https://dx.doi.org/10.1016/j.rser.2023.113612

Similarity learning-based fault detection and diagnosis in building HVAC systems with limited labeled data

Zhe Chen[1], Fu Xiao[1, 2, *] and Fangzhou Guo[1]

[1] Department of Building Environment and Energy Engineering, The Hong Kong Polytechnic University, Hong Kong, China
[2] Research Institute for Smart Energy, The Hong Kong Polytechnic University, Hong Kong, China

Abstract

Machine learning has been widely adopted for fault detection and diagnosis (FDD) in heating, ventilation and air conditioning (HVAC) systems over the past decade due to the ever-increasing availability of massive building operational data. Machine learning-based FDD is flexible and accurate but heavily relies on the availability of sufficient labeled data to develop supervised or unsupervised models. However, collecting labeled data is usually labor-intensive for various types of faulty conditions, significantly limiting the practical implementation of machine learning-based FDD. Therefore, this study proposes a similarity learning-based method using Siamese networks to improve the performance of machine learning-based FDD in applications with limited labeled data. Unlike the conventional supervised approach, the proposed Siamese networks contain two identical LSTM subnetworks which take a pair of multivariate time-series samples from the building energy management system as input. The number of training samples can be significantly augmented by generating pairs randomly. In this way, the generalization ability of the machine learning-based FDD is significantly improved in practical applications. Two case studies were designed and conducted using experimental data when labeled data were limited and imbalanced to validate the proposed similarity learning-based method. In case 1, the proposed method improves the fault diagnostic accuracy by at most 45.7% compared with the baseline model when the number of labeled data is limited. In case 2, the proposed method demonstrated better generalization ability when the labeled data is imbalanced.

* Corresponding author. E-mail address: linda.xiao@polyu.edu.hk

Highlights:

- A similarity learning-based approach is proposed for fault detection and diagnosis in building HVAC systems with limited labeled data.
- The proposed method can augment the number of training samples by generating input pairs from labeled data to improve the model generalization ability.
- Two case studies were conducted to validate the proposed similarity learning-based method when labeled data are limited and imbalanced.
- The fault diagnostic accuracy can be improved by up to 45.7% when the labeled data is limited using the proposed method.

Keywords: Heating, ventilation and air conditioning systems; Fault detection and diagnosis; Building energy management; Similarity learning; Deep learning

Work Count: 6291

Nomenclature

Abbreviation

AHU	Air handling unit
ASHRAE	American society of heating, refrigerating and air-Conditioning engineers
FDD	Fault detection and diagnosis
GAN	Generative adversarial network
HVAC	Heating, ventilation and air Conditioning
LSTM	Long short-term memory
PIR	Performance improvement rate
SVM	Support vector machine

Symbols

b	Biases in neural networks
c_t	Cell state at time step t
$class(\cdot)$	Classification function
f_t	Forget gate at time step t
$f(\cdot)$	LSTM subnetwork
F_i	Fault label
h_{t-1}	Hidden state at time step $t-1$
i_t	Input gate at time step t
m	Window size
n	Number of features
N_t	Number of samples per class in the training set
N_s	Number of samples per class in the support set
o_t	Output gate at time step t
$ReLU(\cdot)$	Rectified linear unit function
$sim(\cdot)$	Similarity function
$tanh(\cdot)$	Hyperbolic tangent function
W	Weights in neural networks
X_i	Input sample
$\sigma(\cdot)$	Sigmoid function
$\odot$	Element-wise multiplication

1. Introduction

The building sector has become one of the world's leading carbon emitters and energy consumers, accounting for approximately 40% of global total energy consumption [1]. Therefore, advanced building energy management technologies are crucial to improving energy efficiency and reducing carbon emissions in the building sector. It is estimated that improvements in controls in commercial buildings can save approximately 4%–5% of U.S. energy consumption [2]. Regarding energy savings in the building sector, heating, ventilating, and air conditioning (HVAC) systems typically hold the most significant energy-saving potential due to their complex system interactions and various operating faults. The efficiency of HVAC systems can be significantly impacted by poor maintenance, component performance issues, and sensor errors [3]. To address

these challenges, fault detection and diagnosis (FDD) is an advanced technology embedded in building energy management systems. FDD algorithms detect operational faults and determine their causes, thereby minimizing unnecessary energy consumption. For example, FDD can reduce energy consumption by 15%–30% in air handling units (AHUs), which are essential components in HVAC systems [4].

Over the past decade, FDD in HVAC systems has been widely studied, with three main approaches: expert rule-based, physical model-based, and machine learning-based approaches [5]. Expert rule-based FDD relies on domain knowledge to construct expert rules with predefined threshold values for decision-making [6,7]. This approach is easy to develop and implement but is less accurate due to oversimplicity [8]. Physical model-based FDD heavily relies on domain knowledge to develop physical models and fault indicators to measure the difference between the measurements and the indicators predicted by the physical models [9]. Compared to the rule-based approach, the physical model-based approach is more reliable because it excels in dealing with dynamic operations. However, this approach requires detailed physical information about the system and is time-consuming to build and validate physical models, making it more information-demanding and labor-intensive. Machine learning-based FDD relies on historical data to develop supervised [10] or unsupervised models [11]. Compared to expert rule-based and physical model-based approaches, the machine learning-based approach is easier to develop because it utilizes historical operational data and does not require detailed information about the HVAC system. FDD is usually treated as a classification task in machine learning, which can be binary (for fault detection) or multi-class classification (for fault diagnosis) [12]. Binary classification detects whether a working condition is normal or faulty [13], while multi-class classification aims to determine whether the working condition is normal or suffers various faults[14,15]. The machine learning-based approach can address performance shifts of equipment during long-term operation by updating the model automatically, which is hardly achievable using expert rule-based or physical model-based approaches [16]. Previous studies have developed machine learning-based models using popular algorithms such as support vector machines [17,18], decision trees [19,20], artificial neural networks [21,22], and deep neural networks [23] for FDD in HVAC systems [5]. These studies demonstrated high accuracy and exemplified the great potential of the machine learning-based approach in FDD, given adequate labeled data, including both normal and faulty data.

However, the practical deployment of machine learning-based FDD encounters significant challenges because the labeled data is often limited in real-world scenarios. On the one hand, traditional ML algorithms usually require a large amount of labeled data to prevent overfitting due to the curse of dimensionality in machine learning (the demand for labeled data increases exponentially with dimensionality) [24]. On the other hand, faulty data usually come from limited maintenance records, which are insufficient to train and validate data-driven models. Therefore, it is important to develop a new FDD method with high generalization ability for HVAC systems in which only a few labeled samples are available. To improve the performance of machine learning-based FDD under limited labeled data, semi-supervised learning methods have recently been adopted for FDD in HVAC systems. Semi-supervised learning enhances the performance of supervised learning by leveraging large amounts of unlabeled data with high confidence scores [25]. The confidence score of an unlabeled sample is the probability associated with the output class from supervised models. In [26], the support vector machine (SVM) was employed for calculating the confidence score. If the predicted class of an unlabeled sample has a higher probability than a predefined threshold, the sample will be assigned with the class label and added to the training set for the next round of training. By adopting the semi-supervised SVM, the results showed above 80% accuracy on the test data for AHU fault diagnosis. Similarly, a generative adversarial network (GAN)-based semi-supervised learning framework was proposed to improve the chiller fault diagnostic accuracy under limited and imbalanced labeled data [27]. The proposed framework could generate artificial fault samples to balance the training data, and the fault diagnostic accuracy reached 90% when each fault type had 30 samples. Li et al. compared GAN-based semi-supervised learning with a supervised baseline [28,29], and the proposed semi-supervised method showed a 3%–10% improvement in FDD accuracy. It was also found that when the number of labeled samples decreased, the proposed method gained more accuracy improvement. Fan et al. designed case studies to test the generalization ability of semi-supervised learning in detecting unseen faults in AHU operations [4,30]. The results showed that semi-supervised learning improved the fault detection rate by about 10% when the size of labeled data is small. The performance of semi-supervised learning gradually approached the baseline with the increase of labeled data.

Similarity learning is a novel supervised learning method for classification problems to enhance the generalization ability of classification problems, and it has been successfully adopted in a

variety of FDD applications, such as bearing fault diagnosis [31,32], power system fault diagnosis [33,34], robot fault diagnosis [35,36], etc. Similarity learning measures the similarity between the new samples and labeled samples to make FDD predictions. Traditionally, the similarity of a pair of samples can be calculated in two ways. First, similarity can be calculated from the geometric differences, such as Euclidean distance [37,38], Manhattan distance [39], and cosine distance [40]. Second, correlation coefficients such as Pearson correlation coefficient [41] and Spearman's rank coefficient [42] can serve as similarity metrics. On the other hand, similarity learning learns the similarity by training neural networks in a supervised learning way. The distance-based and correlation-based similarity measures work well when used to compare univariate time-series samples. However, these similarity metrics cannot reasonably measure the similarity between two multivariate time-series samples because it is difficult to determine the weights of each variable [43]. Despite the success of similarity learning in many FDD applications, only a few studies have adopted similarity learning in building energy management. Tan et al. proposed a sensor fusion framework for detecting occupancy in residential buildings [44]. In the proposed framework, similarity learning was adopted to classify indoor image data into two classes, i.e., occupied and vacant. The results proved the value of similarity learning, especially given limited labeled data. In view of the boom of deep learning and the success of similarity learning in various industries, similarity learning is a promising tool for detecting and diagnosing faults in building HVAC systems, particularly when labeled data is limited.

