SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2024, Vol. 67, Iss. 3, 132404:1–132404:17
https://doi.org/10.1007/s11432-023-3896-4

# A deep learning model enabled multi-event recognition for distributed optical fiber sensing

Yujiao LI[1,2], Xiaomin CAO[1,2], Wenhao NI[1,2] & Kuanglu YU[1,2*]

[1]*Institute of Information Science, School of Computer and Information Technology, Beijing Jiaotong University, Beijing* 100044*, China;*
[2]*Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing Jiaotong University, Beijing* 100044*, China*

Received 12 May 2023/Revised 28 August 2023/Accepted 6 November 2023/Published online 19 February 2024

**Abstract** Fiber optic sensors that utilize backscattered light offer distributed real-time measurements and have been seen tremendous improvements in sensing distance and spatial resolution over the last decades. However, these improvements in sensor capabilities lead to a significant increase in the amount of data that needs to be processed. Traditional processing schemes are no longer adequate, so the development of novel signal processing methods is critical. Phase-sensitive optical time domain reflectometry (Φ-OTDR) is now applied in various applications for multi-event recognition, and it would usually be difficult, sometimes even unrealistic to label all the acquired samples due to its real-time and seamless monitoring nature. To fully take advantage of the information contained within the large number of unlabeled samples, which were formerly not utilized and hence wasted, we propose a semi-supervised model to boost the event classification performance of Φ-OTDR. The model extracts respectively the temporal features and the spatial bidirectional features together with a dual attention mechanism. Its classification accuracy has been improved up to 96.9% with only 1230 labeled samples. In addition, our model shows significant advantages when the number of labeled samples is reduced. Importantly, our method improves the accuracy of multi-event classification without any modification to the optical setup.

**Keywords** Φ-OTDR, event recognition, semi-supervised learning, mean teacher, MT-ACNN-SA-BiLSTM

## 1 Introduction

Distributed optical fiber sensing can continuously measure in a large space range, which has great advantages compared with traditional point sensors. It is mainly achieved by analyzing Rayleigh, Brillouin, and Raman backscattering in optical fibers [1]. Compared with other sensors, distributed optical fiber sensor has the advantages of high sensitivity, anti-electromagnetic interference, a relatively simple structure, and most important of all, real-time distributed sensing along tens of km monitor fiber [2]. Phase-sensitive optical time domain reflectometry (Φ-OTDR) is a typical representative of distributed optical fiber sensing systems [3,4]. It is widely used in oil and gas pipeline monitoring, perimeter security, railway transportation, and other fields [5–9]. The tipping point of its wide application is around the corner, and the remaining issue is how to precisely and efficiently classify various vibration events as people are trying to monitor the longer range with higher spatial resolution in real-time with Φ-OTDR. This is the reason why early research on Φ-OTDR primarily focused on improving the optical system. However in recent years, due to the rapid development of artificial intelligence technology, the research focus has shifted towards various data processing methods, particularly automated monitoring approaches employing end-to-end deep learning models [10]. For example, researchers input images obtained after the Fourier transform or space-time transform of the original signals into a two-dimensional convolutional neural network (2D-CNN) [11–13] to extract features associated with disturbance events, and then the classification is performed with a fully connected (FC) layer. The Φ-OTDR signal can also be well processed by a one-dimensional convolutional neural network (1D-CNN) [14] with a long short-term memory

* Corresponding author (email: klyu@bjtu.edu.cn)

 

network (LSTM) which is designed for long sequential data processing [15, 16]. To achieve more efficient and accurate recognition, Guan et al. [17] proposed multi-scale 1D-CNN (MS 1D-CNN) to better extract signal features of events of different scales. Wu et al. [18] proposed a new 1DCNNs-BiLSTM network to extract the bidirectional features of time and space, and claimed that the model has better performance than the CNN network with a single feature and the 2D-CNN model with a spatial-temporal feature. Tian et al. [19, 20] designed a method combining temporal convolutional network (TCN) and BiLSTM based on the attention mechanism, where the TCN model was used to extract temporal features, and the BiLSTM was used to obtain spatial features. The attention mechanism was employed to focus on key features, to reduce the number of parameters, and to speed up the processing.

Despite their outstanding performance, the models mentioned above are all supervised models, which means the training process requires a large number of labeled data. Obtaining a substantial number of labeled samples in the field application of Φ-OTDR is a formidable task due to the time-consuming nature of manual labeling [19]. Therefore, a significant challenge for supervised models is the scarcity of labeled samples, which poses difficulties in meeting the training requirements of the model. At the same time, most models are only designed for certain kinds of specific signals, which means they lack the adaptability to most other intrusion signals, leading to a weak robustness to the ever-changing environment and requirement.

To solve these problems, we propose to use semi-supervised learning (SSL) to improve Φ-OTDR event classification performance. SSL can solve the performance degradation problem of traditional supervised learning when training samples are insufficient, and demonstrate better classification performance than unsupervised learning that only uses unlabeled samples to train the network. In recent years, applications of SSL in this field [21–24] have been in the initial exploration stage. Most approaches require manual feature extraction and treat supervised and unsupervised aspects separately. Therefore, this paper delves into the mentioned issues to conduct an in-depth investigation. To directly combine SSL with the existing disturbances classification network, we introduce the mean teacher (MT) and self-training (ST) semi-supervised frameworks into the Φ-OTDR domain. By comparing ten models under the two frameworks, we propose a semi-supervised model with the best performance, namely MT-ACNN-SA-BiLSTM. Experiments show that this model can effectively combine a large number of unlabeled samples with a small number of labeled samples for model training, which can take full advantage of a large number of unlabeled samples collected by the Φ-OTDR system for auxiliary training, and eventually obtain a better model performance without increasing hardware cost and system complexity.

## 2 Distributed optical fiber sensing system

The structure of the Φ-OTDR system employed in this experiment is shown in Figure 1(a). In this experiment, we use an NKT BASIK E15 laser with an output wavelength of 1550.12 nm at 15 dBm. The continuous wave is amplified by an erbium doped fiber amplifier (EDFA) and a filter is used to suppress the amplified spontaneous emission noise. It is then modulated into repeated light pulses by an acoustic optic modulator (AOM). Then, the light launched into a Corning sm28e+ single-mode sensing fiber through a circulator. The lead 5/10 km bare fiber is put inside a soundproof box for the isolation of vibration, while the last 100 m fiber employed is a segment of armored fiber in contact with disturbance events. Data is collected using the first 50 m of the armored fiber. An anti-reflection connector (FC/APC) is used at the far end, and the reflection is further reduced by wrapping the fiber into small radius rings. The Rayleigh backscattered light is collected by the photodetector (PD) through the circulator. Finally, the signal is sampled by the data acquisition card (DAQ) encapsulated in the PC with a sampling rate of 10 MSamp/s and saved in a server to analyze the intensity information of the scattered light, which contains information on the disturbance events. These signals are then fed into ten SSL models for event identification.

Six types of events are selected as identification targets, which include background noise, digging, knocking, watering, shaking, and walking. Each type of event is collected at different times and locations. The scene photos collected for the six types of events are shown in Figure 1(b). The data for six types of events are recorded in detail as follows:

(1) Background noise. Random noises in an environment that does not artificially increase interference are collected on different dates and at different times of the day.

(2) Digging. A person uses a small shovel to dig near the sensing fiber buried 15 cm down in the

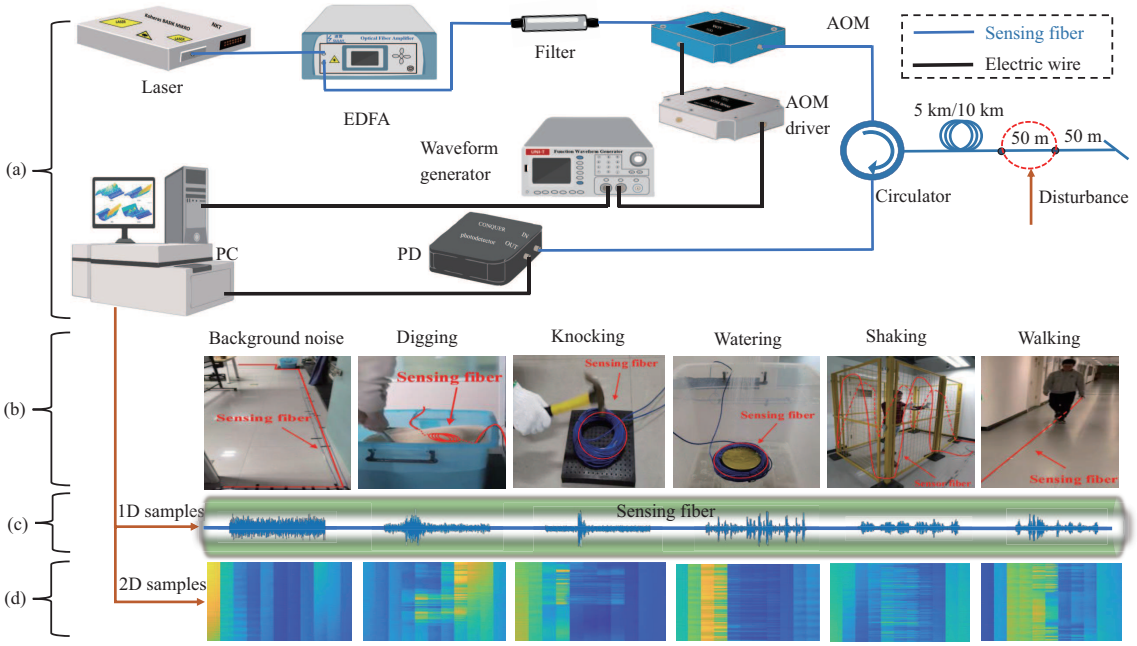


**Figure 1** (Color online) Employed Φ-OTDR system setup and disturbance events. (a) Φ-OTDR system; (b) six types of events; (c) 1D samples; (d) 2D samples.

sandbox. The length of the buried fiber is 10 m.

