Ensemble Based Real-Time Adaptive Classification System for Intelligent Sensing Machine Diagnostics

Minh Nhut Nguyen, Chunyu Bao, Kar Leong Tew, Sintiani Dewi Teddy, *Member, IEEE*, and Xiao-Li Li, *Member, IEEE*

***Abstract*—The deployment of a sensor node to manage a group of sensors and collate their readings for system health monitoring is gaining popularity within the manufacturing industry. Such a sensor node is able to perform real-time configurations of the individual sensors that are attached to it. Sensors are capable of acquiring data at different sampling frequencies based on the sensing requirements. The different sampling rates affect power consumption, sensor lifespan, and the resultant network bandwidth usage due to the data transfer incurred. These settings also have an immediate impact on the accuracy of the diagnostics and prognostics models that are employed for system health monitoring. In this paper, we propose a novel adaptive classification system architecture for system health monitoring that is well suited to accommodate and take advantage of the variable sampling rate of sensors. As such, our proposed system is able to yield a more effective health monitoring system by reducing the power consumption of the sensors, extending the sensors' lifespan, as well as reducing the resultant network traffic and data logging requirements. We also propose an ensemble based learning method to integrate multiple existing classifiers with different feature representations, which can achieve significantly better, stable results compared with the individual state-of-the-art techniques, especially in the scenario when we have very limited training data. This result is extremely important in many real-world applications because it is often impractical, if not impossible, to hand-label large amounts of training data.**

***Index Terms*—Adaptive classifiers, classifiers, data driven diagnostics and prognostics, ensemble learning, sensor data classification.**

Acronyms and Abbreviations

ACS	adaptive classification system
CV	cross-validations
FFT	fast Fourier transform
KNN	k-nearest neighbors
KNN_f	KNN for frequency domain
KNN_t	KNN for time domain
LIBSVM	library for support vector machines
MFS	Machinery Fault Simulator
MLP	multilayer perceptron
MLP_f	MLP for frequency domain
MLP_t	MLP for time domain
RMS	root mean squared
SVM	support vector machine
SVM_f	SVM for frequency domain
SVM_t	SVM for time domain
WSN	wireless sensor network

Notation

C	number of models
d_c^s	binary value: 1 if model c outputs machine state s, and 0 otherwise
F	F-measure, or the harmonic mean of precision and recall
K	kurtosis value
p_c^s	estimated accuracy of model c performed on machine state s
PV	peak value
RMS	root-mean-squared value
S	number of machine states
s_{argmax}	final machine state prediction
$StdDev$	standard deviation
w_c^s	confidence score of model c performed on machine state s

Manuscript received October 27, 2011; revised December 31, 2011; accepted January 26, 2012. Date of publication May 11, 2012; date of current version May 28, 2012. Associate Editor: Q. Miao.

M. N. Nguyen, C. Bao, S. D. Teddy, and X.-L. Li are with the Department of Data Mining, Institute for Infocomm Research, Agency for Science, Technology and Research (A *STAR), Singapore 138632, Singapore (e-mail: mnnguyen@i2r.a-star.edu.sg; cbao@i2r.a-star.edu.sg; sdteddy@i2r.a-star.edu.sg; xlli@i2r.a-star.edu.sg).

K. L. Tew is currently with SAP Research Centre, Singapore (e-mail: kar.leong.tew@sap.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TR.2012.2194352

I. Introduction

Advancements in wireless sensor network (WSN) technology have allowed a wider adoption and application of sensory networks. Currently, we are harnessing the capability of such WSN configurations to collect invaluable monitoring data both in the indoor as well as outdoor environments.

The indoor environment in this case refers to locations such as a manufacturing shop floor [1]–[3], a home for the elderly [4], [5], and healthcare environments [6], [7]. In such environments, sensors are deployed, and the readings collected by the sensors serve as the basis for studying and analyzing the current health of the system or subject. One good example of such a monitoring scenario would be the large-scale deployment of sensors in the manufacturing shop floor. The data collected in this case contain signals and important information about the machine health status and operating conditions. On the contrary, outdoor monitoring systems frequently involve the collection of outdoor environment variables like atmospheric readings. Two examples are the detection of rainfall-induced landslides [8], and avalanche forecasting via the continuous monitoring of the snow coverage [9].

The data collected from the sensors provide an entry point for the prognostics and health management community and data mining community to perform study and in-depth analysis to obtain insights into a particular process. The readings produced by the sensors provide a wealth of information for knowledge extraction, modeling, analysis, and intelligent prediction as well as decision making. There has been a substantial amount of research conducted in diagnostics and prognostics based on sensor data [1]–[7], [10]–[14]. In most of these studies, sensors are usually set to sample their readings at a predefined sampling rate. It is important to note that having a constant sampling rate is a prerequisite for most data modeling and analysis methods as it ensures time consistency and similar data distributions in the collected samples.

Our previous research into machine fault diagnostics for the manufacturing shop floor has led us to the conclusion that a higher sampling rate would usually yield better diagnostics performance [1]. However, in practice, when the machines are running under a healthy state, it is not necessary to collect the sensor readings at such a high sampling rate. More often than not, higher sampling rates are required only when anomalies or faults occur during the machine's operation, to provide the diagnostics system with enough information for fault isolation, analysis, and identification.

Moreover, a higher sampling rate would consequently result in higher energy consumption by the sensors. Direct correlation between the sampling rate and the energy consumption of sensors has been well established [15]–[18]. In any WSN architecture (whether indoors or outdoors), sensors are typically powered through batteries. This approach in turn constrains the sensor lifespan and capability. It has actually been reported that the energy constraint is the main factor preventing the full exploitation of WSN technology [19].

One of the directions of WSN research is thus to improve the power efficiency of the sensors. One common approach is to allow the sampling frequencies to be adaptive as different levels of granularity of information are required in different situations [8], [9], [16], [20]. Exploring the use of such an adaptive sampling rate is a logical approach as it reduces the power consumption of the sensors, and thereby extends the lifespan and usability period of the sensors. On top of this, it also means that only the necessary amount of data are collected, stored, and transmitted, thus freeing up network bandwidth.

On the other hand, such an adaptive sampling rate generates a new set of challenges for existing data modeling and analysis methods. Existing models largely work on a consistent data sampling rate. With an adaptive sampling rate, the models might work under certain scenarios and assumptions, but their performance will be unpredictable. In this paper, we introduce a novel Adaptive Classification System (ACS) architecture that is designed to cater for as well as actively adapt the sampling frequencies of the sensors, while maintaining the fault identification performance of the health monitoring system.

In summary, our ACS is constructed as follows. First, a set of pre-determined sampling rates, low and high, is identified, and models are trained on data collected based on these predefined sensor sampling frequencies. Next, a set of controls is set-up to screen the monitoring results of the models based on the current sampling rate. Lastly, the monitoring results could trigger change in the sampling rate of the sensor as necessary, and in turn, each change in the sampling rate could also trigger change in the model employed to monitor the data. This ensures that a high sampling rate of sensor data is only collected and utilized when necessary.