Another issue in previous studies on using machine learning algorithms to analyze time-series data is the neglect of the high autocorrelation in the time-series data, which often causes biased evaluation or overestimating model performance. Autocorrelation describes the degree of correlation of the same variables between two successive observations. For example, the supply air temperature of AHU at any time should be very close to the previous observation (e.g., the temperature a minute ago) [2]. When developing machine learning models for non-time-series data, training and test data are usually split randomly because data samples are usually independent of each other. However, when such a random split strategy is applied to time-series data, the actual performance of machine learning models might be exaggerated because of the high autocorrelation in the time-series data [45]. For example, if the random split ratio of training/test time-series data is 1:1, it means that every test sample can find similar neighbors in the training data. In this way, the test data have almost identical distribution with the training data, and the out-of-distribution

generalization ability of the proposed methods may not be as good as that shown by test results. Therefore, the temporal train-test split is a preferable method to validate the out-of-distribution generalization ability of the proposed methods [46].

Given the importance of improving FDD performance with limited labeled data and the abovementioned challenges, this study proposes a similarity learning-based fault detection and diagnosis method for building HVAC systems. The temporal data-splitting method is adopted to tackle the issue of the high correlation of training data and test data using random split. Two case studies were conducted to test the effectiveness of the proposed method, with insufficient labeled data and imbalanced data, respectively. The remainder of this paper is organized as follows. Section 2 describes the research methodology developed to address the gaps in current research, and Section 3 describes the two case studies conducted utilizing an open dataset for AHU FDD. The results of the two case studies are presented in Section 4 with in-depth discussions. Section 5 concludes this paper and puts forward the future scopes.

2. Methodology

2.1. Overview of the proposed similarity learning-based method

In traditional supervised multi-classification tasks, the input is a single sample X, and the desired output of the classification function $class(\cdot)$ is the class y to which X belongs, as shown in Eq. (1). If neural networks are used as the classification function, the raw output is the probabilities of the input belonging to each class given by a Softmax activation function. The final classification result is obtained by selecting the class with the maximum probability [47]. Cross-entropy is the most used loss function to train a classification task. Similarity learning differs from traditional supervised learning by transforming the multi-class classification problem into a binary one. Therefore, the loss function used in similarity learning is different. Binary cross-entropy and triplet loss are common loss functions for similarity learning [48]. The classification function of similarity learning $sim(\cdot)$ is formulated in Eq. (2). The similarity function examines the similarity of a pair of input samples, i.e., X_1, X_2 to determine if the two samples belong to different classes. If yes, the desired output of the similarity function is 0; otherwise, the desired output is 1. When the similarity function is obtained, the similarity between new samples and labeled samples can be measured to make FDD predictions. In similarity learning, the labeled samples can be paired

randomly to generate more input pairs for training than using the labeled samples alone. Therefore, the advantage of similarity learning over traditional supervised classification is the larger amount of training data to overcome the problem of limited labeled data and improve the FDD method's generalization ability. This makes similarity learning well-suited to tasks where labeled data are limited or challenging to collect.

$$class(X) = y, y\epsilon\{0,1,\dots,n\} \tag{1}$$

$$sim(X_1, X_2) = \begin{cases} 0, & X_1, X_2 \; belong \; to \; different \; classes \\ 1, & X_1, X_2 \; belong \; to \; the \; same \; class \end{cases} \tag{2}$$

Similarity learning can adopt various architectures and machine learning algorithms to formulate the similarity function, such as Siamese networks [43] and kernel-driven methods [49]. Siamese networks (also called twin neural networks) are widely adopted in similarity learning consisting of two identical neural networks as subnetworks that share the same structure and weights [50]. Siamese networks are typically used for learning the similarity between a pair of samples as input, such as two images or two speech records [51].

The workflow of the proposed similarity learning-based FDD method using Siamese networks is shown in Fig. 1, which consists of two tasks: model training and fault diagnosis. During model training, input pairs are first randomly generated from training data. The various color in Fig. 1 denotes the raw label in the dataset (i.e., Normal, Fault 1,…). The output of the Siamese networks is the similarity of the two samples of an input pair. If the input pair belongs to the same class, the target output is 1. Otherwise, the target output is 0. Finally, the loss function used to update the Siamese networks is binary cross-entropy loss. During fault diagnosis, the test sample is paired with a support set and outputs the most similar class using the trained Siamese networks. Note that the proposed method is applicable to fault detection when the data are only labeled as normal and faulty. The details of model training and fault diagnosis are further elaborated in Sections 2.2 and 2.3, respectively.

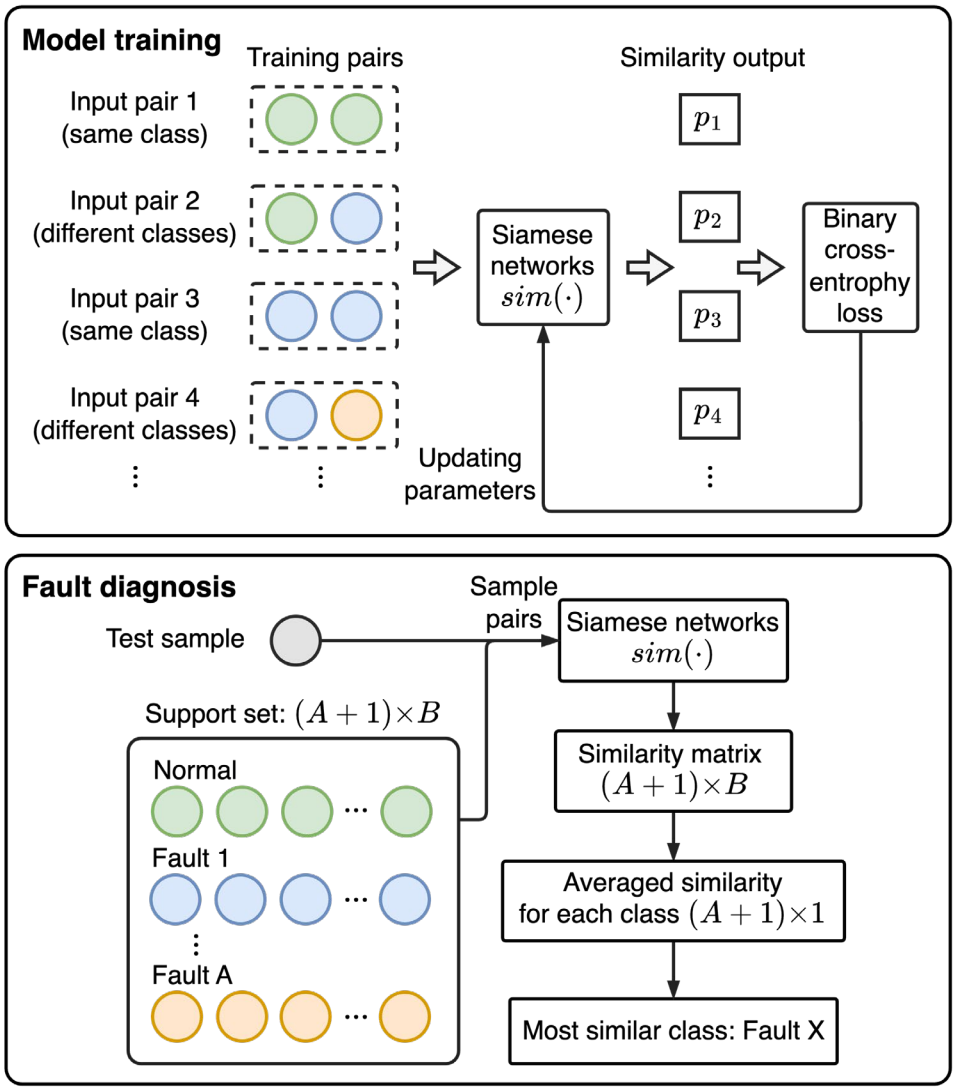

Fig. 1. Workflow of the proposed similarity learning-based FDD method using Siamese networks

2.2. Model training

The structure of the proposed Siamese network is shown in Fig. 2, which contains two identical long short-term memory (LSTM) subnetworks. The input is a pair of multivariate time-series samples, and the output is the similarity of the input pair, ranging from 0 to 1. More details about the proposed method are given in the following subsections.