(3) Knocking. A person hits the sensing fiber on the vibration-proof plate with a hammer. The fiber length is 10 m. This is mainly to simulate the strain damage events.

(4) Watering. To simulate actual rain events, a person waters the sensing fiber evenly using a watering can at a height of 30 cm. The fiber has a length of 10 m.

(5) Shaking. A person stands next to the fence and shakes the sensing fibers fixed at different positions of the fence at a constant speed rate. This is mainly to simulate the actual human intrusion events.

(6) Walking. A person moves along the direction of the sensing fiber. The sensing fiber has a coverage range of 20 m. We collected data on different people walking or running at different times.

To demonstrate the characteristics of six types of signals, one-dimensional samples are shown in Figure 1(c) after differential processing. Background noise observed is a stationary random noise without human interference. For both the digging signals and the knocking signals, the amplitude decays rapidly with time after reaching the highest point. However, the digging signal exhibits a longer duration due to the backflow of sand, while the knocking signal is short-term and decays quickly. For the watering signal, the signal persists for a certain period of time since it is a continuous behavior. The pulses appear at different times and have varying amplitudes, which can be attributed to the randomness of the time and intensity of water droplets poured into the sensing section. The shaking signal is characterized by regular vibration, reflecting the alternating forward and backward shaking behavior that closely matches the temporal relationship of human shaking action. For walking signals, the amplitude of the pulse is related to the cadence of the person. The spatial-temporal signal of each event is shown in Figure 1(d). Each signal consists of 12 adjacent spatial points in the space domain (vertical) and 10000 points in the time domain (horizontal).

It is worth noting that the differential processing is only done here to allow the reader to clearly observe the temporal characteristics of each signal. In subsequent experiments, we will not perform differential processing on the signal before it is input into the model. By directly inputting the collected signals into the model for processing, we can minimize the need for human intervention, thus demonstrating a key advantage of end-to-end signal processing.

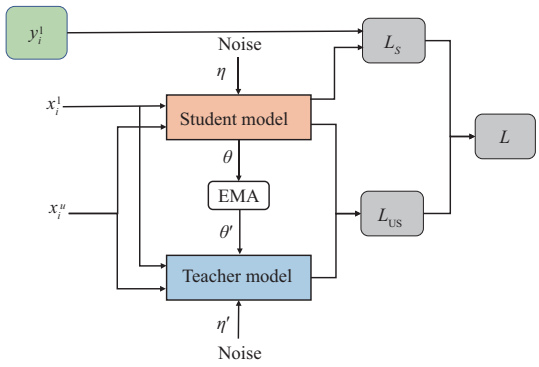


**Figure 2** (Color online) Sketch diagram of the MT model.

## 3 Theory and methodology

### 3.1 SSL

The SSL method could greatly contribute to practical applications, as the unlabeled samples can currently be used for training and avoid the performance degradation problem of the supervised models when training samples are insufficient. Semi-supervised deep learning has two main approaches, the unlabeled data initialization networks [25–28] and the end-to-end semi-supervised depth models [29–31]. The first ones use unlabeled data to initialize the network and then fine-tune the network with the labeled data. This method can be further divided into unsupervised pre-training and pseudo-supervised pre-training. Yang et al. [21] proposed a network based on a sparse stacked autoencoder (SSAE) for Φ-OTDR disturbance event classification, which is an unsupervised pre-training method. However, the unlabeled data initialization networks, which use both labeled and unlabeled data, actually operate in a supervised manner as far as neural networks are concerned. Usually, this method cannot extract the features satisfactorily, and its training still relies on a large number of labeled samples.

For the end-to-end semi-supervised depth models, the deep learning networks are used to predict the unlabeled samples, and the predicted values are used as the pseudo-labels of the unlabeled samples. With the progress of training, the prediction result of unlabeled samples will become more accurate, which makes it more capable of using unlabeled samples to promote the model. End-to-end SSL methods have been extensively utilized in the field of computer science (CS), but their application within the Φ-OTDR domain is limited. Therefore, the following will introduce its development in the field of CS. Lee [26] proposed a Pseudo-Label method, which takes the prediction of unlabeled data by the network as the pseudo-label of unlabeled data to train the network. This idea is simply the ST. In our experiment section, this framework is chosen for comparison. To make this paper more condensed, the details of ST used in this paper are given in Appendix A.1. In 2015, a semi-supervised ladder network [29] was proposed, and it organically combines supervised and unsupervised algorithms to solve compatibility problems and develops an end-to-end semi-supervised depth model. In 2016, the model was developed from a ladder network to temporal ensembling [30], and this development can help deduce network complexity and help migrate the model from the convolutional network. Later, MT [31] was put forward, and it could speed up the training on large data sets with no need to modify the original supervised deep learning network, facilitating the migration of the supervised model. The principle of MT is explained in detail in the next section.

### 3.2 The principle of MT

The MT framework is composed of a teacher network and a student network. The student model and the teacher model update network parameters through two-way propagation, that is, the network weights of the teacher model are calculated by the exponential moving average (EMA) of the student model. The network weights of the student model are updated by the gradient descent of the loss function. MT [31] SSL framework adopts weights average to replace the prediction label average method using the temporal ensembling framework to improve performance. MT adds noise to the samples for data augmentation. The principle of MT is shown in Figure 2.

The loss function $L$ of MT includes both supervised cross entropy loss $L_S$ on the labeled data set, and unsupervised consistency loss $L_{\mathrm{US}}$ on the unlabeled data set, and it can be written as

$$L(\theta) = L_S + \beta L_{\mathrm{US}}, \tag{1}$$

where $\beta$ is the scaling coefficient, and $\theta$ represents the network parameters. $L_S$ could be given as

$$L_S = E_{x_i^l, y_i^l \in D_{L,\eta}}[-y_i^l \log f(x_i^l, \theta, \eta)], \tag{2}$$

where $y_i^l$ is the real label corresponding to the labeled samples $x_i^l$, and prediction value $f(x_i^l, \theta, \eta)$ is predicted by the student model for an added noise $\eta$ on to the sample $x_i^l$. $E$ is the entropy. $L_{\mathrm{US}}$ is defined as the distance between the value $f(x_i^u, \theta', \eta')$ predicted by the teacher model for adding noise $\eta'$ to the sample $x_i^u$ and the prediction value $f(x_i^u, \theta, \eta)$ by the student model.

$$L_{\mathrm{US}} = E_{x_i^u \in D_{L,\eta,\eta'}}[||f(x_i^u, \theta', \eta') - f(x_i^u, \theta, \eta)||^2]. \tag{3}$$

At each iteration, based on the loss function, the network parameters $\theta$ of the student model are updated by the parameters $\theta'$ of the teacher model employing the EMA method. Therefore, the prediction distribution will not have too much noise to ensure that its output distribution is more stable, so that the model has a stronger generalization ability. The calculation formula is shown as

$$\theta'_t = \alpha \theta'_{t-1} + (1-\alpha)\theta_t, \tag{4}$$

where $\alpha$ is the smoothing coefficient, $\theta'_{t-1}$ and $\theta'_t$ represent the network parameters of the teacher model at the $(t-1)$-th and $t$-th iterations, respectively, while $\theta_t$ stands for the network parameters of the student model at the $t$-th iteration.

### 3.3 The teacher model and the student model

MT can facilitate the migration of a supervised deep learning model without modifying the original supervised deep learning network. The teacher model and the student model use the same network structure to utilize unlabeled samples. Therefore, some state-of-the-arts (SOTA) supervised deep learning networks with good performance could be utilized for the teacher and student models. Wu et al. [18] constructed the 1DCNNs-BiLSTM network, which takes into account both features of time and space domains and becomes a classical algorithm. To further improve the classification performance, we introduce two attention modules.

