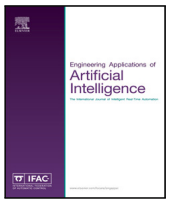
Engineering Applications of Artificial Intelligence 139 (2025) 109584

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



A fault diagnosis framework using unlabeled data based on automatic clustering with meta-learning

Zhiqian Zhao [a], Yinghou Jiao [a,*], Yeyin Xu [b], Zhaobo Chen [a], Enrico Zio [c,d]

[a] School of Mechatronics Engineering, Harbin Institute of Technology, Harbin, Heilongjiang 150000, PR China
[b] School of Astronautics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, PR China
[c] Centre for Research on Risk and Crises (CRC), Mine Paris-PSL University, Rue Claude Daunesse 1, 06904 Sophia Antipolis, France
[d] Energy Department, Politecnico di Milano, Via La Masa 34, 20156 Milan, Italy

ARTICLE INFO

Keywords:
Fault diagnosis
Meta-learning
Contrastive learning
Automatic clustering
Few-shot scenario

ABSTRACT

With the growth of the industrial internet of things, the poor performance of conventional deep learning models hinders the application of intelligent diagnosis methods in industrial situations such as lack of fault samples and difficulties in data labeling. To solve the above problems, we propose a fault diagnosis framework based on unsupervised meta-learning and contrastive learning, which is called automatic clustering with meta-learning (ACML). First, the amount of data is expanded through data augmentation approaches, and a feature generator is constructed to extract highly discriminative features from the unlabeled dataset using contrastive learning. Then, a cluster generator is used to automatically divide cluster partitions and add pseudo-labels for these. Finally, the classification tasks are derived through taking original samples in the partitions, which are embedded in the meta-learner for fault diagnosis. In the meta-learning stage, we split out two subsets from task and feed them into the inner and outer loops to maintain the class consistency of the real labels. After training, ACML transfers its prior expertise to the unseen task to efficiently complete the categorization of new faults. ACML is applied to two cases concerning a public dataset and a self-constructed dataset, demonstrate that ACML achieves good diagnostic performance, outperforming popular unsupervised methods.

1. Introduction

The industrial internet of things (IIOT) has gained significance of traction. Among the developments that IIOT can facilitate, there is great interest on intelligent fault diagnosis and predictive maintenance for industrial equipments (Lei et al., 2020; Feng et al., 2023; Liu and Li, 2024). This is obtained by the availability of massive operating data collected by sensors, which contain information valuable for decision-making. Most fault diagnosis methods are constrained by the need of a large amount of supervised data, but data labeling is costly and time consuming if possible (Zhao et al., 2024; Cha et al., 2024). Given the reliability of industrial equipment, which operates in challenging environments, and the instability of the system operating conditions, accurate data collection in fault states is difficult, which results in scarcity of labeled data (Mueller, 2024; Cha et al., 2018; Ouyang et al., 2024). Data scarcity generally leads to overfitting problems in fault diagnostic models. On the other hand, Unlabeled data are normally collected, and unsupervised learning methods are employed to perform representation learning and information mining on unlabeled data.

When data are scarce and traditional deep learning cannot generalize the tasks, meta-learning methods (Feng et al., 2022a; Lao et al., 2024; Tsialiamanis et al., 2024) with less hyperparametric optimization requirements and strong generalization ability become interesting to improve the performance of models on unknown tasks by drawing on prior knowledge to solve problem of few-shot fault diagnosis. The training of meta-learning can be divided into two phases: inner and outer loops (Zhang et al., 2022; Jia et al., 2024). The goal of the outer loop is to learn how to adapt quickly between multiple tasks and generate a model with better generalization capabilities. A set of model parameters that are more appropriate to the current task scenario are quickly updated and generated based on the current input dataset. The inner loop is based on this set of parameters to fine-tune the training to allow the model to better learn specific task scenarios. The inner loop and outer loop iterations alternate. By applying gradient descent to obtain the optimal initialization parameters, gradient-based meta-learning (Choe et al., 2024; Huisman et al., 2023) can update the initialization parameters by averaging the gradients

* Corresponding author.
E-mail addresses: 21b908069@stu.hit.edu.cn (Z. Zhao), jiaoyh@hit.edu.cn (Y. Jiao), xuyeyin@xjtu.edu.cn (Y. Xu), chenzb@hit.edu.cn (Z. Chen), enrico.zio@polimi.it (E. Zio).

https://doi.org/10.1016/j.engappai.2024.109584
Received 13 July 2024; Received in revised form 25 September 2024; Accepted 28 October 2024
Available online 8 November 2024

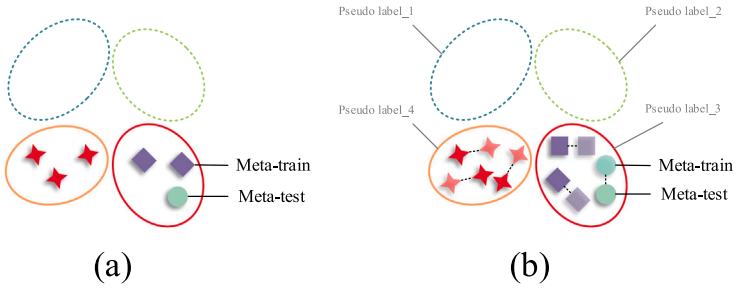


Fig. 1. Explanation of category consistency. Elliptical regions represent clusters with the same pseudo-label, and different shapes represent samples with different true labels. (a) Clustering bias leads to inconsistencies of true categories. (b) Ensuring consistency of true categories through contrastive learning.

between tasks, which will result in all tasks contributing equally to the model optimization (Vettoruzzo et al., 2024). Each task may contribute differently to updating the gradient, and a reasonable distribution of tasks is crucial for the generalization ability of the model. When the distribution of tasks is not reasonable, i.e., the gradients are not consistent, it may lead to a decrease in the generalization ability of the model (Baik et al., 2024). Existing researches on task gradient divergence problem mainly focus on network structure optimization and loss optimization. Network structure optimization methods include task hierarchy and feature fusion, but are less scalable (Hu et al., 2021b; Wang et al., 2021; Feng et al., 2022b); loss optimization methods reduce gradient disagreement by weighting different tasks or improving the loss function, but the optimization of hyperparameters relies on expert knowledge (Feng et al., 2021; Liu et al., 2022). Therefore, a reasonable task distribution should show enough regularity.

Model-Agnostic Meta-Learning (MAML) (Finn et al., 2017) is an optimization-based meta-learning approach that learns useful initialization parameters for quick adaptation to new tasks, but the fast learning process is hindered by the fact that MAML shares initialization in the task distribution (Baik et al., 2020). MAML averages the gradients between tasks to update the initialization parameters, ensuring that each task contributes equally to the optimization of the model. However, as tasks will always be distributed unevenly, each task contributes differently to the gradient update of the meta-model, causing MAML to conflict when the initialization position is altered (Zhao et al., 2023b). Transfer learning can gain a priori knowledge from the source task and improve the performance of the model by fine-tuning it to fit different scenarios. The combination of transfer learning and MAML can utilize the prior from the source tasks to calibrate the model and improve the performance of new tasks (Upadhyay et al., 2021). The difference between inner and outer loop tasks creates the conditions for this combination, and meta-learning performs fault diagnosis by means of meta-features related to transferability. In this paper, a task encoder is embedded in MAML to obtain the meta-representation of each sample while using it to add weights to the tasks. When the gradient of a task differs significantly from the average gradient of all tasks, the weight of such a task is automatically reduced by the task encoder, which in turn weakens its interference in updating the model parameters. Then, the weights obtained for different tasks are linked to the gradients for meta-optimization.

