IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 2, 2017, accepted October 9, 2017, date of publication October 24, 2017, date of current version November 28, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2764474

Data-Driven Inter-Turn Short Circuit Fault Detection in Induction Machines

ZHAO XU[1], CHANGHUA HU[2], FENG YANG[3], SHYH-HAO KUO[3], CHI-KEONG GOH[4], (Senior Member, IEEE), AMIT GUPTA[4], (Senior Member, IEEE), AND SIVAKUMAR NADARAJAN[4], (Student Member, IEEE)

[1]School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710072, China
[2]Xi'an Research Institute of High-Technology, Xi'an 710025, China
[3]Department of Computing Science, Institute of High Performance Computing, Singapore 138632
[4]Advanced Technology Center, Rolls-Royce Singapore, Singapore 797575

Corresponding author: Changhua Hu (hch_reu@sina.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61603303 and Grant 61401363, in part by the Fundamental Research Funds for the Central Universities under Grand G2017KY0008, in part by the Natural Science Basic Research Program of Shaanxi under Grant 2017JQ6005, in part by the China Postdoctoral Science Foundation under Grant 2017M610650, and in part by the Innovation Development Foundation of Loving Students under Grant ASN-IF2015-1502.

ABSTRACT Inter-turn short circuit (ITSC) fault is one of the critical electrical faults in induction motors that affects the reliability of many industrial applications. Although the use of data-driven fault detection techniques have gained much interest, the main deterrent in using these approaches in detecting ITSC faults is in the generalization and robustness of the diagnosis. In this paper, a data-driven on-line fault detection framework, incorporated with multi-feature extraction/selection and multi-classifier ensemble is proposed, capable of detecting ITSC faults in induction motors (IMs) that subjected to variable operating conditions. By using the synchronous time series signals collected from the machines, multiple feature extraction/selection is explored to find the sensitive faulty features, and the different types of classification strategies is used to increase the diversity of single based models. With the increased diversity of the base learners, the fault detection accuracy is expected to be enhanced and the robustness can be guaranteed. The framework was implemented and tested using real data collected from a designed test bed, with the experimental results showing the effectiveness of the framework in detecting ITSC faults in IMs.

INDEX TERMS Data-driven, fault diagnosis, induction motor, inter-turn short circuit.

I. INTRODUCTION

Induction motors (IMs) are widely used in industries for the conversion of electrical energy into mechanical energy. The failure of IMs will leads to machine downtime, economical loss and even threat of human safety. Inter-Turn Short Circuit (ITSC) fault is a common electrical fault in Induction Motors. Typically, ITSC faults are caused by insulation failures, mechanical stress, moisture and partial discharge [1]. These faults usually begin as minor and when left unresolved, causes phase-to-ground and/or phase-to-phase faults, potentially leading to permanent damage in IMs. This, in turn, increases operating cost due to machine downtime, raising the need for effective reliability monitoring and non-invasive fault diagnostics methods, in reducing these unscheduled downtimes and its associated high costs. Thus, the availability of methods, capable of the early detection of these ITSC faults during motor operations, is critical and increasingly demanded in various many industries.

The subject of diagnosing ITSC faults in the early stages, continued to be a challenge, as these IMs can still perform the desired operations, e.g. rotation at the designed speed, and it is difficult to select fault that are sensitive enough to detect the abnormal electrical characteristics in these IMs if the fault severity is too low. Furthermore, these abnormalities are compounded with changes in the mechanical speed of the IM when the voltage of the variable frequency are imposed as well as the connection or disconnection of the loads from the IMs leading to operating point variations. With the aim of detecting ITSC faults at the incipient stage, the available established techniques can be broadly classified into three categories.

The first category is based on signal analyses, which uses spectral tools to underline the occurrence of specific frequency components related to ITSC, using techniques such as motor current signature analysis [2]–[5], negative sequence analysis [6], electromagnetic field monitoring [7], etc.

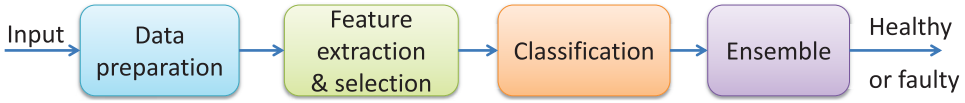


FIGURE 1. Proposed diagnostics condition-monitoring framework.

For example, as an indication of short circuit failure in the windings, it is common to detect changes in the spectrum of the negative sequence voltage. However, the use of such conventional methods is often only for detecting specific known types of abnormalities, therefore unable to detect any new abnormal behaviour present in the system. In addition, if the spectrum of a healthy IM is close or overlapped with that of a faulty IM, it is difficult to distinguish the faulty from the healthy operating conditions. Furthermore, even in safe conditions, the frequency components depend on the speed and power supply, which these strategies are not well adapted to and often only applicable to machines under steady state working conditions i.e., at constant speed and load. In overcoming these limitations, advanced signal processing techniques and high-resolution spectral analysis such as negative- and zero-sequence currents [8] and wavelet based analysis [9] were suggested, with the robustness of these methods still being questioned [10].

The second category uses model-based approaches [11]–[14], which requires physical and mathematical knowledge of the process a prior. The fault diagnosis is realized by generating features such as specific residuals, parameter estimation, state estimation, etc. However, the downside to these categories of methods is that, in many situations, the complexity of the systems under observation makes it almost impossible to derive robust and accurate models for online applications. Moreover, these methods assumes the accurate knowledge of the model parameters, which is not the case in practice, as uncertainties often exist leading to high false alarm [10].

A good alternative to the above-mentioned two categories is the third category, which, prescribes the use of data-driven approaches to detect the ITSC faults. This is undertaken by evaluating the large quantity of available data, collected from non-intrusive and in-expensive sensors, which is already implemented in current IM control system without disturbing the normal operation of the machines. In the areas of online monitoring of IM, several data-driven methods have been applied, with multivariate statistical process monitoring methods and machine learning approaches [15]–[20] to name a few. However, in using these methods, assumptions were often made that all faults are known a priori. With the exception of some common faults, not all faults can be identified before the design of the diagnostic system, thus risking the misclassification of unknown faults. Although fault detection is a reasonably mature field of research, there are very few techniques developed with real-time operations in mind, with the ability to predict incipient faults early enough. Furthermore, in an IM, it can be subjected to various operating states that can be considered normal, arising from the different speed and/or loading conditions, throughout the lifetime of the machine. In detecting these specific mode of failures experienced throughout the lifetime of the machines, the failure detection method must be robust and generalizable over a range of operating conditions.

There were some papers working on motor fault diagnose based on ensemble approach. [21] investigated fault feature extraction of mechanical anomaly on induction motor bearing using ensemble super-wavelet transform. [22] employed vibration signals from normal bearings and bearings with three different fault locations. Bearing fault detection was based on hybrid ensemble detector and empirical mode decomposition. In [23],the stator current signals of induction motors are obtained using the MCSA method and the signals are then processed to produce a set of harmonic-based features for classification using the FMMÂÍCRFE model. Above literatures often uses current or vabration signals as the critical fault indicator, in our experiments, however, no obvious difference was observed in the early stages of ITSC e.g., 2% and 10%. This implies that simply analysing the current or vibration signals will not guarantee the performance of the fault detection. Furthermore, most of existing driven-driven approaches cannot generalize to unseen data, meaning that the testing data must have the same characteristics as the training data. However, as the operating conditions of IMs vary, it is not plausible to exhaust all of the working conditions during training. In addition, the framework trained on previous historical data should correctly predict the data generated after training, which may be affected by noise, resulting from the re-start of the machine, the variations of system dynamics etc.

In mitigating the above identified issues in monitoring ITSC faults, a diagnostic condition monitoring framework, based on an ensemble of data-driven techniques using electrical signals from the IM, is proposed as shown in Fig. 1 and explained in detail in Fig. 9 of Section IV. The framework in Fig. 1 begins with synchronous time series signals collected from the machines, acting as inputs. As mining unknown knowledge from data is one of the most important characteristics of data-driven methods, all of the signals were put together as inputs to proposed framework so that useful signal could be automatically found during training. In the data preparation stage, a sliding window mechanism is used, breaking down the time series data into short segments, before any preprocessing is done. Feature extraction/selection methods were then used to extract features and select the most informative ones for the following tasks. This is then followed by the classification step, where models were built based

on the selected features for fault detection. The purpose of multiple feature extraction/selection is to explore the sensitive faulty features, and the different types of classification strategies is to increase the diversity of single based models. With the increased diversity of the base learners, the ensemble performance is expected to be enhanced and the robustness of the fault detection can be guaranteed. From ensemble learning theory [24], it was argued that the use of multi-learner ensemble improves the performance as compared to single base learners, provided the base learners are accurate and diverse. Thus, the aim of the proposed framework is to develop an on-line monitoring system, capable of detecting ITSC faults in IMs, subjected to variable operating conditions. It it envisaged that the incorporation of these methods in an on-line monitoring setup will make the results sensitive to low severity faults, robust in handling data noise and operating conditions of the IMs and fast in detecting ITSC faults.

