Original Article

Proc IMechE Part C:
J Mechanical Engineering Science
0(0) 1–14

DOI: 10.1177/0954406211429764
pic.sagepub.com
SAGE

Thermal anomaly detection in datacenters

Yang Yuan[1,*], Eun Kyung Lee[2], Dario Pompili[2] and Junbi Liao[1]

Abstract

The high density of servers in datacenters generates a large amount of heat, resulting in the high possibility of thermally anomalous events, i.e. computer room air conditioner fan failure, server fan failure, and workload misconfiguration. As such anomalous events increase the cost of maintaining computing and cooling components, they need to be detected, localized, and classified for taking appropriate remedial actions. In this article, a hierarchical neural network framework is proposed to detect small- (server level) and large-scale (datacenter level) thermal anomalies. This novel framework, which is organized into two tiers, analyzes the data sensed by heterogeneous sensors such as sensors built in the servers and external sensors (Telosb). The proposed solution employs a neural network to learn about (a) the relationship among sensing values (i.e. internal, external, and fan speed) and (b) the relationship between the sensing values and workload information. Then, the bottom tier of our framework detects thermal anomalies, whereas the top tier localizes and classifies them. Our solution outperforms other anomaly-detection methods based on regression model, support vector machine, and self-organizing map, as shown by the experimental results.

Keywords

Thermal-anomaly detection, green datacenter, heterogeneous sensors, neural network, classification

Date received: 5 August 2011; accepted: 21October 2011

Introduction

Cloud computing has emerged as the most popular paradigm to meet the increasing demand for faster computing and high storage capacity. This popularity has resulted in an increase in heat generation in the datacenters of cloud service providers. The large-scale and high server densities in these datacenters also increase the probability of occurrence of anomalous events such as computer room air conditioner (CRAC) fan failure, server fan failure, and workload misconfiguration. Such events will lead to unexpected anomalies like thermal hotspots and fugues.[1] Thermal anomalies can be small (spanning a few servers or racks) or large (spanning many servers or racks) in scale, causing severe performance degradation of server hardware. Hence, these thermal anomalies need to be detected, classified (with respect to the anomalous events that caused them), and localized for timely remedial action.

As different anomalous events cause different scales of thermal anomalies, it is difficult to identify them using few temperature sensors. Thus, in this study, we propose a novel two-tier hierarchical neural network (NN) framework using several heterogeneous sensors organized in a network aimed at detecting, localizing, and classifying the thermal anomalies. Such heterogeneous sensors measure the internal and external temperatures and central processing unit (CPU) fan speed at each server. The bottom tier of our framework analyzes the relationship between the sensed data and workload information on a server using auto-associative neural networks (AANNs)[2] to detect small-scale thermal anomalies and also to perform a preliminary classification of anomalies based on the cause – misconfiguration or fan failure (the CRAC fan and/or server fans). Then, the top

[1]School of Manufacturing Science and Engineering, Sichuan University, People's Republic of China
[2]Center for Autonomic Computing, Department of Electrical and Computer Engineering, Rutgers University, USA
[*]Yang Yuan performed this study while visiting the NSF Center for Autonomic Computing at Rutgers University

Corresponding author:
Yang Yuan, School of Manufacturing Science and Engineering, Sichuan University, Chengdu, Sichuan 610065, People's Republic of China.
Email: yuanyang_1983@yahoo.com.cn

tier aggregates the detection results from different servers and determines whether there are small- or large-scale thermal anomalies. As all the unexpected changes of workload, heat propagation, or environmental temperature have impact on the relationship between the sensed data, the proposed method can detect various types of thermal anomalies. Furthermore, a small amount of data labeled 'anomaly' are used to validate this framework and offer some useful side information, i.e. the specific reaction of each type of data to different types of anomalies. This side information makes it possible to classify the thermal anomalies.

As the thermal anomalies break the relationship between the various workloads (the cause of heat generation) and their corresponding thermal manifestations (measured using temperature sensors) over time, the unexpected change of these relationship can represent the thermal anomalies. Hence, the relationship between the workload and their thermal manifestations can be used to detect the thermal anomalies. However, capturing the relationship encounters two challenges: (1) the relationship is too complex to be properly modeled and the model loses the generalization and (2) events such as misconfiguration, CRAC fan failure, and server fan failure rarely occur in datacenters, causing class imbalance in dataset. The class imbalance in the dataset prevents the computer to explore enough information from the data collected under abnormal circumstance.

To overcome the challenges, we propose a datacenter thermal anomaly detection method based on a machine-learning technique. Specifically, it aims at one-class classification. The machine-learning technique used to do one-class classification can overcome the afore-mentioned challenges because (1) the machine learning allows computers to evolve behaviors based on the data from the sensors when the computer cannot statistically build the model and (2) one-class classification can distinguish one class of samples (normal samples) from all other possible samples (abnormal samples), by learning from a training set labeled 'normal.' AANN is a proper tool[2] to perform one-class classification because it can remove the redundant information in the multiple related data and extract the low-dimensional structure contained in the high-dimensional data. Therefore, the AANN is used to detect the thermal anomalies.

The wireless sensor networks enable continuous monitoring and thermal profiling datacenters using compact sensors. In this research, the sensor motes are placed on the outlet of servers and top of the racks to sense the outlet temperature at the rear side of racks. The base station receives the signals from the external sensor and synchronizes them with the measured internal data such as CPU temperature and CPU fan speed.

Contributions of this article are as follows.

1. We propose a hierarchical NN framework, which enables detection and classification of small-to large-scale anomalies in datacenters using machine-learning-based technique and data obtained from a hybrid (external and internal) sensing infrastructure.
2. Not only does our framework detect hardware anomalies such as server/CRAC fan failures (as most previous works studied), but it also detects misconfiguration of servers and attacks, i.e. misplaced and illegitimate workloads.

The remaining article is organized as follows: first, the related work is introduced in section 'Related work'; the proposed method with a hierarchical framework for thermal anomaly detection is discussed in section 'Proposed solution'; the performance of the proposed method is evaluated and the impact of the feature selection on the detection performance is discussed in section 'Performance evaluation'; and finally, the completed work and the future work are discussed in section 'Conclusions and future work'.

Related work

The methods used in previous research in detecting thermal anomalies in datacenters include threshold-based, modeling-based, and machine-learning-based approaches. The simple threshold-based approach uses a/multiple threshold(s) to make datacenter operate in temperature guidelines. This method focuses on detecting hotspots by setting up thresholds and, hence, prevents servers from overheating.[3] It is worth noting that the hotspots are not equivalent to the thermal anomalies, because in our context, hotspots are the places/servers where the temperature overshoots guidelines, but thermal anomalies are (more generally) the places/servers where thermal behaviors in the datacenter are strange – specifically, where the temperature change does not follow the workloads running on the servers. Modeling-based approach aims at profiling a thermal map depending on the layout of datacenter, the workload distribution, and the cooling setting. Then, it detects thermal anomalies by evaluating deviations of the estimated temperatures (from the thermal map) from actual temperatures.[1,4,5] Machine-learning-based approach is used to learn thermal behaviors in datacenters by training and compare the results with the actual temperatures to detect the anomalies.

In the study by ASHRAE Technical Committees,[6] the thermal guidelines for datacenter were proposed (i.e. temperature, humidity) to operate datacenters. The similar threshold-based approach[7,8] monitored the measured temperature and detected anomalies if the temperature

exceeds the guideline thresholds. However, this method cannot follow unique environmental changes in different datacenters due to the fixed thresholds.