To enable the use of unlabeled data in meta-learning, this paper proposed to apply a clustering method (K-means) (Sinaga and Yang, 2020) to divide multiple cluster-based subsets and automatically, then, pseudo-labels are added to the samples in these subsets for constructing meta tasks. However,

1. When constructing the meta-task, due to the bias of clustering, it is not guaranteed that the meta-training set and the meta-test set have the same true labels if the samples in the subset are selected randomly, as shown in Fig. 1(a).
2. Simply embedding the original sample set into the unsupervised method does not result in distinctly differentiated clusters.

Unsupervised learning does not require expert knowledge in signal processing; it can directly extract abstract features from unlabeled data to determine sample distribution relationships (Aliramezani et al., 2022; Su et al., 2022; Cha and Choi, 2017). Most unsupervised learning fault diagnostic methods cluster unlabeled datasets using a metric-based technique that performs hard assignment by calculating the distance between measurement samples or models to label data samples into specific classes (Hu et al., 2021a; Xiao et al., 2021; Wang and Cha, 2021; Entezami et al., 2023). High-quality feature embedding can be obtained automatically by a carefully designed unsupervised encoder, which eliminates the tediousness of manual feature extraction, and greatly improves the classification accuracy of the downstream model (Zheng and Zhao, 2020). In terms of fault diagnosis, current feature embedding methods only consider the similarity of samples from the same category and do not consider the distance between samples from different categories, whereas the self-supervised method, as a kind of unsupervised learning, does (Huang et al., 2022; Wang et al., 2023).

For unlabeled datasets containing different faults, contrastive learning (Zhao et al., 2023a; Ding et al., 2022; Tang et al., 2024) can aggregate similar fault samples while increasing the differences between samples of different fault types (Wang et al., 2022). The feature generator in this paper is randomly used for data augmentation of vibration signals, ensuring the abundance and diversity of the dataset, and it is combined with the contrastive learning method to generate the more robust feature representation which improves the cluster differentiation. When the same sample is augmented twice, the results obtained by an encoder will be biased so that positive pairs are constructed in this way (Peng et al., 2022), and the remaining samples are used as negative pairs. The raw samples that have matched the pseudo-labels can be split to achieve the same real categories for the data input to the inner and outer loops, as shown in Fig. 1(b). Therefore, it must be ensured that category differentiation is high when adding pseudo-labels and that the data input to the inner and outer loops has the same real categories to obtain models with high classification accuracy.

An unsupervised fault diagnosis framework is designed by combining contrastive learning and the clustering method with meta-learning, automatic clustering with meta-learning (ACML), for solving few-shot fault diagnosis problem with unlabeled data.First, data augmentation mechanisms with contrastive learning are used as a feature generator to obtain high-quality feature representations. Then, the feature set is clustered using the cluster generator K-means, and partitions of the index set are constructed by adding pseudo-labels for original samples. The indexes are grouped according to their labels. Finally, cluster partitions are obtained by iterating the above steps to generate tasks. The original dataset that has obtained pseudo-labels is split in the meta-learner to ensure that samples with the same labels in the inner and outer loops belong to the same real class, the supervised meta-learning method is used to complete the fault diagnosis. Disagreement in the direction of the task update gradient can impair the performance of MAML during the initialization position update of the parameters, and

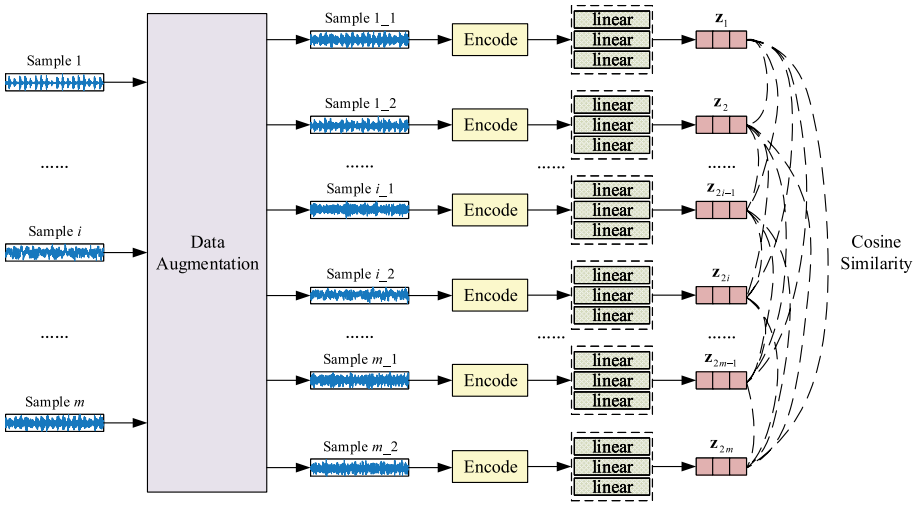


Fig. 2. The structure of the feature generator.

Adding weights for the tasks is proposed to obtain dynamic meta-representation, which weakens the effect of irrelevant information that hinders fast adaptation to obtain gradient agreement. The following are the primary contributions of this paper:

1. Fault diagnosis for unseen tasks is accomplished with unlabeled datasets. It is experimentally verified that ACML can achieve better performance than popular unsupervised meta-learning methods.
2. The purpose of the unsupervised method embedded in meta-learning is achieved by using contrastive learning and the clustering method, which also makes the ACML ensure a high degree of category differentiation when pseudo-labels are automatically added.
3. During the meta-learning, the accuracy of the meta-learning model is improved by separating the original dataset set into two sets and feeding them into the inner and outer loops as meta-training and meta-test data, respectively, ensuring category consistency.

2. Automatic clustering with meta-learning

2.1. Feature generator

In this paper, Gaussian noise (Li et al., 2020), Laplacian noise (Chen et al., 2022), mask noise (Li et al., 2020), and signal translation (Li et al., 2020) are selected for data augmentation, the structure of the feature generator is shown in Fig. 2. This is a simple and successful feature generation method in which the same data are utilized to randomly generate positive sample pairs using two of the above four data augmentation methods (Russell et al., 2024). Each data augmentation method has a trigger probability of 0.25, and the features are then extracted in an encoder. The detailed parameters of the four data augmentation methods are shown in Table 1. Given a mask $\mathbf{M}$, where the probability of each data point of the sample becoming 0 is η, otherwise the probability is 1, a_{tran} is the maximum translation length, $a_{tran} > 0$, where a positive value indicates forward translation and a negative value indicates backward translation. The above steps are repeated for each sample, making the similar features closer and rejecting the dissimilar ones.

For the dataset $\{\mathbf{X}_i\}_{i=1}^{m}$, An encoder $\mathbf{z}_i = f(\mathbf{X}_i)$ is designed, and then the samples that complete data augmentation are fed to the encoder to obtain $\mathbf{z}_i{}_i^m$. The deep residual shrinkage network (DRSN) (Zhao et al., 2019) is chosen as the encoder $f(*)$, which consists of 18 convolutional layers. The output layer of the DRSN is replaced by the projection head (Chen et al., 2020; Sun et al., 2023), which is a multilayer perceptron with 3 layers of dense, and the activation function is ReLU. The capacity of the projection head is increased by replacing the output layer of the DRSN with three linear layers. The nonlinear transformation projection head can improve the quality of the convolutional representation of its previous layer. Cosine similarity is used to determine the similarity of different samples.

Table 1
Detailed parameters of the data augmentation methods.

Method	Equation or explanation	Value
Gaussian noise	$\tilde{\mathbf{X}} = \mathbf{X} + \mathbf{g}, \mathbf{g} \sim N(\mu, \sigma^2)$	$\mu = 0, \sigma^2 = 0.03$
Mask noise	$\tilde{\mathbf{X}} = \mathbf{M} \cdot \mathbf{X}$	$\eta = 0.1$
Laplace noise	$\tilde{\mathbf{X}} = \mathbf{X} + \mathbf{L}, \mathbf{L} \sim \varsigma(\mu, b^2)$	$\mu = 0, b^2 = 0.02$
Signal translation	$[-a_{tran}, a_{tran}]$	$a_{tran} = 90$

Table 2
Structure of the projection.

