) / Actual (columns)

FIGURE 9: Confusion matrix for 12 classes. The first column on the right shows the number of instances of each class that are correctly classified.

LSTM model, with LSTM-FCN yielding more than 98% accuracy. The proposed detection model also largely outperforms SVM and hierarchical LSTM models. The results show that the model is capable of extracting discriminative features from the raw temporal signal, detecting the fault, and classifying its severity with high accuracy.
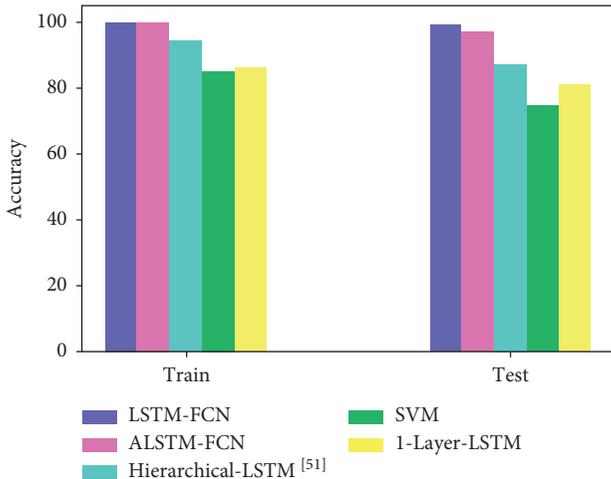
FIGURE 10: Performance comparison of different models.

## 5. Conclusion

In this study, a new approach for the detection and classification of induction machine unbalance fault in different load levels is presented. In this regard, a LSTM recurrent neural network method with temporal convolution is applied for fault detection and classification of its severity for unbalanced winding fault (UWF) detection in the rotor windings of WRIG in different load levels by means of stator current signature. In comparison with traditional fault detection approaches, the proposed method minimizes the need for expert supervision. The proposed method does not need a preprocessing phase and directly uses time series of electrical signals for fault detection and classifications. The results show that the faulty and healthy cases in different load levels can be separated with a high accuracy (98.92%). The proposed method is compared with five different recent methods (LSTM-FCN, ALSTM-FCN, 1-layer LSTM, support vector machine (SVM), and hierarchical LSTM) to show the effectiveness of the proposed method. The accuracy values of some individual classes in the proposed approach may be further improved. To this end, one can use several preprocessing steps to extract more complicated features, or use the extracted features from the final layers of the deep network in another classifier. These are left for future work.

## Data Availability

No data were used to support this study.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

[1] M. H. Marzebali, S. H. Kia, H. Henao, G.-A. Capolino, and J. Faiz, "Planetary gearbox torsional vibration effects on wound-rotor induction generator electrical signatures," *IEEE Transactions on Industry Applications*, vol. 52, no. 6, pp. 4770–4780, 2016.

[2] M. H. Marzebali, J. Faiz, G.-A. Capolino, S. H. Kia, and H. Henao, "Planetary gear fault detection based on mechanical torque and stator current signatures of a wound rotor induction generator," *IEEE Transactions on Energy Conversion*, vol. 33, no. 3, pp. 1072–1085, 2018.

[3] G. H. Bazan, P. R. Scalassara, W. Endo, A. Goedtel, W. F. Godoy, and R. H. C. Palácios, "Stator fault analysis of three-phase induction motors using information measures and artificial neural networks," *Electric Power Systems Research*, vol. 143, pp. 347–356, 2017.

[4] W. Zhang, X. Li, and Q. Ding, "Deep residual learning-based fault diagnosis method for rotating machinery," *ISA Transactions*, vol. 95, pp. 295–305, 2019.

[5] S. Pang, X. Yang, X. Zhang, and X. Lin, "Fault diagnosis of rotating machinery with ensemble kernel extreme learning machine based on fused multi-domain features," *ISA Transactions*, vol. 98, pp. 320–337, 2020.

[6] M. Defdaf, F. Berrabah, A. Chebabhi, and B. D. E. Cherif, "A new transform discrete wavelet technique based on artificial neural network for induction motor broken rotor bar faults diagnosis," *International Transactions on Electrical Energy Systems*, vol. 31, no. 4, Article ID e12807, 2021.

[7] M. B. Gülmezoğlu and S. Ergin, "An approach for bearing fault detection in electrical motors," *European Transactions on Electrical Power*, vol. 17, no. 6, pp. 628–641, 2007.