Experimental studies were conducted on the proposed framework, using healthy and faulty ITSC data, generated from a three-phase powered IM. The results from the studies showed that faulty condition can be distinguished from healthy ones even at low severity as the unseen faults can be detected under the different working conditions.

The rest of this paper is organized as follows. Section II presents the experimental setup and the description of the data collected from IMs. Section III presents the data preprocessing and feature extraction and selection methods. Section IV presents the classification techniques and proposed ensemble framework for the condition monitoring system. Finally, the results in Section V showed the effectiveness of the data-driven fault detection in IMs.

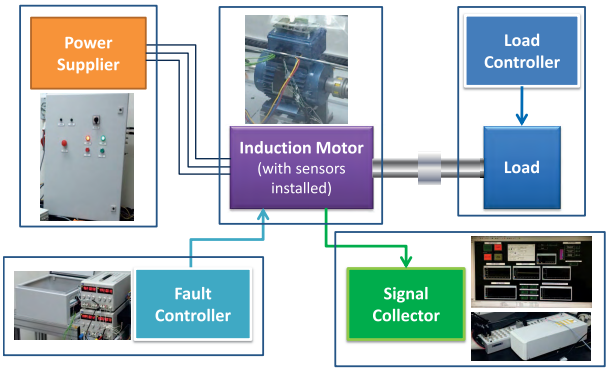


FIGURE 2. Description of the experimental test bed.

II. EXPERIMENTAL SETUP

A. TEST BENCH DESCRIPTION

Fig. 2 illustrates the set up of the experiment test bed conducted with the IMs. It mainly consists of four parts:

- Power supplier: controls the fundamental frequency of the current/voltage in order to vary the mechanical speed of the IM;
- Generator: controls the load applied to IM;
- ITSC controller: constructss various types of ITSC faults;
- Signal collector: collects various signals from the sensors.

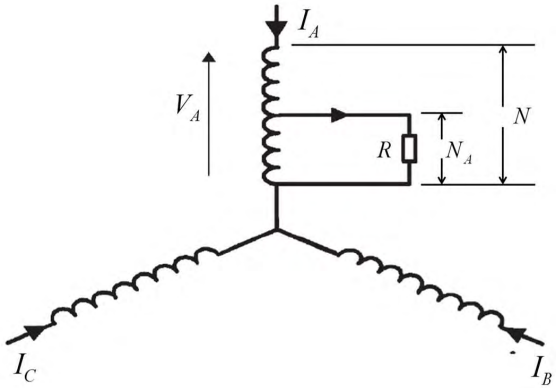


FIGURE 3. Configuration of the three-phase stator windings with ITSC fault.

Fig. 3 illustrates how an ITSC fault is simulated in the controller. The quantity $x_a = N_A/N$ represents the relative fraction of the fault in phase A, defined as the ratio between the shorted turns N_A and total turns N in each stator winding. Defined as short percentage (Short%), it represents the percentage of the stator windings that are short circuited in the test run. The higher the percentage, the more severe the fault is. In simulating the degradation of the fault from incipient to severe, the resistor R is attached to A stator coil. Different values of Short% and R represents the different levels of fault severities that the IM is subjected to. Based on domain knowledge, the fault severity will increase with the increase in Short% and/or decrease in R. This phenomenon is illustrated in Fig. 4 where it can be clearly seen that decreasing fault severities implies the increasing difficulties in detecting the faults.

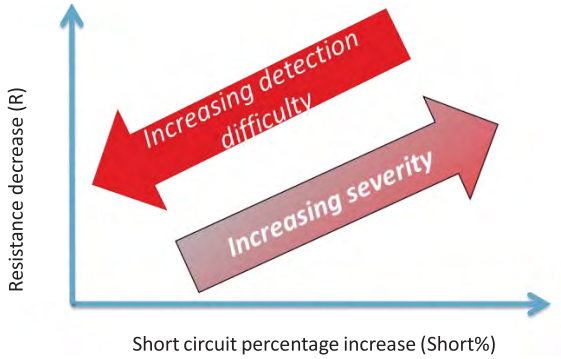


FIGURE 4. Fault condition and severity.

B. DATA DESCRIPTION

The aforementioned test bed is then used for experimental investigations. The condition of an IM with ITSC is expressed

by 2-tuple of (speed, load). The 'speed' values represent the nominal speed and corresponds to the actual fundamental frequency of the current/voltage of IM, controlled by the power supplier. The value of speed is varied from 600 rpm to 1400 rpm. The 'load' represents how much work the motor is outputting, with values ranging from 0 N·m and 5 N·m of the IM loading conditions. The sensors pick up data under the different working conditions from the IM, including: phase current from all three phase (I_A, I_B, I_C) and phase voltage from all three phase (V_A, V_B, V_C).

TABLE 1. Data description.

	Faulty data	Healthy data
Short%	2%, 5%, 8%, 10%	0%
Resistance (ohm)	50, 25, 12, 6, 3, 2, 0.8	Infinity
Collected signals	I_A, I_B, I_C, V_A, V_B, V_C	
Speed (rpm)	600, 800, 1200, 1400	
Load (N•m)	0, 5	

Table 1 shows the description of three phase currents and voltages collected from the experiments, with the Short% set to four different values. When the IM is healthy, ITSC is 0% and R is initially set to positive infinity to collect the healthy data. After approximately 300 seconds, ITSC is manually set with a certain value of Short% with the value of R gradually lowered, represent the degradation process. Other faulty data were also collected at the same time, i.e., from (0%, infinity) to (2%, 50 ohm) and then (2%, 0.8 ohm) and so on. The sensors collect various types of data at the sampling frequency of 5 kHz.

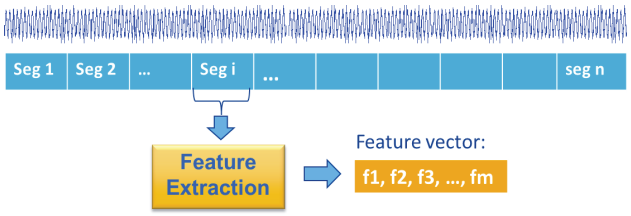


FIGURE 5. Illustration of raw signal segmentation for feature extraction.

III. FAULT INDICATOR

A. DATA PREPARATION

As discussed in Section I, in order to capture and identify specific trends or patterns in the data, a sliding window mechanism is used. As illustrated in Fig. 5 i.e. how a raw signal is segmented for the purpose of feature extraction, the mechanism breaks down the long time series data into short segments before any preprocessing is done, thus providing more insight into the machine behavior. Since the sensors acquire data at the frequency of 5 kHz, this implies that there are 5000 data points per second. In our experiments, the window size is set at 2500 data points, or 0.5 second. This value was chosen so as to achieve a good balance between efficiency and effectiveness. If the window size is too big, the time information will be lost as all the data within the window will be considered as a single instance, i.e. the time resolution of the data processing suffers. On the other hand, if the window size is too small, the extracted information from the signal may not be accurate as the window contains too few cycles, compromising the accuracy of the frequency spectrum analysis.

B. FEATURE EXTRACTION

To further extract useful information from current (I_A, I_B, I_C) and voltage (V_A, V_B, V_C) in diagnosing the ITSC fault, various feature extraction techniques were employed. Each technique will be used to process all the data from a single window into a set of feature value. By building new features from the original time series, feature extraction will reduce the massive time series data points into a manageable synoptic data structure whilst preserving most of the characteristics of the time series. In addition, the use of feature extraction provides an opportunity to incorporate domain knowledge into the data.