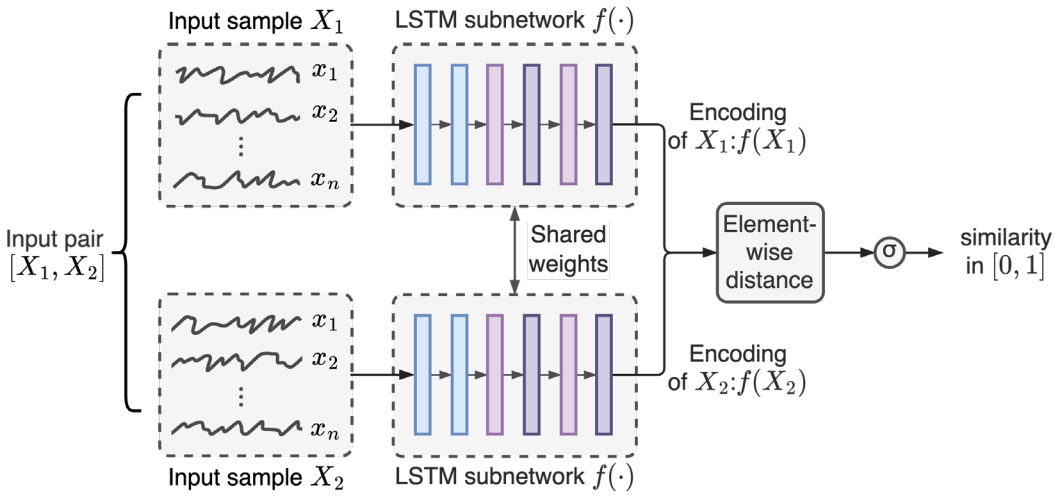

Fig. 2. Structure of the proposed Siamese network

2.2.1. Input pair generation

The input samples X_1, X_2 are a pair of multivariate time-series sampled from historical operation data using a moving window. Fig. 3 exemplified how samples are obtained from a labeled multivariate time-series (e.g., labeled as Normal, Fault 1, Fault 2…). There are n features in the raw multivariate time-series, and the window samples data by moving itself from left to right. The window size m refers to the length of each sample, and the window stride is the number of time steps by which the window is shifted each time. The fault labels for the samples are identical to the label of the raw time-series. After all raw multivariate time-series are sampled, different samples with the shape of $\times n$ are obtained.

Next, pairs of similar and dissimilar samples are generated by randomly pairing the above samples. As defined in Eq. (2), if both samples in the pair belong to the same class, the pair is labeled as similar ("1"). Otherwise, the pair is labeled as dissimilar ("0"). The number of similar and dissimilar pairs in the training dataset should be roughly the same to ensure that the distribution of similar and dissimilar pairs is balanced when creating sample pairs for training the Siamese network. If the distribution is imbalanced, the networks may not be able to learn the differences between the two classes of samples effectively. A binary random variable that can be 1 or 0 with equal probability generates similar or dissimilar pairs to achieve the balance.

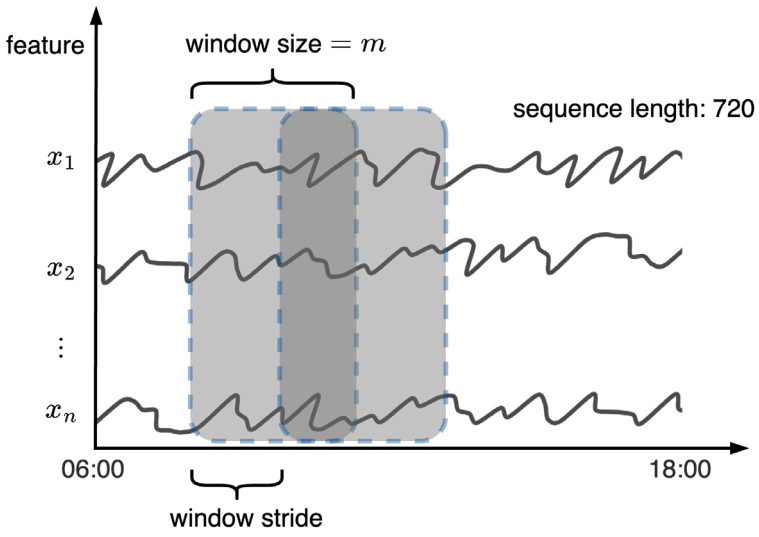

Fig. 3. Moving window sampling for multivariate time-series

2.2.2. LSTM subnetworks for encoding multivariate time-series

Once a set of labeled pairs have been generated, Siamese networks can be trained by providing each pair to the LSTM subnetworks as input and using the labels to compute the loss. The Siamese networks learn to compare the two input samples and predict whether they are similar or dissimilar based on their features and learned relationship. The architecture of LSTM subnetworks is shown in Table 1, consisting of two LSTM layers, two one-dimensional batch normalization layers, and two fully connected layers.

Table 1 Architecture of LSTM subnetworks in the proposed Siamese network

Layer	Parameters	Output size
LSTM layer 1	hidden_size: 75	m×75
LSTM layer 2	hidden_size: 50	50
BatchNorm1d layer 1	num_features: 50	50
Fully connected layer 1 (ReLU)	out_features: 40	40
BatchNorm1d layer 2	num_features: 40	40
Fully connected layer 2 (ReLU)	out_features: 20	20

(1) LSTM layers

LSTM is a type of recurrent neural network that can capture long-term dependencies in data [52]. LSTM networks are well-suited to modeling sequences of data, such as time-series, natural language text, and audio data. An LSTM layer contains LSTM cells, which are composed of an input gate that determines what new information to store in the memory, a forget gate that determines what information to throw away from the cell's memory and an output gate that outputs

the information stored in the memory, as shown in Eqs. (3)-(5). A set of weights controls these gates learned during training. The cell state and hidden state are in Eqs. (6) and (7).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

$$f_t = \sigma\big(W_f \cdot [h_{t-1}, x_t] + b_f\big) \tag{4}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{6}$$

$$h_t = o_t \odot tanh(c_t) \tag{7}$$

where x_t is the input to the LSTM cell at time step t, and b are the trainable weights and biases of the LSTM layer, $\odot$ is the mathematical operator for the element-wise multiplication, and $\sigma(\cdot)$ and $tanh(\cdot)$ are the sigmoid and hyperbolic tangent activation functions, respectively.

(2) Batch normalization and fully connected layers

The batch normalization layer normalizes each batch of data by shifting and scaling the previous layer's output, as shown in Eq. (8). The batch normalization can improve overall performance by stabilizing the learning process and accelerating convergence [53]. A fully connected layer is used after the batch normalization layer to generate a more abstract representation of the input data, as given by Eq. (9).

$$y = \frac{x - \mu}{\sqrt{\sigma + \epsilon}} \times \gamma + \beta \tag{8}$$

where μ and σ are the mean and standard deviation of the batch, respectively, ϵ is a small constant (1×10^{-5} in this study), added to prevent division by zero, and γ and β are trainable parameters.

$$y = ReLU(Wx + b) \tag{9}$$

where W and b are the weights and biases of the fully connected layer, respectively, and $\mathrm{ReLU}(\cdot)$ is the rectified linear unit activation function.

After the Siamese subnetworks, a pair of 20-dimensional vectors $f(X_1), f(X_2)$ representing the encodings of the input pair X_1, X_2 are obtained, where $f(\cdot)$ denotes the LSTM subnetwork.

2.2.3. Binary cross-entropy loss

During training, Siamese networks are fed with pairs of input samples with labels indicating whether the pair is similar or dissimilar. The networks compute the similarity between the two time-series in the pair, and the loss is computed using the similarity and the pair's true label. The networks are then optimized using this loss, with the goal of minimizing the overall loss in the training data. This helps the networks learn to predict the similarity between pairs of input samples accurately. Siamese networks typically use binary cross-entropy loss or contrastive loss [54]. This study adopts the binary cross-entropy loss because the proposed similarity learning-based method treats fault diagnosis as a binary classification problem (i.e., either similar or dissimilar).

Before calculating binary cross-entropy, the encodings $f(X_1), f(X_2)$ are squashed into the range [0,1] using a fully connected layer with the sigmoid activation function. First, the absolute elementwise difference between the encodings $f(X_1), f(X_2)$ is calculated. After that, the fully connected layer is applied, and the output of the layer is interpreted as the probability that the input pair belongs to the same class. The process is given as:

$$d = |f(X_1) - f(X_2)| \tag{10}$$

$$\hat{y} = \sigma(W \cdot d + b) \tag{11}$$

where $\hat{y}$ is the output of the layer, W is the weight matrix, d is the absolute elementwise difference between encodings of input X_1, X_2, and b is the bias vector.

The binary cross-entropy loss is computed as the average of the loss values for each sample in the dataset, as shown in Eq. (12). Like the cross-entropy loss, the binary cross-entropy loss penalizes confident but incorrect predictions, encouraging the model to output probabilities closer to the true labels. This helps improve the model's performance and accuracy.