[8] J. Faiz and H. Nejadi-Koti, "Eccentricity fault diagnosis indices for permanent magnet machines: state-of-the-art," *IET Electric Power Applications*, vol. 13, no. 9, pp. 1241–1254, 2019.

[9] M. K. Saini and A. Aggarwal, "Detection and diagnosis of induction motor bearing faults using multiwavelet transform and naive Bayes classifier," *International Transactions on Electrical Energy Systems*, vol. 28, no. 8, Article ID e2577, 2018.

[10] S. Seto, W. Zhang, and Y. Zhou, "Multivariate time series classification using dynamic time warping template selection for human activity recognition," in *Proceedings of the 2015 IEEE Symposium Series on Computational Intelligence*, pp. 1399–1406, IEEE, Cape Town, South Africa, December 2015.

[11] C. Orsenigo and C. Vercellis, "Combining discrete svm and fixed cardinality warping distances for multivariate time series classification," *Pattern Recognition*, vol. 43, no. 11, pp. 3787–3794, 2010.

[12] T. Jaakkola, M. Diekhans, and D. Haussler, "A discriminative framework for detecting remote protein homologies," *Journal of Computational Biology*, vol. 7, no. 1-2, pp. 95–114, 2000.

[13] Z. Xing, J. Pei, and E. Keogh, "A brief survey on sequence classification," *ACM Sigkdd Explorations Newsletter*, vol. 12, no. 1, pp. 40–48, 2010.

[14] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: a unified approach to action segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 47–54, Amsterdam, Netherlands, October 2016.

[15] W. Pei, H. Dibeklioglu, D. M. Tax, and L. van der Maaten, "Multivariate time-series classification using the hidden-unit logistic model," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, 2018.

[16] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018.

[17] P. Kumar and A. Shankar Hati, "Convolutional neural network with batch normalisation for fault detection in squirrel cage induction motor," *IET Electric Power Applications*, vol. 15, no. 1, pp. 39–50, 2021.

[18] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[19] S. Zhang, S. Zhang, B. Wang, and T. G. Habetler, "Deep learning algorithms for bearing fault diagnostics-a comprehensive review," *IEEE Access*, vol. 8, pp. 29857–29881, 2020.

[20] Z. Chen, C. Li, and R. V. Sanchez, "Gearbox fault identification and classification with convolutional neural networks," *Shock and Vibration*, vol. 2015, Article ID 390134, 10 pages, 2015.

[21] S. Tao, T. Zhang, J. Yang, X. Wang, and W. Lu, "Bearing fault diagnosis method based on stacked autoencoder and softmax regression," in *Proceedings of the 34th Chinese Control Conference (CCC)*, pp. 6331–6335, IEEE, Hangzhou, China, July 2015.

[22] Z. Chen and W. Li, "Multisensor feature fusion for bearing fault diagnosis using sparse autoencoder and deep belief network," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 7, pp. 1693–1702, 2017.

[23] W. Abed, S. Sharma, R. Sutton, and A. Motwani, "A robust bearing fault detection and diagnosis technique for brushless DC motors under non-stationary operating conditions," *Journal of Control, Automation and Electrical Systems*, vol. 26, no. 3, pp. 241–254, 2015.

[24] W. Mao, Y. Liu, L. Ding, and Y. Li, "Imbalanced fault diagnosis of rolling bearing based on generative adversarial network: a comparative study," *IEEE Access*, vol. 7, pp. 9515–9530, 2019.

[25] M. He and D. He, "Deep learning based approach for bearing fault diagnosis," *IEEE Transactions on Industry Applications*, vol. 53, no. 3, pp. 3057–3065, 2017.

[26] R. Razavi-Far, E. Hallaji, M. Farajzadeh-Zanjani et al., "Information fusion and semi-supervised deep learning scheme for diagnosing gear faults in induction machine systems," *IEEE Transactions on Industrial Electronics*, vol. 88, no. 8, pp. 6331–6342, 2018.

[27] B. Luo, H. Wang, H. Liu, B. Li, and F. Peng, "Early fault detection of machine tools based on deep learning and dynamic identification," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 1, pp. 509–518, 2018.

[28] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019.

[29] Z. Xu, C. Li, and Y. Yang, "Fault diagnosis of rolling bearings using an improved multi-scale convolutional neural network with feature attention mechanism," *ISA Transactions*, vol. 110, pp. 379–393, 2021.

[30] C. Lu, Z. Wang, and B. Zhou, "Intelligent Fault diagnosis of rolling bearing using hierarchical convolutional network based health state classification," *Advanced Engineering Informatics*, vol. 32, pp. 139–151, 2017.