Modeling-based approach models the environment (i.e. workload distribution, the layout of the datacenter, and the cooling system parameters) and estimates the temperature based on simulations. The results of the simulations are compared with the actual temperature to detect anomalies. The research described in Romadhon et al.,[9] Wang et al.,[10] and Tang[11] showed the performance of using a modeling tool, computation fluid dynamics tool, to predict the temperature. However, it requires extreme computation time because the modeling is computationally intensive. Another modeling-based method was to employ a regression model to estimate temperature using historic data.[12] The regression model-based approach in Haaland et al.[12] is not computationally intensive, but modeling the relationship between the thermal features affecting thermal changes is hard with this method because regression uses one feature to interpolate data points.

The machine-learning approach aimed at detecting anomalies by learning relationships among the thermal features, and learning whether the relationships are normal or abnormal by labeling. In the study of Moore et al.,[4] a datacenter is divided into contiguous blocks, and in each block, NN is used to learn thermal changes in datacenters and predict the thermal changes with inputs (the workload and power of CRAC unit) and output (outlet temperature). The differences between predicted and actual temperatures are used to detect the thermal anomaly. However, the granularity of the block is too large to detect and localize small-scale anomalies. In Wang et al.,[13] a NN was designed and with the workload, the temperature at time t as input and that at time $t+1$ as an output. However, this approach also requires a stable environment and the prediction mechanism cannot be adapted by various workloads in datacenters.

In the works of Depren et al.[14] and Ma et al.,[15] other learning techniques such as self-organizing map (SOM) and one-class support vector machine (SVM) are applied in detecting network anomalies. However, only network intrusion was considered in this article and no application of SOM and one-class SVMs on thermal anomaly detection was discussed. The AANN used in Marwah and Sharma[1] is a promising machine-learning technique since it can explore the low-dimensional structure contained in the multiple features collected in datacneters.[16,17]

Background of AANN

AANN is referred to as a multi-layer NN because AANN is composed of multiple layers of nodes connected each other. In NN architecture, each connection between the neighboring layers has a weight that scales data passing through it. Output data from the first layer (input layer) are inserted as inputs of the consecutive next layers (hidden layers). Then, nodes in the next layers sum the data fed to them and scale the data using a 'squashing' function and process them until data reach the last layer (output layer).[17]

There are two phases used to do one-class classification: training and testing periods. Training phase is used to train the AANN and testing phase to apply the trained AANN for real-time detection. During the training phase, each pair of an input and target output set to the AANN is the same. The differences between the actual and the desired outputs are fed back to the AANN as inputs (hence, it is called backpropagation), and the weights are adjusted as follows

$$\omega_{ij}(k+1) = \omega_{ij}(k) - \eta \cdot \frac{\partial e_k}{\partial \omega_{ij}} \tag{1}$$

where ω_{ij} represents the weight on the connection from layer i to j and e_k is the output error of AANN changing at the kth iteration.

After the AANN is trained, actual data are inserted as inputs for testing and the reconstruction errors (error between the inputs and outputs) calculated. Generally, errors are low when data for testing belong to the same class as the data for training, and high otherwise. Also, the AANN uses low-dimensional feature vectors abstracted from high-dimensional feature vectors. In the study of Jothilakshmi et al.,[18] AANNs were used to detect anomalies to capture the low-dimensional distribution of the feature vectors. The low-dimensional distribution was represented as a certain pattern and the data deviating from this pattern were identified as the anomalies.

Proposed solution

In this section, a two-tier hierarchical NN framework is proposed to detect, localize, and classify thermal anomalies, as shown in Figure 1, because thermal change corresponding to large- and small-scale problems need to be detected, respectively, in the two tiers. The bottom tier is composed of distributed processing nodes (AANN nodes) and the top tier a few central processing nodes. In the bottom tier, each AANN is enabled to one server. The features related with the thermal change are internal temperature, external temperature, and CPU fan speed. They are measured by heterogeneous sensors and sent to each server's corresponding AANN. Each AANN node in the bottom

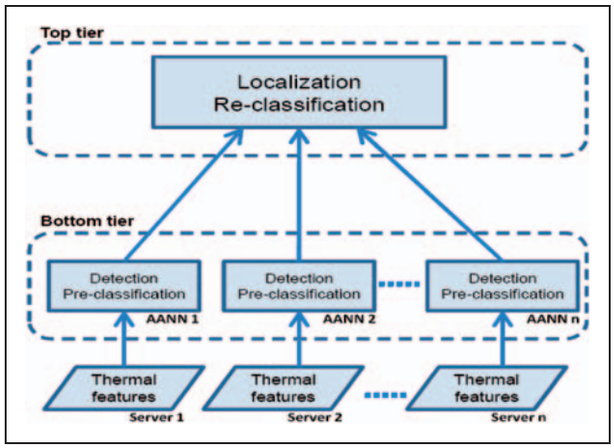

Figure 1. Information flow chart of our two-tier hierarchical NN framework. Thermal features are used to detect small-scale anomalies in the bottom tier, and the information is collected to detect large-scale anomalies in the top tier.

tier uses the multiple thermal features on a server in a specific location for training, and the trained AANN is used to detect and classify the small-scale thermal anomalies (i.e. misconfiguration of workloads and server fan failure) in small-scale. Then, the top tier aggregates information of these thermal anomalies (i.e. duration, intensity, location) from the nodes in the bottom tier, and localize and classifies the large-scale thermal anomalies (i.e. CRAC fan failure) in large-scale. An AANN node can be implemented as a part of the same server where the thermal features are collected, or it could be in a remote entity that processes thermal features depending on the processing capabilities in a datacenter.

Feature selection based on preliminary observation

We preliminarily observe data collected under 'normal' and 'abnormal' durations in our datacenter. The data under 'normal', 'misconfiguration', 'CRAC fan failure', and 'server fan failure' durations are shown in Figure 2. The preliminary observation on the dataset facilitates determining appropriate features for thermal anomaly detection.

Figure 2(a) shows how every thermal feature reacts to misconfiguration and the top subplot shows the configured workload and the workload actually running on the server. If a workload runs on a server where the workload is not supposed to run (misconfiguration), the heat mainly generated from the CPU unexpectedly changes the internal temperature, CPU fan speed, and external temperature. As the heat propagates from inside to outside, these thermal features are consecutively affected from inside to outside with different delays. Figure 2(a) also shows that the internal temperature is well related with the workload actually running. The change of internal temperature against the workload actually running can offer the intuition knowledge that the relationship between the internal temperature and the workload are sensitive to the misconfiguration. Although the workload still affects the CPU fan speed and external temperature of this server, the CPU fan speed and external temperature are also affected by the heat propagation or environmental temperature. Therefore, internal temperature is the feature most sensitive to the misconfiguration.

Figure 2(b) shows the reactions of all the features to the CRAC fan failure. The CRAC fan failure increases the temperature in and around the server. Hence, all the features are changed by the CRAC fan failure. As the external temperature is related with the heat propagation and the environmental temperature, it is the feature most sensitive to the CRAC fan failure. Internal temperature is more related with workload. Hence, its spike caused by the CRAC fan failure is not as clear as that of external temperature. The CPU fan speed is more sensitive to the heat propagation than the internal temperature is. Hence, it still facilitates the detection of the CRAC fan failure.

Figure 2(c) shows the reactions of all the features to server fan failure. Internal temperature is only sensitive to the server fan failure when the server fan fails during server-busy duration. Theoretically, the CPU fan speed and external temperature change because of the heat propagation when server fan failure occurs. However, none of them change clearly enough to be solely used for detection. Hence, all the features should be used to get combinatorial reaction and improve the accuracy of detecting server fan failure.

Based on the preliminary observation on Figure 2, anomalous events, and the features and scopes most affected by the anomalous events are summarized in Table 1. Hence, we can use the features in Table 1 to detect different types of thermal anomalies. Using these multiple features has the following advantages:

1. Information from heterogeneous sensors facilitates the detection of more types of anomalies since they have included the major features to different anomalies.
2. Although more anomalies can be detected using multiple features, false alarm rate will not increase because the multiple features can be used to get the best overall result of detection. For example, if the environmental temperature changes during normal operation, only using the external temperature is likely to produce false alarm, but internal temperature is not sensitive to normal.