1) FFT

Fourier Transform (FT) is one of the most popular techniques used in analyzing time series and FFT (Fast Fourier Transform) is its fast implementation. Given a vector $(x_1, \ldots x_N)$, the time series is represented by its spectrum as

$$\mathrm{x}(k) = \sum_{j=1}^{N} x_j \omega_N^{(j-1)(k-1)} \tag{1}$$

where $\omega_N = e^{(-2\pi i)/N}$ is the Nth root of unity. After FT, the coefficients obtained are complex values which are not suitable for most of the existing classifiers. A following step to get real-valued features is often required. Existing means include calculating the real value, absolute value and image value, etc. Here, the absolute values of FT coefficients are used as the extracted features. Based on the corresponding frequency components of the features, two types of FFT-based feature extraction technique can be used:

- Basic FFT (FFT-B): FFT-B uses all frequency components as the extracted features. There are no pre-assumptions made on the input signal and therefore FFT-B can be applied to any type of time series.
- Harmonic FFT (FFT-H): FFT-H uses the harmonic frequency component as the extracted feature. An assumption made on the input signal is that there is a fundamental frequency in the signal and its harmonics play an important role in analyzing the signal. This is true for the phase currents and/or voltages in an IM and FFT-H can be used to pick out the most important frequency components. Another merit of FFT-H is that the features are independent of the fundamental frequency, i.e. when

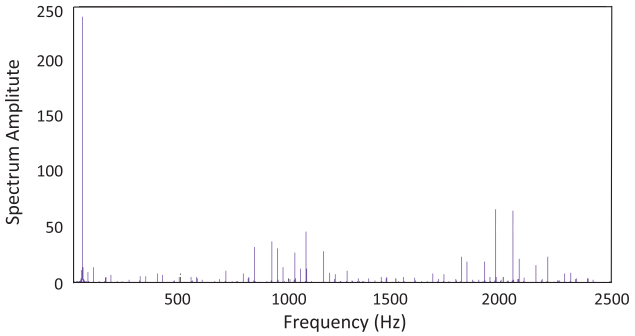


FIGURE 6. Amplitude of single-sided spectrum from V_A.

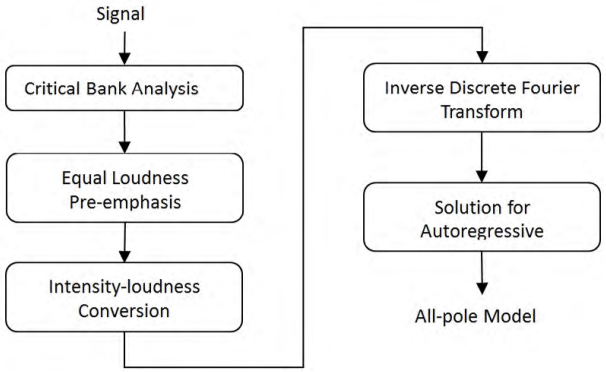


FIGURE 7. Procedure of perceptual linear prediction.

the fundamental frequency changes, features from FFT-H are in the same position and have the same physical meanings.

In essence, FFT-H uses the magnitudes of the harmonics as the extracted features. For illustrationi, Fig. 6 shows an example of the single-sided spectrum amplitude of a phase A voltage, V_A, under ÂąÂőspeed =1200 rpm and load = 0 N·mÂąÂŕ. The spectrum can be approximately divided into three ranges, namely low (0 to 500 Hz), medium (500 to 1500 Hz) and high (1500 to 2500 Hz) frequencies. The highest amplitude of the spectrum is at the fundamental frequency (40Hz) and based on domain knowledge, these first several harmonics (and half harmonics) carry important information about the signal. Due to the inverter generating the signals, it is observed that there also exist two symmetric centers in the single-side spectrum, at 1002.5 Hz and 2005 Hz respectively. In ensuring that as much key information in the extracted features are retained, the harmonics around these two centers were also used as features.

2) PERCEPTUAL LINEAR PREDICTION

Perceptual Linear Prediction (PLP) is a popular feature extraction technique in audio signal processing [25]. In PLP, the information of the human hearing system is analysed such that only the perceptually relevant details remain. During the processing, a critical band analysis is followed by equal-loudness pre-emphasis and intensity to loudness compression, which are done in frequency domain [26]. In this way, the warped frequency perception, the nonlinear and frequency-variant human loudness perception are all modeled. The signals are transformed back to the time domain after the above propcessing and regular linear prediction analysis is implemented. The block diagram of the feature extraction propcess is shown in Fig. 7.

3) WAVELET TRANSFORM

From its first introduction by Grossmann and Morlet [27] in the mid-1980s, wavelet transform has been widely applied in various applications such as pattern recognition, processing and synthesizing various signals (e.g., speech), image analysis, etc. The wavelet transform of a one-dimensional signal involves its decomposition over a basis obtained from a wavelet, possessing some specific properties, by dilations and translations. Each of the functions of this basis emphasizes both a specific spatial (temporal) frequency and its localization in physical space (time). Thus, the signal could be simultaneously analyzed in time and frequency spaces. Wavelet transform decomposes the input signal into approximation and detail space represented by wavelet coefficients in a series of sub-bands [28], and these coefficients can be used for feature extraction. In our experiments, the wavelet packet transform is used with the mother wavelet of 'db4'. After the decomposition, the following features are extracted: maximum, minimum, average power, mean, the standard deviation as well as the mean of the absolute values of the wavelet coefficients in each sub-band.

C. FEATURE SELECTION

In the feature extraction stage, many features will be generated, forming a summary of the original signals. However, not all of the extracted features carries useful information for classification. Feature selection is the following step , with the aim of selecting the most useful features for a specific task [29], [30]. In addition, it has been argued that many practical machine learning methods will degrade in performance when unnecessary features exist [31]. Thus, in most circumstances, feature selection seeks to find a subset of relevant or influential features from all original features under some evaluation criterion [29].

Since IM operates under varying conditions of changing speed and load, this makes the fault detection task even more challenging. Feature extraction and selection play key roles by transforming the original time series into suitable space and selecting the features which are invariant to the changing operation conditions of IM. Here, for the purpose of fault detection, Fisher's ratio is used as feature selection method. It calculates the ratio of squared inter-class divergence to intra-class spread of a feature x_j by:

$$\mathrm{FR}(x_j) = \frac{\left(m_{j(1)} - m_{j(2)}\right)^2}{\sigma^2_{j(1)} + \sigma^2_{j(2)}} \tag{2}$$

where $m_{j(c)}$ and $\sigma^2_{j(c)}$ are the instance mean and variance of feature x_j respectively, and $c = 1, 2$ represent the two classes. The large value of the FR indicates the strong discriminative power of a feature. As seen from above equation, Fisher's ratio evaluates the importance of the features independently, which makes it computationally simple and able to compare the importance of any two features.

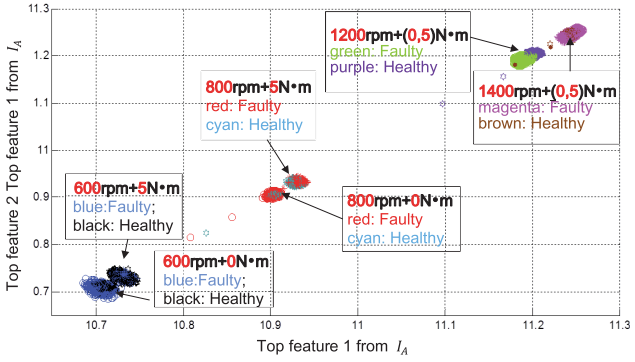


FIGURE 8. Scatter plot of data with FFT-H features.

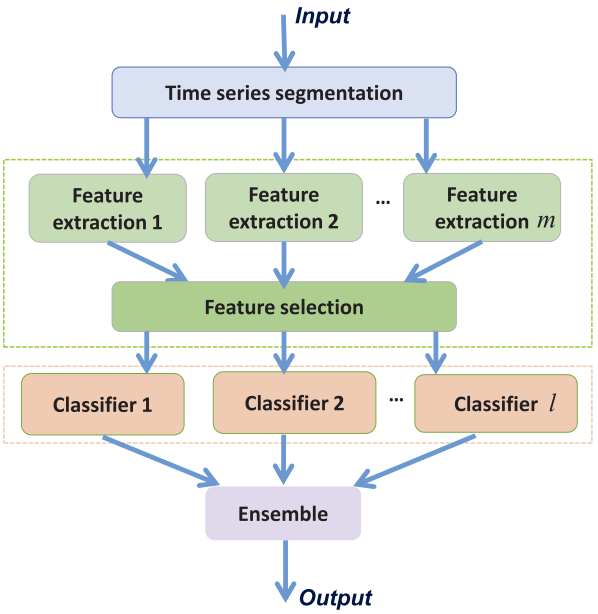


FIGURE 9. Framework for fault detection in ITSC.