It is worth noting that our event classification model employs a different strategy from Wu. Wu et al. [18] utilized multiple identical CNNs to process temporal signals from different sensing points (fiber sections) in parallel. This strategy allows each CNN to specialize in processing signals from a specific sensing point, thereby better preserving the temporal features of that point. However, this method results in a larger model size, requiring more computational resources and longer processing time. In contrast, our disturbance classification model, as shown in Figure 3(a), employs only one CNN to extract temporal features from all sensing points, reducing the model size and improving processing speed. This strategy not only saves computational resources and processing time but also leverages the spatial distribution and temporal structure features of the vibration signals, leading to better classification performance. Therefore, we concatenate the signals from 12 nodes and input them into a 1D-CNN network to extract temporal features from all sensing points. Then, the extracted 1D-CNN feature vectors are fed into a bidirectional LSTM network to further explore spatial correlations among different node signals. Finally, the extracted spatio-temporal features are stacked and input into an FC layer for event classification.

#### 3.3.1 *Squeeze-and-excite (SE) module*

The disturbance information contained by signals at different time points is different, resulting in different channels having different influences on the classification. Therefore, in this paper, the channel attention mechanism is introduced in time domain convolution to lay on those channels with higher impacts thereby ensuring better performance.

By inserting a lightweight SE module [32] into the model, channel dependencies are established at a low computational cost to fuse global contextual information by squeezing the feature map. The SE

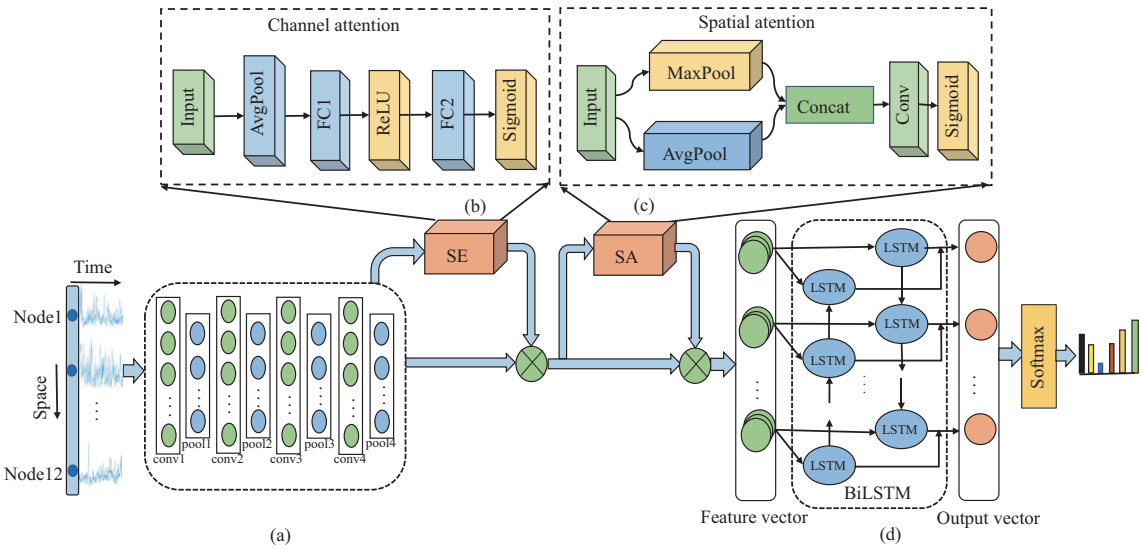


**Figure 3** (Color online) Structure diagram of the teacher model and the student model using ACNN-SA-BiLSTM. (a) 1DCNN; (b) bidirectional LSTM; (c) spatial attention; (d) channel attention.

module is shown in Figure 3(b). The excitation operation uses two FC layers to obtain the nonlinear relationship between channels and determine the weights of different channels. The first FC layer uses the ReLU activation function while the second uses the Sigmoid activation function. The SE module achieves primary and secondary priorities by assigning weights among different channels.

#### 3.3.2 *Spatial attention (SA) module*

To help the model better focus on disturbance information when we extract spatial bidirectional relationships, an SA mechanism [33] is added between 1D-CNN and BiLSTM. The SA module is shown in Figure 3(c). The time-domain features extracted by 1D-CNN constitute a spatial sequence according to the position on the fiber under test, and they are input into SA to extract key information. Because the light intensity dynamic range at the disturbed location is usually larger, a maximum pooling layer is employed. The output result of the maximum pooling layer and the average pooling layer is cascaded and input into the convolutional layer to extract the attention weights.

In our proposed model, ACNN-SA-BiLSTM, the channel attention mechanism, and the SA mechanism are introduced at the same time. The network structure is shown in Figure 3. The teacher model and the student model use this model to further optimize the feature extraction abilities, improve the processing capability of long-distance data, and speed up the network fitting speed.

### 3.4 Φ-OTDR event classification method based on MT

In Subsection 3.2, we explain the principles of MT as the overall SSL framework. In Subsection 3.3, to further enhance the ability of both the teacher model and the student model in extracting temporal and spatial features, we introduce two attention mechanisms and design the ACNN-SA-BiLSTM model, in which the self-attention mechanism of automatic extraction and updating is applied to the 1DCNN-BiLSTM network. We name this network, the MT-ACNN-SA-BiLSTM model, as shown in Figure 4. By combining SSL with supervised models, this method is suitable for online learning and large data sets, especially when not all the samples are labeled, it can significantly improve the accuracy and robustness of the model.

## 4 Experiment

### 4.1 Experimental dataset

With the Φ-OTDR system as shown in Figure 1, we collected and used 12334 samples containing six typical events, i.e., digging, knocking, watering, shaking the fence, and walking, along with background

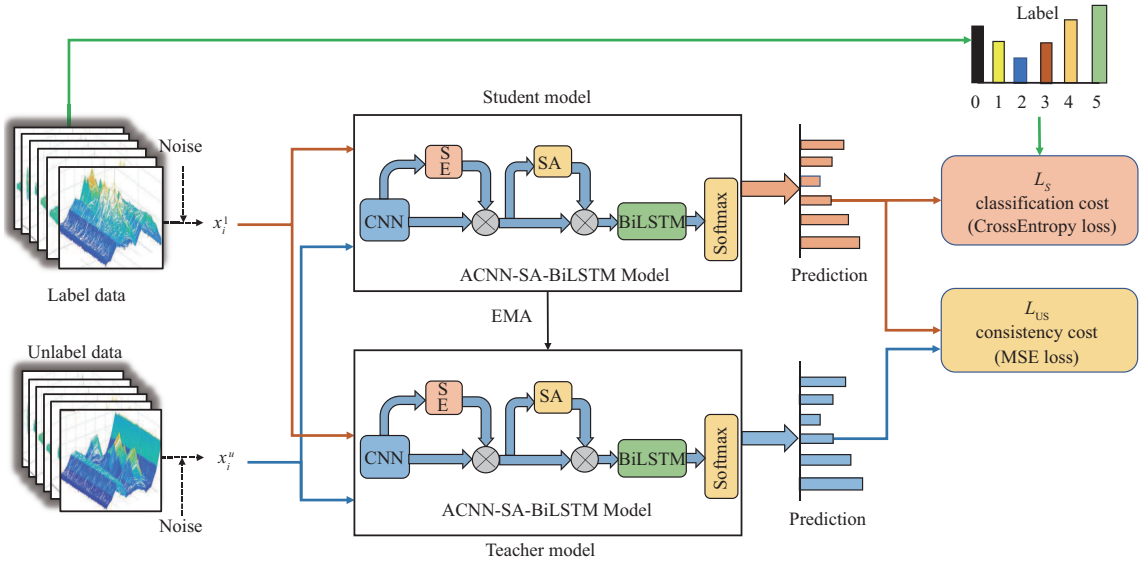


**Figure 4** (Color online) Structure of MT-ACNN-SA-BiLSTM model.

**Table 1** Numbers of six sample categories

| Category | Label | Number of samples in the training/test set |
|---|---|---|
| Background noise | 0 | 2357/589 |
| Digging | 1 | 2010/502 |
| Knocking | 2 | 2024/506 |
| Watering | 3 | 1802/451 |
| Shaking | 4 | 2182/546 |
| Walking | 5 | 1959/490 |
| Total | – | 12334/3084 |

noise. Those samples are respectively labeled with labels from 0 to 5 as shown in Table 1, which also includes the number of samples in each category. The data is unified into $12 \times 10000$, where each sample consists of 10000 points in the time domain and 12 adjacent space points in the space domain. The dataset is open to the public and can be downloaded from GitHub [34].

Base on the above experimental data, the data set required by the semi-supervised deep learning method is also further prepared. For different experiments, we randomly set different proportions of invalid labels according to the experimental requirements. These invalid labels are represented as $-1$, while the labels of the remaining data are retained. In this way, labeled data and unlabeled data required for semi-supervised training are prepared, which is convenient to distinguish the losses of labeled data and unlabeled data.