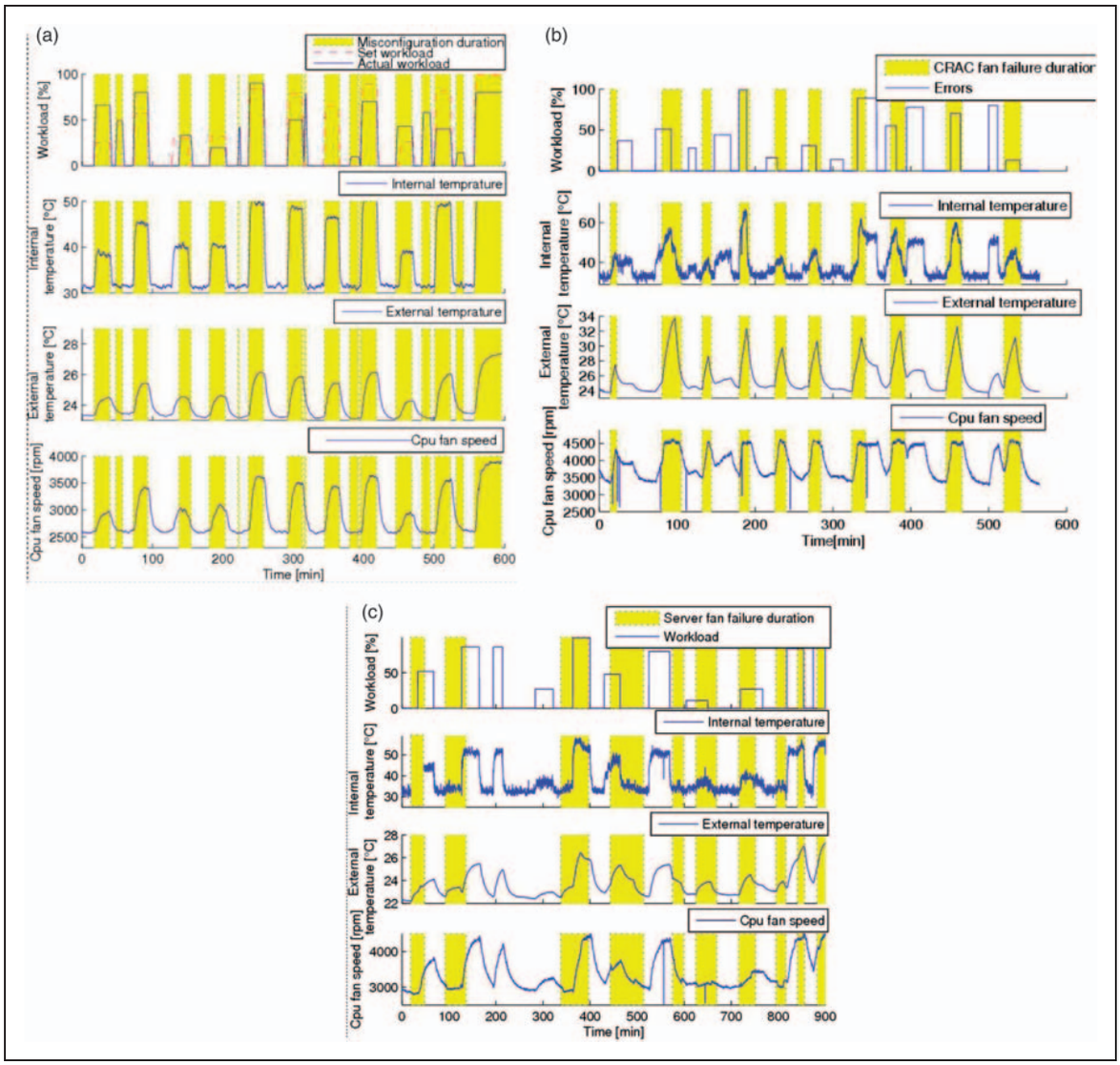

Figure 2. (a) Misconfiguration features, (b) CRAC fan failure features, and (c) server fan failure features.

Table 1. Anomalous events in datacenters and their symptoms and scopes.

Anomalous events	Features most affected by the anomalous events	Scope
Misconfiguration	Internal temperature	With a rack or multiple racks
CRAC fan failure	External temperature and CPU fan speed	CRAC's region of influence
Server fan failure	Internal temperature, CPU fan speed, and external temperature	Server

3. Environmental change only if the true anomalies occur. Hence, combining the different features can decrease the false alarm.
4. Classifying the anomalies will be simpler than using a single feature since different features have their special reactions to various types of anomalies. For example, when the workload unexpectedly changes, the CPU temperature is most heavily affected. When the fan failure occurs, the external temperature is most heavily affected as the heat propagates from inside to outside. Hence, observation on the reaction of each feature to various types of anomalies can offer some side information to classify different types of anomalies.

Small-scale thermal anomaly detection and classification

The bottom tier of our framework first detects small-scale thermal anomalies. A large amount of data should be processed, but the thermal features are often related to each other and they contain redundant information in high-dimensional space of thermal features (four dimensions in our case), which is hard to analyze. The AANN employs the high-dimensional features and maps them onto a low-dimensional subspace (two dimensions in our case). The projections of the features onto the low-dimensional subspace are called optimal features because they represent the entire data and it is easier to identify the data indicating 'normal' or 'abnormal' in low-dimensional subspace. In the bottom tier, an AANN is enabled to a server to detect thermal anomalies in small-scale.

The research described in Bianchini et al.[19] shows that as the number of the layers of the AANN increases, the complexity of AANN will increase and if the number of the layers is less than 5, the AANN can only get the optimal features in linear subspace. Hence, we designed a five-layer AANN for our solution as the five-layer architecture is sufficient to get the optimal features in nonlinear subspace and more layers will increase the complexity of training the AANN. The architecture of the five layers AANN is as shown in Figure 3. Generally, the AANN learns the relationship in the data during training phase. The data labeled 'normal' are used to train the AAAN. Besides, a small dataset labeled 'anomaly' is used to validate the trained AANN and get some side information. It is worth noting that although the amount of the data labeled 'abnormal' is not sufficient, they can be used to validate the AANN and offer some useful information for further classification.

During the training phase, the data labeled 'normal' are used as the inputs and the target outputs simultaneously for training. The four-dimensional (4D) inputs, i.e. (workload, internal temperature, external temperature, and CPU fan speed) are mapped onto two-dimensional (2D) subspace through the input, the second, and the middle layers of the AANN. The middle layer is called the bottleneck layer because it has only two nodes whereas other layers in the AANN have more nodes than the bottleneck layer. Then, the data in the 2D subspace are mapped onto 4D space through the fourth and the output layers. For every iteration, when training samples are inserted for training, the AANN calculates the error between its input and its output and adjusts its weights based on the error with equation (1) to minimize the error between the next input and the next output of AANN. The errors between the inputs and the outputs are called reconstruction errors because they indicate the ability of the AANN to reconstruct its inputs belonging to certain class. Reconstruction errors are calculated by equation (2) as

$$e_k = \frac{\sum_{i=1}^{m} (\mathrm{In}_k^i - \mathrm{Out}_k^i)}{m}, k \in \{1, \ldots, n\} \tag{2}$$

where n is the number of samples and m the number of features which is 4 in our case.