Network	Output size	Layers
Projection	1024	Input feature representation
	1024	Linear (1024, 1024), BatchNorm1D, ReLU
	512	Linear (1024, 512), BatchNorm1D, ReLU
	64	Linear (512,64), ReLU

$$s(i,\upsilon) = \frac{\mathbf{z}_i^T \mathbf{z}_\upsilon}{\kappa \left\|\mathbf{z}_i\right\| \left\|\mathbf{z}_\upsilon\right\|}, i \neq \upsilon \tag{1}$$

where κ denotes the temperature parameter, and its role is to regulate the level of attention for difficult samples. It is assumed that $\mathbf{z}_i$ and $\mathbf{z}_j$ are positive pairs, and the softmax function is used to obtain the similarity probability of $(\mathbf{z}_i, \mathbf{z}_j)$.

$$g(i) = \frac{e^{s(i,j)}}{\sum_{\upsilon=1}^{2m} e^{s(i,\upsilon)}} \tag{2}$$

where m is the total number of samples. The resulting loss function L (Chen et al., 2020) is

$$l(i) = -\log(g(i)) \tag{3}$$

$$L = \frac{1}{2m} \sum_{\upsilon=1}^{m} [l(2\upsilon - 1, 2\upsilon) + l(2\upsilon, 2\upsilon - 1)] \tag{4}$$

Please refer to Table 2 for the detailed structure of projection.

After the feature generator completes the pretraining, the weights of the encoder and projection head are determined. When the downstream

task is arranged, the projection head is removed. By freezing the weights of the encoder, the output feature maps are used as proxies for representation quality. Rather than feeding $\{\mathbf{z}_i\}_i^m$ into the downstream model, the feature map in the encoder before the projection head is taken as the final representation $U = \{\mathbf{h}_i\}_{i=1}^m$.

2.2. Cluster generator

When there are only feature representations without labels, it is necessary to divide the representations with similar attributes into a cluster through cluster analysis, and then pseudo-labels can be automatically added to this cluster (Ding and Jia, 2021; Hsu et al., 2018; Pan et al., 2024). K-means (Sinaga and Yang, 2020) is applied as a cluster generator to better collect feature distribution information. The unsupervised algorithm K-means is first learned on $\mathbf{U}$ and then the data $\mathbf{h}_i$ is embedded into the space $\mathbf{Z}$ to obtain $\mathbf{z}_i$. To obtain different task sets, p partitions $P_p = (C_c)_p$ are generated by running the cluster generator p times. Denotedly, o_c is the centroid of the cluster C_c, the clustering process can be represented as

$$P = \underset{\{C_c\},\{o_c\}}{\arg\min} \sum_{c=1}^{k} \sum_{\mathbf{z}\in C_c} \|\mathbf{z} - o_c\|^2 \tag{5}$$

Each task is constructed by sampling partitions from $\mu(P)$, for $h_i \in C_c$, specifying pseudo-label $l_i = c$. Significantly, a pseudo label l_i is added to the corresponding X_i by means of h_i. The C is much larger than the true number of categories. Following the addition of pseudo-labels to the cluster features, C different categories are built, multiple samples are taken individually for each of the constructed categories, and C different one-hot vectors are used as task-specific labels. Concomitantly, the better the clustering, the fewer incorrectly labeled categories there will be, and the model will be more accurate.

When the number of clusters is small, K-means clustering, due to its unconstrained nature, often results in clusters containing multiple true labels. This leads to unbalanced tasks and a higher incidence of pseudo-labeling errors. A higher number of clusters improves label consistency within clusters, minimizing pseudo-labeling errors and benefiting meta-learning (Bertugli et al., 2020).

2.3. Meta-learner

By generating labeled tasks, the meta-learner proposed in this paper is the supervised meta-learning method. The core principle of MAML is to find a better initial parameter θ for an unknown task, allowing the model to learn quickly with fewer gradient steps to obtain high accuracy. MAML in the meta-training consists of inner loops and outer loops. In the inner loops, the optimal parameter θ_n' for each task τ_n is discovered, and in the outer loops, the gradient of the optimal parameter θ_n' in each new task needs to be calculated to update the randomly initialized model parameters θ (Zhang et al., 2021). The data input to the inner and outer loops should be different. A gradient descent update mechanism is proposed to obtain an improved MAML (iMAML), as shown in Fig. 3. Two networks with mutually independent parameters are first built: a task encoder t parametrized by θ and a base learner parametrized by ω. By combining these two networks, $\varphi_{\theta,\omega}(x) = \zeta\omega(t\theta(x))$ is obtained. By adding a task encoder in the outer loop to learn the meta-representation of each task instead of learning the model initialization, the adaptive capability of the model can be improved (Zhao et al., 2023b). When performing the gradient computation in the inner loop, only the parameters of the base learner are updated, and then the task encoder and the base learner are updated simultaneously in the outer loop. The task encoder is used to minimize the meta objective, and the parameters of this network are fixed in the meta test phase.

In the task encoder, the ith layer tensor x^i performs the convolution operation as follows,

$$\hat{x}^i = w^i * x^i + b^i \tag{6}$$

where $*$ is a convolution operation, w^i is the convolution weight of the ith layer tensor, b^i is the convolution bias of the ith layer tensor, and $\hat{x}^i$ is the output of the convolution. Further,

$$w^i = w_s^i * w_t^i \tag{7}$$

where w_s^i is the shared weight of all tasks, which is fixed in the inner loop, and w_t^i is the weight of the current task.

For each task $\tau = \{...((\mathbf{x}_i, \bar{\mathbf{x}}_i), l_c)...\}$, $\mathbf{X}$ is divided equally into $\mathbf{x}$ and $\bar{\mathbf{x}}$. N feature samples are taken for analysis. To ensure that the datasets input to the inner and outer loops have the same classification labels, The datasets are split into $D_1^n = \{...(\mathbf{x}_i, l_c)...\}$, $D_2^n = \{...(\bar{\mathbf{x}}_i, l_c)...\}$, and then input D_1^n to the inner loop to train the model. The optimal set of parameters for loss minimization is found as follows

$$\omega_n' = \omega_n' - \alpha\nabla_{\omega_n'} L_{\tau_n}(\varphi_{\theta,\omega_n'}(\{\mathbf{x}_i\}_{i=1}^m, \{l_c\}_{c=0}^C)) \tag{8}$$

where θ_n' is the optimal parameter for task τ_n, α is the hyperparameter, and $\nabla_{\omega_n'} L_{\tau_n}(\varphi_{\theta,\omega_n'})$ is the gradient of task τ_n.

The optimal parameters are found by the above steps for each τ_n in T. In the outer loops, the gradients of these optimal parameters ω_n' are computed. D_1^n is sampled to obtain the task τ_n and inputted into the model for training to update the random initialization parameters θ.

$$\theta = \theta - \beta\nabla_\theta \sum_{\tau_n} L_{\tau_n}(\varphi_{\theta,\omega_n'}(\{\bar{\mathbf{x}}_i\}_{i=1}^m, \{l_c\}_{c=0}^C)) \tag{9}$$

where β is the hyperparameter and $\nabla_\theta \sum_{\tau_n} L_{\tau_n}(\varphi_{\theta,\omega_n'})$ is the gradient of the new task τ_n with respect to the parameters ω_n'. The loss function L_{τ_n} of the meta-learner is shown as follows,

$$L_{\tau_n}(\varphi_{\theta,\omega}) = \sum_{\mathbf{x}_i, l_i \sim \tau_n} l_c \log\left(\varphi_{\theta,\omega}(\mathbf{x}_i)\right) + \left(1 - l_c\right)\log\left(1 - \varphi_{\theta,\omega}(\mathbf{x}_i)\right) \tag{10}$$

The ACML algorithm is detailed in Algorithm 1.