As an example, in our experiments, the features are extracted and selected using FFT-H and Fisher's ratio respectively, based on healthy and faulty data under 1200 rpm. It can be clearly seen that the top 10 features are all from I_A. This result is expected as current is often used as the important health indicator for the IM. Based on the top two features selected above, the scatter plot of the available data under '[600, 800, 1200, 1400] rpm + [0, 5] N·m' with healthy and faulty instances is shown in Fig. 8. It can be clearly seen that the data under different speed are in different clusters. In addition, instances under different load and same speed may also form different clusters in the feature space, e.g., 600 rpm. The top two features are very important features under 1200 rpm in separating faulty from healthy instances. However, this clear separation does not appear for all the speeds considered. There is low consistency between the features under different speed settings.

IV. FAULT DIAGNOSE SYSTEM

In detecting ITSC faults, several challenges needs to be addressed: 1)

1) Detect ITSC in early stage: Although existing literature often uses current (I_A, I_B, I_C) signals as the critical fault indicator, in our experiments, however, no obvious difference was observed in the early stages of ITSC e.g., 2% and 10%. This implies that simply analysing the current signals will not guarantee the performance of the fault detection.
2) Generalize to unseen data: most of existing driven-driven approaches cannot generalize to unseen data, meaning that the testing data must have the same characteristics as the training data. However, as the operating conditions of IMs vary, it is not plausible to exhaust all of the Short% under various conditions during training. Thus, the framework should be generalized to unseen data e.g., train on 2%, 5%, test on 10%.
3) robust to noise: the framework trained on previous historical data should correctly predict the data generated after training, which may be affected by noise, resulting from the re-start of the machine, the variations of system dynamics etc.

In making use of the advantages of the different techniques in detecting ITSC faults in the early stages, a multi-feature extraction/selection multi-model ensemble framework is proposed, as shown in Fig. 9 and was briefly introduced in Section I. As mining unknown knowledge from data is one of the most important characteristics of data-driven methods, all of the signals were put together as inputs to proposed framework so that useful signal could be automatically found during training. The purpose of multiple feature extraction/selection as well as the different types of classification strategies is to increase the diversity of single based models. With the increased diversity of the base learners, the ensemble performance is expected to be enhanced. At the same time, as the complexity of the model grows, an ensemble tends to have a lower bias, which will reduce the variance due to re-sampling and re-weighting in the sample and feature spaces. Therefore, in terms of robustness and/or accuracy, an ensemble learning is usually better than individual learners.

The first step implementing the proposed condition monitoring framework is to determine the specific techniques used in each block. In our experiments, four feature extraction techniques were explored including the PLP, wavelet-based feature extraction, FFT-B and FFT-H as introduced in Section III-B. Having obtained a set of features, the key for condition monitoring of the IM is to build appropriate classifiers, which can then be applied to identify whether instances

are healthy or faulty. In practice, there are different types of classifiers namely linear, non-linear, statistical, kernel based, etc, and each has its own advantages and disadvantages. In out experiments, multiple classifiers were also explored for the purpose of ensemble, including NB, NN [32], [33], linear and nonlinear SVM [34]–[36], linear and nonlinear ELM [37], [38]. All these models need to be trained using training data prior to testing. In our experiments, offline training is employed for the model training process.

The step by step implementation of offline training is as follows:

1) The training data was prepared into healthy and faulty data set.
2) A sliding window mechanism was used to break down the healthy and faulty data set from a long time series into short segments before any preprocessing was undertaken as explained in Section III-A.
3) Each data segment then goes into the four feature extraction blocks and new features were built from the data from the segment by the four extraction methods independently. Specific features extracted from each method were introduced in Section III-B.
4) The extracted features from the four feature extraction methods were put together as inputs for feature selection. Fisher's ratio was used as the feature selection method to evaluate the importance of all the features based on (2). Only high ranked features were retained for classification. The number being kept were decided are decided by the criterion that the classification performance will not be much improved by incorporating more features as inputs to classifiers.
5) The selected features then acts as inputs to each classifier. Models were trained off-line and the parameters of each classifier were decided. The performance of the classifiers were then evaluated by using the definitions of TPR and TNR as introduced in Section V-A.
6) The individual outputs of each classifier were then combined at the ensemble stage using the majority voting approach.

During offline training, the parameters of each block in the framework were decided and models were saved for online testing. The parameters of the classifiers were given in Table 2 with Table 3 showing the four techniques and their extracted features as well as the number of extracted features from each input signals. For on-line testing, as long as a data segment is available as inputs, the processing in Fig. 9 will take place step-by-step and the ensuing results will displayed. Since online testing is based on the saved models during offline training and only one data segment is processed at a time, the detection of the ITSC faults were found to be very fast.were found to be very fast. Besides giving improved performance, the proposed framework was found to be easy to extend and modify, with the possibility of incorporating new data and/or new techniques in each stage.

TABLE 2. Classifier parameters.

Classifier	Parameters
Naïve Bayes	Priori: [0.5 0.5]
SVM	RBF kernel, C = 100, γ = 0.01
NN	Back-propagation in conjunction with Levenberg-Marquardt, number of hidden neurons = 100
ELM	RBF kernel, C = 0.5, γ = 0.01
LinearSVM	C = 100
LinearELM	C = 0.5, Activation function = 'sigmoid', number of hidden neurons = 1000

TABLE 3. Experimental setting for feature extraction.

Method	Extracted features	Number of features from each input
PLP	Auto-regression coefficients	11
Wavelet	The features as introduced in Section III	55
FFT-B	The top 100 frequency components	100
FFT-H	The 34 half-harmonics in the low frequency, the 33 half-harmonics around the first switch frequency (1002.5 Hz) and the 33 half-harmonics around the second switch frequency (2005 Hz)	100

V. EXPERIMENTAL VERIFICATION

A. PERFORMANCE EVALUATION

Classification performance evaluation is an important stage in the construction of a fault detection system as it acts as an index feeding back to the previous stages for improving the outputs from each stage or the whole system. It is also a way to reflect the ability of the classification system, helping to build confidence for users. For a binary classification problem, without the loss of generality, assuming healthy class is 'positive' and faulty class is 'negative', each classified instance will belong to one of the following four categories:

- True Positive (TP): Instance is predicted as 'positive' when it is actually positive;
- False Positive (FP): Instance is predicted as 'positive' when it is actually negative;
- True negative (TN): Instance is predicted as 'negative' when it is actually negative;
- False negative (FN): the instance is predicted to be 'negative' while it is actually positive.

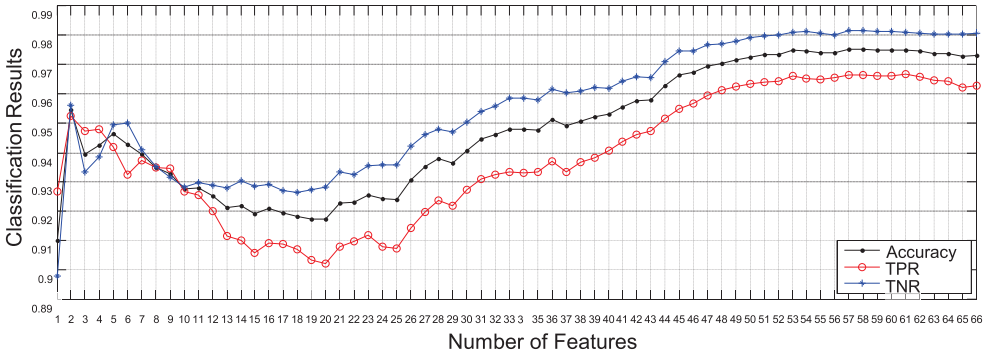


FIGURE 10. Classification results under different number of features by using PLP.

In this paper, the True Positive Rate (TPR) and True Negative Rate (TNR) are utilized as measures to evaluate the performance of the fault detection. TPR is the proportion of correctly predicted positive instances defined as:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3}$$

TNR measures the proportion of negative instances that are correctly identified defined as:

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{4}$$

TABLE 4. Performance evaluation methods.