The adoption of our ACS model provides a more logical, sophisticated mechanism to adapt the sampling frequencies of the sensors. It allows for more advanced condition monitoring paradigm, and for suitable architecture to be configured to adapt the sensors' sampling frequencies based on the perceived system status. In addition, it fulfills the need and requirements of reducing the power consumption and network bandwidth utilization.

Note that previous research on machine fault diagnostics employed traditional supervised learning techniques for classification that typically rely on large amounts of labeled examples from predefined classes for their learning process.. In practice, this approach is not practical because collecting and labeling large sets of data for training are often very expensive if not impossible [23]. Sometimes, the data from a certain class of problems could be rare because most of the time machines operate under normal conditions (denoting the normal class), and only a few examples (representing machines breaking down, or the defective class) are available to train a model to recognize an abnormal status [24]. As such, we also propose an ensemble based learning method to find an optimal solution for the real situations where we have only a small set of training data.

We observe that the proposed ensemble based classifier offers our ACS system much better performance measures compared to the individual traditional state-of-the-art classifiers such as multilayer perceptron (MLP), support vector machine (SVM), or k-nearest neighbors (KNN). This result occurs because, through combining the outputs of several classifiers, we are able to minimize the potential bias and risk of individual predictions, and the expected errors by our ensemble approach can be expected to be reduced. One similar example in a real life medical diagnosis domain is that people consult many doctors before undergoing major surgery, and then based on their diagnosis they can get a fairly accurate diagnosis.

The rest of the paper is organized as follows. Section II describes some of the state-of-the-art sensor-based monitoring methodologies. We investigate their performance measures in utilizing sensor readings of different sampling rates. We also discuss the pros and cons of real-time sampling rate adjustment of the sensors. Section III presents a detailed description of the architectural design of our proposed ACS model that aims to address the issues and harness the power of adaptive sensor sampling rate. We also describe our ensemble learning method in details. In Section IV, a use case scenario is presented to demonstrate the applications of our ACS model as well as illustrate the effectiveness of our proposed techniques through comprehensive experiments. Finally, Section V concludes this paper.

II. Existing Approach

The main procedure for sensor-based data modeling and analysis is summarized as follows. 1) Suitable sensors (vibration, temperature, humidity, etc.) are selected and strategically positioned on the subject of study. 2) The sensors are then configured to operate at a suitable sampling rate along with other sensing parameters. 3) The layout of the sensor network is then drawn up to decide on locations, which might contain the critical information for monitoring purposes. 4) Data are then accumulated over a period of time, and stored for study and analysis. 5) Preprocessing (e.g. fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies) [22] and signal transformation are performed via algorithms such as Fast Fourier Transform (FFT) for frequency domain analysis, or other time and time-frequency based analysis to obtain features that can be used to construct the data models and classifiers. 6) Training of the selected models and classifiers would then take place. 7) Results and performance measures of the models are then analysed and studied, together with the subject matter experts, to derive insights into the process, to understand the underlying fault behavior of the system, or to even exploit the knowledge discovered to optimize the manufacturing process.

Some common scenarios of applications that are based on the above-mentioned approaches are in manufacturing, healthcare systems, home for the elderly, aerospace, disaster prediction and prevention, and intelligent transportation systems. To date, most of such applications employ a constant setting of sensor parameters, including the sensor's sampling rate. While this approach is acceptable for short monitoring duration, it is not desirable when monitoring is to be performed over an extended continuous period of time.

The lack of sensor's sampling rate adaptation would signify 1) an increase in the power consumption, 2) a reduction in the sensor lifespan, and 3) an increase in the network traffic incurred to transfer the collected data. There has been consistent effort to address these three problems by the WSN research community. Such works are important as some sensors are due to be placed in environments where it is impossible to connect them to a power source.

To address the issues mentioned above, we propose a novel ACS architecture for system health monitoring. We will introduce our proposed ACS architecture in details in the next section. An example of the useful applications of ACS would be the sensor-based monitoring of the machines in a manufacturing process. In such cases, if the machine is not in operation (powered off, idle, or under maintenance) or in a functional condition, it would be desirable to either switch off the sensors, or to sample sensor readings at a significantly lower sampling rate.

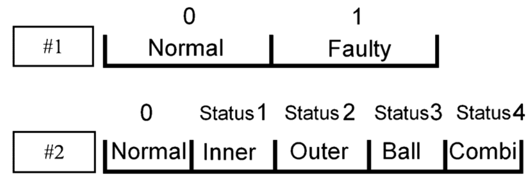

Fig. 1. Setup of Adaptive Classification System. Classifier #1 differentiates between 0—Normal and 1—Faulty states. Classifier #2 differentiates among 0—Normal, Status 1—Inner fault, Status 2—Outer fault, Status 3—Ball fault, and Status 4—Combination fault.

Sensors powered by alternative power sources (e.g. batteries) would have a limited life span. To assure the functionality of sensors, periodic checks and maintenance are needed. For the scale of the sensor deployment environments such as hi-tech manufacturing, the sheer effort required to maintain the sensors is tremendous. Thus, if a sensor works at a relatively low sampling rate under a normal status, and only increases the sampling rate when necessary, extending its lifetime and lowering its cost can be achieved. Due to these needs and requirements, we believe that there will be an increase in the adoption of WSN with adaptive sampling rates. It is therefore imperative that a suitable modeling and monitoring system is developed which will not only react to the changes in the sensor's sampling rate but also to take advantage of it.

III. Proposed Solution

This section presents our proposed solution: the ACS architecture. ACS reacts to different sampling rates, and reaps the benefits of the adaptive sensor sampling rate as applied in a manufacturing environment.

In the existing studies on the adaptive sensor sampling rate strategy [9], [15]–[20], the changes in the sampling rate of the sensors are mostly governed by a predefined rule based system. Such an arrangement is not suitable for complex monitoring scenarios or sensor setup, as the decision to change the sensors' sampling rate is logically dependent of the predefined rules, and not representative of the actual machine conditions.

Rather than designing the whole system as a reactive system, which switches the models to be used according to the sensors' sampling rate, our ACS model adopts a proactive approach. In our model, the required sampling rate of the sensors is decided by the monitoring output of the model. This design is significantly different from the existing approach. Each of the models and classifiers embedded within the ACS will provide a monitoring and classification result based on the sensor readings. One scenario is illustrated in Fig. 1. In this setup, we employ two models, namely classifier #1 and classifier #2, where classifier #1 is trained with a low sampling rate data, but classifier #2 is trained with a high sampling rate data.

While the machine is in a normal state, the sensors would be sampling at a lower rate. The first classifier #1 is to be used at times when the sampling rate is low, and its task is to monitor and classify machine conditions among two states, namely, 0—Normal, and 1—Faulty. When this classifier #1 outputs a monitoring result of 1, which corresponds to a faulty state, it prompts the sensor node to switch to the higher sampling rate to trigger the in-depth investigations. This approach not only allows more detailed information to be collected, it also allows detailed fault identification to be performed by the second classifier.