Algorithm 1 Automatic Clustering with Meta-learning

Require: Unlabeled dataset: $D = \{\mathbf{X}_i = (\mathbf{x}_i, \bar{\mathbf{x}}_i)\}_{i=1}^m$, Cluster generator algorithm: δ, Feature Generator Algorithm: ψ, hyperparameters: α, β.

```
Run the algorithm ψ on D to generate the feature sets U = {h_i}_{i=1}^m;
Run the algorithm δ on the observations {h_i} of the feature set U to generate partitions {P_p = (C_c)_p};
Construct task τ = {...(X_i, l_c)...} from the clustering partition {P_p}, where l_c the pseudo-label;
while not done do
    randomly initialize ω;
    Sample a batch of tasks T;
    for each τ_n in T do
        Sample dataset from the constructed task τ_n and split them into D_1 = {...(x_i, l_c)...} and D_2 = {...(x̄_i, l_c)...};
        ω'_n = ω;
        Calculate ∇_{ω'_n} L_{τ_n}(φ_{θ,ω'_n}) using D_1 ;
        Update the model to obtain the adaptive parameters ω'_n = ω'_n − α∇_{ω'_n} L_{τ_n}(φ_{θ,ω'_n}) for τ_n;
    end for
    Update θ ← β∇_θ Σ_{τ_n} L_{τ_n}(φ_{θ,ω'_n}) using D_2 ;
end while
Obtain optimized parameters θ and task-specific weights ω.
```

The framework of ACML is shown in Fig. 4. In the meta-testing phase, the prior knowledge gained in the unsupervised meta-training is extended to unseen tasks. The structure of meta learner is shown in Table 3, where N indicates the number of categories.

Flowchart of the proposed model is shown in Fig. 5. In the meta-testing phase, the optimized parameters learned from meta-training are used. A small number of gradient updates are performed based on the support set to update the model parameters to quickly adapt to the new tasks, and then the updated parameters are used for fault prediction on the query set.

3. Experimental verification

For the purpose of validating the performance of this method, the Case Western Reserve University (CWRU) dataset (Smith and Randall, 2015) and a rotor dataset obtained through a self-built test bench are

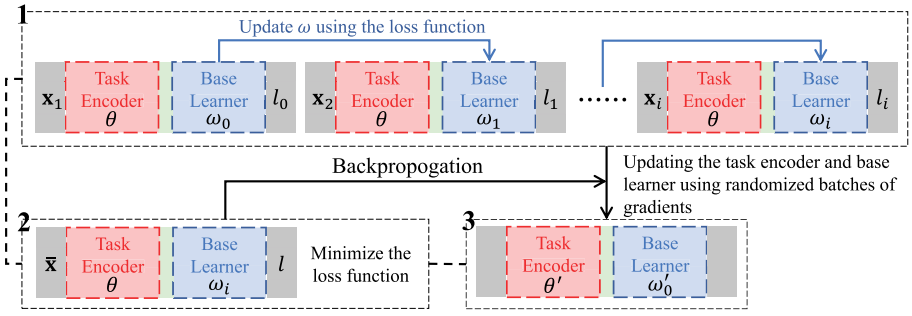


Fig. 3. Gradient descent update mechanism.

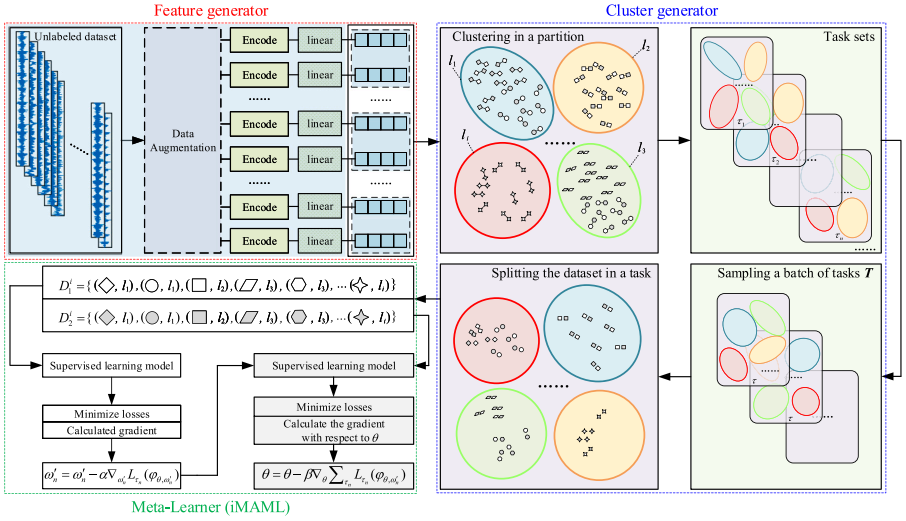


Fig. 4. The framework of ACML. The feature generator is used to obtain more discriminative feature embeddings, prompting downstream module to obtain more accurate pseudo-labels; the cluster generator is employed to obtain pseudo-labels and also to match these pseudo-labels for the original samples. and unseen faults are diagnosed by the trained meta-learner.

Table 3
Structure of the meta learner.

Network	Output size	Layers
Task encoder	1 × 1024	Input feature representation
	64 × 512	Conv1D(3 × 1, stride 1, padding 1), BatchNorm1D, ReLU, MaxPool (2 × 1, stride 2)
	64 × 256	Conv1D(3 × 1, stride 1, padding 1), BatchNorm1D, ReLU, MaxPool (2 × 1, stride 2)
	64 × 128	Conv1D(3 × 1, stride 1, padding 1), BatchNorm1D, ReLU, MaxPool (2 × 1, stride 2)
Base learner	64 × 64	Conv1D(3 × 1, stride 1, padding 1), BatchNorm1D, ReLU, MaxPool (2 × 1, stride 2)
	64 × 32	Conv1D(3 × 1, stride 1, padding 1), BatchNorm1D, ReLU, MaxPool (2 × 1, stride 2)
	64 × 16	Conv1D(3 × 1, stride 1, padding 1), BatchNorm1D, ReLU, MaxPool (2 × 1, stride 2)
	64 × 8	Conv1D(3 × 1, stride 1, padding 1), BatchNorm1D, ReLU, MaxPool (2 × 1, stride 2)
	512	Flatten
	N	Linear (512, N), SoftMax

employed to validate the effectiveness of ACML in few-shot scenarios. There is no supervised information added to the two datasets, and only the samples are normalized. There is no overlap between the source and target domain data.

3.1. Datasets

(1) Case Western Reserve University (CWRU) dataset. This dataset is widely used in the field of fault diagnosis for rotating machinery. The vibration data of the bearings at the drive end and the fan end under different operating conditions (0 HP, 1 HP, 2 HP and 3 HP) are used to study four states: normal (NOR), inner ring fault (IF), outer ring fault (OF) and rolling element fault (RF). Each fault is characterized by four diameters, which are 0.007, 0.014 and 0.021 inch respectively. The sampling frequency is 12 kHz. As shown in Table 4, 76 fine-grained states are considered in this experiment, and only the vibration signal is normalized. The data set was randomly divided into a meta-training set consisting of 62 categories and a meta-test set consisting of 14 unseen categories. In the training set, the number of samples per category is 220, and similarly, the number of samples per category in the test set is also 220.