## 4.2 Model parameter settings

The hyperparameter settings of the MT-ACNN-SA-BiLSTM model in the experiment are shown in Table 2. The basic hyperparameters of the MT framework are consistency cost weight and EMA decay rate $\alpha$. These two parameters, after referring to the dynamic parameter settings in [31], showed a negligible difference in performance improvement from setting them to fixed values. Therefore, to reduce the model complexity and improve the calculation speed, we directly set the two parameters to 0.2 and 0.95, respectively. Gaussian noise with a mean value of 0 and variance of 1 is added to the training data. Meanwhile, we use dropout to enable the model to learn more abstract invariances. The parameter setting of CNN and BiLSTM in the ACNN-SA-BiLSTM model is adopted and modified from [18], where a four-convolution layer is used. To improve the convergence speed of the network and alleviate overfitting, batch normalization (BN) is added after each convolution layer. The SE module is inserted at the end of the 1D-CNN structure, the number of output channels after the last BN layer is set to 256, and the dimension reduction ratio is 15. The convolution kernel size of the SA module is set to 7. The input channel of SA is 2 since the feature description of SA is obtained by cascading the output of the

**Table 2** Hyperparameter settings of the MT-ACNN-SA-BiLSTM model

| Components of the model | Hyperparameter | Value |
|---|---|---|
| MT | Training epochs | 60 |
| | Learning rate | 0.001 |
| | Dropout | 0.04 |
| | Gaussian noise | $\sigma = 1$ |
| | EMA decay rate | $\alpha = 0.95$ |
| | Consistency weight | weight = 0.2 |
| CNN | Conv1 | $k = 1 \times 25 \times 32$, padding = 12, stride = 1 |
| | Pool1 | $k = 1 \times 25 \times 32$, stride = 25 |
| | Conv2 | $k = 1 \times 25 \times 64$, padding = 12, stride = 1 |
| | Pool2 | $k = 1 \times 10 \times 64$, stride = 10 |
| | Conv3 | $k = 1 \times 5 \times 128$, padding = 2, stride = 1 |
| | Pool3 | $k = 1 \times 8 \times 128$, stride = 8 |
| | Conv4 | $k = 1 \times 5 \times 256$, padding = 2, stride = 1 |
| | Pool4 | $k = 1 \times 4 \times 256$, stride = 4 |
| Channel attention | FC1 | in_features = 256, out_features = 15 |
| | FC2 | in_features = 15, out_features = 256 |
| SA | Conv | $k = 1 \times 7 \times 1$, padding = 3, stride = 1 |
| BiLSTM | Hidden | 256 |

maximum pooling layer and the average pooling layer. On setting and activating the convolutional layer, it is necessary to pay attention to the matching problem between weights and input dimension, and to keep the output channel dimension as 1.

### 4.3 Evaluation metrics

To achieve high accuracy in the classification recognition task, some indicators below are chosen to evaluate the model performance.

Confusion matrix is an error matrix commonly used to visually evaluate the performance of a classification algorithm. The main diagonal represents the number of correctly classified samples, while the other cells indicate the number of misclassified samples. The darker the color is the higher the sample number would be. The confusion matrix is represented by $C$. Define the confusion matrix $C = [d_{ij}]$, $i \in [0, 5]$, $j \in [0, 5]$. Where, $i$ represents the $i$-th row of $C$, and $j$ represents the $j$-th column.

The nuisance alarm rate (NAR) and the false negative rate (FNR) are the major indicators to evaluate the reliability and possibility of Φ-OTDR in practical application. We define NAR to be the ratio of the number of false alarms in the system to the total number of alarms, while FNR is the ratio of the number of missed alarms to the total number of disturbances. Therefore, NAR and FNR could be calculated from the confusion matrix as shown below:

$$\mathrm{NAR} = \frac{\sum_{j=1}^{5} d_{0j}}{\sum_{i=0}^{5}\sum_{j=1}^{5} d_{ij}}, \tag{5}$$

$$\mathrm{FNR} = \frac{\sum_{i=1}^{5} d_{i0}}{\sum_{i=1}^{5}\sum_{j=0}^{5} d_{ij}}. \tag{6}$$

Accuracy, precision, recall, and F1-score are also selected to evaluate the model recognition performance.

### 4.4 Comparative experiments

Three sets of comparative experiments are designed to verify the effectiveness of our model.

**Experiment 1: comparison of classification effects of five models under MT framework and other semi-supervised models.** To verify the effectiveness of the MT semi-supervision framework, five models under two different frameworks are selected for comparative experiments. In this experiment, the proportion of label samples is set to be 10% of the total number of samples. Therefore, 1230 labeled

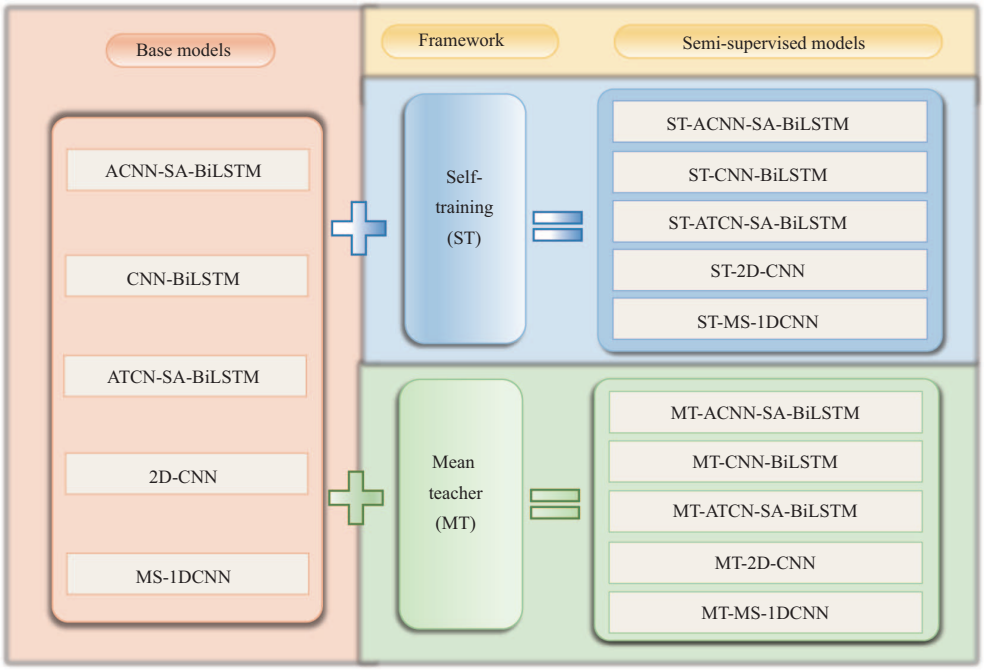


**Figure 5** (Color online) Models under the MT and ST frameworks.

samples and 11104 unlabeled samples are used for training. In addition to MT, another major semi-supervised framework, ST, is also investigated for comparison. ST refers to the Pseudo-Label model [19]. The five models are ACNN-SA-BiLSTM, MS-1DCNN [17], CNN-BiLSTM [18], ATCN-SA-BiLSTM [20] and 2D-CNN [34]. The first of these five models is our improved model, whose parameters are described in Subsection 4.2. The latter four are the SOTA models with desirable performance. MS 1D-CNN is an improved version of the 1D-CNN model, which adopts multi-scale convolution kernels to obtain richer features of signals on different scales, thus improving the classification performance. Both 1DCNNs-BiLSTM and ATCN-SA-BiLSTM models use special networks to respectively extract the temporal and the spatial features of the Φ-OTDR signal. The model parameter settings are referred to in the original literature [17–19], respectively. The parameter setting of 2D-CNN is detailed in Appendix B. 2D-CNN is a classical CNN model for processing 2D signals. We have chosen these models because they are the typically popular ones in data processing for Φ-OTDR. The parameter settings of the five models under the MT and ST frames are the same. To facilitate the explanation of the results, the names of the five models under the MT and ST frameworks are shown in Figure 5. By comparing the performance of the same model under two different frameworks, we can verify that the MT framework can improve the performance of each model.

In addition, we also compare the classification performance of semi-supervised methods in our domain on our dataset. These include semi-supervised support vector machine (SVM) methods such as transductive SVM (TSVM) [24], Laplacian SVM (LapSVM) [24], and SSAE [21] method.

**Experiment 2: ablation experiments in MT and ST frameworks.** To further objectively illustrate the effectiveness of each attention mechanism on the model, we set up the ablation experiment. At the same time, we verify the optimization effect of the model under two frameworks respectively. The dataset used in this experiment is the same as in Experiment 1.

Under the MT and ST frameworks, the channel attention module and the SA module of ACNN-SA-BiLSTM model are removed respectively to obtain MT-CNN-SA-BiLSTM, MT-ACNN-BiLSTM, ST-CNN-SA-BiLSTM, and ST-ACNN-BiLSTM models. Then we compare the performance indicators of the eight models: MT-CNN-BiLSTM, MT-ACNN-BiLSTM, MT-CNN-SA-BiLSTM, MT-ACNN-SA-BiLSTM, ST-CNN-BiLSTM, ST-ACNN-BiLSTM, ST-CNN-SA-BiLSTM, and ST-ACNN-SA-BiLSTM. The accuracy, NAR, FNR, and training time of the model under two frameworks with and without the two attention mechanisms are analyzed and compared.