This second classifier #2 differentiates among Normal, fault status 1: Inner Race fault, fault status 2: Outer Race fault, fault status 3: Ball fault, and fault status 4: Combination fault. Here the combination fault has combined three different faults, namely, inner race fault, outer race fault and ball fault. This detailed analysis and identification of fault is only possible when the model has been trained with data with higher granularity. When the fault has been rectified and the machine returned to normal state, the ACS will inform the sensor node switch back to a lower sampling rate.

By assigning the control of changing the sampling frequencies to our ACS model, we allow a more sophisticated, precise diagnostic process on when to intelligently change the sampling frequencies. In addition, whenever necessary, the models and classifiers can be trained to provide in-depth monitoring and analysis of machine health status like in the case of the second model. With this functionality, the proposed ACS model can lower the network traffic and power consumption. Thus, the model can extend the usability lifespan of battery-powered sensors when applied to real-world applications, such as continuous health-monitoring of aerospace equipment (engines, bearings, gear boxes etc), to enable the assessment of system status and prevent catastrophic failures. In addition, manufacturing also needs to equip sensors to collect data for monitoring the system health to prevent machines from breakdown. Other applications include disaster monitoring systems (e.g. fire alarm), patient health monitoring, and environmental monitoring (for chemical plant accidents, air and water quality), etc.

Another useful feature that can be incorporated within our system would be a feature to allow for asynchronous alert systems for the end users based on the monitoring results. As the monitoring results are captured and sent to the sensor node for sampling rate adjustment, messages can also be sent to web services or other devices to notify the end users of the machine status accordingly. This ability also enables other more factory-specific alert actions such as messaging, lighting control, and alarms to be activated based on machine health status.

In ASC, one challenge is how to build the accurate models and classifiers. Typically, supervised learning methods are applied to learn these models from labeled training data. As discussed in the Introduction section, current supervised learning methods for fault diagnostics have to rely on the *sufficient* training examples to build accurate classifiers. However, obtaining large amounts of training data can be a labor-intensive, time-consuming process, and it also highly depends on the expertise of domain experts. Thus, in many real-world applications, we have to face the challenge of learning with the limited training examples. In this paper, we build our ensemble-based classifier which integrates multiple state-of-the-art classifiers such as MLP, SVM, KNN, etc., where each of them can be trained using the features extracted from the raw sensor signals.

To minimize the bias and risk of individual classifiers, we introduce an ensemble based learning method to our ACS model. The strategy of the ensemble systems is to create many classifiers, and combine their outputs such that the combination improves upon the performance measures of single classifiers. The intuition is that if each classifier makes different errors, then a strategic combination of these classifiers can reduce the total error. The overall principal in ensemble systems is therefore to make each classifier as unique as possible, particularly with respect to misclassified instances [25]. In this paper, our ensemble-based ASC model is built by integrating multiple, diverse classifiers, i.e. MLP, SVM, KNN models for time domain, and MLP, SVM, KNN models for frequency domain, and consists of two steps, i.e., compute confidence scores, and integrate for prediction.

For each machine state s, the *confidence score* of a model c is computed as

$$w_c^s = log\frac{p_c^s}{1 - p_c^s} \tag{1}$$

This estimated accuracy p_c^s can be computed using the training data with n-fold cross-validations (CV). In particular, in an n-fold CV, we can randomly divide the training data into n equal partitions; and each time, one partition is reserved for testing, and the remaining $n - 1$ partitions are used for computing accuracy for that particular round. This process can be repeated n times until each partition has been used as a test partition once. By averaging the n accuracies, a good estimation of p_c^s can be computed. Note that the higher the confidence score w_c^s, the more confident the classifier c can be used to classify the state s.

The confidence scores computed in the first step indicate that some classifiers are more qualified than others in classifying a particular data set. In the following step, we integrate the classifiers by accumulating the voting. During the voting, we will take the confidence scores into consideration. In particular, the final machine state prediction s_{argmax} of our ensemble system is the state that satisfies

$$s_{\text{argmax}} = \underset{1 \leq s \leq S}{\text{argmax}} \sum_{c=1}^{C} w_c^s d_c^s \tag{2}$$

We will show in our experiments that the proposed ensemble based classifier indeed achieves much better performance measures compared to the individual traditional state-of-the-art classifiers.

The next section will discuss and evaluate the performance of our ACS model, as well as ensemble learning method based on data collected from sensors operating under varying sampling rates.

IV. Use Case Scenario

In this section, we present use-case scenarios employed to validate the effectiveness of our proposed system. In Section IV-A, we explain our experimental setup, data collection process, various feature extraction methodologies, and learning models and classifiers, as well as the evaluation

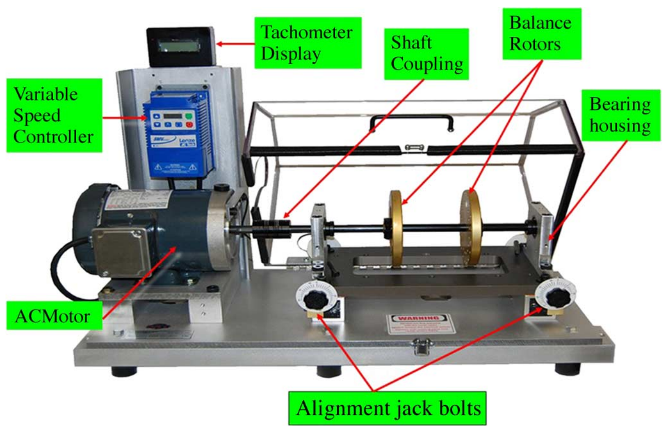

Fig. 2. Main components of the MFS-Lite Fault Simulator.

metric. In Section IV-B, we perform comprehensive experimental studies to investigate the classification performance measures for binary classification, multi-class classification, ACS without ensemble-based and with ensemble-based learning for intelligent sensing machine fault diagnostics.

A. Experimental Setup, Data Collection, Feature Extraction, Classification Model, and Evaluation Metric

1) Machinery Fault Simulator: Due to the lack of real machinery data to validate our proposed system, an industrial Machinery Fault Simulator (MFS) is employed in our study to generate the different sets of normal, idle, and faulty machine data. The MFS that we employed is the MFS Lite from SpectraQuest, which provides an innovative tool for studying signatures of common machinery faults without having to compromise factory production or profits. MFS allows us to introduce various faults, either individually or jointly in a controlled environment, and is ideal as a test bed for our system.

The MFS setup is shown in Fig. 2. The hardware components of the MFS consist of a motor, two balance rotors, speed controller, tachometer display, and two bearing housings. To simulate the effect of different bearing faults, the good bearing of the MFS is subsequently replaced with a known defective bearing. Based on the bearing geometry, MFS provides the following four types of faulty bearings: outer race defects (outer), inner race defects (inner), ball spin defects (ball), and combined defects (combination) faults.

Data Collection: In our data collection process, the sensors are placed near the bearing housing of the defective bearing to obtain the most pronounced fault signatures in the vibration signal (Fig. 3). Two channels of accelerometer readings, which capture the horizontal and vertical vibration signals, are acquired for each simulation. The fault simulations are done in three different machine operating speeds: 800 rotations per minute (rpm), 1780, rpm and 3600 rpm. For each machine speed, 100 of 1 second window samples are collected under three sampling frequencies: 256 Hz (low), 1024 Hz (medium), and 5120 Hz (High). A one-second snapshot of the vibration signals collected via the front right sensor is shown in Fig. 4.