(2) Rotor dataset. The rotor test bench is shown in Fig. 6. It consists of motor, coupling, sliding bearing and rod-fastened rotor. The displacement signals in the two orthogonal directions of the rotating shaft are measured by eddy current sensors, and the vibration signals on the bearing support at both ends are measured by accelerometers. This experiment consists of four different states: normal condition (NC), misalignment (Mis), rub-impact (Rub), and a combination of rub-impact and misalignment (Com). Mis has three levels, 0.3 mm, 0.6 mm and 0.9 mm. The above four states were collected in the test cases of 50

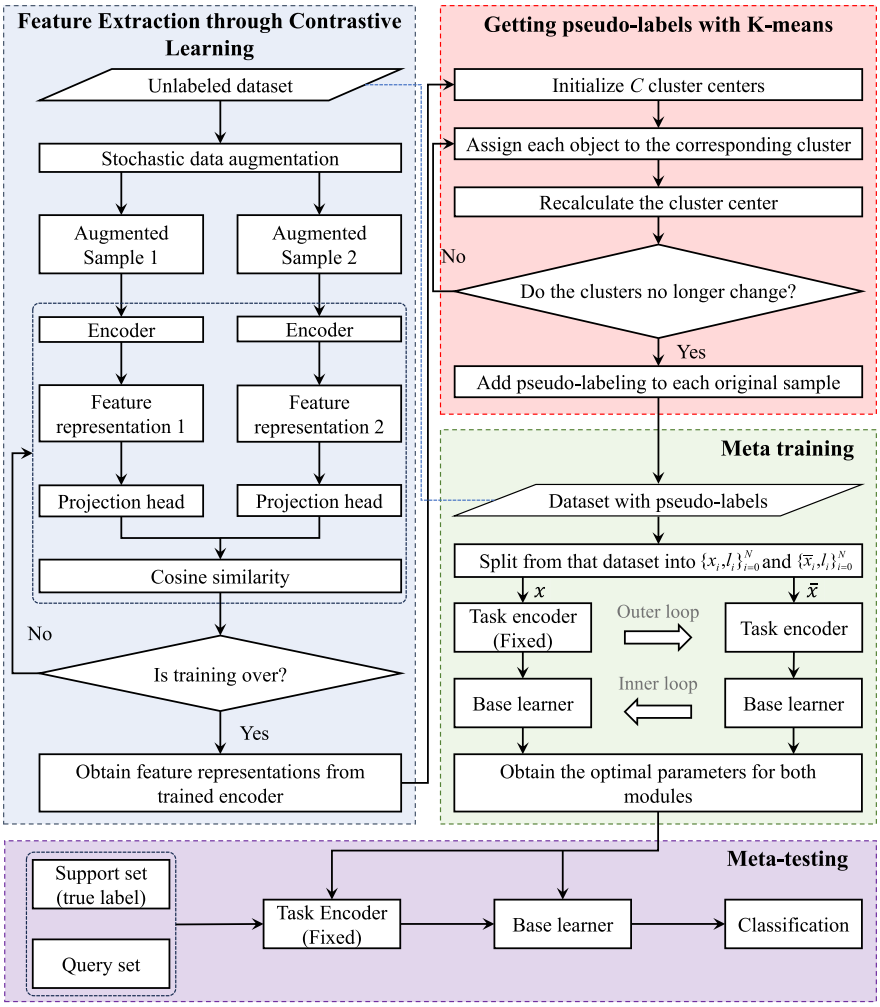


Fig. 5. Flowchart of the proposed model.

Table 4
Fault information of CWRU dataset.

Location	Speed (rpm)	Load (hp)	Status	Fault size (inch)	Category	Train/Test
Drive end	1797, 1772, 1750, 1730	0,1, 2,3	NOR, IF, RF, OF	0.007, 0.014, 0.021, 0.028	$4+15\times4=64$	220/220
Fan end			OF	0.007, 0.014, 0.021	$3\times4=12$	220/220

Table 5
Fault information of rotor dataset.

Location	Speed (rpm)	Status	Category	Train/Test
Pre-70	800, 1000, 1200, 1400		$4\times8=32$	200/200
Pre-90		NC, Rub, Mis-0.3, Mis-0.6, Mis-0.9, Com-0.3, Com-0.6, Com-0.9	$4\times8=32$	200/200
Pre-50	400, 600, 800, 1000,1200		$5\times8=40$	200/200

N m (Pre-50), 70 N m (Pre-70), and 90 N m (Pre-90), and the sampling frequency was set to 3.2 KHz. Also, the experiment considered different rotational speeds under various preload states, as shown in Table 5. The experiment has 104 fine-grained faults. The dataset was randomly divided into a meta-training set with 84 categories and a meta-test set with 20 unseen categories. In the training set, the number of samples per category is 200 and also, the number of samples per category in the test set is 200.

The fault is modeled in accordance with Fig. 7. Samples with various working conditions are chosen, utilizing only the vibration data in the X-direction of the right bearing support as input. The vibration signals at different preload as well as at different rotational speeds are shown in Fig. 8. Vibration data is highly affected by noise. At low speed, the vibration amplitude is greater, and a larger preload reduces the amplitude. Rub-impacts produce distorted vibration waveforms with clipping. In coupling fault, the effect of misalignment dominates.

3.2. Implementation details

To assess the diagnostic capability of ACML, training is conducted using the unlabeled dataset, and then unseen faults are diagnosed by sampling 1 or 5 examples per state throughout the meta-learning process. First, the CWRU dataset and the rotor dataset are enhanced with high-quality abstract features created by the feature generator; at this stage, the number of input sample data points is 2048, and the number of output feature vectors $\mathbf{h}_i$ is 1024. The dimension of $\mathbf{x}$ and $\bar{\mathbf{x}}$ is 1024. The cluster generator is run iteratively to construct

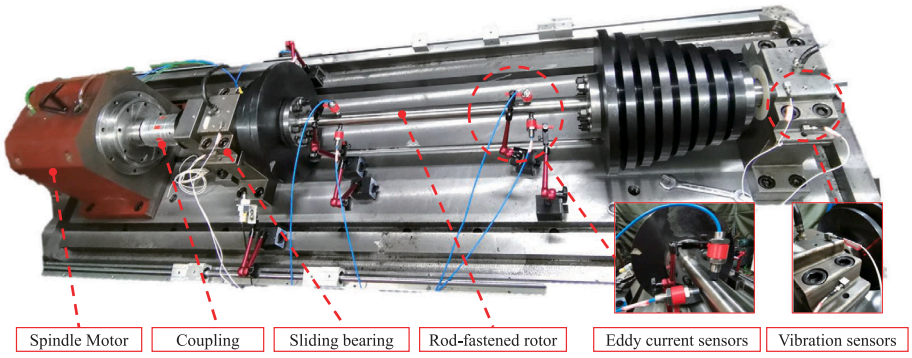


Fig. 6. Rotor test bench.

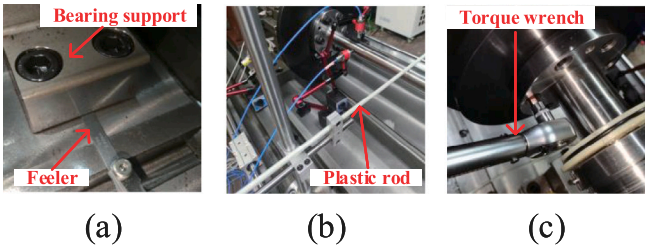


Fig. 7. Different forms of faults. (a) The bearing support is padded with feelers to create the misalignment fault; (b) the plastic rod is used to contact the rotor to create the rub impact fault; and (c) various preloads are applied to the rod-fastened rotor using a torque wrench.

Table 6
Hyperparameter settings of ACML.

Parameter	Value	Parameter	Value
Feature generator optimizer	Adam	Meta-batch	16
Feature generator learning rate	0.005	Meta-iterations	600
Temperature	0.1 or 0.2	Tasks	1000
Outer loop optimizer	Adam	Partitions	50
Outer loop learning rate	0.001	Inner loop learning rate	0.05
Inner loop optimizer	SGD	Inner gradient step	1

tasks, which are fed into the meta-learner iMAML for fault diagnosis. Hyperparameter settings of ACML are described in Table 6.