**Experiment 3: comparing the supervised model and the semi-supervised model for different numbers of labeled samples.** To facilitate a detailed comparison of the classification performance

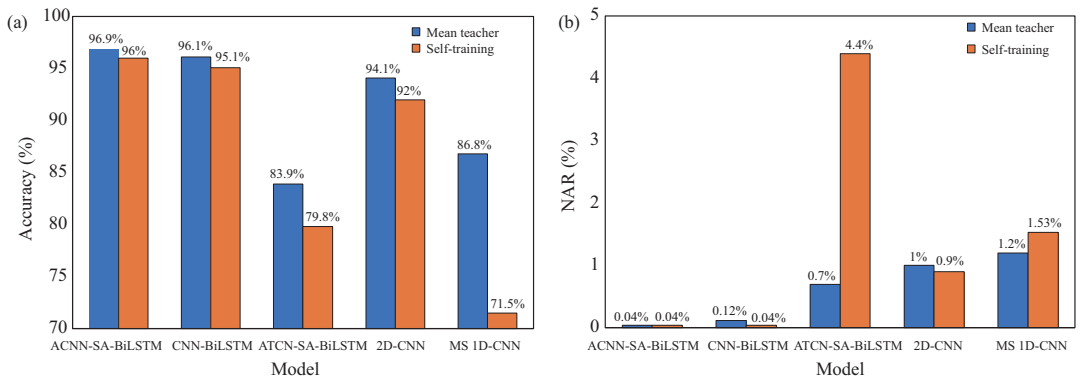


**Figure 6** (Color online) Comparison of (a) accuracy and (b) NAR under the two SSL frameworks.

across six categories with a limited number of labeled datasets, we set the number of labeled samples as 100, 250, 500, 1000, and 1230 for comparison experiments. The classification effect of the supervised ACNN-SA-BiLSTM model and the semi-supervised MT-ACNN-SA-BiLSTM model under the same number of labels are compared, to better verify the network performance when we train with different numbers of labeled samples. This experiment mainly simulates the situation that labeled samples are difficult to obtain and unlabeled samples exist in large quantities in field practice.

## 4.5 Result analysis

### 4.5.1 *The result of Experiment 1*

For Experiment 1, it can be seen from Figure 6(a) that the classification accuracy of the ACNN-SA-BiLSTM model is the highest under two semi-supervised frameworks, among the five models. The accuracy rate can reach more than 96%. Among the ten models corresponding to the two semi-supervised frameworks, the MT-ACNN-SA-BiLSTM model with a 96.9% classification accuracy is significantly better than other models. In general, the accuracy of the five models under the MT is higher than that under the ST, e.g., the accuracy of the MT-ACNN-SA-BiLSTM model is 0.9% higher than that of the ST-ACNN-SA-BiLSTM model. The MT-MS-1DCNN model demonstrates the most significant accuracy improvement within the MT framework, with a remarkable 15.3% higher accuracy compared to the ST-MS-1DCNN model. These results indicate that the MT semi-supervision framework is more effective. The NAR is shown in Figure 6(b). The NAR of the ACNN-SA-BiLSTM model and the ST-CNN-BiLSTM model are the lowest.

The differences in the performance of ten models are illustrated clearly in Figure 7. It can be seen that the MT-ACNN-SA-BiLSTM model's accuracy, precision, recall, F1-score, and NAR indexes are the most conspicuous. In addition, the NAR and FNR of the ST-ATCN-SA-BiLSTM model are the highest.

To provide a more intuitive representation of the recognition effect of the algorithm on each type of event, we also obtain the confusion matrixes of those ten models on the six-category dataset, as shown in Figure 8. The confusion matrix can be used to visualize the performance of the algorithm. Each column represents the predicted value, and each row represents the actual value. The value on the diagonal represents the number of correct predictions, and the value of the non-diagonal is the number of wrong predictions. In general, the deeper the diagonal color of the confusion matrix, the more accurate the prediction results. Non-diagonal numbers greater than 21 are indicated by green dotted lines. Hence, out of the ten models, the MT-ACNN-SA-BiLSTM model demonstrates superior recognition performance. At the same time, the precision, recall, and F1-score of each event are calculated according to the confusion matrix, as shown in Table 3.

It is shown that the MT-ACNN-SA-BiLSTM model has the best recognition effect on background noise, knocking, digging, watering, shaking, and walking events. ST-CNN-BiLSTM has the best recognition effect on background noise. The recognition rate of all models for the walking event is relatively low, which shows the difficulties of walking event recognition, the possible reason is that walking events contain samples with different frequencies, and some of them may contain only a single stampede. The MT-ACNN-SA-BiLSTM model can improve the recognition rate of the walking event. The precision is up to 97.6%, recall is up to 93.1%, and F1 is up to 95.3%. When it comes to events like knocking and digging, in reality, they bear a closer resemblance to walking, and our algorithm demonstrates a strong

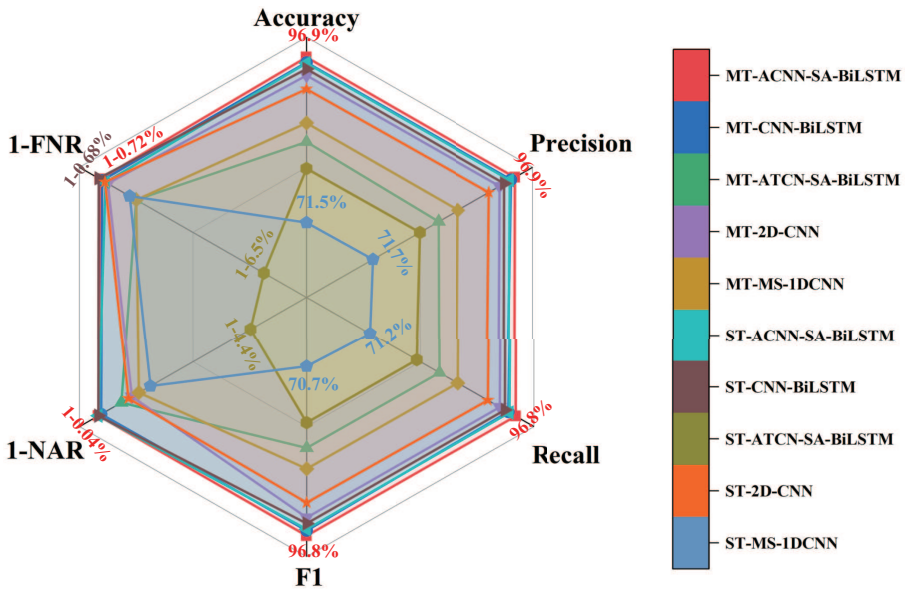


**Figure 7** (Color online) Performance comparison of 10 models.

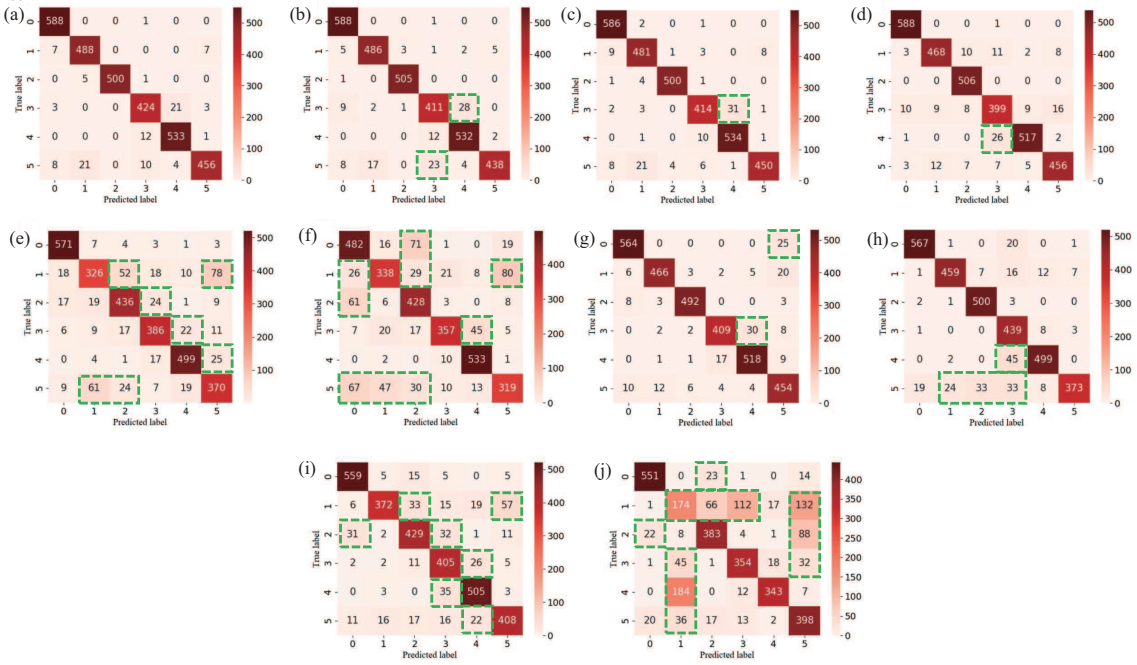


**Figure 8** (Color online) Confusion matrixes of 10 models. (a) MT-ACNN-SA-BiLSTM; (b) ST-ACNN-SA-BiLSTM; (c) MT-CNN-BiLSTM; (d) ST-CNN-BiLSTM; (e) MT-ATCN-SA-BiLSTM; (f) ST-ATCN-SA-BiLSTM; (g) MT-2DCNN; (h) ST-2DCNN; (i) MT-MS-1DCNN; (j) ST-MS-1DCNN.