Fig. 3. Experimental setup. The sensors are placed in the red circles respectively.

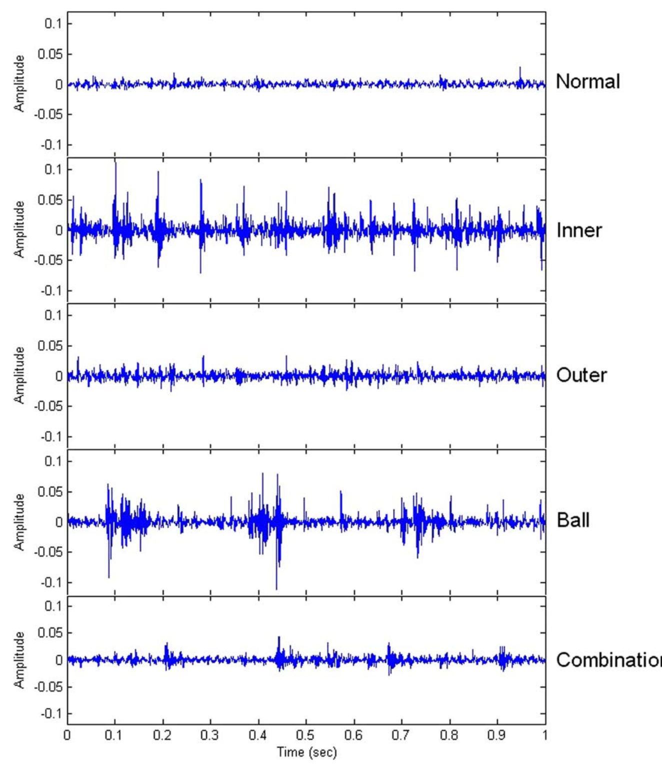

Fig. 4. Snapshot of vibration signals collected via the right front channel.

In this study, we investigate the performance of the ACS, which reacts to different sampling rates, and reaps the benefits of the adaptive sensor sampling rate as applied to diagnose faulty bearing conditions based on vibration data collected via accelerometers. Five datasets, each representing different machine states (normal, ball fault, inner fault, outer fault, and combination fault), are collected with different sampling rates for each machine operating speed. This summed up into a total of 1500 seconds worth of sensor data; and among them, 50% of the data are used for training the models and classifiers, and 50% of the data are used for testing purposes. The data are subsequently windowed for analysis. A one-second snapshot of the vibration signals collected via the right front channel is shown in Fig. 4.

Feature Extraction: Vibration monitoring and analysis is a commonly used technique for bearing fault detection. There

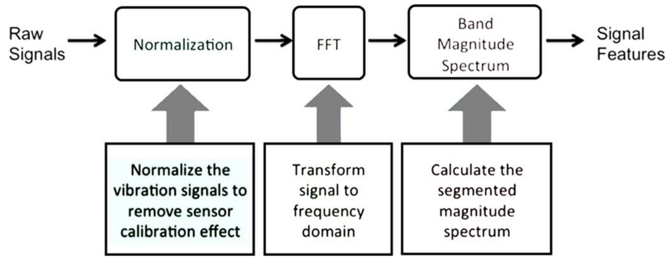

Fig. 5. Frequency domain features for bearing detection.

are two main approaches to the analysis of bearing vibration signatures: time-domain, and frequency-domain analyses [21]. In time-domain analysis, four time-domain features are extracted from the raw vibration signal to characterize each of the bearing faults. They are peak value, kurtosis, Root-Mean-Squared (RMS) value, and standard deviation. They are computed as in (3) –(6).

1 Peak Value

$$PV = \frac{\max(x_i) - \min(x_i)}{2} \tag{3}$$

2 Root-Mean-Square (RMS) Value

$$\mathrm{RMS} = \sqrt{\frac{1}{N} \cdot \sum_i (x_i)^2} \tag{4}$$

3 Standard Deviation

$$\mathrm{StdDev} = \sqrt{\frac{1}{N} \cdot \sum_i (x_i - \bar{x})^2} \tag{5}$$

4 Kurtosis Value

$$K = \frac{\frac{1}{N}\sum_i (x_i - \bar{x})^4}{\mathrm{RMS}^4} \tag{6}$$

These four features are chosen as they have been widely used to characterize bearing faults in industrial machines [1].

The frequency-domain analysis approach, on the other hand, involves examining the vibration spectrum to discover the presence of defect frequencies in the vibration signal [21]. In this study, we calculate the band magnitude spectrum for each segment of the vibration signal, and use this as the set of features for learning different machine states, as shown in Fig. 5. Specifically, the raw vibration signal is first normalized to the $[-1, 1]$ range using the formula

$$\mathrm{NorData} = 2 \cdot \frac{x - \min(x)}{\max(x) - \min(x)} - 1 \tag{7}$$

Fast Fourier Transform (FFT) is then used to transform the normalized data from the time domain to the frequency domain. Finally, the input signal in the frequency domain is transformed into discrete band-magnitude features with the bandwidths of 32 Hz, 50 Hz, and 100 Hz.

Classification Models: For the data modeling and monitoring (classification) process, three classifiers are employed: MLP, SVM, and KNN. Each classifier is applied to both the time-domain and frequency-domain to build different classification models for bearing fault detection. This results into six bearing detection models, namely, MLP for the time domain $(\mathrm{MLP\_t})$, MLP for the frequency domain $(\mathrm{MLP\_f})$, SVM for the time domain $(\mathrm{SVM\_t})$, SVM for the frequency domain $(\mathrm{SVM\_f})$, KNN for the time domain $(\mathrm{KNN\_t})$, and KNN for the frequency domain $(\mathrm{KNN\_f})$.

2) Evaluation Metric: In this study, we use the F-measure to evaluate the performance measures of different classification models. The F-measure is the harmonic mean of precision (denoted as p) and recall (denoted as r), and it is defined as

$$F = 2 \cdot \frac{p \cdot r}{p + r} \tag{8}$$

In other words, the F-measure reflects an average effect of both precision (p) and recall (r). The F-measure is large only when both precision and recall are good. This is suitable for our purpose to accurately classify each particular machine state. Having either too small a precision or too small a recall is unacceptable, and would be reflected by a low F-measure.

Note that in this paper we will compute the F-measure for each class or machine state. We will report both the F-measure for each class, as well as the average F-measure of all the classes and states, which represents the overall performance measure.

B. Comprehensive Studies of Bearing Fault Detection

In this subsection, we perform comprehensive experiments to evaluate the performance measures of four different types of classification models: binary classification, multi-class classification, ACS system without ensemble-based learning, and ensemble based ACS system for intelligent sensing machine fault diagnostics. To provide a holistic picture of these techniques, we build these models using different sampling rates (low frequency 256 Hz, medium frequency 1024 Hz, and high frequency 5120 Hz), with different machine operating speeds (800 RPM, 1780 RPM, and 3600 RPM), and with different feature representation methods (for both time domain and frequency domain feature extractions). In each of the following testing cases, the classification models are trained and evaluated with data collected on their respective sampling frequencies.