[31] W. Zhang, C. Li, G. Peng, Y. Chen, and Z. Zhang, "A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load," *Mechanical Systems and Signal Processing*, vol. 100, pp. 439–453, 2018.

[32] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network- based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5990–5998, 2017.

[33] W. Zhang, F. Zhang, W. Chen, Y. Jiang, and D. Song, "Fault state recognition of rolling bearing based fully convolutional network," *Computing in Science & Engineering*, vol. 21, no. 5, pp. 55–63, 2018.

[34] W. Qian, S. Li, J. Wang, Z. An, and X. Jiang, "An Intelligent fault diagnosis framework for raw vibration signals: adaptive overlapping convolutional neural network," *Measurement Science and Technology*, vol. 29, no. 9, Article ID 095009, 2018.

[35] L. Eren, T. Ince, and S. Kiranyaz, "A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier," *Journal of Signal Processing Systems*, vol. 91, no. 2, pp. 179–189, 2019.

[36] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," 2014, https://arxiv.org/abs/1409.2329.

[37] M. Heydarzadeh, M. Nourani, J. Hansen, and S. Hedayati Kia, "Non-invasive gearbox fault diagnosis using scattering transform of acoustic emission," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 371–375, New Orleans, LA, USA, March 2017.

[38] J. E. Garcia-Bracamonte, J. M. Ramirez-Cortes, J. de Jesus Rangel-Magdaleno, P. Gomez-Gil, H. Peregrina-Barreto, and V. Alarcon-Aquino, "An approach on MCSA-based fault detection using independent component analysis and neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 5, pp. 1353–1361, 2019.

[39] C. Morales-Perez, J. Rangel-Magdaleno, H. Peregrina-Barreto, J. P. Amezquita-Sanchez, and M. Valtierra-Rodriguez, "Incipient broken rotor bar detection in induction motors using vibration signals and the orthogonal matching pursuit algorithm," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 9, pp. 2058–2068, 2018.

[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[41] W. Khan, A. Daud, F. Alotaibi, N. Aljohani, and S. Arafat, "Deep recurrent neural networks with word embeddings for Urdu named entity recognition," *ETRI Journal*, vol. 42, no. 1, pp. 90–100, 2019.

[42] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2019, https://arxiv.org/abs/1409.0473.

[43] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, https://arxiv.org/abs/1312.4400.

[44] F. Chollet, "Keras GitHub," 2015, https://github.com/fchollet/keras.

[45] M. Abadi, P. Barham, J. Chen et al., "TensorFlow: a system for large-scale machine learning tensorflow: a system for large-scale machine learning," in *Proceedings of the OSDI'16: 12th USENIX conference on Operating Systems Design and Implementation*, pp. 265–284, Savannah, GA, USA, November 2016.

[46] I. Bello, B. Zoph, V. Vasudevan, and Q. V. Le, "Neural optimizer search with reinforcement learning," in *Proceedings of the 34th International Conference on Machine Learning*, pp. 459–468, Sydney, Australia, August 2017.

[47] X. Zhang, X. Feng, P. Xiao, G. He, and L. Zhu, "Segmentation quality evaluation using region-based precision and recall measures for remote sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 102, pp. 73–84, 2015.

[48] K. Ting, "Sensitivity and Specificity," in *Encyclopedia of Machine Learning*Springer, Berlin, Germany, 2011.

[49] D. J. Hand and R. J. Till, "A simple generalisation of the area under the ROC curve for multiple class classification problems," *Machine Learning*, vol. 45, no. 2, pp. 171–186, 2001.

[50] L. V. D. Maaten and G. Hinton, "Visualizing data using $t$-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.

[51] L. Yu, J. Qu, F. Gao, and Y. Tian, "A novel hierarchical algorithm for bearing fault diagnosis based on stacked LSTM," *Shock and Vibration*, vol. 2019, Article ID 2756284, 10 pages, 2019.

[52] S.-D. Wu, P.-H. Wu, C.-W. Wu, J.-J. Ding, and C.-C. Wang, "Bearing fault diagnosis based on multiscale permutation entropy and support vector machine," *Entropy*, vol. 14, no. 8, pp. 1343–1356, 2012.

[53] Y.-K. Gu, X.-Q. Zhou, D.-P. Yu, and Y.-J. Shen, "Fault diagnosis method of rolling bearing using principal component analysis and support vector machine," *Journal of Mechanical Science and Technology*, vol. 32, no. 11, pp. 5079–5088, 2018.