capability to distinguish between these events. Overall, the MT-ACNN-SA-BiLSTM model has the best performance on event recognition. MT method demonstrates superior performance over the ST method, a distinction that can be theoretically explained. The MT model introduces consistency regularization, enhancing data distribution learning and generalization. In contrast, ST relies on potentially inaccurate pseudo-labels, leading to noise interference. Additionally, the 'teacher-student' structure in MT promotes stability, improving learning efficiency and generalization, unlike ST, which can suffer from instability due to uncertain pseudo-labels.

Our experiments are conducted on a desktop computer with an 11th Gen Intel (R) Core (TM) i7-11700KF @ 3.60 GHz CPU and GeForce RTX 3080 GPU, running a Linux system. The training time of ten models is shown in Table 4. The average training time for each epoch of the MT-ACNN-SA-BiLSTM

**Table 3** Performance comparison of the 10 models under two semi-supervised frameworks[a)]

| Model | Event type | MT | | | | ST | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 | Accuracy |
| ACNN-SA-BiLSTM | 0 | **0.970** | **0.998** | 0.984 | **0.969** | 0.962 | **0.998** | 0.980 | 0.960 |
| | 1 | **0.966** | **0.972** | **0.969** | | 0.962 | 0.968 | 0.965 | |
| | 2 | **1.000** | 0.988 | 0.994 | | 0.992 | 0.998 | **0.995** | |
| | 3 | 0.946 | 0.940 | **0.943** | | 0.917 | 0.911 | 0.914 | |
| | 4 | 0.955 | 0.976 | **0.966** | | 0.940 | 0.974 | 0.957 | |
| | 5 | 0.976 | **0.931** | **0.953** | | **0.984** | 0.894 | 0.937 | |
| CNN-BiLSTM | 0 | 0.967 | 0.995 | 0.981 | 0.961 | 0.972 | **0.998** | **0.985** | 0.951 |
| | 1 | 0.939 | 0.958 | 0.949 | | 0.957 | 0.932 | 0.945 | |
| | 2 | 0.990 | 0.988 | 0.989 | | 0.953 | **1.000** | 0.976 | |
| | 3 | **0.952** | 0.918 | 0.935 | | 0.899 | 0.885 | 0.892 | |
| | 4 | 0.943 | **0.978** | 0.960 | | **0.970** | 0.947 | 0.958 | |
| | 5 | 0.978 | 0.891 | 0.947 | | 0.946 | **0.931** | 0.938 | |
| ATCN-SA-BiLSTM | 0 | 0.919 | 0.969 | 0.944 | 0.839 | 0.750 | 0.818 | 0.782 | 0.798 |
| | 1 | 0.765 | 0.649 | 0.703 | | 0.788 | 0.673 | 0.726 | |
| | 2 | 0.816 | 0.862 | 0.838 | | 0.744 | 0.846 | 0.792 | |
| | 3 | 0.848 | 0.856 | 0.852 | | 0.888 | 0.792 | 0.837 | |
| | 4 | 0.904 | 0.914 | 0.909 | | 0.890 | 0.976 | 0.931 | |
| | 5 | 0.746 | 0.755 | 0.751 | | 0.738 | 0.656 | 0.695 | |
| 2D-CNN | 0 | 0.959 | 0.958 | 0.958 | 0.941 | 0.961 | 0.963 | 0.962 | 0.920 |
| | 1 | 0.963 | 0.928 | 0.945 | | 0.943 | 0.914 | 0.928 | |
| | 2 | 0.976 | 0.972 | 0.974 | | 0.926 | 0.988 | 0.956 | |
| | 3 | 0.947 | 0.907 | 0.926 | | 0.790 | **0.973** | 0.872 | |
| | 4 | 0.930 | 0.949 | 0.939 | | 0.947 | 0.914 | 0.930 | |
| | 5 | 0.875 | 0.927 | 0.900 | | 0.971 | 0.761 | 0.854 | |
| MS 1D-CNN | 0 | 0.918 | 0.949 | 0.933 | 0.868 | 0.926 | 0.935 | 0.931 | 0.715 |
| | 1 | 0.930 | 0.741 | 0.825 | | 0.389 | 0.347 | 0.367 | |
| | 2 | 0.850 | 0.848 | 0.849 | | 0.782 | 0.757 | 0.769 | |
| | 3 | 0.797 | 0.898 | 0.845 | | 0.714 | 0.785 | 0.748 | |
| | 4 | 0.881 | 0.925 | 0.903 | | 0.900 | 0.628 | 0.740 | |
| | 5 | 0.834 | 0.833 | 0.834 | | 0.593 | 0.819 | 0.688 | |

a) Bolded numbers signify the optimal values within the metrics.

model is 341.0 s. The MT method has a slightly longer average training time per epoch compared to the ST method. This can be explained by the additional computational overhead introduced by the 'teacher-student' structure in the MT method. While this structure adds a bit of extra computation, it is important to note that the difference in training time, although somewhat longer, is not substantial.

The classification results of TSVM, LapSVM, and SSAE methods are shown in Table 5. The recognition accuracy for TSVM and LapSVM is 95.2% and 91.0% respectively, while SSAE achieves an accuracy of 84.1%. Among the three models, TSVM exhibits the lowest FNR and NAR. Worth mentioning is the fact that these methods necessitate a preliminary manual feature extraction process, relying on human expertise. Furthermore, the SSAE method's training process is segregated into unsupervised and supervised phases, impeding seamless integration. In contrast, the MT method enables end-to-end processing, presenting a more straightforward approach. In conclusion, based on the experimental analysis conducted above, the MT-ACNN-SA-BiLSTM model demonstrates the best recognition performance among these models.

#### 4.5.2 *The result of Experiment 2*

For Experiment 2, the results of ablation experiments of the ACNN-SA-BiLSTM model under two semi-supervised frameworks are shown in Table 6. It can be seen that the accuracy of the CNN-BiLSTM model can be improved by adding a channel attention mechanism and an SA mechanism. Moreover, the accuracy could be further improved if both modules are employed. For the MT semi-supervision framework, the accuracy is improved by 0.8% while NAR is reduced by 0.08%. What is more, the training time of the MT-ACNN-SA-BiLSTM model is almost the same as that of the MT-CNN-BiLSTM model, or even less.

**Table 4** Training time of 10 models

| Model | Average training time per epoch (s) | Model | Average training time per epoch (s) |
|---|---|---|---|
| MT-ACNN-SA-BiLSTM | 341.0 | ST-ACNN-SA-BiLSTM | 278.9 |
| MT-CNN-BiLSTM | 344.6 | ST-CNN-BiLSTM | 273.3 |
| MT-ATCN-SA-BiLSTM | 339.8 | ST-ATCN-SA-BiLSTM | 343.3 |
| MT-2D-CNN | 365.8 | ST-2D-CNN | 262.6 |
| MT-MS 1D-CNN | 342.2 | ST-MS 1D-CNN | 213.2 |

**Table 5** Performance comparison of semi-supervised algorithms in the Φ-OTDR domain

| Model | Accuracy | Precision | Recall | F1 | NAR | FNR |
|---|---|---|---|---|---|---|
| TSVM | 0.952 | 0.952 | 0.950 | 0.950 | 0.0024 | 0.0084 |
| LapSVM | 0.910 | 0.910 | 0.904 | 0.905 | 0.0024 | 0.0160 |
| SSAE | 0.841 | 0.842 | 0.841 | 0.839 | 0.0375 | 0.0375 |

**Table 6** Results of ablation experiments under MT and ST frameworks[a)]

| Framework | Model | Accuracy (%) | NAR (%) | FNR (%) | Training time (s) |
|---|---|---|---|---|---|
| MT | MT-CNN-BiLSTM | 96.1 | 0.12 | 0.80 | 344.6 |
| | MT-ACNN-BiLSTM | 96.3 | 0.12 | 0.88 | **339.7** |
| | MT-CNN-SA-BiLSTM | 96.4 | 0.12 | **0.60** | 339.9 |
| | MT-ACNN-SA-BLSTM | **96.9** | **0.04** | 0.72 | 341.0 |
| ST | ST-CNN-BiLSTM | 95.1 | **0.04** | 0.68 | **273.3** |
| | ST-ACNN-BiLSTM | 95.9 | 0.24 | **0.28** | 276.0 |
| | ST-CNN-SA-BiLSTM | 95.3 | **0.04** | 0.88 | 275.6 |
| | ST-ACNN-SA-BiLSTM | **96.0** | **0.04** | 0.92 | 278.9 |

a) Bolded numbers signify the optimal values within the metrics.