Evaluation method	Description
10-fold Cross-Validation	Training and testing data are from same data set, splitting in a 10-fold cross-validation (10CV) way
Time-Generalization	Training data are generated in the past time while testing data are from future time, i.e. testing data will not appear before training data
Short%-Generalization	Training and testing data are from data under different Short% values

Table 4 lists the three evaluation methods that are used in the experiments to evaluate the performance of the classification. The 10CV is to randomly split the data into 10 equal folds and each fold is then selected as testing data and the rest as training data. For time series data, the performance evaluated by using 10CV method is used optimistic as data appeared in past and some future time would be used to make predictions for the data to appear in-between the past and future. However, classification model trained using previous historical data may not be able to correctly predict the data generated after training which may be affected by re-start of the machine, the variations of system dynamics etc. In order to verify whether the data variations can be handled, 'Time-Generalization' is used to ensure that the testing data will always appear in time after training. In this way, the impact of time sequence will also be investigated. In addition, most of existing driven-driven approaches is unable to generalize to unseen data, implying that the testing data must have the same characteristics as the training ones. However, it is often not plausible to exhaust all of the Short% during training. Therefore, the 'Short%-Generalization' is employed to test the generalization from a group of Short% values to another unseen Short% value, e.g., train on 2%, 5%, test on 10%.

IMs work in varying operating conditions, e.g. changing load or speed. Using the above-mentioned, the following sections will test: 1) if the framework can detect the ITSC in early stage,. e.g., 2% ITSC fault; 2) time-generalization ability (robustness); 3) short percentage generalization ability in multi-load and multi-speed scenarios. The analysing results will be shown in Sections V-B, V-C and V-D. The results will be summarized in Section V-F. Note that the following sections only show the representative results due to the space limitation, which are usually the worst case or the most difficult testing scenarios based on the data listed in Table 1.

B. MULTI-LOAD SCENARIO

Load can be connected or disconnected from the IMs, leading to operating point variations. It is clear from Fig. 8 that the instances under load and no load conditions may form different clusters even under the same speed. Whilst it is not plausible to validate against all possible conditions, the ability to generalize and detect the unseen behaviors is important. In the following part, the above issue is addressed by selecting the appropriate feature extraction methods, classifiers and their combinations as in the framework 9 for the framework shown in Fig 10.

1) 10-FOLD CROSS-VALIDATION UNDER MULTI-LOAD SCENARIO

One of the key task of the proposed framework is the ability to generalize unseen scenarios which assumes there exists an underlying set of rules relating the parts of the input signals or features that does not change when the condition of IM changes. This relationship, if captured by the classifiers, will enable the framework to function under a range of in a wide range of operating conditions. This investigation is thus focused on finding the investigate feature extraction method

that can identify signals with little variance at different working conditions across the different settings.

The previously discussed four feature extraction techniques, namely PLP, Wavelet-based feature extraction, FFT-B and FFT-H were selected and implemented. For comparisons of the performance between the different feature extraction methods, all available healthy (0% short circuit + Inf ohm) and faulty (2%,5%,8%,10%) data under each operation condition (speed, load) were used in the 10CV evaluation way to estimate the performance of TPR and TNR. NB was the chosen classifier due to its simplicity and potentially high generalization ability. The accuracy (ACC) is used as another metric to select the number of features according to the classification performance. ACC is the probability that the classifier classifies a randomly selected instance to the correct class, which is the proportion of the total number of predictions that are correct. It is determined using the equation:

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}}$$

Table 3 shows the results from the 4 feature extraction techniques and their corresponding number of extracted features from the experiments. The number of features are selected based on the classification performance. For example, the classification results of 'speed =1200 rpm and load = [0, 5] N·m' by using PLP and NB is plotted as an example in Fig. 10, which showed that the classification performance does not indicate any obvious change after 55 features. In our experiemnts, 11 features from total 66 features were selected from each input signals for classification.

TABLE 5. Performance of the four feature extraction techniques.

Operation conditions		PLP	FFT-B	FFT-H	Wavelet-based
Speed	Load	TPR/TNR	TPR/TNR	TPR/TNR	TPR/TNR
1400	0	.92/.94	**.50**/1.0	**.67**/1.0	**.75**/.99
1400	5	.85/.85	.80/.94	**.75**/.97	**.60**/.99
1400	0, 5	.85/.80	**.55**/.92	**.68**/.95	**.59**/.99
1200	0	.97/.99	.99/.99	1.0/1.0	1.0/1.0
1200	5	.99/.99	**.78**/.95	.99/.98	.98/.99
1200	0, 5	.96/.98	.84/.81	.96/1.0	.92/.98
800	0	**.75**/.95	**.52**/1.0	**.77**/1.0	**.60**/.95
800	5	.92/.86	.98/.95	.97/.99	**.65**/.99
800	0, 5	**.76**/.85	.99/**.50**	**.77**/.99	**.55**/.91
600	0	.85/.95	**.74**/1.0	.84/.99	.85/.92
600	5	.96/.98	.94/.98	.86/.98	.82/.99
600	0, 5	**.75**/.80	.80/1.0	.82/.80	**.76**/.81

Table 5 shows the averaged TPR and TNR for the four feature extraction techniques, with the performance values < 0.8 being highlighted. The observations from the table are as follows:

- PLP and FFT-H perform better than the other two methods, PLP with better acceptability;
- FFT-B performed the worst with some of the values very close to 0.5 (random guess);
- Wavelet-based feature extraction showed low performance in TPR with some of them close to 0.6.

In addition, when comparing the results from FFT-B and FFT-H, it is obvious that FFT-H performed better. This is reasonable and within expectation. The fundamental frequency will change with the load under the same speed. The true fundamental frequency will change over time even under the same speed and load. Therefore, FFT-H which chooses the features based on the actual fundamental frequencies will be more reliable than FFT-B in a fundamental frequency-dynamic environment. Based on the above analysis and comparison, it is concluded that PLP and FFT-H be the first choice and Wavelet-based feature extraction the second choice. FFT-B will not be suitable for fundamental frequency-changing signals.

2) TIME-GENERALIZATION (ROBUSTNESS) UNDER MULTI-LOAD SCENARIO

The instances in the data set appear in time order. When 10CV is used, the time order is randomized when generating the training and testing data, i.e., instances in training data may appear after testing instances which is not case in practice. Thus, the experiment here is designed to testify. From the results obtained in Section V-B.1, it was shown that FFT-H is a generally good feature extraction technique as the data under 'speed = 1400 rpm' has more overlaps between the faulty and the healthy data in the feature space. Thus, in the experiments, the data under 'speed = 1400 rpm + load = [0 5] N·m' was chosen, with FFT-H employed to extract features. In addition to the extracted features, the nominal fundamental and actual fundamental frequencies from fourier transform were also used as two additional features. This provided the ability to handle variations in the fundamental frequency, are mainly caused by the changes of load over time. Six classifiers are used in the experiments including NB, SVM, NN, ELM and the linear forms of SVM and ELM, namely, linearSVM and linearELM. In the experiments, the instances under each condition (load, Short%, resistance) were first separated into two parts based on their appearing time. Within the available data, the first part is used as training which includes instances appearing during the first half of data capture time, whilst the second part is used as testing which includes those appeared after the first part. In obtaining a robust estimation of the performance, in each round, 90% of the training instances were randomly selected to train a classifier which was then tested on the testing data. Table 6 shows the performance of the classifiers, with 'P' and 'N' being the number of positive and negative instances respectively.

The observations from the table are as follows:

- NB seems to be the worst classifier as compared to the other classifiers since it has the worst TPR;

- TNR values are shown to be generally good from the six classifiers;
- In essence, the it can be concluded that both nonlinear (SVM, NN, ELM) and linear classifiers (linearSVM linearELM) can be used for monitoring the working conditions;
- Although not indicated in the table, it should be noted that linearSVM is more computationally intensive than the rest. Specifically, in linearSVM requires several hundred seconds for training whilst the others needs less than 20 seconds.

TABLE 6. Performance of time-generalization under multi-load scenario.

Testing Data		TPR/TNR by different classifiers					
P	N	NB	SVM	NN	ELM	linearSVM	linearELM
1254	1536	**.64**/.999	.92/.99	.84/.97	.86/.99	.93/.97	.85/.95

TABLE 7. Four experimental scenarios for Short% generalization.