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \, log \, \hat{y}_i + (1 - y_i) \, log(1 - \hat{y}_i)] \tag{12}$$

where N is the number of samples in the dataset, y_i is the true label of the i-th sample ("0" or "1" representing different and same labels, respectively), and $\hat{y}_i$ is the predicted probability of the i-th sample belonging to the same class ("0" or "1").

2.3. Fault diagnosis

The next step involves predicting the labels of test samples using the Siamese network. A support set needs to be constructed, which provides a set of reference samples with labels. The support set can be used to pair up with one test sample as the inputs of the Siamese networks. As shown in Fig. 1, if an FDD problem has A types of faults, the number of classes is $A + 1$, including the normal class for each class, B samples are randomly chosen from the training data to construct the support set. Then, a similarity matrix is calculated by comparing the similarity of the input sample with each sample in the support set. Finally, the similarity of all test samples to the samples in each class is averaged to find the most similar class of the input sample.

3. Case studies

In this research, two case studies are conducted to evaluate the proposed Similarity learning-based FDD method under two typical scenarios: insufficient labeled data and imbalanced labeled data. Section 3 introduces the experiment data from the American society of heating, refrigerating and air-conditioning engineers (ASHRAE) project RP-1312. Sections 3.2 and 3.3 explain the settings of the two case studies.

3.1. Dataset description

AHU operational data from the ASHRAE project RP-1312 [55] are used in both case studies to verify the proposed method's performance. The schematic of AHU in the ASHRAE RP-1312 is shown in Fig. 4. The outdoor air is initially combined with the return air, after which it undergoes treatment sequentially using a set of heating and cooling coils. Temperature, flowrate, and pressure sensors were installed, and the experimental data were collected under normal and various fault conditions. The faults were manually introduced in the AHU, such as the stuck air damper and the leaking heating valve. As listed in Table 2, five types of faults happened in different components of AHU. These faults were tested under various intensities and had 15 fault classes. The data in normal and fault classes are labeled as $F_0, F_1, \ldots, F_{15}$, accordingly. The operation duration of AHU in this experiment was from 6:00 to 18:00, and data were collected with a one-minute interval. A 12-hour time-series was collected during a one-day experiment. Each faulty class contains 720 data points, while the normal class contains 720×4 data points. Each fault scenario was tested on different days, while the normal scenario was tested on four days. The 15 features used for fault diagnosis are listed in Table 3, mainly including temperatures, flow rates, and fan power signals.

Fig. 4. Schematic of AHU in the ASHRAE RP-1312

Table 2 Details of typical AHU faults in the ASHRAE RP-1312

Fault type	Fault (class) label	Fault detail	Experiment date
Normal	F_0	No fault	2007-08-19/25, 2007-09-04, 2007-09-10
Type A	F_1	Exhaust air damper stuck (100% open)	2007-08-20
	F_2	Exhaust air damper stuck (0% open)	2007-08-21
Type B	F_3	Outdoor air damper stuck (0% open)	2007-08-26
	F_4	Outdoor air damper leaking (45% open)	2007-09-05
	F_5	Outdoor air damper leaking (55% open)	2007-09-06
Type C	F_6	Heating coil valve leaking (0.4 GPM)	2007-08-28
	F_7	Heating coil valve leaking (1.0 GPM)	2007-08-29
	F_8	Heating coil valve leaking (2.0 GPM)	2007-08-30
Type D	F_9	AHU duct leaking (after supply fan)	2007-09-07
	F_{10}	AHU duct leaking (before supply fan)	2007-09-08
	F_{11}	AHU duct leaking (before supply fan)	2007-09-09
Type E	F_{12}	Cooling coil valve stuck (100% open)	2007-08-31
	F_{13}	Cooling coil valve stuck (0% open)	2007-08-27
	F_{14}	Cooling coil valve stuck (15% open)	2007-09-01
	F_{15}	Cooling coil valve stuck (65% open)	2007-09-02

Table 3 Features used for AHU fault diagnosis in the ASHRAE RP-1312

Feature name	Description	Unit
SA-TEMP	Supply air temperature	°F
OA-TEMP	Outdoor air temperature	°F
MA-TEMP	Mixed air temperature	°F
RA-TEMP	Return air temperature	°F
SA-CFM	Supplied air flow rates	CFM
RA-CFM	Return air flow rates	CFM
OA-CFM	Outdoor air flow rates	CFM
SF-WAT	Supply fan power	W
RF-WAT	Return fan power	W
SF-STS	Supply fan status	On = 1, Off = 0
RF-STS	Return fan status	On = 1, Off = 0
SF-DP	Differential pressure of supply fan	Pa
RF-DP	Differential pressure of return fan	Pa
CHWC-VLV	Cooling coil valve position	%
HWC-VLV	Heating coil valve position	%

3.2. Data-splitting

As discussed in the introduction, the temporal train-test split is preferable for time-series data. The temporal train-test split involves selecting a certain number of time steps from the beginning of the series as the training data and the rest as the test data. There is also a constraint that the test data must be chronologically after the training data. This constraint ensures that the model has not seen any data from the future when making predictions on the test data.

As shown in Fig. 5, for each day's experimental data, the data from 6:00 to 14:00 was used as the training period, and the data from 14:00 to 18:00 was used as the test period. The data samples were generated using the method introduced in Section 2.2.1. The window stride size is set to 1 minute for the training period. In the following two cases, only a part of the training data was randomly selected to simulate the insufficient labeled and imbalanced data scenarios. The selection methods for each case will be introduced in Sections 3.3 and 3.4, respectively. For the test period, the window stride is 5 minutes. In both case studies, the window size m is 10.

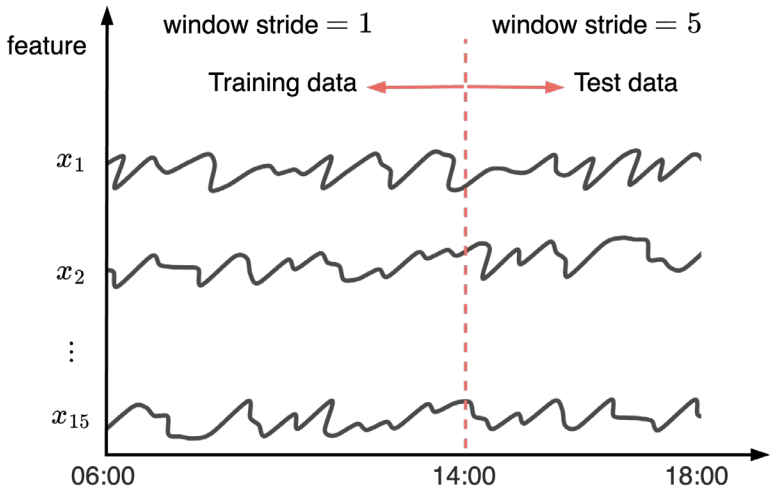

Fig. 5. Data-splitting for case studies

3.3. Case 1: Insufficient labeled data

As listed in Table 2, there are 16 normal/faulty working conditions in the dataset. The purpose of case 1 is to test the performance of the proposed method when only a small amount of labeled data is available for each fault, which is a common situation when developing data-driven models for FDD in practical HVAC systems. In this case, the performance of the proposed method is tested by adjusting the number of samples per class in the training set N_t in the range of 5–100. In addition, the number of samples in each class in the support set N_s is also adjustable in the range of 1–10. When $N_s < N_t$, N_s samples are selected from each training class without replacement; when $N_s \geq N_t$, all N_t samples in each class used for training are selected. During the experiments, the specified number of samples were randomly selected from the training samples generated in Section 3.2. The experiments were repeated ten times using different random seeds to eliminate the effect of randomness on the results.

3.4. Case 2: Imbalanced labeled data

Case 2 is designed to test the performance of the proposed method under data imbalance, for example, when a particular fault class has less labeled data than other fault classes. This situation is also common as some faults occur more frequently than other faults during the operation of HVAC systems. In Case 2, each type of fault in Table 2 was tested under data imbalance. The imbalance ratio of each imbalanced fault class and the remaining classes (faulty/normal

conditions) is 1:10. The number of labeled samples for the imbalanced fault is 10, and the number of samples for the remaining faulty/normal conditions is 100. Similar to Case 1, the experiments were repeated ten times to avoid sampling bias.

3.5. Baseline model

To compare the proposed similarity learning-based method with the traditional data-driven approach, the same architecture of LSTM subnetworks is used as the baseline. The baseline model treats fault diagnosis as a multi-class classification problem. Therefore, the input and output of the baseline model differ from those of the Siamese networks. As shown in Fig. 6, the input is a single sample, and the output is the probability of the sample belonging to each class, i.e., p_i. The probability vector is obtained using a fully connected layer with the Softmax activation function after the LSTM. Finally, the baseline model is optimized using the cross-entropy loss.