1) Binary Classification: We investigate the performance measures of the six trained models ($\mathrm{MLP\_t}$, $\mathrm{MLP\_f}$, $\mathrm{SVM\_t}$, $\mathrm{SVM\_f}$, $\mathrm{KNN\_t}$ and $\mathrm{KNN\_f}$) when they are employed to detect a machine-bearing's binary conditions, i.e. whether the machine is currently operating under a healthy condition, or whether a fault or a combination of faults has happened. Therefore, for the purpose of this study, all the faulty bearing states mentioned above (i.e. outer, inner, ball, and combination faults) are combined as a single label of "Defective" or "Fault" class, while the normal state corresponds to the "Normal" class.

The three models' detection and classification performance measures of time domain models ($\mathrm{MLP\_t}$, $\mathrm{SVM\_t}$, and $\mathrm{KNN\_t}$) for three machine operating speeds (800 RPM, 1780 RPM, and 3600 RPM), and three sampling rates (low, medium, and high) are shown in Table I. We observe that both $\mathrm{MLP\_t}$ and $\mathrm{SVM\_t}$ are able to achieve 100% of the F-measure for all operation speeds with all sampling rates. This result indicates that the healthy bearing condition is well differentiated with

TABLE I
BINARY CLASSIFICATION PERFORMANCE MEASURES OF TIME DOMAIN MODE

Sensor Sampling Rate		Low (256Hz)			Medium (1024Hz)			High (5120Hz)		
Machine Speed (RPM)		800	1780	3600	800	1780	3600	800	1780	3600
Normal(%)	MLP_t	100	100	100	100	100	100	**100**	**100**	**100**
	SVM_t	100	100	100	100	100	100	**100**	**100**	**100**
	KNN_t	40.4	56.2	86.6	**70.7**	**63.9**	96	65.3	63.3	**100**
Fault(%)	MLP_t	100	100	100	100	100	100	**100**	**100**	**100**
	SVM_t	100	100	100	100	100	100	**100**	**100**	**100**
	KNN_t	87.1	90.5	96.8	**92.8**	91.3	99	91.2	**93.1**	**100**
Average(%)	MLP_t	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	SVM_t	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	KNN_t	**63.8**	**73.3**	**91.7**	**81.7**	**77.6**	**97.5**	**78.3**	**78.2**	**100**

TABLE II
BINARY CLASSIFICATION PERFORMANCE MEASURES OF FREQUENCY DOMAIN MODE

Sensor Sampling Rate		Low (256Hz)			Medium (1024Hz)			High (5120Hz)		
Machine Speed (RPM)		800	1780	3600	800	1780	3600	800	1780	3600
Normal(%)	MLP_f	75.8	63.9	85.4	100	94	99	**100**	**100**	**100**
	SVM_f	69.5	60.4	84.8	100	91.5	98	**100**	**100**	**100**
	KNN_f	67.5	59.1	84.1	96.1	87.9	95.9	**100**	**100**	**100**
Fault(%)	MLP_f	94.3	91.3	96.8	100	98.5	99.8	**100**	**100**	**100**
	SVM_f	92.8	90.6	96.6	100	98	99.5	**100**	**100**	**100**
	KNN_f	93.5	91.3	96.6	99	97.3	99	**100**	**100**	**100**
Average (%)	MLP_f	**85.1**	**77.6**	**91.1**	**100**	**96.1**	**99.4**	**100**	**100**	**100**
	SVM_f	**81.2**	**75.5**	**90.7**	**100**	**94.8**	**98.7**	**100**	**100**	**100**
	KNN_f	**80.5**	**75.2**	**90.3**	**97.5**	**92.6**	**97.5**	**100**	**100**	**100**

general defective conditions using time domain features. The KNN based model $\mathrm{KNN\_t}$, as expected, performed worse than the other two strong classifiers $\mathrm{MLP\_t}$ and $\mathrm{SVM\_t}$.

The other three models' detection and classification performance measures for frequency domain models ($\mathrm{MLP\_f}$, $\mathrm{SVM\_f}$, and $\mathrm{KNN\_f}$) for three machine operating speeds (800, 1780, and 3600), and three sampling rates (low, medium, and high) are shown in Table II. The three frequency domain models obtain good results using high sampling rates, but present quite poor results when we apply low sampling rates. Such results are not surprising as some of the bearing faults can only be discovered in the signals with high sampling rates.

2) Multi-Class Classification: Binary classification can only distinguish between the machine's normal and faulty states. More often than not, in practice, we may need to identify the specific types of the faults. As such, we investigate the performance measures of the six models and classifiers in recognizing the types of bearing faults. In particular, we have five classes in our classification: normal, outer fault, inner fault, ball fault, and combination fault. In this paper, we build the multi-class classifiers for all three machine operating speeds and all the three sampling rates. Note that the multi-class SVM is implemented using the library for support vector machines (LIBSVM) [26].

The evaluation results are shown in Tables III and IV for time domain, and frequency domain modes respectively. Compared to the classification models of time domain, the models of the frequency domain with high sampling rates produce consistently better classification results across three machine speeds. In particular, SVM for the frequency domain ($\mathrm{SVM\_f}$) model with high sampling rates (5120 Hz) archives average F-measures of 99%, 99%, and 100% for the three machine speeds 800 rpm, 1780 rpm, and 3600 rpm, respectively, which are significantly better than the results with low sampling rates, indicating that a high sampling rate would yield better diagnostics performance. We also observe that both $\mathrm{MLP\_f}$ and $\mathrm{KNN\_f}$ (MLP and KNN with frequency domains) are able to achieve satisfactory results using a high sampling rate (5120 Hz), while they are slightly worse than the $\mathrm{SVM\_f}$ model.

3) ACS System Without Ensemble-Based Learning: In this case study, we evaluate the performance of our ACS system without ensemble-based learning, where the classification system will adapt the sampling frequencies of the sensors according to the observed machine condition. When the sampling rate changes, the ACS will consequently employ the corresponding classifier that is best suited for the sampling frequencies to perform the next classification.