	Training Data (faulty)	Testing Data (faulty)
Case 1	[0, 5] N·m + [2%, 5%, 8%]	[0, 5] N·m + [10%]
Case 2	[0, 5] N·m + [2%, 5%, 10%]	[0, 5] N·m + [8%]
Case 3	[0, 5] N·m + [2%, 8%, 10%]	[0, 5] N·m + [5%]
Case 4	[0, 5] N·m + [5%, 8%, 10%]	[0, 5] N·m + [2%]

3) SHORT PERCENTAGE-GENERALIZATION UNDER MULTI-LOAD SCENARIO

The result in Table 6 showed the performance of fault detection using different classifiers. Here, the ability of the framework to detect ITSC faults in the early stages, based on the performance of different Short%, is evaluated. As it is not possible to test all of the Short% during training and testing due to heavy computational load, it is, therefore, important that the framework is incorporated with generalization capability. In test out whether a model trained on a group of Short% can pick out faults in another Short%, four scenarios, listed in Table 7, were designed , where one Short% is taken out for testing and the remaining for training. Similar to the previous experiments, data under 'speed = 1400 rpm + load = [0 5] N·m' were used and FFT-H was employed to extract features. All instances from the data set were first separated into two groups namely training and testing based on their appearing time.

Table 8 shows the ensuing performance including TPR and TNR for each case using the six different classifiers. From the table, the following can be observed:

- The testing performance from most classifiers is shown to be satisfactory in terms of TPR and TNR when Short% equals 5%, 8% and 10%. An exception is in case 2, where NN obtained very poor result. NB performed the worst with un-acceptable TPR (<50%) in cases 3 and 4;

TABLE 8. Performance of short%-generalization evaluation.

	Testing Data				TPR/TNR by different classifiers					
	Load	Short%	P	N	NB	SVM	NN	ELM	linearSVM	linearELM
Case 1	0	0% +10%	270	882	**.66**/.99	.96/.97	.91/.91	.91/.97	.98/.94	.92/.91
	5	0% +10%	283	654	1.0/1.0	1.0/.99	1.0/.99	1.0/1.0	1.0/.99	.89/.91
Case 2	0	0% +8%	361	882	.78/.99	.90/.97	.97/**.44**	.91/.96	.92/.95	.80/.90
	5	0% +8%	311	654	1.0/1.0	1.0/.99	.95/**.50**	1.0/1.0	1.0/.99	.82/.95
Case 3	0	0% +5%	135	882	**.44**/.99	.95/.98	.84/92	.88/.98	.96/.95	.87/.93
	5	0% +5%	154	654	**.46**/1.0	.99/.99	.96/.99	.98/1.0	.99/.99	.92/.97
Case 4	0	0% +2%	168	882	**.13**/.99	.76/.99	**.66**/.72	**.54**/.99	.77/.97	**.57**/.97
	5	0% +2%	162	654	**.17**/1.0	**.41**/1.0	**.65**/.74	**.46**/1.0	**.34**/1.0	**.46**/.98

- Separating 2% faults from healthy data is found to be very difficult as indicated from the TPR values. This is reasonable since 2% faults is in the very early stages of ITSC faults and it belongs to low severity fault.
- Training on lower severity fault and testing on higher severity fault is possible, providing a mean to detect ITSC faults in very early stages.
- The two linear classifiers, linearSVM and linearELM, showed similar performance as compared to the nonlinear classifiers, therefore, only nonlinear classifiers were considered in the following sections.

C. MULTI-SPEED SCENARIO

In this section, the affect of the detecting performance on changing speed, under multi-load multi-speed scenarios, is evaluated. From Fig. 8, it is clearly seen that the data under different speed form different clusters, which makes it difficult to find features that are speed-invariant for the classifiers to distinguish faulty from healthy instances. It is more complicated scenarios than those in Section V-B.

1) TIME-GENERALIZATION (ROBUSTNESS) UNDER MULTI-SPEED SCENARIO

(1200, 1400) rpm of the mixed data gives an appropriate coverage of the speed range for testing the multi-speed scenarios. In addition, it is observed from the result in Table 5 that the healthy and faulty data at 1200 rpm data speed can be separated easily in two dimensional feature space while those of 1400 rpm overlap heavily, making it difficult for the classification. In our simulations, the first half of the data from both 1200 rpm and 1400 rpm were used as training data and second half as testing. The same data were also tested using 10-fold CV and 4 classifiers were also explored, namely, SVM, NN, ELM and NB. As 10-fold CV gives an optimistic estimation of the performance to time series and NB continues to perform badly, NB will not be considered in the following sections.

TABLE 9. Performance of time-generalization under multi-speed scenario.

Testing Data			TPR/TNR by different classifiers			Ensemble
Speed	P	N	SVM	NN	ELM	
1200	1090	1531	1.0/1.0	**.75**/.999	.998/1.0	.998/1.0
1400	1254	1536	.89/.96	**.63**/.98	.83/.99	.84/.99

Comparing Table 9 under multi-load multi-speed scenario with Table 6 under multi-load single-speed scenario, the following were observed:

- All classifiers performed better on 1200 rpm data compared to 1400 rpm data, which is in agreement with the classification results observed in Section V-B.1;
- Compared with Table 6 under the multi-load single-speed scenario, the performance is decreased when training and testing with multi-speed. However, the performance was found to still be satisfactory. This implies that the framework can be applied on changing speed and load conditions with the selected feature and classifiers;
- SVM and ELM perform well and NN seems to be the worst among the three classifiers.

TABLE 10. Four experimental scenarios for short% generalization.

	Training Data (faulty)	Testing Data (faulty)
Case 1	[1200, 1400] rpm + [0, 5] N·m + [2%, 5%, 8%]	[1200, 1400] rpm + [0, 5] N·m + [10%]
Case 2	[1200, 1400] rpm + [0, 5] N·m + [2%, 5%, 10%]	[1200, 1400] rpm + [0, 5] N·m + [8%]
Case 3	[1200, 1400] rpm + [0, 5] N·m + [2%, 8%, 10%]	[1200, 1400] rpm + [0, 5] N·m + [5%]
Case 4	[1200, 1400] rpm + [0, 5] N·m + [5%, 8%, 10%]	[1200, 1400] rpm + [0, 5] N·m + [2%]

2) SHORT%-GENERALIZATION UNDER MULTI-SPEED SCENARIO

The experiments conducted here are similar to those in Section V-B.3, with the difference being the data from two speed values at 1200 rpm and 1400 rpm were also included in both training and testing. Table 10 lists the four cases of the experimental scenarios.

The corresponding results from the 4 cases are shown in Table 11, inclusive of the overall and detailed performances. From the results, the following were observed:

- The performance of NN is not stable, as it can detect 5% in single speed scenario but unable to do in multi-speed scenarios. The performance of SVM and ELM decreased as compared with the results in Table 11, however, the results in classifying the different Short% is acceptable with the lowest values of TPR and TNR are both greater than 0.8 in most of the cases;

TABLE 11. Performance of short%-generalization under multi-speed scenario.

	Testing Data				TPR/TNR by different classifiers				Ensemble
	Speed	Short %	P	N	SVM	NN	ELM	linearELM	
Case 1	1200	0% + 10%	573	1531	.83/.999	**.61**/.999	.80/.999	.80/.98	.81/.999
	1400	0% + 10%	553	1536	.98/.90	.95/.92	.96/.93	.90/.87	.97/.92
Case 2	1200	0% + 8%	535	1531	.96/.1.0	.95/.**25**	.99/1.0	.88/.99	.98/1.0
	1400	0% + 8%	672	1536	.92/.89	.99/.**23**	.98/.90	.97/.84	.97/.88
Case 3	1200	0% + 5%	541	1531	.997/1.0	.99/.**50**	.996/1.0	.91/.98	.997/.999
	1400	0% + 5%	660	1536	.92/.94	.94/.**48**	.89/.96	**.76**/.91	.91/.95
Case 4	1200	0% + 2%	536	1531	.99/1.0	.99/.998	.99/1.0	.88/.99	.99/1.0
	1400	0% + 2%	630	1536	**.47**/.98	**.48**/.97	**.43**/.99	**.39**/.96	**.44**/.99

- It is easier to detect ITSC fault under no-load scenario for multi-speed case as compared with having load, i.e. 5 N·m in the experiments, which is not obvious in single-speed cases.
- The performance of detecting 2% ITSC is relatively good when the speed is 1200 rpm, but the detection failed at 1400 rpm. This is because there is much overlap between the healthy data and 2% faulty data of 1400 rpm.
- Training on lower severity fault and testing on higher severity fault is possible in multi-speed cases, which provides a mean to detect the ITSC faults in very early stage.