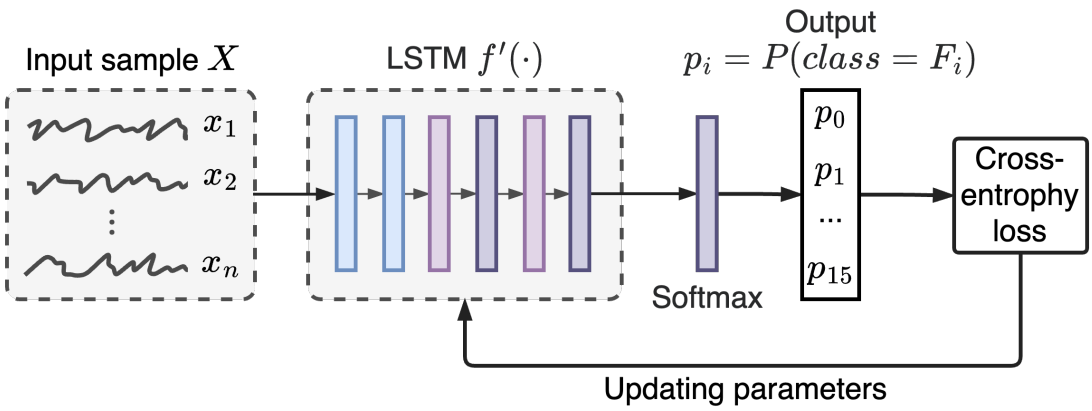

Fig. 6. Baseline model for case studies

3.6. Evaluation metric

The performance improvement rate is used as an evaluation metric to measure the performance difference between the proposed similarity learning-based method and the baseline model, which is defined as:

$$PIR = \frac{Accuracy_{Siamese} - Accuracy_{Baseline}}{Accuracy_{Baseline}} \times 100\% \quad (12)$$

where $Accuracy_{Siamese}$ is the accuracy of the proposed Siamese networks on the test set; $Accuracy_{Baseline}$ is the accuracy of the baseline model on the test set.

4. Results and discussions

This section presents the results of Case 1 and Case 2 in Sections 4.1 and 4.2, respectively. Case 1 analyzes the impacts of the number of samples per class in the training set and the support set on fault diagnostic accuracy using the proposed Siamese network. Then the performance improvement rate versus the baseline model is evaluated. In Case 2, each imbalanced fault's diagnostic accuracy and the remaining classes is analyzed.

4.1. Case 1: Insufficient labeled data

4.1.1. Impact of the number of samples per class in the training set

Table 4 and Fig. 7 show how the fault diagnostic accuracy on the test data changes in different settings (the number of samples per class in the support set N_s and the number of samples per class in the test set N_t) are used for training Siamese network. Note that all experiments were repeated ten times using different random seeds, and Fig. 7 shows the average accuracy. Generally, there is a positive correlation between fault diagnostic accuracy and N_t under a given number of N_s, because more labeled data can provide extra information for the model and lead to better generalization performance. A significant increase in the fault diagnostic accuracy can be observed when N_t increases from 5 to 10. The fault diagnostic accuracy becomes stable when N_t reaches a certain critical value. At this point, adding more labeled data to the training set does not improve the model's overall accuracy. Nevertheless, according to Table 4, a higher number of samples per class in the training set has a lower standard deviation of accuracy in the ten random experiments. Therefore, adding more labeled data to the training set can improve the model's stability and robustness, although the overall accuracy does not improve.

4.1.2. Impact of the number of samples per class in the support set

According to Table 4 and Fig. 7, there is a positive correlation between fault diagnostic accuracy and N_s under a given number of N_t. The more samples per class in the support set, the Siamese networks can make more reliable classification results based on the similarity matrix. Compared with the performance leap owing to the increase of N_t at the early stage, the performance improved with the increase of N_s is not significant, especially when $N_t \geq 20$. For example, when $N_t = 20$, the difference in fault diagnostic accuracy from $N_s = 1$ to $N_s = 10$ is only 4.1% (from 86.3% to 90.4%). It means that despite only one sample per class in the support set, Siamese networks can

make accurate enough FDD results. Similarly, the correlation between the standard deviation of accuracy and N_s is also not as strong as the correlation between the standard deviation of accuracy and N_t. It means that the stability of Siamese networks is more sensitive to the number of samples per class in the training set than the number of samples per class in the support set.

Another finding is that when N_t is large, the improvement of fault diagnostic accuracy becomes small as N_s increases from 1 to 10. For example, the difference in model accuracy from $N_s = 1$ to $N_s = 10$ is 12.5% (from 60.6% to 73.1%) when N_t is 5, but is only 4.4% (from 87.6% to 92.0%) when $N_t = 100$. Thus, there is a trade-off between the selection of N_t and N_s. Fewer samples in the support set are needed when more samples are used to train the model.

Table 4 Accuracy of fault diagnosis on the test set using Siamese networks (%)

N_t	Number of samples per class in the support set N_s										
	1	2	3	4	5	6	7	8	9	10	Avg.
5	60.6	66.3	69.9	73.7	73.1	73.1	73.1	73.1	73.1	73.1	70.9
	(7.6)	*(12.5)*	*(6.3)*	*(6.5)*	*(7.4)*	*(7.4)*	*(7.4)*	*(7.4)*	*(7.4)*	*(7.4)*	*(7.7)*
10	77.1	84.4	85.1	84.7	86.2	86.1	85.5	86.1	86.1	86.2	84.8
	(11.4)	*(7.9)*	*(5.8)*	*(5.0)*	*(5.7)*	*(5.4)*	*(5.5)*	*(6.1)*	*(6.2)*	*(6.1)*	*(6.5)*
20	86.3	89.8	87.4	90.0	89.9	90.5	90.6	89.7	89.6	90.4	89.4
	(5.4)	*(6.6)*	*(5.2)*	*(5.4)*	*(5.0)*	*(6.1)*	*(4.8)*	*(5.7)*	*(6.1)*	*(5.8)*	*(5.6)*
30	86.2	89.6	90.3	90.6	87.3	90.0	90.7	89.1	90.7	90.1	89.5
	(6.0)	*(5.6)*	*(4.8)*	*(4.6)*	*(6.0)*	*(4.3)*	*(3.9)*	*(5.1)*	*(4.1)*	*(4.2)*	*(4.9)*
40	88.9	89.3	88.3	89.9	89.3	91.1	90.9	90.9	91.0	90.9	90.1
	(7.1)	*(4.7)*	*(3.8)*	*(4.1)*	*(5.0)*	*(4.3)*	*(5.2)*	*(4.1)*	*(4.5)*	*(4.7)*	*(4.8)*
50	87.1	89.4	90.9	90.6	90.3	92.4	91.3	89.9	90.0	91.1	90.3
	(6.9)	*(7.2)*	*(3.9)*	*(6.7)*	*(5.7)*	*(4.2)*	*(4.8)*	*(6.1)*	*(4.5)*	*(4.2)*	*(5.4)*
60	86.9	88.2	91.4	90.8	91.8	91.9	91.3	91.7	91.6	90.9	90.7
	(5.3)	*(5.2)*	*(5.1)*	*(5.9)*	*(4.0)*	*(4.1)*	*(5.1)*	*(5.0)*	*(4.9)*	*(4.8)*	*(4.9)*
70	87.6	90.4	88.0	88.7	91.0	90.1	90.8	89.6	90.3	89.6	89.6
	(7.1)	*(4.6)*	*(6.6)*	*(5.1)*	*(4.8)*	*(4.4)*	*(3.3)*	*(5.0)*	*(5.5)*	*(4.9)*	*(5.1)*
80	89.1	91.8	91.4	92.3	92.4	92.0	91.9	92.7	92.5	91.0	91.7
	(4.3)	*(4.7)*	*(3.5)*	*(4.7)*	*(3.6)*	*(3.2)*	*(5.1)*	*(3.5)*	*(3.7)*	*(5.1)*	*(4.1)*
90	90.9	92.1	88.3	92.0	90.7	91.8	90.4	91.0	92.0	91.6	91.1
	(3.9)	*(3.6)*	*(5.6)*	*(3.4)*	*(4.1)*	*(3.4)*	*(4.2)*	*(4.1)*	*(3.4)*	*(4.2)*	*(4.0)*
100	87.6	91.9	92.7	91.5	92.8	92.0	93.1	92.4	92.8	92.0	91.9
	(5.2)	*(4.0)*	*(2.7)*	*(3.0)*	*(2.0)*	*(2.3)*	*(2.6)*	*(2.5)*	*(2.4)*	*(2.5)*	*(2.9)*
Avg.	84.4	87.6	87.6	88.6	88.6	89.2	89.1	88.7	89.1	88.8	88.2
	(6.4)	*(6.1)*	*(4.8)*	*(4.9)*	*(4.8)*	*(4.5)*	*(4.7)*	*(5.0)*	*(4.8)*	*(4.9)*	*(5.1)*

Note: N_t refers to the number of samples per class in the training set; the unit of numbers is %; numbers in parentheses refer to the standard deviation of accuracy in ten repeated experiments using different random seeds.