For the ST framework, compared with the baseline model, the accuracy of the ACNN-SA-BiLSTM model is improved by 0.9% and the total training time is slightly increased. Overall, the classification effect of the model is improved after the employment of the attention mechanism. The accuracy can be improved without increasing the training time under the MT framework, which is a plus.

#### 4.5.3 *The result of Experiment 3*

For Experiment 3, we set different numbers of labels to compare the supervised model and the semi-supervised model. The results are shown in Figure 9. It can be seen that the classification effect of the semi-supervised MT-ACNN-SA-BiLSTM model is better than that of the supervised ACNN-SA-BiLSTM model on the whole. When an ample number of labeled samples ($> 1000$) is available, the classification accuracy of the supervised model and semi-supervised model has a smaller difference. However, as the number of labeled samples decreases and the number of unlabeled samples increases, the semi-supervised deep learning model has more advantages. When the number of labeled samples decreased to 100, the performance of the supervised ACNN-SA-BiLSTM model decreased significantly (accuracy decreased to 77.2%, and NAR increased to 3.07%), while the classification accuracy of semi-supervised MT-ACNN-SA-BiLSTM model remained above 83.5%, which is 6.3% higher than that of the supervised, and its NAR is only 0.69%, one magnitude smaller than its counterpart.

The experiment shows that: (1) with a large number of labeled samples, both the supervised and the semi-supervised models can extract event features correctly, so better and similar classification performance can be obtained on our dataset, though we need to mention the semi-supervised is 0.2% better in accuracy; (2) when the number of labeled samples decreases, the impact on MT-ACNN-SA-BiLSTM model is smaller. Especially when the number of labeled samples decreases to 100, the advantage of a semi-supervised deep learning model (MT-ACNN-SA-BiLSTM) is more obvious. The reason is quite straightforward, the SSL models could utilize the unlabeled samples while their supervised counterparts cannot. Since NAR is significantly reduced in SSL, it is obvious that the optimization using SSL is the key to distinguish between disturbance and non-disturbance events. It should also be noted that with a high NAR in supervised models, the Φ-OTDR field application would be greatly hindered.

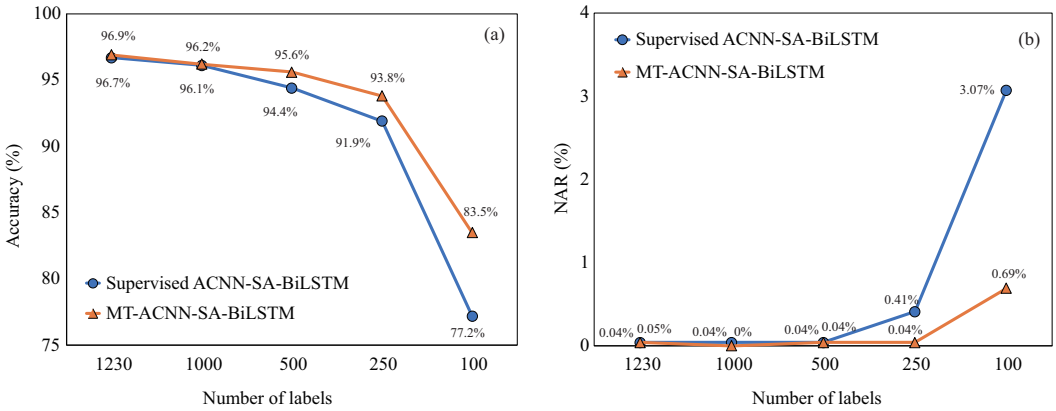


**Figure 9** (Color online) Comparison of (a) accuracy and (b) NAR of ACNN-SA-BiLSTM using supervised and semi-supervised (MT) methods.

## 5 Conclusion

We propose an MT framework for Φ-OTDR events classification, which fully utilizes the unlabeled samples that were not used in previous research and thus discarded. Based on the MT semi-supervision framework, the trained models can combine a large number of unlabeled samples and a small number of labeled samples for training, so that unlabeled samples can be fully utilized for auxiliary training to enhance classification performance. Then, an MT-ACNN-SA-BiLSTM model is put forward and compared with other SOTA models. In the proposed model, both time domain and space domain features are taken into account, and the channel attention mechanism and the SA mechanism are introduced to further optimize the ability of feature extraction. This model can improve the processing capability of samples with long-distance information, accelerate the speed of network fitting, and optimize the performance of event classification. The model is used to classify a dataset with six kinds of events, and the results show that the recognition accuracy of our model reaches 96.9%, which is the best among the five SOTA models. We also report that when the number of labeled samples decreases while the number of unlabeled samples increases, the SSL model is always better and it will gradually show relatively better classification performance than the supervised models.

The method proposed in this work can be an appropriate approach to tackle the practical difficulties. We have placed particular emphasis on the method of data processing for the collected signals because this approach is highly versatile when it comes to classifying interference events. Obtaining labeled samples in OTDR field applications is difficult and time-consuming, while the number of unlabeled samples is huge. Given the recent effectiveness of supervised methods in this field, the semi-supervised framework based on MT can facilitate the transfer of supervised deep learning models. Another significant advantage of our method is that it is especially suitable for training with new data collected in real-time, which can improve the accuracy and robustness of the model and is more suitable for complex online monitoring scenarios. Meanwhile, the tedious and time-consuming labor work can be saved by labeling the real-time collected samples.

**Acknowledgements** This work was supported in part by Fundamental Research Funds for Central Universities (Grant No. 021314380211), National Key Research and Development Program of China (Grant No. 2021YFB2900704), and Outstanding Chinese and Foreign Youth Exchange Program of China Association for Science and Technology. We are especially grateful to Ms. Manling TIAN, whose previous research on the semi-supervised model provided strong support for our research in this paper. At the same time, we would like to thank Ms. Shuman SUN for her help in setting 2D-CNN model's parameters.

**References**

1 Huang L J, He Z Y, Fan X Y. Simplified single-end Rayleigh and Brillouin hybrid distributed fiber-optic sensing system. Sci China Inf Sci, 2023, 66: 129404
2 Kandamali D F, Cao X, Tian M, et al. Machine learning methods for identification and classification of events in Φ-OTDR systems: a review. Appl Opt, 2022, 61: 2975–2997
3 Liang Y X, Wang Z N, Lin S T, et al. Optical-pulse-coding phase-sensitive OTDR with mismatched filtering. Sci China Inf Sci, 2022, 65: 192303
4 Li H, Fan C Z, Liu T, et al. Time-slot multiplexing based bandwidth enhancement for fiber distributed acoustic sensing. Sci China Inf Sci, 2022, 65: 119303

5 Tan D, Tian X, Sun W, et al. An oil and gas pipeline pre-warning system based on Φ-OTDR. In: Proceedings of the 23rd International Conference on Optical Fibre Sensors, Santander, 2014. 1269–1272

6 Juarez J C, Taylor H F. Field test of a distributed fiber-optic intrusion sensor system for long perimeters. Appl Opt, 2007, 46: 1968–1971

7 Juarez J C, Maier E W, Choi K N, et al. Distributed fiber-optic intrusion sensor system. J Lightwave Technol, 2005, 23: 2081–2087

8 Juarez J C, Taylor H F. Polarization discrimination in a phase-sensitive optical time-domain reflectometer intrusion-sensor system. Opt Lett, 2005, 30: 3284–3286

9 Qin Z, Chen L, Bao X. Wavelet denoising method for improving detection performance of distributed vibration sensor. IEEE Photon Technol Lett, 2012, 24: 542–544

10 Wu H J, Liu X Y, Rao Y J. Processing and application of fiber optic distributed sensing signal based on Φ-OTDR (in Chinese). Laser Optoelectron Prog, 2021, 58: 1306003

11 Xu C, Guan J, Bao M, et al. Pattern recognition based on time-frequency analysis and convolutional neural networks for vibrational events in $\varphi$-OTDR. Opt Eng, 2018, 57: 1

12 Shi Y, Wang Y, Zhao L, et al. An event recognition method for Φ-OTDR sensing system based on deep learning. Sensors, 2019, 19: 3421

13 Sun Q, Li Q, Chen L, et al. Pattern recognition based on pulse scanning imaging and convolutional neural network for vibrational events in Φ-OTDR. Optik, 2020, 219: 165205