D. ENSEMBLE FOR FAULT DETECTION

Ensemble is expected to achieve improved performance under the condition that the multiple 'weak' classification models are very different in predicting testing instances. Here 'weak' means that a classifier's performance is low but should be higher than random guesses (i.e. TPR and TNR > 0.5). For consideration of simplicity, majority voting is used as the ensemble method. In all experiments, we never see a case where all classifiers produce good results. Each time some fail to classify acceptably. In practice, the condition for improved ensemble is hard to be fulfilled and ensemble generally obtains an intermediate performance of the multiple classifiers. This is also the case in our experiments. See for example in Table 9 under speed $=$ 1400 rpm, the highest and lowest TPR are 0.89 (by SVM) and 0.63 (by NN), while the ensemble TPR of the three classifiers is 0.84. Similar phenomenon can be observed from the TNR values in this case. Even though ensemble does not guarantee improvement in classification performance compared to each base classifier, it can generally be expected that the robustness of the classification can be improved by producing the ensemble results.

Based on their performance as shown and discussed in the above sections, the three nonlinear classifiers SVM, NN

and ELM are selected as the base classifiers for ensemble. The experiments are done in the same scenarios as in the Section V-C by the evaluation of Time-Generalization and Short%-Generalization. Ensemble results of the two scenarios are shown in last columns of Table 9 and Table 11.

TABLE 12. Comparison of performance of time-generalization under multi-speed scenario.

Testing Data			Ensemble	FFT-H+DAE+SVM
Speed	P	N		
1200	1090	1531	.998/1.0	.991/.952
1400	1254	1536	.84/.99	.76/.99

E. COMPARE WITH DEEP NEURAL NETWORK BASED FAULT DETECTION

Deep neural networks are very efficient in complex nonlinear system modeling and have recently gained much success on various applications, such as classification, speech recognation, etc. Deep autoencoder (DAE) is one type of deep neural network that can automaticly learn the concise representations of the big data in either unsuperivsed or supervised way. In this work, we compare our proposed methods with DAE based fault detection. The DAE is used to extracted features from the data and SVM is used as classifiers. We tried to used the normalized raw data ($I_A, I_B, I_C, V_A, V_B, V_C$) for fault detection, but the results are not acceptable with unbalanced TPR and TNR. So we transformed the raw data into frequency domain by FFT-H, then the DAE is performed for feature extraction followed by SVM as classifer. The DAE has 4 layers, and each layer is composed of 1200-500-200-1200 nodes. After feature extraction, the outputs of the third layer are used as the inputs for SVM to train the classifier. The parameters settings of DEA are {activation function: sigmoid; learning rate:0.0005; momentum: 0.5; regularization: 0.0001; iteration 2000}. The parameters settings of SVM are {RBF kernel, $C = 100, \gamma = 0.2$}. We compare the DAE based fault detection with our methods for the two scenarios in Tables 9 and 11 which also transform the raw data with FFT-H as inputs. The comparison results are shown in Table 12 and 13 respectively. It can be observed that the performance of our proposed method is better that DAE based method in most cases.

F. DISCUSSION

The proposed framework evaluates the large amounts of data without disturbing the normal operation of the IM in real-time. It is not restricted to the domain knowledge and experience but provide the possibility to mining the unknown information from the data by simply testing the different combinations of inputs, features and classifiers which is too complicated to build the mathematical models.

Most of the existing driven-driven approaches are evaluated by 10CV, which does not consider the system dynamic variation with time. In addition, detecting ITSC of IM working under multi-load multi-speed scenarios is not well studied. To solve the above problems, multiple scenarios have been designed and tested based on the designed test bed as shown in Figure 2. Using the selected features in Table 3 and trained models with the parameters settings in Table 2, the condition monitoring system can work in a wide range of operating conditions as long as the operating condition (speed, load) of the testing data is included in the training stage. In addition, the good performance in Table 5 and 9 indicates that the proposed framework is robust to the system noise and small system dynamic variations resulted from re-start or running of the IM, etc. Table 8 and 11 test the generalization from one group of Short% to another different Short%. This means that a model trained on the training data does not have all knowledge about testing data and needs to depend on its generalization ability for prediction. The good performance of implementations implies that the framework has the ability to predict data from an unknown Short%. The is promising since it is not possible to exhaust all the Short% during training, we can always train the framework with low severity fault to guarantee the testing of higher severity fault.

TABLE 13. Comparison of performance of short%-generalization under multi-speed scenario.

	Testing Data				Ensemble	FFT-H+DAE+SVM
	Speed	Short%	P	N		
Case 1	1200	0%+ 10%	573	1531	.81/1.0	.82/.98
	1400	0% + 10%	553	1536	.97/.92	.76/.99
Case 2	1200	0% + 8%	535	1531	.98/1.0	.95/1.0
	1400	0% + 8%	672	1536	.97/.88	.66/.99
Case 3	1200	0% + 5%	541	1531	1.0/1.0	.90/.95
	1400	0% + 5%	660	1536	.91/.95	.96/.74
Case 4	1200	0% + 2%	536	1531	.99/1.0	.99/.97
	1400	0% + 2%	630	1536	.44/.99	.94/.41

VI. CONCLUSION

In this paper, we have designed and described a condition monitoring framework with multi-feature extraction/selection multi-model ensemble. In addition to 10-fold Cross-Validation which is used as a general method in data mining to evaluate the testing performance, two other evaluations were also carried out for assessing the model generalization abilities including the generalization of predicting new time series appearing in a future time (Time generalization), predicting new time series of a different short circuit percentage (Short% generalization). Experimental studies based on the proposed framework were conducted on real data which includes healthy data and the ITSC faulty data generated from a three-phase powered IM. With the selected inputs and techniques in the framework, the results from the empirical studies showed that it is possible to distinguish the faulty condition from the healthy ones in early stage and can also detect the unseen fault under different working conditions.

ACKNOWLEDGEMENT

The authors would like to acknowledge the support received from the Rolls-Royce Singapore Advanced Technology Centre (ATC) for the work done under the Computational Engineering Lab (CEL).

REFERENCES

[1] P. K. Dagadkar and C. V. Honade, "Monitoring of power transformer incipient fault," *Int. J. Innov. Res. Sci. Technol.*, vol. 2, no. 3, pp. 187–189, 2015.

[2] H. V. Padullaparti, P. Chirapongsananurak, M. E. Hernandez, and S. Santoso, "Analytical approach to estimate feeder accommodation limits based on protection criteria," *IEEE Access*, vol. 4, pp. 4066–4081, 2016.

[3] J.-H. Jung, J.-J. Lee, and B.-H. Kwon, "Online diagnosis of induction motors using MCSA," *IEEE Trans. Ind. Electron.*, vol. 53, no. 6, pp. 1842–1852, Dec. 2006.

[4] A. Khlaief, M. Boussak, and M. Gossa, "Open phase faults detection in pmsm drives based on current signature analysis," in *Proc. 19th Int. Conf. Elect. Mach. (ICEM)*, Sep. 2010, pp. 1–6.

[5] S. M. A. Cruz and A. J. M. Cardoso, "Stator winding fault diagnosis in three-phase synchronous and asynchronous motors, by the extended Park's vector approach," *IEEE Trans. Ind. Appl.*, vol. 37, no. 5, pp. 1227–1233, Sep. 2001.

[6] J. L. Kohler, J. Sottile, and F. C. Trutt, "Condition monitoring of stator windings in induction motors. I. Experimental investigation of the effective negative-sequence impedance detector," *IEEE Trans. Ind. Appl.*, vol. 38, no. 5, pp. 1447–1453, Sep. 2002.

[7] H. Henao, C. Demian, and G.-A. Capolino, "A frequency-domain detection of stator winding faults in induction machines using an external flux sensor," *IEEE Trans. Ind. Electron.*, vol. 39, no. 5, pp. 1272–1279, Sep./Oct. 2003.

[8] M. B. K. Bouzid and G. Champenois, "New expressions of symmetrical components of the induction motor under stator faults," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 4093–4102, Sep. 2013.

[9] J. Cusidó, L. Romeral, J. A. Ortega, J. A. Rosero, and A. G. Espinosa, "Fault detection in induction machines using power spectral density in wavelet decomposition," *IEEE Trans. Ind. Electron.*, vol. 55, no. 2, pp. 633–643, Feb. 2008.

[10] B. Aubert, J. Régnier, S. Caux, and D. Alejo, "Kalman-filter-based indicator for online interturn short circuits detection in permanent-magnet synchronous generators," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1921–1930, Mar. 2015.