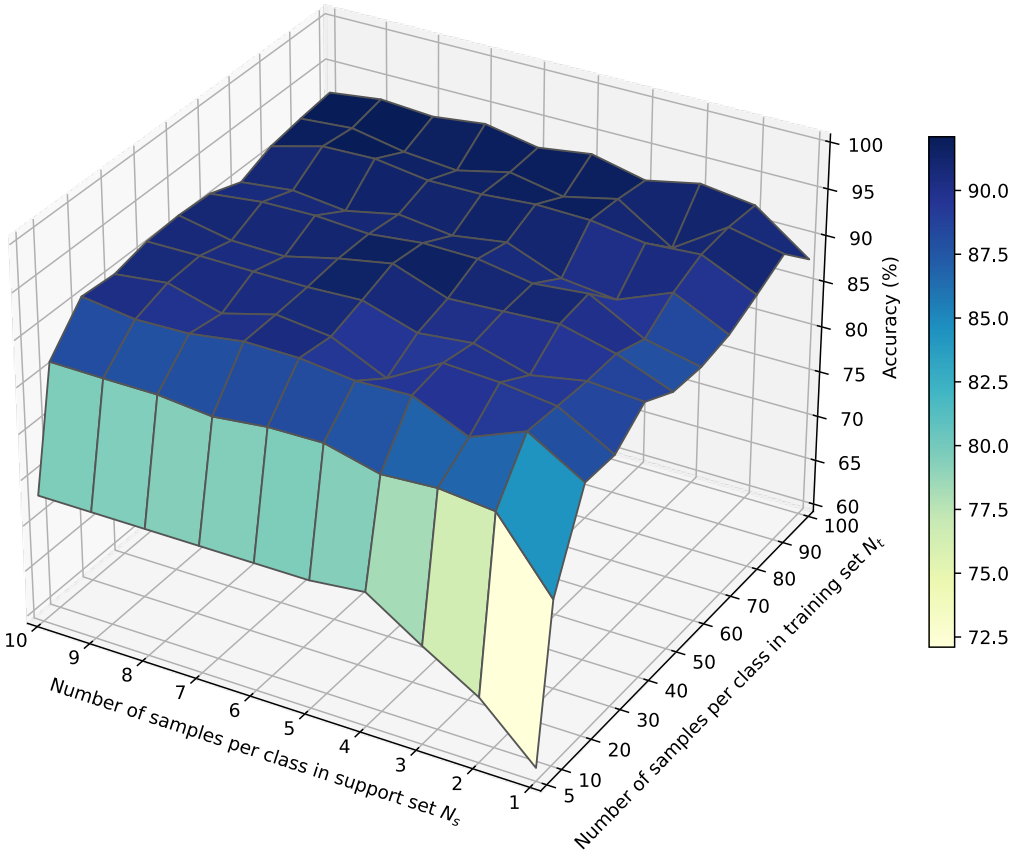

Fig. 7. Average accuracy of fault diagnosis on the test set using the proposed Siamese network

4.1.3. Performance improvement rate

The performance improvement rate of the proposed Siamese networks compared with the baseline model is shown in Fig. 8. The number of samples per class in the support set N_s is fixed to 10. Eq. (12) defines a positive PIR as the Siamese networks perform better than the baseline model. When the number of samples per class in the training set N_t is 5, an average of 45.7% PIR can be observed. It means that the proposed Siamese networks can boost fault detection accuracy when the number of labeled data is insufficient. With the increase of N_t, the PIR decreases fast ($N_t \leq 20$) and fluctuates in the range between 0%–10% ($N_t \geq 30$). In general, the proposed Siamese networks outperform the baseline model, especially when the number of labeled data is small ($N_t \leq 20$).

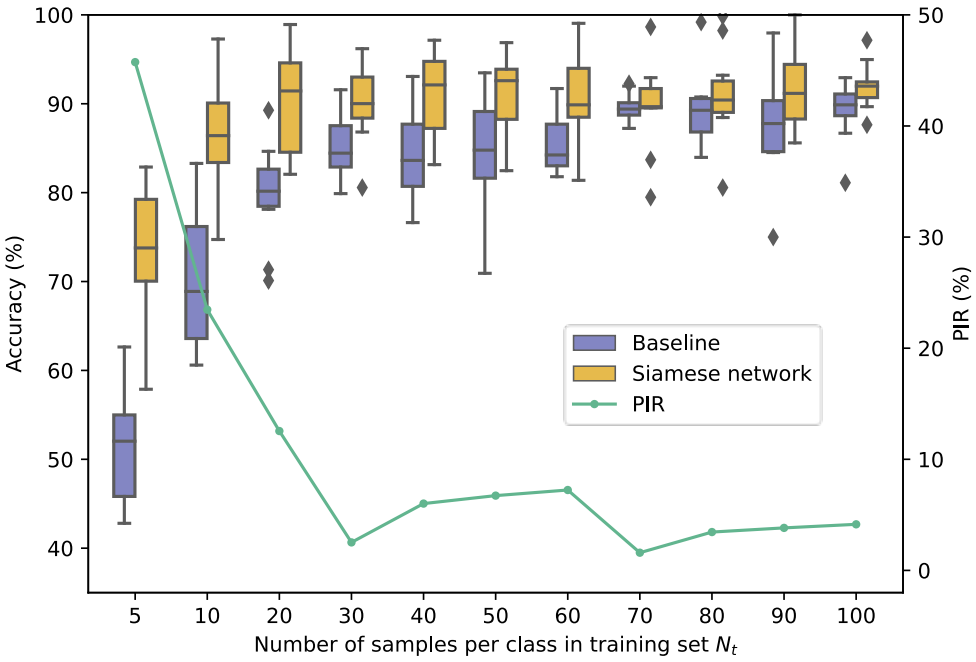

Fig. 8. Performance improvement rate of the proposed Siamese networks ($N_s = 10$)

4.2. Case 2: Imbalanced labeled data

Fig. 9 shows the testing accuracy of each imbalanced fault in case 2. For example, when F_2 (exhaust air damper is stuck to 0% position) is the class with the imbalanced fault in the training data, the fault diagnostic accuracy of F_2 on the test data using the Siamese networks and baseline model is 82.6% and 73.0%, respectively. The proposed Siamese networks outperform the baseline model in all experiments except F_{13}, F_{14}, and F_{15}. Furthermore, the lowest accuracy of the imbalanced fault using Siamese networks is 56.5% compared with 13.9% using the baseline mode (when F_5 is the imbalanced fault in the training data).

Fig. 9 shows the testing accuracy of the remaining classes. For example, when F_2 (exhaust air damper is stuck to 0% position) is the imbalanced fault in the training data, the fault diagnostic accuracy of F_2 on the test data using the Siamese networks is 82.6% and the diagnostic accuracy of the remaining classes is 88.7%. As for the baseline model which is trained using labeled samples instead of sample pairs, the fault diagnostic accuracy of F_2 and the remaining classes is 73.0% and 87.9%, respectively. As is seen in Fig.10, the proposed Siamese networks have higher diagnostic accuracy than the baseline model in all experiments. Compared with the results in Table 4, the data imbalance decreases the FDD accuracy in general. Still, the proposed Siamese networks perform betters than the baseline model in both imbalanced fault and the remaining classes.