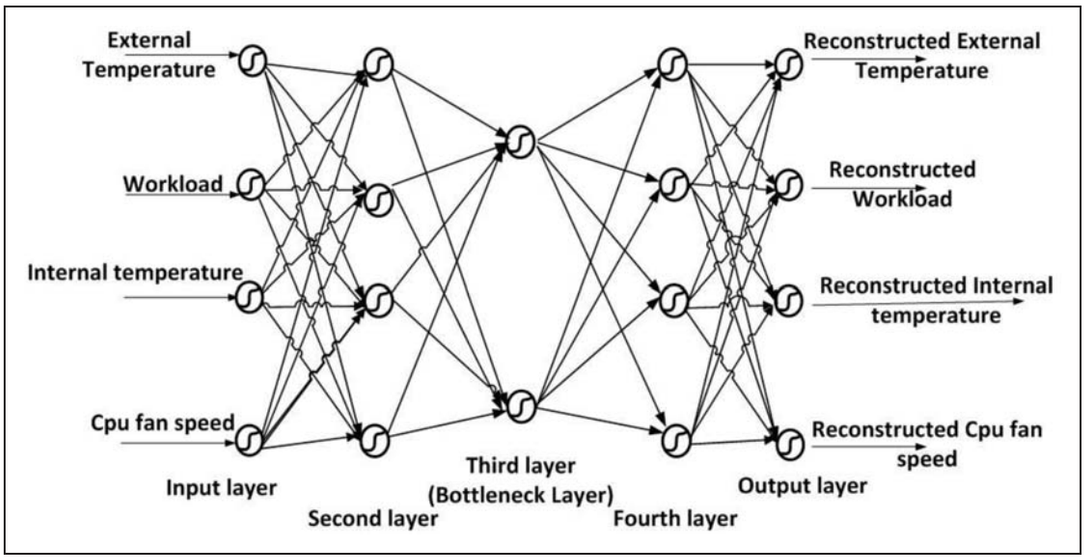

Figure 3. A five-layer AANN.

The two nodes in the bottleneck layer map the data from the 4D space onto 2D subspace and the 2D data contain the most significant information of the input since the 2D data need be used to reconstruct the input in the output layer. When the reconstruction error is minimized, it means that the 2D data have contained enough abstracted information from inputs and they can be used to reconstruct the input in the output layer. Therefore, the 2D data are the optimal features in low-dimensional subspace extracted from high-dimensional space. In this way, the AANN is adapted to reconstruct the input in its output layer when the inputs are collected under normal circumstances.

After the AANN is trained well, the samples for training are input to the AANN again to represent the ability of the AANN to capture the pattern of the data labeled 'normal'. The projections of data for training onto 2D subspace are represented with the optimal features, as shown in Figure 4(a). All the optimal features are organized with a certain trend. It indicates that the AANN has captured the pattern contained in the data for the sake of clearly training it. Therefore, the reconstruction errors between the inputs for training and the outputs are close to 0.

The intuition analysis about the relationship between the types of anomalies and the features most affected can give deeper knowledge about the characteristics of the features, i.e. the reconstruction errors of each feature have different reactions to various types of anomalies.

The intuition analysis about detection results according to normal and various types of abnormal duration gives the domain knowledge as follows: (1) The reconstruction errors during anomaly duration are higher than the reconstruction errors during normal duration since the weights of the AANN have been adjusted to the data for training and only the optimal features of 'normal' set can be used to reconstruct the input; (2) The reconstruction errors of workload and internal temperature are higher than those of other features when the thermal anomalies are caused by misconfiguration. The reason is that when the misconfiguration occurs, the internal temperature unexpectedly changes and the relationship between the configured workload and the measured internal temperature clearly deviates from that during the normal duration; (3) The reconstruction errors of external temperature are higher than the reconstruction errors of workload and internal temperature when fan failure events occur. The reason is that when the fan failure occurs, the external temperature increases more clearly than other features because it is affected by the environmental temperature or heat propagation from the inside to outside of the server; and (4) The thermal change in certain server caused by the server fan failure does not have heavy impact on the thermal change of its neighboring servers.

The training set and a small set of data labeled 'abnormal' are used to validate the mentioned domain knowledge. The domain knowledge is illustrated in Figure 4. Figure 4(a) indicates that any optimal feature which deviates from the pattern extracted from the training set is corresponding to the 'abnormal' situation. The reconstruction errors according to abnormal situation are clearly high and a threshold can be set based on the maximum reconstruction errors and relaxed later to decrease the false alarm. An adaptive threshold selection is used to select the maximum error as initial threshold and expand it with the standard deviation error during test. *Threshold* at time t is set as follows

$$Threshold = \max(e_n) + std(e_{1,2,\cdots,t}), t > n \tag{3}$$

When any sample whose reconstructed error e through AANN is higher than *Threshold*, it is detected as an 'anomaly'.

Although the adaptive threshold selection can decrease the false alarm caused by the normal environmental change, the AANN will still outdate periodically. This fact requires AANN to be periodically retrained with recent data. Retraining of the NN is implemented by Algorithm 1.

Figure 4(b) shows that the reconstruction errors of the external temperature are the highest when CRAC fan failure occurs and the reconstruction errors of the

Algorithm 1. Retraining using intensity of potential risk of exceeding the threshold.

INIT PARAMETER DECISION:
e_k = {the kth member in the list of the reconstruction errors($1 \times n$)}
Threshold = {upper limit, the threshold to detect the thermal anomaly}
a = {the factor multiplied with the threshold to generate a lower limit}
b = {the counter to count the numbers of the errors consecutively higher than the lower limit}
c = {the indicator to decide when to retrain the AANN}
Initialize parameters: $a = 0{:}96$, $b = 0$, $c = 100$
RETRAINING DECISION:
1: for $k = 1 \rightarrow n$ do
2: if $e_k > a \times$ *Threshold* and $e_k <$ *Threshold* then
3: if $b < c$ then
4: $b \Leftarrow b + 1$
5: else
6: retrain the AANN with recent 2×10^4 data
7: end if
8: else if $e_k \geq$ *Threshold* then
9: thermal anomaly occurs
10: else
11: $b \Leftarrow 0$
12: end if
13: end for