In this paper, supervised meta-learning methods MAML (Finn et al., 2017), ProtoNet (Feng et al., 2022a) and few-shot defect recognition (FSDR) (Liu et al., 2023), self-supervised methods 1D-BYOL (Peng et al., 2022), MOCO (He et al., 2020), and unsupervised meta-learning method pseudo-labeling framework based on a clustering-friendly feature embedding (PL-CFE) (Dong et al., 2022) are considered for comparative experiments with the present method, and the details of the comparison methods are shown in Table 7.

3.3. Results and analysis

The diagnostic results using different methods on the two datasets are reported in Table 8. For the supervised fault diagnosis model, the number of state types is known. In Table 8, (N, K) means that each time N classes are sampled from the meta-dataset, there are K samples in each category. In the training phase, the number of clusters C on the CWRU dataset is set to 100 and the number of clusters C on the rotor dataset is set to 250. The temperature κ is 0.1 for the CWRU dataset and 0.2 for the rotor dataset.

As shown in Table 8, for diagnosis of unseen faults under CWRU dataset, ACML can obtain an average accuracy of 91.66%, which is around 1.46% higher than ProtoNet and about 0.06% lower than MAML. When only 5 examples were sampled, ACML achieves an average accuracy of 97.01%, which is about 1.54% higher than MAML and ProtoNet. The Diagnostic results of each method under the rotor dataset are shown in Table 8, ACML achieves similar performance to supervised meta-learning. For the tasks of diagnosing unseen faults in both datasets, ACML can obtain an average accuracy of 79.15%, which achieves higher accuracy than the self-supervised learning methods 1D-BYOL, MOCO. 1D-BYOL and MOCO obtain the predicted labels by obtaining the similarity between the support set and the query set, but there is no processing for the label consistency. So, the diagnostic performance of 1D-BYOL, MOCO is lower than that of ACML. It proves the effectiveness of constructing meta-learning-based fault diagnosis task on the basis of self-supervised learning. After pre-training with contrastive Learning, an average accuracy of 79.25% on both datasets was achieved by FSDR by calculating the distance between the labeled data and the prototype. Because of the effect of labeling, FSDR exceeds the accuracy of the present method in the 1-shot tasks. The present method can achieve similar results as the combined supervised meta-learning and contrastive learning method FSDR. PL-CFE obtains embedded features by contrast learning, k-means is run on them to generate clusters for pseudo-labeling, and finally entropy is used to construct clean few-shot tasks. An average accuracy of 77.19% is achieved by this method on both datasets. In the 1-shot tasks, the higher accuracy obtained by PL-CFE over ACML is due to the fact that fewer samples indicate more confusing information about the faults, and the introduction of entropy in PL-CFE removes samples that are meaningless for updating the fault diagnosis performance. Whereas, in the 5-shot tasks, the labeling consistency of the samples dominates the diagnostic performance because of the increase in the number of samples, allowing ACML to obtain a higher accuracy than PL-CFE.

In this paper, t-SNE (Anowar et al., 2021) is used to visualize the task set in the meta-testing phase, as shown in Figs. 9 and 10. In the 5-K meta-test, feature representations with the same label are clustered into groups, and clear boundaries are revealed between each group, which is particularly evident in the CWRU dataset. The distribution of source and target domains is more inconsistent in the rotor dataset, but the features of each state begin to separate during the unsupervised domain adaptation. Therefore, in those comparison experiments, ACML outperforms other unsupervised approaches in few-shot situations and can achieve classification accuracy comparable to supervised meta-learning methods.

As shown in Fig. 11, the diagnostic performance of the two datasets is demonstrated for different number of clusters. When the number of clusters is less than the true category, the performance of the model is poor. And when the number of clusters is equal to the real categories, the accuracy of the model is less than the performance when the number of clusters is greater than the real categories. This is because a decrease in the number of clusters leads to a decrease in category consistency in the feature representation. As the number of clusters increases, the predictive accuracy of the model increases and

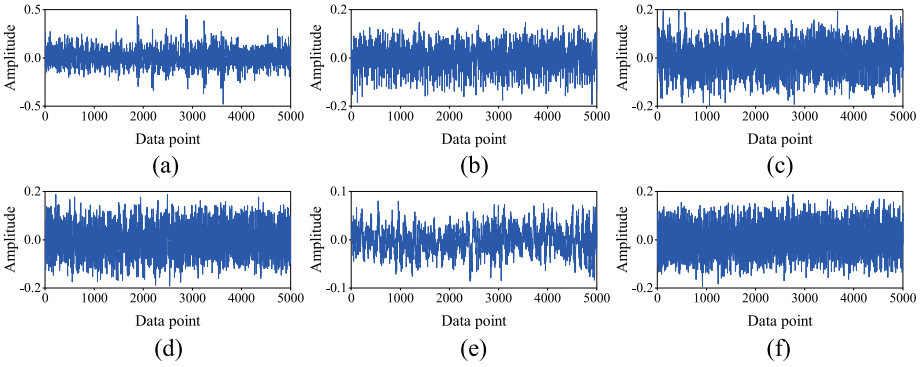


Fig. 8. Vibration signals of rotor data sets at different speeds and different preloads.(a) Normal condition and Preload 50 N m, speed is 800 rpm; (b) Normal condition and Preload 90 N m, speed is 800 rpm; (c) Normal condition and Preload 90 N m, speed is 1200 rpm; (d) Misalignment, speed is 1000 rpm; (e) Rub-impact, speed is 1000rpm; (d) Rub-impact and misalignment, speed is 1000 rpm.

Table 7
Description of the comparison methods.

Method	Type	Feature generator	Meta learner	Main setting
MAML	Supervised	N/A	MAML	MAML framework is the same as base learner, refer to Table 3 for detailed parameters.
ProtoNet	Supervised	Base learner	ProtoNet	The optimizer is Adam, with a learning rate of 0.001, the distances are calculated between the classlevel prototypes and the embeddings of the query points for classification.
1D-BYOL	Self-supervised	DRSN	N/A	The data augmentation method as well as the temperature coefficients are the same as ACML; momentum is 0.995 and queue size is 16384; the optimizer is SGD, the learning rate is 0.01, and the predicted labels are obtained by calculating the similarity between the support set and the query set (Jang et al., 2023).
MOCO	Self-supervised	DRSN	N/A	The data augmentation method is same as ACML, the optimizer is SGD, the learning rate is 0.01; momentum is 0.999 and queue size is 16384; the predicted labels are obtained by calculating the similarity between the support set and the query set (Jang et al., 2023).
FSDR	Supervised	1D-BYOL	DRSN	The relevant parameters are consistent with 1D-BYOL; the DRSN is pre-trained using the comparative learning method 1D-BYOL; the parameters of the DRSN obtained through pre-training will initialize the meta-learner in the meta-learning phase; the diagnostic results are obtained by calculating the distance between the query set and the prototype.
PL-CFE	Self-supervised	Contrastive learning	iMAML	The data augmentation method as well as the temperature coefficients are the same as ACML; randomly select N clusters as base clusters and select candidate clusters using K nearest neighbors; the clusters with the highest entropy are selected using the evaluation model to filter out the noisy samples.

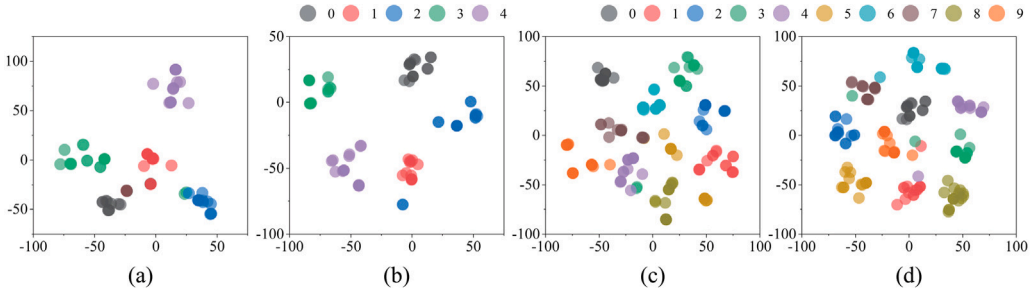


Fig. 9. The visualization of ACML for CWRU dataset in different tasks. (a) 5-way 1-shot task, (b) 5-way 5-shot task, (c) 10-way 1-shot task, (d) 10-way 5-shot task.

then decreases. This decrease is attributed to the problem of limited diversity in feature representations as the number of clusters increases. Therefore, the number of clusters was chosen to be 100 for the CWRU dataset and 250 for the rotor dataset.