[11] F. Duan and R. Živanović, "Condition monitoring of an induction motor stator windings via global optimization based on the hyperbolic cross points," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1826–1834, Mar. 2015.

[12] C. H. De Angelo, G. R. Bossio, S. J. Giaccone, M. I. Valla, J. A. Solsona, and G. O. García, "Online model-based stator-fault detection and identification in induction motors," *IEEE Trans. Ind. Electron.*, vol. 56, no. 11, pp. 4671–4680, Nov. 2009.

[13] R. Isermann, "Model-based fault-detection and diagnosis—Status and applications," *Annu. Rev. Control*, vol. 29, no. 1, pp. 71–85, 2005.

[14] S. Bachir, S. Tnani, J. C. Trigeassou, and G. Champenois, "Diagnosis by parameter estimation of stator and rotor faults occurring in induction machines," *IEEE Trans. Ind. Electron.*, vol. 53, no. 3, pp. 963–973, Jun. 2006.

[15] C. Li *et al.*, "Bearing fault diagnosis using fully-connected winner-take-all autoencoder," *IEEE Access*, to be published.

[16] S. Yin, S. X. Ding, X. Xie, and H. Luo, "A review on basic data-driven approaches for industrial process monitoring," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6418–6428, Nov. 2014.

[17] S. Toma, L. Capocchi, and G.-A. Capolino, "Wound-rotor induction generator inter-turn short-circuits diagnosis using a new digital neural network," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 4043–4052, Sep. 2013.

[18] B. Yao, P. Zhen, L. Wu, and Y. Guan, "Rolling element bearing fault diagnosis using improved manifold learning," *IEEE Access*, vol. 5, pp. 6027–6035, 2017.

[19] M. B. K. Bouzid, G. Champenois, N. M. Bellaaj, L. Signac, and K. Jelassi, "An effective neural approach for the automatic location of stator interturn faults in induction motor," *IEEE Trans. Ind. Electron.*, vol. 55, no. 12, pp. 4277–4289, Dec. 2008.

[20] A. Siddique, G. S. Yadava, and B. Singh, "Applications of artificial intelligence techniques for induction machine stator fault diagnostics: Review," in *Proc. 4th IEEE Int. Symp. Diagnostics Elect. Mach., Power Electron. Drives (SDEMPED)*, Aug. 2003, pp. 29–34.

[21] W. He, Y. Zi, B. Chen, F. Wu, and Z. He, "Automatic fault feature extraction of mechanical anomaly on induction motor bearing using ensemble super-wavelet transform," *Mech. Syst. Signal Process.*, vols. 54–55, pp. 457–480, Mar. 2015.

[22] G. Georgoulas, T. Loutas, C. D. Stylios, and V. Kostopoulos, "Bearing fault detection based on hybrid ensemble detector and empirical mode decomposition," *Mech. Syst. Signal Process.*, vol. 41, nos. 1–2, pp. 510–525, 2013.

[23] M. Seera, C. P. Lim, S. Nahavandi, and C. K. Loo, "Condition monitoring of induction motors: A review and an application of an ensemble of hybrid intelligent models," *Expert Syst. Appl.*, vol. 41, no. 10, pp. 4891–4903, 2014.

[24] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*. Berlin, Germany: Springer, 2000, pp. 1–15.

[25] G. Zweig and M. Picheny, "Advances in large vocabulary continuous speech recognition," *Adv. Comput.*, vol. 60, pp. 249–291, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0065245803600124

[26] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *J. Acoust. Soc. Amer.*, vol. 87, no. 4, pp. 1738–1752, 1990.

[27] A. Grossmann and J. Morlet, "Decomposition of hardy functions into square integrable wavelets of constant shape," *SIAM J. Math. Anal.*, vol. 15, no. 4, pp. 723–736, 1984.

[28] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989.

[29] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Jan. 2003.

[30] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[31] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, 1997.

[32] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Netw.*, vol. 4, no. 2, pp. 251–257, 1991.

[33] F. Scarselli and A. C. Tsoi, "Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results," *Neural Netw.*, vol. 11, no. 1, pp. 15–37, 1998.

[34] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[35] R. G. Brereton and G. R. Lloyd, "Support vector machines for classification and regression," *Analyst*, vol. 135, no. 2, pp. 230–267, 2010.

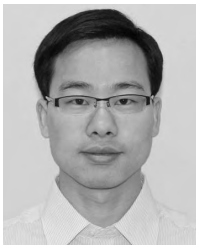
[36] C.-W. Hsu *et al.* (2010) *A Practical Guide to Support Vector Classification*. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

[37] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, 2006.

[38] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.

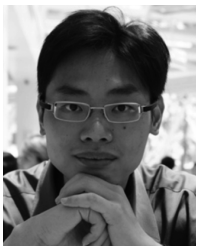
ZHAO XU received the B.Eng. and M.Eng. degrees in automatic control from Northwestern Polytechnical University, Xi'an, China, in 2005 and 2008, respectively, and the Ph.D. degree in control and instrumentation from Nanyang Technological University, Singapore, in 2013. She was a Research Scientist with the Institute of High Performance Computing from 2012 to 2014. Since 2015, she has been an Assistant Professor with the School of Electronics and Information, Northwestern Polytechnical University. Her research interests include control, data-based diagnosis, and prognosis.

CHANGHUA HU was a Visiting Scholar with the University of Duisburg in 2008. He is currently a Professor with the High-Tech Institute, Xi'an, China. He has published two books, and about 100 articles. His research interests include fault diagnosis and prediction, life prognosis, and fault tolerant control. He received the Changjiang Scholar by the Chinese Ministry of Education in 2013.



FENG YANG received the B.Eng. degree in information engineering and the M.Eng. degree in control engineering from Xi'an Jiaotong University, Xi'an, China, and the Ph.D. degree in machine learning and bioinformatics from Nanyang Technological University (NTU), Singapore, in 2012. From 2011 to 2012, he was a Research Associate with NTU. In 2012, he joined the Institute of High Performance Computing, Agency for Science, Technology, and Research, Singapore, where he is currently a Scientist with the Department of Computing Science. His research interests include machine learning, signal processing, information retrieval, big data, and data-based diagnosis and prognosis.

SHYH-HAO KUO is involved in parallel and concurrent programming languages, and software refactoring and algorithm transformations software engineering methods for concurrent programming optimization algorithms.



CHI-KEONG GOH (SM'14) received the B.Eng. and Ph.D. degrees in electrical engineering from the National University of Singapore, Singapore, in 2003 and 2007, respectively.

He is currently the Team Lead in data analytics and optimization with the Rolls-Royce Advanced Technology Centre, Singapore. His current research interests include evolutionary computation and data analytics and their application.

Dr. Goh has served as a reviewer for various international journals, such as the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, PART B: CYBERNETICS, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, PART C: APPLICATIONS AND REVIEWS, and *Neurocomputing*.

AMIT GUPTA (SM'09) received the bachelor's degree in electrical engineering from IIT Roorkee and the Ph.D. degree in electrical engineering from the National University of Singapore. From 2000 to 2012, he was with Bechtel Corporation, Samsung Heavy Industries, Delphi Automotive Systems, and Vestas Wind Systems. Since 2012, he has been the Chief of the Rolls-Royce Electrical, Rolls-Royce Singapore Pte. Ltd. He is currently the Director of the Electrical Programme at the Rolls-Royce@NTU Corporate Laboratory. He is fellow of the IET and a Chartered Engineer from the Engineering Council U.K. He is an Associate Editor of the IEEE TRANSITIONS ON POWER ELECTRONICS and plays an active role in organizing electrical power engineering conferences in Asia.

SIVAKUMAR NADARAJAN (S'12) received the bachelor's degree in electrical and electronics engineering and the master's degree in electrical drives and controls from the Pondicherry Engineering College, Pondicherry University, Puducherry, India, in 2002 and 2004, respectively. He is currently pursuing the Ph.D. degree with the National University of Singapore, Singapore, where he has been involved in modeling, design, and demonstrating condition monitoring techniques for brushless synchronous generators.

From 2004 to 2009, he was a Lecturer with SVCE, Sriperumbudur, India, a Research Engineer with M S Elevators (Toshiba joint venture), Malaysia, and a Software Engineer with Delphi Automotive Systems, Singapore. Since 2009, he has been with Rolls-Royce Singapore Pte. Ltd, Singapore. His research interests include electric drives, power electronic converters, condition monitoring, and modeling of electrical machines.