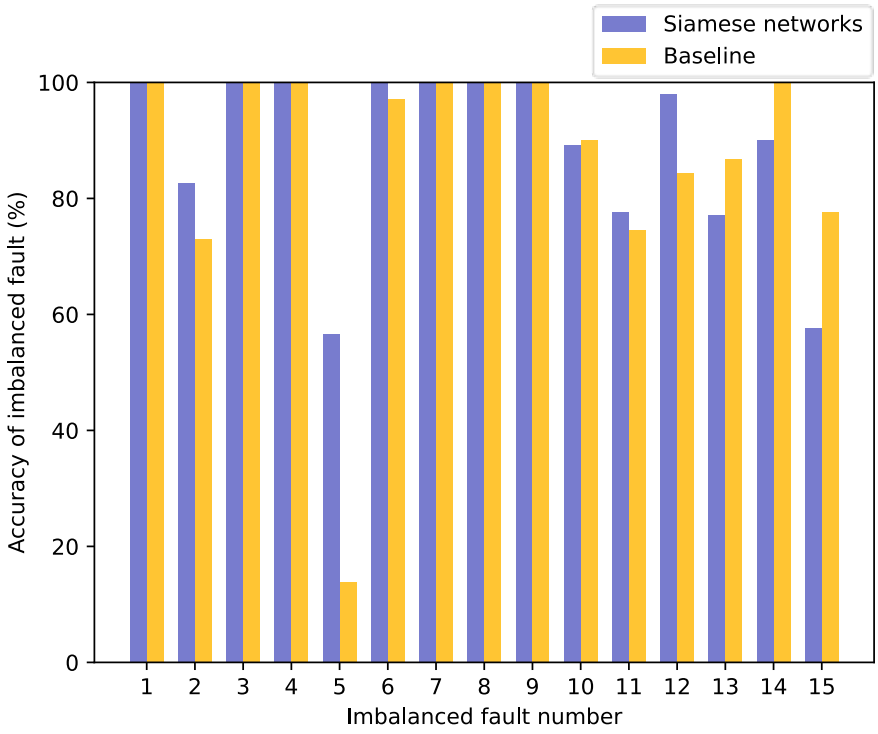

Fig. 9. Fault diagnostic accuracy of each imbalanced fault

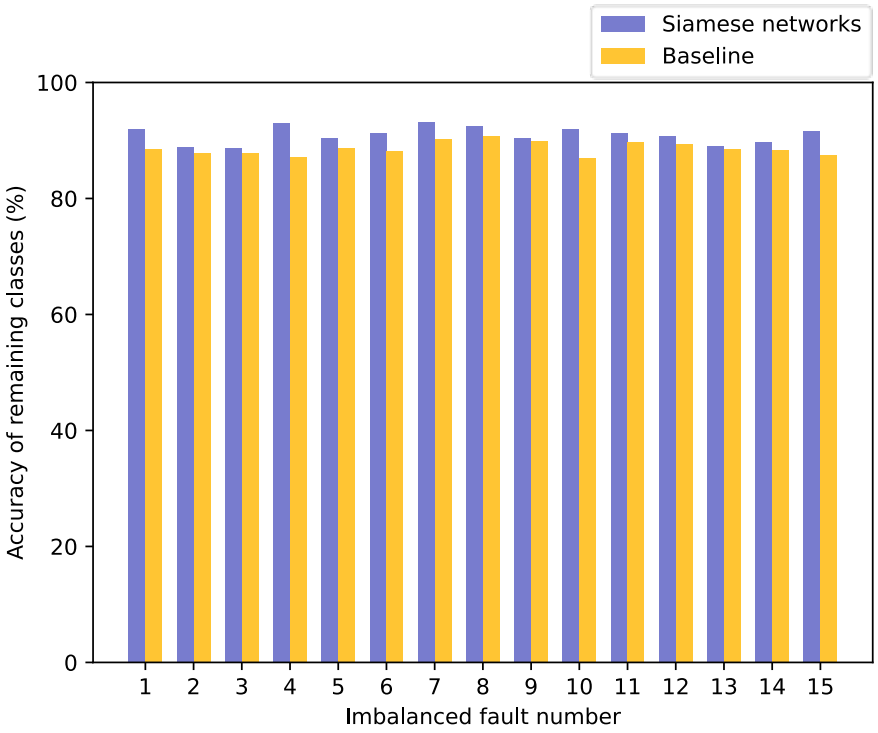

Fig. 10. Fault diagnostic accuracy of remaining classes

5. Conclusions

HVAC systems have the most significant potential for energy savings in buildings. Fault detection and diagnosis is an advanced technology that detects operational faults and determines their causes. Although the machine learning-based FDD approach is flexible and has high accuracy, the lack of labeled data may limit the practical deployment of machine learning-based models. To address the challenge, this study proposes the Siamese network, a similarity learning method, to improve the FDD performance under limited labeled data. Similarity learning is a novel supervised learning method for classification problems. Similarity learning can enlarge the training dataset and enhance the model generalization ability compared to traditional supervised classification methods. The Siamese network proposed in this paper contains two identical LSTM subnetworks. The Siamese network is trained by pairs of multivariate time-series samples from the building energy management system and learns a similarity function.

The case studies examine to evaluate the proposed similarity learning-based FDD method under two typical scenarios during the operation of HVAC systems: when labeled data are limited and imbalanced. The normal and faulty AHU operation data collected from the ASHRAE RP-1043 was used in the case studies. The data were treated using a temporal train-test split to ensure the generalization ability of the proposed method is properly evaluated. Case 1 tested the performance of the proposed method when all faults have a limited amount of labeled data. Compared to the baseline model, the proposed similarity learning-based method improved fault diagnostic accuracy by at most 45.7% when only a limited number of labeled data is available. In case 2, the performance of the proposed method when labeled data were imbalanced was tested. The results show that the proposed method outperforms the baseline model for the imbalanced fault and the remaining classes. The case studies demonstrate the great potential of the proposed FDD method in practical application when labeled data are usually limited or imbalanced during the operation of HVAC systems. Overall, the proposed similarity learning-based method is effective and promising in detecting and diagnosing faults in HVAC systems, showing better generalization ability than the traditional method.

Acknowledgment

The authors gratefully acknowledge the support of this research by Innovation and Technology Fund (ITP/002/22LP) of the Hong Kong SAR and the Research Grants Council of the Hong Kong SAR (C5018-20GF).

References

[1] Chen Y, Guo M, Chen Z, Chen Z, Ji Y. Physical energy and data-driven models in building energy prediction: A review. Energy Rep 2022;8:2656–71. https://doi.org/10.1016/j.egyr.2022.01.162.

[2] Granderson J, Lin G, Harding A, Im P, Chen Y. Building fault detection data to aid diagnostic algorithm creation and performance testing. Sci Data 2020;7:65. https://doi.org/10.1038/s41597-020-0398-6.

[3] Mirnaghi MS, Haghighat F. Fault detection and diagnosis of large-scale HVAC systems in buildings using data-driven methods: A comprehensive review. Energy Build 2020;229:110492. https://doi.org/10.1016/j.enbuild.2020.110492.

[4] Fan C, Liu Y, Liu X, Sun Y, Wang J. A study on semi-supervised learning in enhancing performance of AHU unseen fault detection with limited labeled data. Sustain Cities Soc 2021;70:102874. https://doi.org/10.1016/j.scs.2021.102874.

[5] Zhao Y, Li T, Zhang X, Zhang C. Artificial intelligence-based fault detection and diagnosis methods for building energy systems: Advantages, challenges and the future. Renew Sustain Energy Rev 2019;109:85–101. https://doi.org/10.1016/j.rser.2019.04.021.

[6] Wang H, Feng D, Liu K. Fault detection and diagnosis for multiple faults of VAV terminals using self-adaptive model and layered random forest. Build Environ 2021;193:107667. https://doi.org/10.1016/j.buildenv.2021.107667.

[7] Wang H, Chen Y, Chan CWH, Qin J. An online fault diagnosis tool of VAV terminals for building management and control systems. Autom Constr 2012;22:203–11. https://doi.org/10.1016/j.autcon.2011.06.018.

[8] Capozzoli A, Piscitelli MS, Brandi S, Grassi D, Chicco G. Automated load pattern learning and anomaly detection for enhancing energy management in smart buildings. Energy 2018;157:336–52. https://doi.org/10.1016/j.energy.2018.05.127.

[9] Deshmukh S, Samouhos S, Glicksman L, Norford L. Fault detection in commercial building VAV AHU: A case study of an academic building. Energy Build 2019;201:163–73. https://doi.org/10.1016/j.enbuild.2019.06.051.

[10] Chakraborty D, Elzarka H. Early detection of faults in HVAC systems using an XGBoost model with a dynamic threshold. Energy Build 2019;185:326–44. https://doi.org/10.1016/j.enbuild.2018.12.032.

[11] Yan K, Huang J, Shen W, Ji Z. Unsupervised learning for fault detection and diagnosis of air handling units. Energy Build 2020;210:109689. https://doi.org/10.1016/j.enbuild.2019.109689.

[12] Li G, Zheng Y, Liu J, Zhou Z, Xu C, Fang X, et al. An improved stacking ensemble learning-based sensor fault detection method for building energy systems using fault-discrimination information. J Build Eng 2021;43:102812. https://doi.org/10.1016/j.jobe.2021.102812.

[13] Turner WJN, Staino A, Basu B. Residential HVAC fault detection using a system identification approach. Energy Build 2017;151:1–17. https://doi.org/10.1016/j.enbuild.2017.06.008.

[14] Schein J, Bushby S. A Hierarchical Rule-Based Fault Detection and Diagnostic Method for HVAC Systems. HVACampR Res 2006;12:111–25. https://doi.org/10.1080/10789669.2006.10391170.

[15] Chen Z, Xu P, Feng F, Qiao Y, Luo W. Data mining algorithm and framework for identifying HVAC control strategies in large commercial buildings. Build Simul 2021;14:63–74. https://doi.org/10.1007/s12273-019-0599-0.

[16] Chen Z, Xiao F, Guo F, Yan J. Interpretable machine learning for building energy management: A state-of-the-art review. Adv Appl Energy 2023;9:100123. https://doi.org/10.1016/j.adapen.2023.100123.

[17] Han H, Gu B, Hong Y, Kang J. Automated FDD of multiple-simultaneous faults (MSF) and the application to building chillers. Energy Build 2011;43:2524–32. https://doi.org/10.1016/j.enbuild.2011.06.011.

[18] Capozzoli A, Lauro F, Khan I. Fault detection analysis using data mining techniques for a cluster of smart office buildings. Expert Syst Appl 2015;42:4324–38. https://doi.org/10.1016/j.eswa.2015.01.010.

[19] Li D, Zhou Y, Hu G, Spanos CJ. Fault detection and diagnosis for building cooling system with a tree-structured learning method. Energy Build 2016;127:540–51. https://doi.org/10.1016/j.enbuild.2016.06.017.