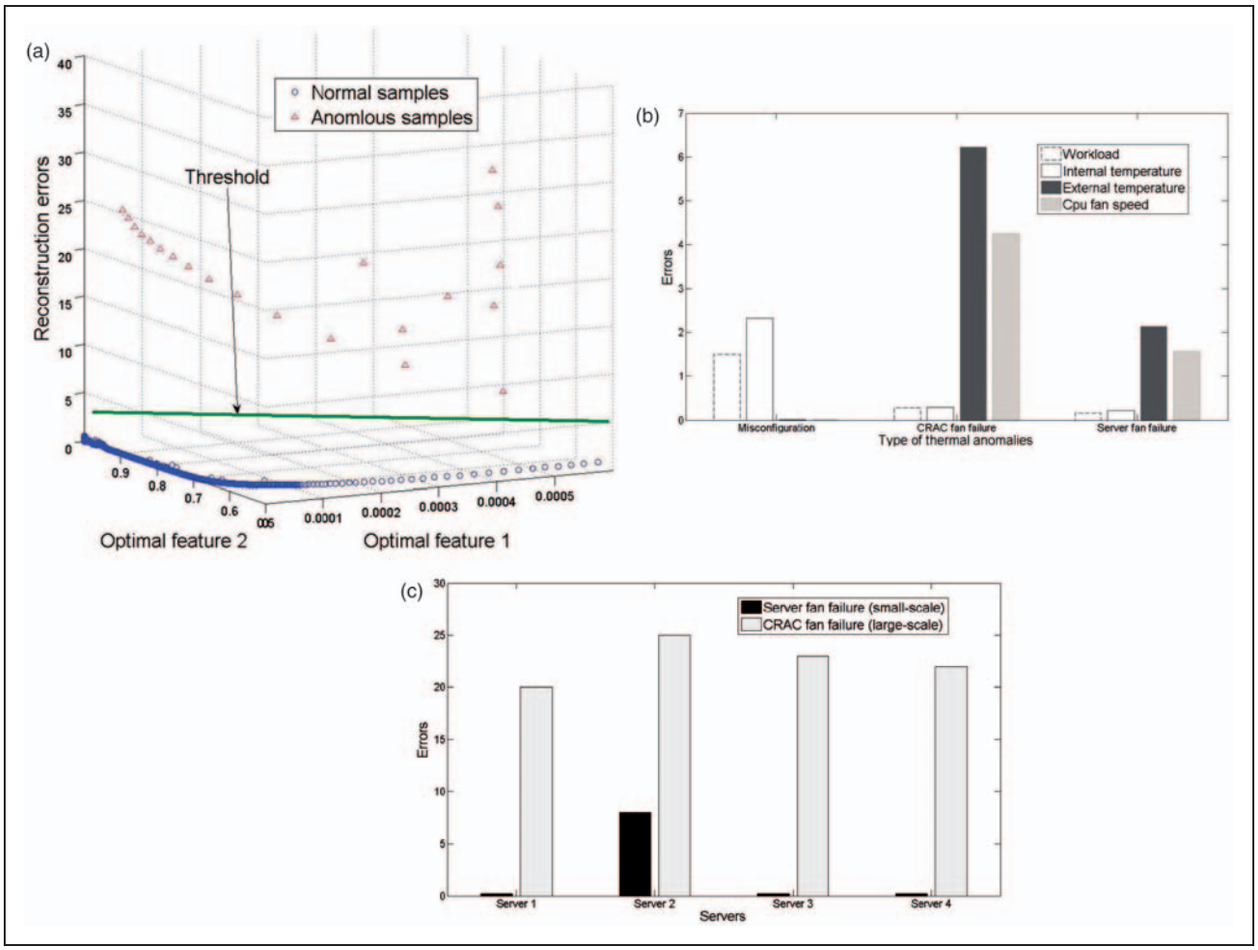

Figure 4. (a) Projections of features onto 2D space and their errors, (b) reconstruction errors of different inputs to AANN against different types of anomalies, and (c) reconstruction errors of the AANNs of neighboring servers.

workload and internal temperature are higher than those of external temperature when misconfiguration occurs. This domain knowledge can work as a intuition proof to separate the misconfiguration from fan failure events (CRAC fan failure or server fan failure) although the data labeled 'anomaly' are insufficient for training.

Large-scale thermal anomaly detection and classification

The top tier distinguishes the small-scale fan failure anomaly (server fan failure) and the large-scale fan failure anomaly (CRAC fan failure). Figure 4(c) shows that the reconstruction errors of AANNs enabled to neighboring servers are in the similar level when CRAC fan failure occurs and the reconstruction errors of AANN enabled to certain server are higher than those of its neighbor servers' when server fan failure occurs in that server. The reason is that the server fan failure does not affect its neighboring server. This domain knowledge is as shown in Figure 4(c). The location of the server fan failure will be recorded if it is indicated that the server fan failure occurs and the range of the anomalies will be recorded if the CRAC fan failure occurs.

Performance evaluation

In real datacenters, as the proposed method aims at detecting both the small- and large-scale thermal anomalies, one AANN is enabled to each server. The back-propagation algorithm is adopted during training.

The experiments were implemented in the datacenter of Rutgers University. The training efforts increase while the datacenter size becomes larger. However, the training efforts do not linearly increase against the datacenter size. The reason is that algorithm 1 is implemented for different AANNs and the AANNs are not activated for training simultaneity. Hence, the

increments of the 'training' efforts are smaller than the increments of datacenter size. In our experiments, 2×10^4 data were used for each time of training. The servers are mounted onto 19-inch racks. For thermal anomaly detection, the sensors and the sensed features are given in Table 2. Figure 5(a) shows the deployment of 11 nodes of Telosb motes of the wireless sensor network in the datacenter. The top mote works as a base station and the other motes as sensor motes. Besides the external sensors, the internal sensors which are built-in components in each server are also used. The external sensors at the outlet sense the outlet temperature which is mainly affected by the heat propagation and environmental temperature. The location of fan around the heated elements is shown in Figure 5(b), and the internal temperature sensors are located under the fan. The internal sensors sense the CPU temperature (internal temperature) and CPU fan speed which are mainly affected by the workload and heat propagation. To combine the features sensed by the internal and external sensors, a server is connected to the base station and the internal features are consecutively sensed and sent to the server connected to the base station. The internal information is recorded in the server connected to the base station. The signals containing the outlet temperature are sent to the base station every 3 s via wireless sensor network. Once the base station receives the signal from the external sensors, the server connected to it reads the outlet temperature from the signal and the latest internal information from the record. In this way, the external and internal features are combined together for analysis.

The noise contained in the raw data degrades the classification performance and the data need be preprocessed to filter the noise. The moving average technique is used here where the moving window size is 50, i.e. the window time is 150 s. To enhance the accuracy of AANN, we normalized the input and output instances to [0; 1] by the following equation

$$f(x_i) = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \tag{4}$$

where x_i is the variable in the dataset at the ith iteration and $x_{\min}$ and $x_{\max}$ are the minimum and maximum of the dataset.

Table 2. Sampled features and their corresponding sensors.

Features	Sensors used to collect different types of data
Outlet temperature (°C)	Sensors located at the outlet of the rack
CPU temperature (°C)	Built-in sensors on the CPU
CPU fan speed (r/min)	Sensors on the motherboard

Result of thermal anomaly detection

The anomalous events in Table 1 are generated. The events are insufficient for training and only used to evaluate the performance of the thermal anomaly

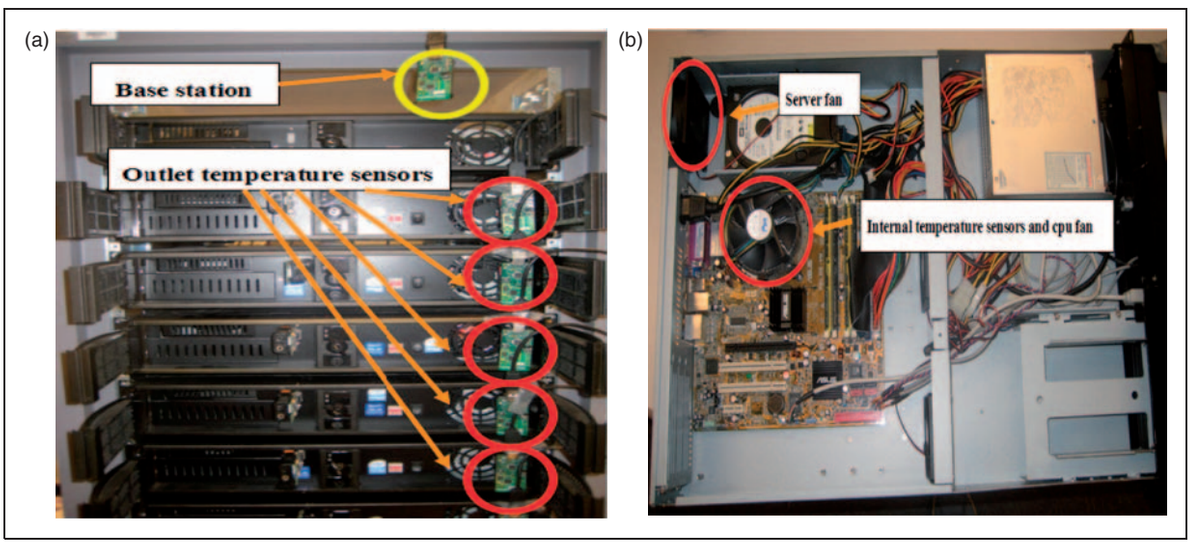

Figure 5. Location of sensors and fans: (a) wireless sensor network monitoring the outlet temperature of servers and (b) internal temperature sensors and fans around the heated elements on the motherboard.

detection. The experimental results are shown in Figure 6: (1) misconfiguration events were generated by running the workload different from the configured workload on certain servers. Figure 6(a) shows that the reconstruction errors of AANN in the misconfiguration duration are clearly higher than those in normal duration; (2) CRAC fan failure was generated by turning off the CRAC fan or decreasing its speed. Figure 6(b) shows that the reconstruction errors of AANN in the CRAC fan failure duration are clearly higher than those in the normal period and most reconstruction errors caused by CRAC fan failure are higher than those caused by misconfiguration events; and (3) fan failure was generated by stopping the running of the server fan. Figure 6(c) shows that the reconstruction errors of AANN in the server fan failure duration are higher than those in the normal duration and it is hard to separate the server fan failure events from the misconfiguration events only with the mean square errors because some reconstruction errors caused by server fan failure are close to those caused by misconfiguration.