In our test case, when the machine is in the normal state, the sampling rate of the sensors is set to be 256 Hz. The low sampling rate time domain $\mathrm{SVM\_t}$ is trained to differentiate between the normal and faulty machine states. This classifier is the one that has been trained in the binary classification study,

TABLE III
MULTI-CLASS CLASSIFICATION RESULTS OF TIME DOMAIN MODE

Sensor Sampling Rate		Low (256Hz)			Medium (1024Hz)			High (5120Hz)		
Machine Speed (RPM)		800	1780	3600	800	1780	3600	800	1780	3600
Normal(%)	MLP_t	100	98	95.8	100	99	100	**100**	**100**	**100**
	SVM_t	61.7	75	96.2	7.7	**85.1**	93.2	**68**	16.1	**98**
	KNN_t	40	56.2	86.6	**69.8**	**63.9**	89.8	66.7	63.3	**100**
Inner(%)	MLP_t	68.8	58.4	64.5	86.5	68.7	91.1	**93.1**	**96.2**	**96**
	SVM_t	75.9	53.3	63.8	83.5	76.7	83.6	**86.8**	**95.2**	**95.2**
	KNN_t	57.4	44.6	61.9	35.9	67.9	77.6	**60.7**	**90.6**	**92**
Outer(%)	MLP_t	67.8	72.1	55.8	80	70.9	91.6	**86.5**	**97.1**	**95.2**
	SVM_t	0	78.1	89.3	**65.4**	**88.5**	92.9	11.1	62.3	**98**
	KNN_t	62.7	66.1	80	**76.5**	66	90.2	60.9	**76**	**100**
Ball(%)	MLP_t	51.1	50.7	45.8	65.9	62.2	85.1	**71.1**	**92.6**	**85.7**
	SVM_t	49.5	39.5	53.7	68.2	44.7	76.9	**78.3**	**94.7**	**89.1**
	KNN_t	35.2	29.2	56.9	44.2	32.9	62	**44.9**	**43.5**	**76.2**
Combination(%)	MLP_t	78.3	96.9	79.1	**91.1**	99	98	89.8	**98**	**90.7**
	SVM_t	55.4	53	76.7	84.6	75	94.8	**91.1**	**100**	**93.6**
	KNN_t	44.2	49	43.3	67.3	60.4	57.1	**78.8**	**67.2**	**73.7**
Average(%)	MLP_t	**73.2**	**75.2**	**68.2**	**84.7**	**79.9**	**93.1**	**88.1**	**96.8**	**93.5**
	SVM_t	**48.5**	**59.8**	**75.9**	**61.9**	**74**	**88.3**	**67.1**	**73.7**	**94.8**
	KNN_t	**47.9**	**49**	**65.7**	**58.8**	**58.2**	**75.3**	**62.4**	**68.1**	**88.4**

TABLE IV
MULTI-CLASS CLASSIFICATION RESULTS OF FREQUENCY DOMAIN MODE

Sensor Sampling Rate		Low (256Hz)			Medium (1024Hz)			High (5120Hz)		
Machine Speed (RPM)		800	1780	3600	800	1780	3600	800	1780	3600
Normal(%)	MLP_f	32.8	30.3	76.5	86.4	89.6	97	**100**	**100**	**100**
	SVM_f	76.8	53.8	85.7	99	91.7	98	**100**	**100**	**100**
	KNN_f	70.2	53.3	84.5	96.1	87.9	95.9	**100**	**97.4**	**100**
Inner(%)	MLP_f	4.2	39.7	5.6	88	92.5	72	**100**	**100**	**100**
	SVM_f	61.9	74.1	42	98	96.2	87	**100**	**100**	**100**
	KNN_f	59.3	72.4	38.2	96.2	94.3	83.3	**100**	**97.6**	**95.2**
Outer(%)	MLP_f	34.1	47	60.5	88.3	86	71.7	**95.2**	**100**	**97.6**
	SVM_f	81.1	64.2	80	**98**	91.3	98	97.6	**100**	**100**
	KNN_f	78.9	63.7	76.5	**97**	89.9	96.1	95.2	**97.6**	**100**
Ball(%)	MLP_f	54	51	52.1	72.5	82.8	71.7	**94.7**	**97.4**	**97.4**
	SVM_f	53.2	49.4	52.7	76.9	87.9	82.2	**97.4**	**97.4**	**100**
	KNN_f	42.9	30.4	49	73.1	90.3	65	**91.9**	**94.7**	**94.7**
Combination(%)	MLP_f	3.8	43.2	0	79.2	94.2	81.8	**100**	**97.6**	**100**
	SVM_f	29.3	53.8	20.8	83.3	94.3	90.2	**100**	**97.6**	**100**
	KNN_f	40	52.9	27.7	76	97	75	**97.6**	**97.6**	**100**
Average(%)	MLP_f	**25.8**	**42.2**	**38.9**	**82.9**	**89**	**78.8**	**98**	**99**	**99**
	SVM_f	**60.5**	**59.1**	**56.2**	**91.1**	**92.3**	**91.1**	**99**	**99**	**100**
	KNN_f	**58.2**	**54.6**	**55.2**	**87.7**	**91.9**	**83.1**	**96.9**	**97**	**98**

where it has an average of 100% in terms of the F-measure (see Table I). When the machine condition has been identified to be in a faulty state, the ACS will inform the sensor node to switch to a higher sampling rate of 5120 Hz. Subsequently, the next monitoring task will be executed by the appropriate model, in this case the SVM_f model, which is trained under a 5120 Hz data sampling rate using frequency domain features.

From Table V, we can see that our ACS achieves very good results with 100% F-measures for all machine states with a machine speed of 3600 rpms. In fact, only four numbers in Table V are not 100%: for the outer faults, we have 97.4% for both machine speeds 800 rpm and 1780 rpm; for ball, and combination faults we have 97.6% for machine speed 800 rpm, and 1780 rpm, respectively. The overall average results are 99% for machine

TABLE V
ADAPTIVE CLASSIFICATION RESULTS WITHOUT ENSEMBLE-BASED LEARNING

	Adaptive Classification System without Ensemble-based Learning		
Machine Speed (RPM)	800	1780	3600
Normal (%)	100	100	100
Inner(%)	100	100	100
Outer (%)	97.6	100	100
Ball (%)	97.4	97.4	100
Combination (%)	100	97.6	100
Average (%)	**99**	**99**	**100**