[20] Yan R, Ma Z, Kokogiannakis G, Zhao Y. A sensor fault detection strategy for air handling units using cluster analysis. Autom Constr 2016;70:77–88. https://doi.org/10.1016/j.autcon.2016.06.005.

[21] Tassou SA, Grace IN. Fault diagnosis and refrigerant leak detection in vapour compression refrigeration systems. Int J Refrig 2005;28:680–8. https://doi.org/10.1016/j.ijrefrig.2004.12.007.

[22] Zhou Q, Wang S, Xiao F. A Novel Strategy for the Fault Detection and Diagnosis of Centrifugal Chiller Systems. HVACampR Res 2009;15:57–75. https://doi.org/10.1080/10789669.2009.10390825.

[23] Shahnazari H, Mhaskar P, House JM, Salsbury TI. Modeling and fault diagnosis design for HVAC systems using recurrent neural networks. Comput Amp Chem Eng 2019;126:189–203. https://doi.org/10.1016/j.compchemeng.2019.04.011.

[24] Verleysen M, François D. The Curse of Dimensionality in Data Mining and Time Series Prediction. In: Cabestany J, Prieto A, Sandoval F, editors. Comput. Intell. Bioinspired Syst., Berlin, Heidelberg: Springer; 2005, p. 758–70. https://doi.org/10.1007/11494669_93.

[25] van Engelen JE, Hoos HH. A survey on semi-supervised learning. Mach Learn 2020;109:373–440. https://doi.org/10.1007/s10994-019-05855-6.

[26] Yan K, Zhong C, Ji Z, Huang J. Semi-supervised learning for early detection and diagnosis of various air handling unit faults. Energy Build 2018;181:75–83. https://doi.org/10.1016/j.enbuild.2018.10.016.

[27] Yan K, Chong A, Mo Y. Generative adversarial network for fault detection diagnosis of chillers. Build Environ 2020;172:106698. https://doi.org/10.1016/j.buildenv.2020.106698.

[28] Li B, Cheng F, Cai H, Zhang X, Cai W. A semi-supervised approach to fault detection and diagnosis for building HVAC systems based on the modified generative adversarial network. Energy Build 2021;246:111044. https://doi.org/10.1016/j.enbuild.2021.111044.

[29] Li B, Cheng F, Zhang X, Cui C, Cai W. A novel semi-supervised data-driven method for chiller fault diagnosis with unlabeled data. Appl Energy 2021;285:116459. https://doi.org/10.1016/j.apenergy.2021.116459.

[30] Fan C, Liu X, Xue P, Wang J. Statistical characterization of semi-supervised neural networks for fault detection and diagnosis of air handling units. Energy Build 2021;234:110733. https://doi.org/10.1016/j.enbuild.2021.110733.

[31] Fang Q, Wu D. ANS-net: anti-noise Siamese network for bearing fault diagnosis with a few data. Nonlinear Dyn 2021;104:2497–514. https://doi.org/10.1007/s11071-021-06393-4.

[32] Yang Y, Wang H, Liu Z, Yang Z. Few-shot Learning for Rolling Bearing Fault Diagnosis Via Siamese Two-dimensional Convolutional Neural Network. 2020 11th Int. Conf. Progn. Syst. Health Manag. PHM-2020 Jinan, 2020, p. 373–8. https://doi.org/10.1109/PHM-Jinan48558.2020.00073.

[33] Zhang J, Wang Y, Zhu K, Zhang Y, Li Y. Diagnosis of Interturn Short-Circuit Faults in Permanent Magnet Synchronous Motors Based on Few-Shot Learning Under a Federated Learning Framework. IEEE Trans Ind Inform 2021;17:8495–504. https://doi.org/10.1109/TII.2021.3067915.

[34] de Oliveira LM, Mendes da Silva FE, da Cunha Pereira GM, Oliveira DS, Marcelo Antunes FL. Siamese Neural Network Architecture for Fault Detection in a Voltage Source Inverter. 2021 Braz. Power Electron. Conf. COBEP, 2021, p. 1–4. https://doi.org/10.1109/COBEP53665.2021.9684063.

[35] Utkin LV, Zaborovsky VS, Popov SG. Siamese neural network for intelligent information security control in multi-robot systems. Autom Control Comput Sci 2017;51:881–7. https://doi.org/10.3103/S0146411617080235.

[36] Oh Y, Kim Y, Na K, Youn BD. A deep transferable motion-adaptive fault detection method for industrial robots using a residual–convolutional neural network. ISA Trans 2022;128:521–34. https://doi.org/10.1016/j.isatra.2021.11.019.

[37] Chen Z, Chen Y, Xiao T, Wang H, Hou P. A novel short-term load forecasting framework based on time-series clustering and early classification algorithm. Energy Build 2021;251:111375. https://doi.org/10.1016/j.enbuild.2021.111375.

[38] Chen Z, Chen Y, He R, Liu J, Gao M, Zhang L. Multi-objective residential load scheduling approach for demand response in smart grid. Sustain Cities Soc 2022;76:103530. https://doi.org/10.1016/j.scs.2021.103530.

[39] Kar P, Shareef A, Kumar A, Harn KT, Kalluri B, Panda SK. ReViCEE: A recommendation based approach for personalized control, visual comfort & energy efficiency in buildings. Build Environ 2019;152:135–44. https://doi.org/10.1016/j.buildenv.2019.01.035.

[40] Lin G, Claridge D. Two Similarity Measure Approaches to Whole Building Fault Diagnosis 2012.

[41] Sha H, Xu P, Hu C, Li Z, Chen Y, Chen Z. A simplified HVAC energy prediction method based on degree-day. Sustain Cities Soc 2019;51:101698. https://doi.org/10.1016/j.scs.2019.101698.

[42] Miller C. What's in the box?! Towards explainable machine learning applied to non-residential building smart meter classification. Energy Build 2019;199:523–36. https://doi.org/10.1016/j.enbuild.2019.07.019.

[43] Chicco D. Siamese Neural Networks: An Overview. In: Cartwright H, editor. Artif. Neural Netw., New York, NY: Springer US; 2021, p. 73–94. https://doi.org/10.1007/978-1-0716-0826-5_3.

[44] Tan SY, Jacoby M, Saha H, Florita A, Henze G, Sarkar S. Multimodal sensor fusion framework for residential building occupancy detection. Energy Build 2022;258:111828. https://doi.org/10.1016/j.enbuild.2021.111828.

[45] Splitting data randomly can ruin your model | Data Science 2021. https://datascience.stanford.edu/news/splitting-data-randomly-can-ruin-your-model (accessed November 23, 2022).

[46] Ma R, Boubrahimi SF, Hamdi SM, Angryk RA. Solar flare prediction using multivariate time series decision trees. 2017 IEEE Int. Conf. Big Data Big Data, 2017, p. 2569–78. https://doi.org/10.1109/BigData.2017.8258216.

[47] ElBedwehy MN, Behery GM, Elbarougy R. Face Recognition Based on Relative Gradient Magnitude Strength. Arab J Sci Eng 2020;45:9925–37. https://doi.org/10.1007/s13369-020-04538-y.

[48] Beuve N, Hamidouche W, Deforges O. DmyT. Proc. 1st Workshop Synth. Multimed. - Audiov. Deep. Gener. Detect., New York, NY, USA: Association for Computing Machinery; 2021, p. 17–24. https://doi.org/10.1145/3476099.3484316.

[49] Kang Z, Peng C, Cheng Q. Kernel-driven similarity learning. Neurocomputing 2017;267:210–9. https://doi.org/10.1016/j.neucom.2017.06.005.

[50] Chakladar DD, Kumar P, Roy PP, Dogra DP, Scheme E, Chang V. A multimodal-Siamese Neural Network (mSNN) for person verification using signatures and EEG. Inf Fusion 2021;71:17–27. https://doi.org/10.1016/j.inffus.2021.01.004.

[51] Li Y, Chen CLP, Zhang T. A Survey on Siamese Network: Methodologies, Applications, and Opportunities. IEEE Trans Artif Intell 2022;3:994–1014. https://doi.org/10.1109/TAI.2022.3207112.

[52] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Comput 1997;9:1735–80. https://doi.org/10.1162/neco.1997.9.8.1735.

[53] Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. Proc. 32nd Int. Conf. Mach. Learn., PMLR; 2015, p. 448–56.

[54] Lian Z, Li Y, Tao J, Huang J. Speech Emotion Recognition via Contrastive Loss under Siamese Networks. Proc. Jt. Workshop 4th Workshop Affect. Soc. Multimed. Comput. First Multi-Modal Affect. Comput. Large-Scale Multimed. Data, New York, NY, USA: Association for Computing Machinery; 2018, p. 21–6. https://doi.org/10.1145/3267935.3267946.

[55] Wen J, Li S. Tools for evaluating fault detection and diagnostic methods for air-handling units, ASHRAE RP-1312 Final Report. Am Soc Heat Refrig Air Cond Eng Inc Atlanta GA USA 2011.