Comparison using receiver operating characteristic

Receiver operating characteristic (ROC),[20] which is a popular classification performance metric, is applied to evaluate the performance of anomaly detection. ROC was originally applied within the medical field. Hence, the samples representing 'abnormal' and 'normal' behaviors are referred to as the positive and negative samples. The ROC curve heavily relies on notations as sensitivity and specificity which are used to calculate the different measurements of the quality of the test. The ROC parameters, i.e. TP, FP, TN, and FN are defined in Table 3. The sensitivity (SE), as calculated in equation (5), also called true positive rate (TPR), is the probability of having a positive test among the samples which have positive diagnosis. The ROC curve has the sensitivity plotted vertically and has the horizontal axis called the false positive rate (FPR) as calculated by equation (6)

$$\mathrm{SE} = \mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{5}$$

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{6}$$

The sensitivity and specificity are calculated against each threshold and the resulting points are plotted as a ROC curve. The research on ROC has indicated that the larger the area under ROC curve (AUC) is, the better the performance of the detection is.[20]

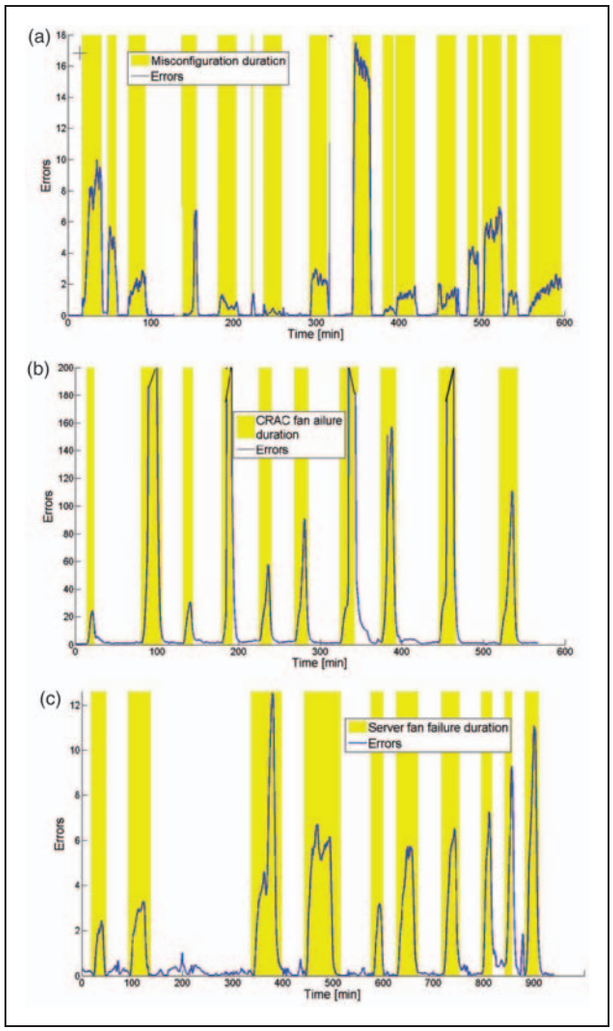

Figure 6. (a) Reconstruction errors during misconfiguration duration, (b) reconstruction errors during CRAC fan failure duration, and (c) reconstruction errors during server fan failure duration.

Table 3. Definition of ROC parameters.

		Actual label	
		Positive	Negative
Estimated label	Positive	TP	FN
	Negative	FP	TN

Note: TP: true positive; FP: false positive; FN: false negative; and TN: true negative.

Impact of uncertainty in sensor readings on the detection performance. As the outlet temperature is sensed by the temperature sensors on the Telosb mote and CPU temperature is read by the sensors attached

to the CPU, the causes of uncertainty in sensor readings in the datacenter can be summarized into three groups:

1. The disturbance from the external environment, such as electromagnetic interference, change of environmental temperature, or opened door of datacenter.
2. The degraded stability of sensors caused by the thermal drift of sensors.
3. Noise caused by random errors.

Generally, the uncertainty would degrade the detection performance, i.e. decrease the TPR and increase the FPR. Hence, some techniques are used to minimize the impact of these kinds of uncertainties on the thermal anomaly detection. Moving-average technique is used to remove the noise in the raw data and minimize the effect of the noise. Also, the impact of the thermal drift of sensor can be removed by calibration. The impact of environmental temperature can be minimized using the re-training algorithm 1 since the scheme can adapt itself to the environment. The case that the door of datacenter is opened mainly affects the region where temperature is heavily affected by the air from outside. The electromagnetic interferences from power switch, transformer, or other equipments make the sensor read the data incorrectly or miss some data. However, during the training phase, the uncertainty caused by opened door or electromagnetic interference can be minimized using a large amount of data for training because these two cases can be maintained only for a short time and other data collected under 'normal' circumstance can make the AANN only adapt to the significant 'normal' pattern in the whole dataset. The FPR will increase when these two cases occur in test phase. Hence, these two cases should be carefully concerned at the design stage of the datacenter.

Comparison between the results with different features. Each type of anomaly has different impacts on the various features. Hence, different scenarios in Table 4 are used to validate this domain knowledge.

Figure 7(a) shows that the ROC curves for the detection of misconfiguration in scenarios 1 and 2 outperform that in scenario 3. Using only the external temperature, AUC is obtained as 0.92. However, by introducing the internal temperature, it is much better. Using CPU fan speed does not make much sense after the internal temperature has been used. The reason is that the internal temperatures are most sensitive to the workload change and therefore change the CPU fan speed. After the internal temperature is introduced, the AANN treats the relationship between workload and CPU fan speed as redundant information.

Figure 7(b) indicates that the thermal anomaly caused by CRAC fan failure changes the various

Table 4. Scenarios with different features.