The feature representations after contrastive learning were visualized as shown in Fig. 12, where the first 10 categories in the CWRU dataset were selected for visualization, while the first 19 categories in the rotor dataset were selected for visualization. The results show that samples with similar features are aggregated together in the absence of labels, providing a basis for obtaining more accurate pseudo-labels in downstream tasks.

A comparative analysis was conducted by applying data augmentation to clusters with inconsistent sample sizes. Table 9 presents the performance comparison between unbalanced and balanced task settings. The results demonstrate that models trained on unbalanced data outperform those trained on balanced clusters in terms of generalization ability, owing to the preservation of higher task diversity. Data augmentation shows a limited impact on model performance, and this

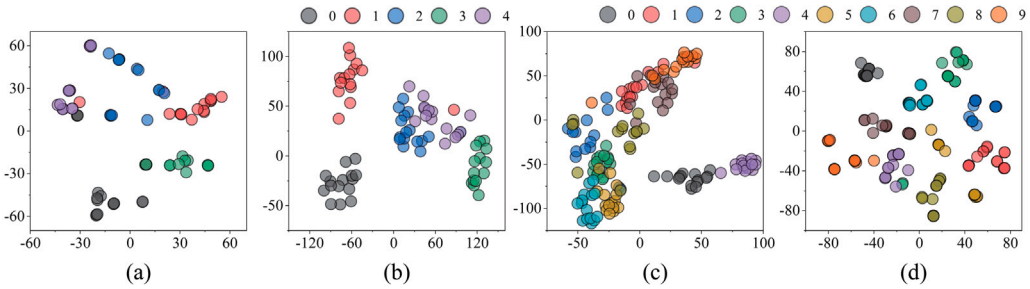


Fig. 10. The visualization of ACML for rotor dataset in different tasks. (a) 5-way 1-shot task, (b) 5-way 5-shot task, (c) 10-way 1-shot task, (d) 10-way 5-shot task.

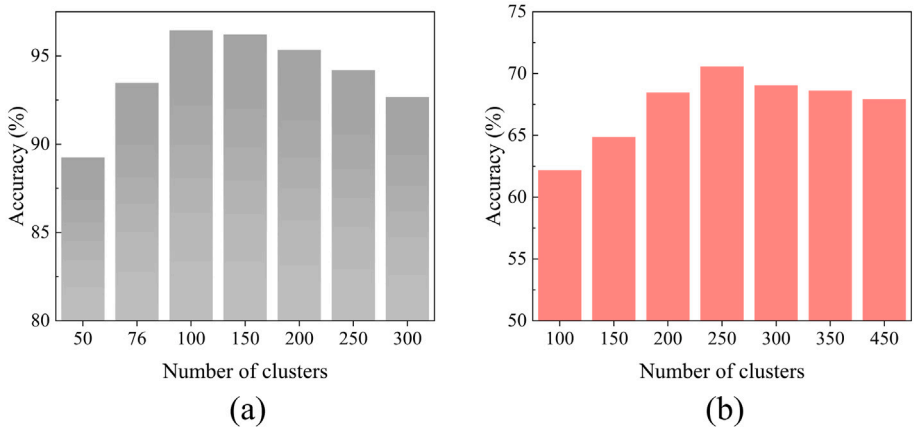


Fig. 11. Comparison of model performance with different number of clusters. (a) CWRU dataset, (b) rotor dataset.

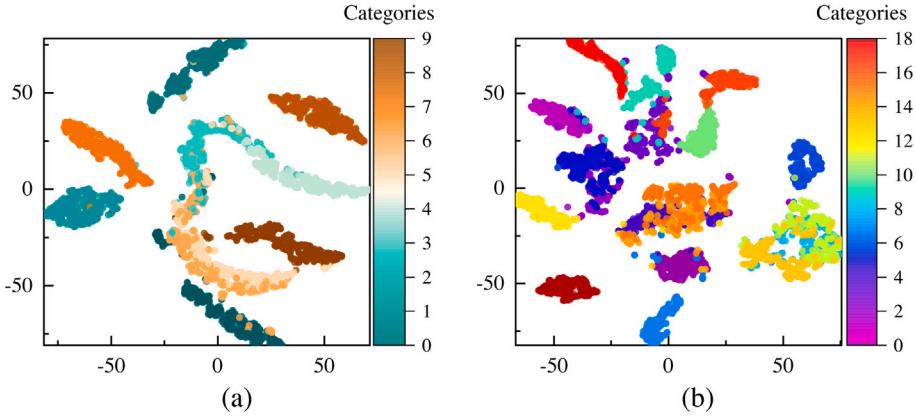


Fig. 12. Visualization of feature representation after contrastive learning. (a) CWRU dataset, (b) Rotor dataset.

Table 8
Diagnostic results using different methods for the two datasets.

Tasks	CWRU dataset				Rotor dataset			
	(5,1)	(5,5)	(10,1)	(10,5)	(5,1)	(5,5)	(10,1)	(10,5)
1D-BYOL	87.07	95.88	85.30	90.92	66.56	69.13	56.56	68.20
MOCO	86.74	95.46	84.15	89.91	64.25	68.50	56.67	66.75
FSDR	**91.34**	96.72	85.97	96.03	64.80	71.96	**59.61**	67.53
PL-CFE	87.45	92.19	**86.78**	91.30	**67.51**	70.36	57.00	64.89
ACML	87.03	**97.57**	85.62	**96.44**	66.85	**72.33**	56.74	**70.58**
MAML	89.41	96.73	86.07	94.68	65.56	71.85	60.42	70.05
ProtoNet	87.36	97.44	82.98	93.02	68.94	73.24	58.57	71.42

balancing method increases intra-class inconsistency with an average diagnostic accuracy of 77.71%. Therefore, the diversity of clusters is relatively more important in this method than the balance of the number of samples in the cluster.

Table 9
Cluster balancing with data augmentation for performance comparison with ACML.

Tasks	CWRU dataset				Rotor dataset			
	(5,1)	(5,5)	(10,1)	(10,5)	(5,1)	(5,5)	(10,1)	(10,5)
Data augmentation	86.32	95.86	84.69	95.01	64.93	70.84	55.42	68.61
ACML	**87.03**	**97.57**	**85.62**	**96.44**	**66.85**	**72.33**	**56.74**	**70.58**

3.4. Discussion

The temperature κ determines how much the contrastive loss focuses on the difficult negative samples. After the pre-training phase, all convolutional layers are frozen, and iMAML is added to find the

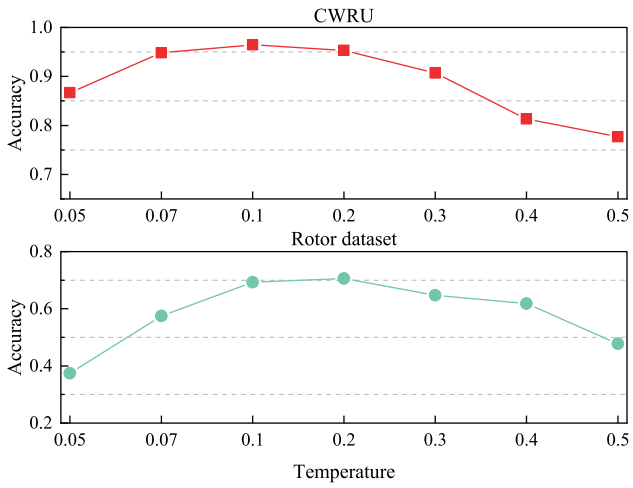


Fig. 13. Comparison of model performance at different temperatures.

ideal temperature κ so that the downstream model obtains better performance. Fig. 13 shows the classification performance for the two datasets, respectively, with the model reaching its best performance when κ is 0.1 or 0.2. The model tends to produce more uniform embedding distributions at low temperatures, focusing on challenging negative samples that are closely related to the sample. Large values of temperature tend to be more tolerant of samples with consistent states, whereas they may produce embeddings with insufficient uniformity. Therefore, for the analysis of the CWRU dataset, κ was chosen to be 0.1, and for the analysis of the rotor dataset, κ was chosen to be 0.2.