14 Chen J, Wu H, Liu X, et al. A real-time distributed deep learning approach for intelligent event recognition in long distance pipeline monitoring with DOFS. In: Proceedings of International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Zhengzhou, 2018. 290–2906

15 Chen X, Xu C. Disturbance pattern recognition based on an ALSTM in a long-distance $\varphi$-OTDR sensing system. Micro Opt Tech Lett, 2020, 62: 168–175

16 Manie Y C, Li J W, Peng P C, et al. Using a machine learning algorithm integrated with data de-noising techniques to optimize the multipoint sensor network. Sensors, 2010, 20: 1070

17 Wu J, Guan L, Bao M, et al. Vibration events recognition of optical fiber based on multi-scale 1-D CNN. Opto-Electron Eng, 2019, 46: 180493

18 Wu H, Yang M, Yang S, et al. A novel DAS signal recognition method based on spatiotemporal information extraction with 1DCNNs-BiLSTM network. IEEE Access, 2020, 8: 119448

19 Tian M, Dong H, Cao X, et al. Temporal convolution network with a dual attention mechanism for $\varphi$-OTDR event classification. Appl Opt, 2022, 61: 5951–5956

20 Tian M, Dong H, Yu K. Attention based Temporal convolutional network for Φ-OTDR event classification. In: Proceedings of the 19th International Conference on Optical Communications and Networks (ICOCN), Qufu, 2021. 1–3

21 Yang Y, Zhang H, Li Y. Long-distance pipeline safety early warning: a distributed optical fiber sensing semi-supervised learning method. IEEE Sens J, 2021, 21: 19453–19461

22 He J, Hu X, Zhang D, et al. Semi-supervised learning for optical fiber sensor road intrusion signal detection. Appl Opt, 2022, 61: C65

23 Wang S, Liu F, Liu B. Semi-supervised deep learning in high-speed railway track detection based on distributed fiber acoustic sensing. Sensors, 2022, 22: 413

24 Cao X. Recognition of Φ-OTDR disturbed signal based on semi-supervised learning. Dissertation for the Master's Degree. Beijing: Beijing Jiaotong University, 2023

25 Bachman P, Alsharif O, Precup D. Learning with pseudo-ensembles. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, 2014. 3365–3373

26 Lee D H. Pseudo-label: the simple and efficient semi-supervised learning method for deep neural networks. In: Proceedings of Workshop on Challenges in Representation Learning, 2013

27 Li Z, Ko B, Choi H. Pseudo-labeling using Gaussian process for semi-supervised deep learning. In: Proceedings of IEEE International Conference on Big Data and Smart Computing (BigComp), Shanghai, 2018. 263–269

28 Wu H, Prasad S. Semi-supervised deep learning using pseudo labels for hyperspectral image classification. IEEE Trans Image Process, 2017, 27: 1259–1270

29 Rasmus A, Berglund M, Honkala M, et al. Semi-supervised learning with ladder networks. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, 2015

30 Laine S, Aila T. Temporal ensembling for semi-supervised learning. In: Proceedings of International Conference on Learning Representation (ICLR), 2016

31 Tarvainen A, Valpola H. Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017

32 Hu J, Shen L, Albanie S, et al. Squeeze-and-excitation networks. IEEE Trans Pattern Anal Mach Intell, 2020, 42: 2011–2023

33 Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module. In: Proceedings of 15th European Conference, 2018. 3–19

34 Cao X, Su Y, Jin Z, et al. An open dataset of $\varphi$-OTDR events with two classification models as baselines. Results Opt, 2023, 10: 100372

# Appendix A Details of the ST framework

## Appendix A.1 Principle of ST

In this work, the ST framework is adopted from [26]. The framework is combined with the five models mentioned in Subsection 4.4, and the sketch diagram of the training process is shown in Figure A1. First, the model is trained with labeled data. Then the trained model is used to predict pseudo-labels for the unlabeled data. Finally, the pseudo-labeled data is treated as labeled data, and the model is trained and updated with labeled data.

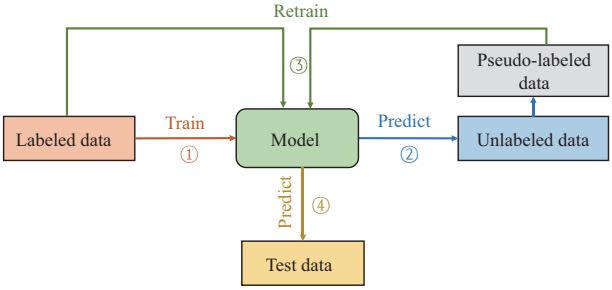


**Figure A1** (Color online) Sketch diagram of the ST framework.

The overall loss includes two parts. The first item is the loss of label data, and the second is the loss of pseudo-labeled data. Hence the overall loss function is

$$L = \frac{1}{n}\sum_{m=1}^{n}\sum_{i=1}^{C} L(y_i^m, f_i^m) + \alpha(t)\frac{1}{n'}\sum_{m=1}^{n'}\sum_{i=1}^{C} L(y_i'^m, f_i'^m), \tag{A1}$$

where $n$ is the number of mini-batch in labeled data, $n'$ is the number of mini-batch in unlabeled data, $f_i^m$ is the output units of $m$'s sample for labeled data, $y_i^m$ is the label of the labeled data, $f_i'^m$ is the output units for unlabeled data, $y_i'^m$ is the pseudo-label of the unlabeled data. $\alpha(t)$ is a balance coefficient. The selection of $\alpha(t)$ is very important for network performance. The model uses annealing algorithm [19] to calculate $\alpha(t)$, which is defined as

$$\alpha(t) = \begin{cases} 0, & t < T_1, \\ \dfrac{t - T_1}{T_2 - T_1}\alpha_f, & T_1 \leqslant t < T_2, \\ \alpha_f, & T_2 \leqslant t, \end{cases} \tag{A2}$$

where $t$ is the number of training. $T_1$ and $T_2$ are the separation points for the initial phase when the pseudo-labeled data are not involved in the network training and the fine-tuning phase, respectively.

## Appendix A.2 Parameter setting of the ST framework

On setting parameters for the ST network, several points should be taken into account. First of all, there is an important factor that in Lee [26]'s Pseudo-Label model training, the ratio of labeled data to unlabeled data in each batch is 1:1. However, in Φ-OTDR disturbance event classification, the ratio of both in the dataset cannot be guaranteed to be the same. Therefore, it is necessary to do a calculation based on the number of real label data ($N_{\text{label}}$) in the batch. When $N_{\text{label}} = N_{\text{batch}}$, the loss between the prediction result and its label value should be calculated according to the original method. When $N_{\text{label}} < N_{\text{batch}}$, the label value of $-1$ should be changed to the prediction with the highest probability. After the pseudo label obtained, the loss can be calculated according to (A1). It is important to note another parameter, the cross entropy loss function. The ignore index in the function should be set to be $-1$, meaning that when the label value of the input data is $-1$, the loss would not be calculated. This is because when unlabeled data is trained, the network assumes that its true label value is $-1$. If the loss between $-1$ and prediction results is directly calculated, the network will be optimized in the wrong direction. The ST parameters obtained from the actual training results are shown in Table A1. Note here, according to the principle of ST, the unlabeled data does not participate in the network training in the first round but participates in training from round 2 to 30 with the loss weight gradually increasing. After 30 rounds, the unlabeled loss weight is fixed at 0.3. The cosine annealing algorithm is used to adjust the learning rate, $T_{\max}$ is set to 5, and $\eta_{\min}$ is set to 0.

**Table A1** Hyperparameter settings of the ST network

| Components of the model | Hyperparameter | Value |
|---|---|---|
| Loss weight of unlabeled data | $T_1$ | 1 |
| | $T_2$ | 30 |
| | $\alpha_f$ | 0.3 |
| Cosine annealing algorithm | $T_{\max}$ | 5 |
| | $\eta_{\min}$ | 0 |

## Appendix B Parameter of the 2D-CNN model

The 2D-CNN network in this paper uses four convolutional layers, and BN is added after each convolutional layer. Dropout is also used in the model. The parameters are given in Table B1.

**Table B1** Hyperparameter settings of the 2D-CNN model

| Model | Hyperparameter | Value |
|---|---|---|
| 2D-CNN | Conv1 | $k = 3 \times 3$, stride $= 1 \times 1$, padding $= 1$ |
| | Pool1 | $k = 2 \times 400$, stride $= 2$ |
| | Conv2 | $k = 3 \times 3$, stride $= 1 \times 1$, padding $= 1$ |
| | Pool2 | $k = 2 \times 400$, stride $= 2$ |
| | Conv3 | $k = 3 \times 3$, stride $= 1 \times 1$, padding $= 1$ |
| | Pool3 | $k = 1 \times 400$, stride $= 2$ |
| | Conv4 | $k = 3 \times 3$, stride $= 1 \times 1$, padding $= 1$ |
| | Pool4 | $k = 1 \times 200$, stride $= 2$ |
| | dropout | 0.5 |
| | Linear | Classifier:$1 \times 6$ |