Scenario 1	Detection using external temperature, internal temperature, and CPU fan speed
Scenario 2	Detection using external and internal temperatures
Scenario 3	Detection only using external temperature

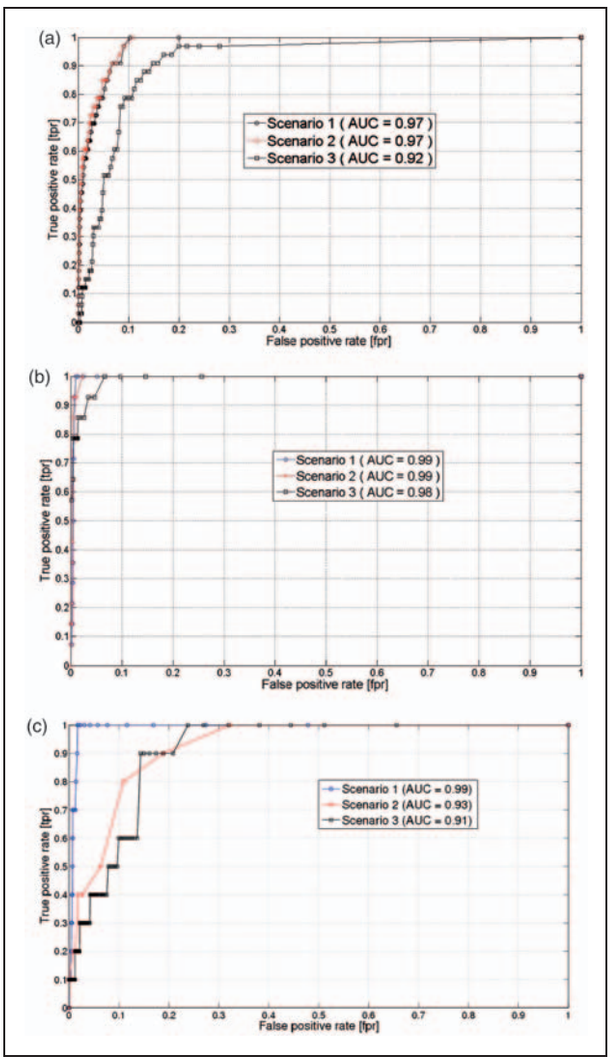

Figure 7. (a) ROC curve of the misconfiguration detection results with AANN using different features, (b) ROC curve of CRAC fan failure detection results with AANN using different features, and (c) ROC curve of server fan failure detection results with AANN using different features.

features clearly enough and the detection performance in any scenario gets the AUC close to 1.0. The external temperatures are impacted most heavily by the CRAC fan failure because it is affected by not only the heat propagation but also the environment around the servers.

Figure 7(c) shows that the performance of detecting server fan failure is not highly improved when only certain feature is used because the server fan failure mainly affects the heat transferring from the inside to the outside of the server. Hence, using all the features related with the heat transfer improves the performance of detecting server fan failure.

Figure 7 indicates that using the multiple features highly improves the detection performance. Furthermore, using the multiple features does not require additional work to be done to adjust the architecture of AANN to detect the various anomalies.

Comparison between different methods. We compared our method with the following methods: (1) regression model-based method: regression model is a function which can adjust its parameters based on the historical workload and estimate the next features such as external temperature, internal temperature, and CPU fan speed based on their historical data. If the differences between the measured features and the estimated features are higher than the threshold, the anomalies are detected; (2) SOM is a competitive network composed of components called neurons. At each epoch, the neurons whose weight is nearest to the input for training will become winner and its weight will be adjusted so that the similar neurons form a cluster. If the distance between project of the measured data and the center of the cluster is longer than a certain threshold, it is classified as an anomaly. We decide the threshold by calculating the mean value of distances between each datum and the center of the cluster and relax it by 75%; (3) one-class SVM: it transforms the feature into higher dimensional space via the kernel function and separates the different kinds of data with the hyperplanes. Hence, the hyperplane works as a boundary between the data indicating 'normal' and 'abnormal' situations. The kernel function used here is the radial basis function since it can classify the data in high-dimensional space even if the data are not linearly separable in the high-dimensional space.

AANN-based detection in the proposed method, regression model-based detection, one-class SVM-based detection, and SOM-based detection are compared on detecting the misconfiguration. Figure 8(a) shows that the regression model-based method gives superior performance compared to other methods in detecting misconfiguration. By analysis, the reason obtained is that the relationship between the features and the workload broken by misconfiguration is clear enough for statistical modeling.

One-class SVM performs the worst in misconfiguration detection since the mapping of data from low-dimensional to high-dimensional space introduces more redundant information which is not appropriate for the thermal anomaly detection in datacenters.

Figure 8(b) shows that AANN-based detection in the proposed method outperforms other methods in detecting CRAC fan failure. Regression model-based detection gets the worst performance since it is susceptible to both temperature change caused by the CRAC fan failure and normal change of environmental temperature, which gives it the highest FPR.

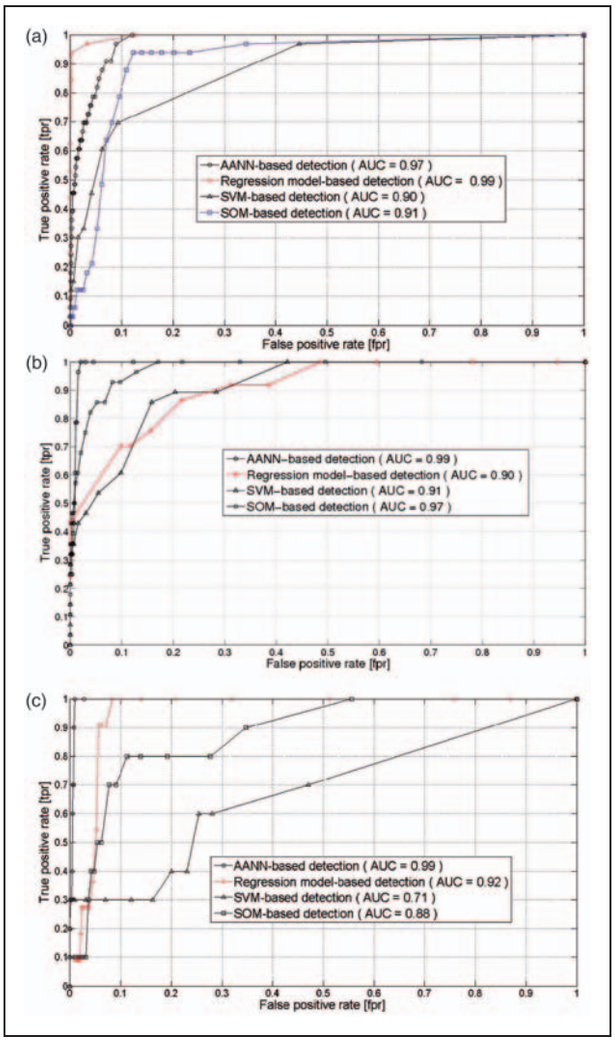

Figure 8. (a) ROC curve of the misconfiguration detection result with different approaches, (b) ROC curve of CRAC fan failure detection result with AANN using different approaches, and (c) ROC curve of server fan failure detection result with AANN using different approaches.

Figure 8(c) shows that the proposed ANN-based method gives superior performance compared to the other methods in detecting server fan failure because it captures the implicit changes of the relationship between different features correctly and gets lowest FPR. SOM-based detection performs the worst since there may not be any appropriate knowledge discovered in the features for competitive learning.

The result indicates that AANN-based method performs the best in detecting fan failure anomalies and regression model-based method the best in detecting misconfiguration anomalies. The reason is that the change of the external temperature caused by the misconfiguration is stable enough for modeling and the regression model-based method can clearly separate the normal duration and misconfiguration anomalies. However, when CRAC fan failure occurs, the changes of the relationship among the workloads and the features take place too implicitly to be modeled with regression. Another drawback of the regression model-based method is that it is susceptible to the environmental temperature. This fact indicates that the AANN-based method is appropriate for anomaly detection and it may be improved when combined with regression model-based method.

Comparison between one-class and multi-class classification-based detections. Misconfiguration, server fan failure, and CRAC fan failure are rare events in the datacenters and the amount of the data collected in each events duration is not enough for training, i.e. it cannot accurately detect the anomalies. Multi-class classification-based detection uses workload and features labeled 'normal' and 'abnormal,' i.e. external temperature, internal temperature, and CPU fan speed as inputs and the combination of indexes 1 and 2 in Table 5 as output.

Table 5. The meaning of the output of the multi-class classification NN.

[Index 1 Index 2]	[0 0]	[1 0]	[0 1]	[1 1]
Classification result	Normal	Misconfiguration	CRAC fan failure	Server fan failure

Note: CRAC: computer room air conditioner.