TABLE VI
CLASSIFICATION RESULTS OF THE PROPOSED ENSEMBLE BASED CLASSIFIER

Training Percentage	Testing Percentage	Machine Speed (RPM)	MLP_f (%)	SVM_f (%)	KNN_f (%)	MLP_t (%)	SVM_t (%)	KNN_t (%)	Ensemble (%)	Improvement (%)
80	20	800	86.1	**96**	90.8	88.9	66.3	58.7	**98**	2.08
		1780	90	**95**	90.8	86.3	83.7	57.4	**99**	4.21
		3600	84.5	**92.8**	85.1	91.6	88.6	81.4	**96**	3.45
70	30	800	86.8	**95.3**	87.6	84.8	65.5	56.4	**95.9**	0.63
		1780	88.1	**90.4**	89.8	85.9	87.7	60.4	**97.6**	7.96
		3600	75.6	**94.1**	84	91.6	89.2	77.9	**95.1**	1.06
60	40	800	89.1	**90.5**	87.7	84.1	62.5	55.2	**93.1**	2.87
		1780	86.3	**91.3**	89.9	82.1	84.6	62.5	**97.5**	6.79
		3600	78.7	90.4	82.2	**93.5**	89.3	77.2	**96.5**	3.21
50	50	800	86.6	**91.1**	87.7	86.3	61.9	58.8	**93.5**	2.63
		1780	86.2	**92.3**	91.9	83.2	74	58.2	**97.6**	5.74
		3600	72.7	**91.1**	83.1	90.7	88.3	75.3	**95.2**	4.50
40	60	800	81.7	**89.5**	86.7	80.6	61.1	59.3	**93.2**	4.13
		1780	84.7	**91.9**	83.9	82.6	65	55.9	**95.9**	4.35
		3600	73.4	**92**	83.6	90.7	89.9	73.6	**93.8**	1.96
30	70	800	78.1	**88.1**	84.1	86.8	61.2	58.6	**92.4**	4.88
		1780	75.6	**87.6**	80	85.5	69.4	55.4	**94**	7.31
		3600	65.4	87.6	79.8	**89.9**	89.1	74	**94.2**	4.78
20	80	800	73.1	**88**	84.2	84.8	61	55	**92.5**	5.11
		1780	72.4	82.8	81.6	**83.5**	64.2	58.8	**89.7**	7.43
		3600	69.6	81.5	76	**84.7**	**84.7**	69.9	**90.7**	7.08
10	90	800	62.6	77.4	78.4	**81**	50.7	51.4	**89.7**	10.74
		1780	61	**80.5**	77.2	80.2	65.2	49.3	**85.7**	6.46
		3600	60	74.7	74.2	77.3	**84.6**	67.1	**91.4**	8.04
5	95	800	45.6	**76.1**	70.7	73.9	52.6	40.8	**81.2**	6.70
		1780	61.9	77.8	76.3	**78.4**	63.1	51.4	**85.5**	9.06
		3600	57	73.1	70.6	68.8	**74.4**	59.5	**82.5**	10.89

speed 800 rpm, and 1780 rpm; and 100% for machine speed 3600 rpm.

Moreover, by applying the ACS system, we are able to address the issues of 1) extending usability of battery powered sensors via better power efficiency, 2) reducing network traffic load, 3) allowing classifiers to maintain performance while sampling frequency varies, and 4) allow for a better, higher level of control system over the change in sampling rate via classifiers.

4) Ensemble Based ACS System: Finally, we study whether our proposed Ensemble based ACS system can enhance the classification results, especially when we have relatively less training examples available. For this purpose, we changed the percentages for training and testing in the first two columns in Table VI. We listed the results of six individual classifiers and our proposed ensemble based method (represented by Ensemble in Table VI) with different machine speeds (here we fixed the medium sampling rate at 1024 Hz in our experiments). We observe that our proposed ensemble based method produces consistently better classification results across three machine speeds compared to those of individual classifiers, regardless of the percentages of training and test data. When we have relatively less training data, our proposed method can achieve bigger improvements, indicating that we can effectively minimize the potential bias and risk of individual classifiers, and thus reduce the expected errors. For example, for the challenging situation with 5% training data and 95% test data, we can achieve 6.70%, 9.06%, and 10.89% better results than the second best models for machine speeds 800 rpm, 1780 rpm, and 3600 rpm, respectively.

In summary, the proposed ensemble based ACS model outperforms individual methods in various training and test data percentages in term of prediction accuracy, robustness, and stability. As pointed out in [25], the performance measure of an ensemble system is not always better than that of the best individual classifier. It mainly depends on two key components of that ensemble system. They are the diversity of the individual classifiers, and the strategy to combine the output of these classifiers. In this paper, the superiority of our proposed ensemble based ACS model comes from its six diversity classifiers, as well as our effective integration mechanism, which can reduce the classification errors by eliminating the potential bias and risk of individual classifiers.

Recall that we have employed n-fold cross validation for estimating the confidence scores for each individual classifiers for different states. Given that we could have very limited training samples when the training percentage is small (e.g. 5%), we choose a small $n = 2$ to avoid tiny partitions in our implementation.

V. Conclusion

The deployment and usage of sensors for system health monitoring are common and widespread in modern manufacturing. Their main objective is to provide a non-intrusive method for remote monitoring of machine performance. Hence, the reduction in power consumption in sensor-based machine health monitoring is undoubtedly of exceptional benefit to the manufacturing domain.

With the decline in the cost, and technical barrier to the installation and setup of WSN, more sensors are being added and deployed onto manufacturing floors. As the number of sensors deployed increases, there is the need to maximize the usability lifespan of battery powered sensors, and reduce the amount of network traffic bandwidth consumed by the transmission of sensor data.

The recent developments in the sensor community to address the above-mentioned issues have generally moved in two main directions. The first direction is to identify and create new algorithms that allow for a reduction in sensors' sampling rate while maintaining the resolution and monitoring capability of the system. The second approach attempts to establish the capabilities of the sensor nodes to adapt the sampling frequencies of its attached sensors according to the requirements.

In light of such recent developments, we have developed the ensemble based ACS approach to take advantage of these latest sensor capabilities. We have developed a system that integrates various models and classifiers trained under different scenarios, and adapt the sampling frequencies of the sensors according to the observed machine status. We have also demonstrated the use of such an ACS configuration in monitoring the health status of a machine. The results produced have shown that such an ACS configuration is able to maintain the accuracies of its health monitoring effort while at the same time reduce the sensors' data rate, as compared to when a constant high sampling rate is employed for the monitoring task.

As part of our future work, we will investigate the optimal set of sensor's sampling frequencies range for which models and classifiers have to be trained on. It will be highly impractical to require models to be trained at all possible sampling frequencies. This effort is such that we will be able to avoid overloading the ACS with too many models and classifiers.

References

[1] K. L. Tew, S. D. Teddy, P. Y. Watt, and X.-L. Li, "A systematic approach towards manufacturing health management," *Prognostics and System Health Management Conference (PHM-Shenzhen)*, pp. 1–6, May 24–25, 2011.

[2] G. Niu, D. Anand, and M. Pecht, "Prognostics and health management for energetic material systems," *Prognostics and Health Management Conference, 2010. PHM '10*, pp. 1–7, January 12–14, 2010.

[3] S. Massam and S. McQuillan, "Verification of PHM capabilities: A joint customer/industrial perspective," in *Aerospace Conference Proceedings, 2002. IEEE*, 2002, vol. 6, pp. 2799–2813.

[4] C. Phua, V. S.-F. Foo, J. Biswas, A. Tolstikov, A.-P.-W. Aung, J. Maniyeri, M.-H. That, and A. K.-W. Chu, "2-layer Erroneous-Plan Recognition for dementia patients in smart homes," in *11th International Conference on e-Health Networking, Applications and Services, 2009. Healthcom 2009*, December 16–18, 2009, pp. 21–28.

[5] K. Sim, G.-E. Yap, C. Phua, J. Biswas, A.-P.-W. Aung, A. Tolstikov, W. Huang, and P. Yap, "Improving the accuracy of erroneous-plan recognition system for activities of daily living," in *e-Health Networking Applications and Services (Healthcom), 2010 12th IEEE International Conference on*, July 1–3, 2010, pp. 28–35.

[6] S. K. Sahoo, W. Lu, S. D. Teddy, D. Kim, and M. Feng, "Detection of atrial filbrillation from non-episodic ECG data: A review of methods," in *Annual International Conference of the Engineering in Medicine and Biology Society(EMBC)*, 2011.

[7] M. Feng, Z. Zhang, F. Zhang, Y. Ge, L. Y. Loy, K. Vellaisamy, W. Guo, P. L. Chin, N. K. K. King, B. T. Ang, and C. Guan, "iSyNCC: An intelligent system for patient monitoring & clinical decision support in neuro-critical-care," in *Annual International Conference of the Engineering in Medicine and Biology Society(EMBC)*, 2011.

[8] M. V. Ramesh and P. Ushakumari, "Threshold based data aggregation algorithm to detect rainfall induced landslides," in *Proceedings of 2008 International Conference on Wireless Networks (ICWN)*, 2008, vol. 1, pp. 255–261.

[9] C. Alippi, G. Anastasi, C. Galperti, F. Mancini, and M. Roveri, "Adaptive sampling for energy conservation in wireless sensor networks for snow monitoring applications," in *IEEE International Conference on Mobile Adhoc and Sensor Systems, 2007. MASS 2007*, October 2007, pp. 1–6.

[10] D. Jiang and C. Liu, "Machine condition classification using deterioration feature extraction and anomaly determination," *IEEE Transactions on Reliability*, vol. 60, no. 1, pp. 41–48, March 2011.

[11] Y. Z. Rosunally, S. Stoyanov, C. Bailey, P. Mason, S. Campbell, G. Monger, and I. Bell, "Fusion approach for prognostics framework of heritage structure," *IEEE Transactions on Reliability*, vol. 60, no. 1, pp. 3–13, March 2011.

[12] Y. Zhang, G. W. Gantt, M. J. Rychlinski, R. M. Edwards, J. J. Correia, and C. E. Wolf, "Connected vehicle diagnostics and prognostics, concept, and initial practice," *IEEE Transactions on Reliability*, vol. 58, no. 1, pp. 286–294, 2009.

[13] T. J. Sheskin, "Fault diagnosis with imperfect tests," *IEEE Transactions on Reliability*, vol. R-30, no. 2, pp. 156–160, 1981.

[14] P. Yip, "Estimating the number of errors in a system using a martingale approach," *IEEE Transactions on Reliability*, vol. 44, no. 2, pp. 322–326, 1995.

[15] W. R. Dieter, S. Datta, and K. K. Wong, "Power reduction by varying sampling rate," in *Proceedings of the 2005 International Symposium on Low Power Electronics and Design, 2005. ISLPED '05*, August 8–10, 2005, pp. 227–232.

[16] Y. Imaoka and Y. Sanada, "Experimental Investigation of Sampling Rate Selection with Fractional Sampling for IEEE802.11b WLAN System," in *2009 IEEE 70th Vehicular Technology Conference Fall (VTC 2009-Fall)*, September 20–23, 2009, pp. 1–5.

[17] I. Crk, F. Albinali, C. Gniady, and J. Hartman, "Understanding energy consumption of sensor enabled applications on mobile phones," in *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, September 3–6, 2009, pp. 6885–6888.

[18] M. Qiu and E. H. Sha, "Energy-aware online algorithm to satisfy sampling rates with guaranteed probability for sensor applications," in *IEEE International Conference on High Performance Computing and Communications*, 2007, pp. 156–167.

[19] T. Kurp, R. X. Gao, and S. Sah, "An adaptive sampling scheme for improved energy utilization in wireless sensor networks," in *Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE*, May 3–6, 2010, pp. 93–98.

[20] R. Rieger and J. T. Taylor, "An adaptive sampling system for sensor nodes in body area networks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17, no. 2, pp. 183–189, April 2009.

[21] N. Tandon and A. Choudhury, "A review of vibration and acoustic measurement methods for the detection of defects in rolling element bearings," *Tribology International*, vol. 32, pp. 469–480, 1999.

[22] J. Han, M. Kamber, and J. Pei*, Data Mining: Concepts and Techniques*, 3rd ed. New York: Morgan Kaufmann, 2011.

[23] M. N. Nguyen, X.-L. Li, and S.-K. Ng, "Positive unlabeled leaning for time series classification," *International Joint Conferences on Artificial Intelligence*, pp. 1421–1426, 2011.

[24] H. Cao, X.-L. Li, Y. K. Woon, and S.-K. Ng, "SPO: Structure preserving oversampling for imbalanced time series classification," in *IEEE International Conference on Data Mining*, 2011.

[25] R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits and Systems Magazine*, vol. 6, pp. 21–45, 2006.

[26] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *Computer and Information Science*, vol. 104, pp. 1–30, 2001.

Minh Nhut Nguyen received B.S. and M.S. degrees in computer engineering from Ho Chi Minh City University of Technology, Vietnam, in 2001, and 2005, respectively; and the Ph.D. degree in computer science from Nanyang Technological University, Singapore in 2008. He is currently a Research Scientist of the Institute for Infocomm Research, A *STAR, Singapore. His research interests include machine learning for time series and stream data, pattern recognition, and neural networks.

Chunyu Bao received the Ph.D. degree in Computer Engineering from National University of Singapore, Singapore (2009). He is currently a Research Scientist in the Data Mining Department, Institute for Infocomm Research, Agency for Science, Technology and Research (A *STAR). His current research interests include artificial neural networks, emotion synthesis on 3D face models, and machine learning on large scale data.

Kar Leong Tew received his Bachelor of Science degree (First Class Honors) from the University Of Exeter, UK in 2005. He is currently working as a Senior Researcher in SAP AG working in the domain of Business Network Orchestration. His research interests are in the domain of data mining, network analysis, and information visualization.

Sintiani Dewi Teddy (S'03–M'06) received both her B.Eng. (First Class Honors), and Ph.D. degrees in computer engineering from Nanyang Technological University, Singapore, in 2003, and 2008 respectively. She is currently a Principal Investigator with the Data Mining Department of the Institute for Infocomm Research (A *STAR), Singapore. Her current research interests include the cerebellum and its computational models, artificial neural networks, the study of brain-inspired learning memory systems, computational finance, and autonomous control of bio-physiological processes.

Xiao-Li Li (M'08) is currently a principal investigator in the Data Mining Department at the Institute for Infocomm Research, Agency for Science, Technology and Research (A *STAR), Singapore. He also holds an appointment of adjunct professor in the School of Computer Engineering, Nanyang Technological University. His research interests include data mining, machine learning, and bioinformatics. He has been serving as a member of technical program committees in the leading data mining and machine learning conferences KDD, ICDM, CIKM, SDM, AAAI, and PKDD/ECML etc. He is the Editor-in-Chief of the International Journal of Knowledge Discovery in Bioinformatics (IJKDB). In 2005, he received the best paper award in the 16th International Conference on genome informatics (GIW 2005). In 2008, he received the best poster award in the 12th Annual International Conference Research in computational molecular biology (RECOMB 2008). In 2011, he received the Best Paper Runner-Up Award in the 16th International Conference on Database Systems for Advanced Applications (DASFAA 2011). To learn more about Dr. Xiao-Li Li, please visit his Web page: http://www1.i2r.a-star.edu.sg/ xlli/.