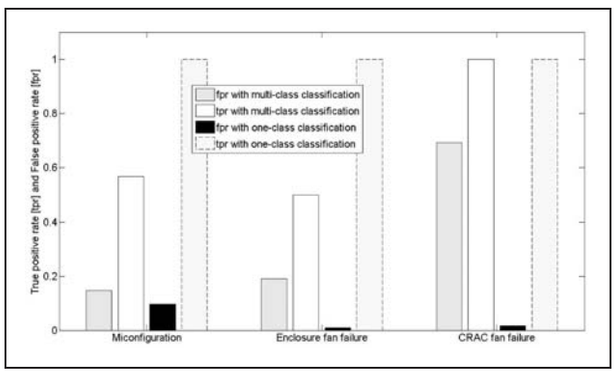

Figure 9. Performance comparison between multi-class and one-class classification-based detections.

The proposed method is compared with the multi-class classification-based detection in Figure 9. It is shown that the multi-class classification-based detection has higher FPR than the proposed method.

Conclusions and future work

In our solution, a two-tier hierarchical NN framework is proposed to detect thermal anomalies. The features to the framework, i.e. internal temperature, external temperature, and CPU fan speed are sensed by heterogeneous sensors. The framework extracts the relationship between the features and the thermal change with the AANN to detect the small-scale thermal anomaly at the bottom tier and detect the large-scale thermal anomaly at the top tier. The experiment results show that the proposed method outperforms regression model-based, SVM-based, and SOM-based methods by at most 5%, 21%, and 11%, respectively. The experimental results also show that the proposed method outperforms the multi-class classification-based detection when both the methods are given optimal thresholds. Furthermore, the detection results give a promising method for anomaly classification in future work, i.e. the root cause of thermal anomalies can be identified based on the relationship between the reconstruction errors of the AANN and the types of thermal anomalies. This anomaly classification will be more practical than the multi-class classification since the proposed anomaly classification does not need the large amount of data according to each type of thermal anomaly for training.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

1. Marwah M and Sharma RK. Autonomous detection of thermal anomalies in data centers. In: *Proceedings of the InterPACK Conference (IPACK)*, San Francisco, CA, New York: ASME, paper no. Inter PACK2009-89140, 19–23 July 2009, pp. 777–783.
2. Herrero A, Corchado E and Gastaldo P. Auto-associative neural techniques for intrusion detection systems. In: *Proceedings of the IEEE International Symposium on*

Industrial Electronics (ISIE), Vigo, Spain, Piscataway, NJ: IEEE, 4–7 June 2007, pp. 1905–1910.
3. Wang XD, Wang XR, Xing G, et al. Towards optimal sensor placement for hot server detection in data centers. In: *Proceedings of the IEEE International Conference on Application-specific Systems (ICDCS)*, Minneapolis, MN, Piscataway, NJ: IEEE, 20–24 June 2011, pp. 899–908.
4. Moore J, Chase JS and Ranganathan P. Weatherman: automated, online, and predictive thermal mapping and management for data centers. In: *Proceedings of the IEEE International Conference on Autonomic Computing (ICAC)*, Dublin, Ireland, Piscataway, NJ: IEEE, 12–16 June 2006, pp. 155–164.
5. Sharma F, Shih R, Patel C, et al. Application of exploratory data analysis (EDA) techniques to temperature data in a conventional data center. In: *Proceedings of the InterPACK Conference (IPACK)*, British Columbia, Canada, New York: ASME, paper no. IPACK2007-33700, 19–23 July 2009, pp. 851–857.
6. ASHRAE Technical Committees. *Thermal guidelines for data processing environments*. Atlanta, GA: American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), 2004.
7. Patterson MK. Towards efficient supercomputing: a quest for the right metric. In: *Proceedings of the IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, Denver, CO, Piscataway, NJ: IEEE, 4–8 April 2005, pp. 1–8.
8. Greenberg S, Mills E and Tschudi B. Best practices for data centers: lessons learned from benchmarking 22 data centers. In: *Proceedings of the American Council for an Energy-Efficient Economy (ACEEE)*, Pacific Grove, CA, Washington, DC: ACEEE, 13–18 August 2006, pp. 76–87.
9. Romadhon R, Ali M, Mahdzir AM, et al. Optimization of cooling systems in data centre by computational fluid dynamics model and simulation. In: *Proceedings of the Innovative Technologies in Intelligent Systems and Industrial applications (ITISIA)*, Kuala Lumpur, Malaysia, Piscataway, NJ: IEEE, 25–26 July 2009, pp. 322–327.
10. Wang L, Laszewski GV, Dayal J, et al. Thermal aware workload scheduling with backfilling for green data centers. In: *Proceedings of the IEEE International Performance of Computing and Communications Conference (IPCCC)*, Scottsdale, AZ, Piscataway, NJ: IEEE, 14–16 December 2009, pp. 289–296.
11. Tang Q, Gupta SKS, Stanzione D, et al. Thermal-aware task scheduling to minimize energy usage of blade server based datacenters. In: *Proceedings of the IEEE International Symposium on Dependable, Autonomic and Secure Computing (DASC)*, Indianapolis, IN, Piscataway, NJ: IEEE, 29 September–1 October 2006, pp. 195–202.
12. Haaland B, Min W, Qian PZG, et al. A statistical approach to thermal management of data centers under steady state and system perturbations. *J Am Stat Assoc* 2010; 105(491): 1030–1041.
13. Wang L, Laszewski GV, Dayal J, et al. Task scheduling with ANN-based temperature prediction in a data center: a simulation-based study. *Eng Comput* 2011; 37(2): 1–11.
14. Depren O, Topallar M, Anarim E, et al. An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. *Expert Syst Appl* 2005; 29(4): 713–722.
15. Ma J, Dai G and Xu Z. Network anomaly detection using dissimilarity-based one-class SVM classifier. In: *Proceedings of the International Conference on Parallel Processing Workshops (ICPPW)*, Vienna, Piscataway, NJ: IEEE, Austria, 22–25 September 2009, pp. 409–414.
16. Bratina B, Mukinja N and Tovornik B. Design of an auto-associative neural network by using design of experiments approach. *Neural Comput Appl* 2010; 19(2): 207–218.
17. Kramer MA. Nonlinear principal component analysis using autoassociative neural networks. *AIChE* 1991; 37(2): 233–243.
18. Jothilakshm SI, Ramalingami V and Palanivel S. Speaker diarization using autoassociative neural networks. *Eng Appl Artif Intell* 2009; 22(4–5): 667–675.
19. Bianchini M, Frascon P, Stanzione ID, et al. Learning in multilayered networks used as autoassociators. *IEEE Trans Neural Networks* 1995; 6(2): 512–515.
20. Kerekes J. Receiver operating characteristic curve confidence intervals and regions. *IEEE Geosci Remote Sens Lett* 2008; 5(2): 251–255.

Appendix

Notation

a	the factor multiplied with the *Threshold* to generate lower threshold as a lower limit
b	a parameter to count the numbers of the errors consecutively higher than the upper limit and lower than the lower limit
c	the indicator to decide when to re-train the AANN
e_k	the kth mean reconstruction error between input and output of the AANN
$f(x_i)$	the function to make moving average on the dataset x
FN	false negative
FP	false positive
FPR	false positive rate
In_k^i	the kth value of the ith feature input to the AANN
m	dimension of features
n	numbers of training epochs
Out_k^i	output of AANN when the kth value of the ith feature as the input
SE	sensitivity represented by TPR
Threshold	threshold to detect the anomalies
TN	true negative
TP	true positive
TPR	true positive rate
$x_{\min}$	the minimum value of x
$x_{\max}$	the maximum value of x
ω_j^i	the weight on the connection from ith layer to jth layer