The impact of two parameters (number of partitions, number of clusters) on ACML performance is discussed, as shown in Fig. 14. The clustering generator is iterated P times to obtain P partitions, and C groups are drawn in each partition uniformly without replacement to construct the meta-task. When the number of clusters is the same as the number of real categories, the results are much less accurate than when the number of clusters is greater than the number of real categories. Since K-means clustering is unconstrained, when the number of clusters is small, the chances of each cluster containing multiple different true labels are high, leading to a greater rate of pseudo-labeling errors for the samples. Conversely, more clusters indicate that samples in certain clusters may have the same true label. Although samples with the same true labels are classified into different pseudo-classes and assigned different pseudo-labels, this incorrect pseudo-labeling, during training, does not have a significant impact on the model performance. This is because the chance that the samples fed into the inner and outer loops have the same true label increases during the meta-learning process. However, as the number of clusters rises, the prediction accuracy of the model becomes lower and lower, this is because more clusters will make the feature representation to have a finite diversity problem, and fewer clusters will make the feature representation the lower the category consistency, so the appropriate parameter C is chosen for the different datasets. The prediction accuracy of the model slightly fluctuates as the number of partitions rises.

Ablation experiments were introduced to verify the effect of different modules on the performance of ACML, as shown in Table 10. Relative to ACML, ACML- does not split the samples during the meta-learning process; Stacked Auto-encoder(SAE) (Yang et al., 2021)-ACML utilizes SAE to replace the feature extractor, after which K-means is utilized to add pseudo-labels to the original samples; and ACML (MAML) utilizes MAML to replace the iMAML proposed in this paper. The hyperparameters of the above methods are consistent with ACML. Under different experimental conditions, Fig. 15 intuitively shows the data distribution of different methods in prediction accuracy. In two

Table 10
Diagnostic results of ablation experiments in two datasets.

Tasks	CWRU dataset				Rotor dataset			
	(5,1)	(5,5)	(10,1)	(10,5)	(5,1)	(5,5)	(10,1)	(10,5)
ACML-	85.15	94.96	83.65	93.81	63.75	68.25	48.40	65.24
ACML(MAML)	83.39	90.27	79.38	89.09	60.00	67.38	49.51	62.94
SAE-ACML	76.75	81.63	72.38	79.75	55.73	60.54	42.25	48.71
ACML	**87.03**	**97.57**	**85.62**	**96.44**	**66.85**	**72.33**	**56.74**	**70.58**

different datasets, ACML has higher accuracy than other unsupervised fault diagnosis methods and it has a more concentrated data distribution. Comparing SAE-ACML, it can be shown that the addition of contrastive learning increases ACML by roughly 14.43%, making the generated pseudo-labels more consistent with the actual state. Increasing the gradient agreement of the task causes ACML to rise by around 6.41% when compared to ACML(MAML). It was discovered that maintaining category consistency of the data input to the inner and outer loops resulted in an ACML growth of around 3.71% when compared to ACML-.

There are three modules to improve the performance of ACML: (1) a more robust feature representation is generated using contrastive learning, allowing the model to guarantee a high level of category differentiation when adding pseudo-labels for samples; (2) the input inner and outer loop data are guaranteed to have the same true category; (3) a new update mechanism ensures gradient agreement for tasks.

The sample number of each cluster obtained based on the contrastive learning is shown in Fig. 16. After the feature extractor, the number of samples in each pseudo-category is not the same, and in the iterative process of each epoch, N categories are randomly selected for training. Due to the different number of samples in each category, the probability of randomly sampling the same samples in different batches during the process of randomly selecting K samples from N categories to construct the task is large, which leads to the lack of diversity in the obtained task. In this paper, the above problem is alleviated by splitting the samples. Due to the large differences between different types of faults in the CWRU dataset, the number of samples in different pseudo-classes is similar and the tasks obtained are relatively balanced, so that a good accuracy can be obtained. For the rotor dataset with large noise, the number of samples in the smallest cluster obtained by ACML's clustering generator is only 21, while the number of samples in the largest cluster is 342. The diagnostic performance obtained in this dataset is poor because the clusters are unconstrained and therefore the generated tasks are unbalanced.

The degree of agreement between different batches of tasks is defined as the average angle between the gradient vectors (Zhao et al., 2023b) and Fig. 17 shows the degree of task divergence in layers 1 and 4 of base learner during the meta-updating process. The conflicts between tasks are greatly reduced by the gradient update mechanism proposed in this paper. For MAML, task divergence did not weaken with epoch iteration. On the contrary, the degree of task divergence is reduced by iMAML already at the first layer of convolution. Gradient disagreement between tasks is weaker when the convolution layers are deeper.

An ablation study of the learning rates in the inner and outer loops was performed, and the diagnostic results for both datasets in the 10way-5shot task are shown in Fig. 18. The model performs optimally at $\alpha = 0.001$ and $\beta = 0.05$. The overall performance of the model increases with the learning rate β increases. Learning rate β has a greater effect on the model in the inner loop, relative to α.

4. Conclusion

In this paper, a novel meta-learning fault diagnosis framework, ACML is proposed, which merges contrastive learning and clustering methods into meta-learning for judging the potential information of

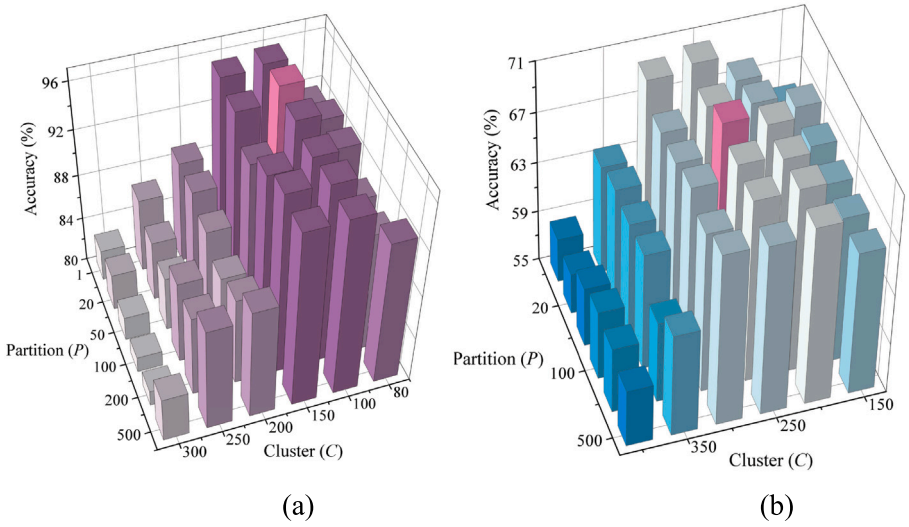


Fig. 14. The impact of two parameters on ACML performance. (a) Impact of parameters partition and cluster on fault diagnosis performance of CWRU dataset, (b) Impact of parameters partition and cluster on fault diagnosis performance of rotor dataset.

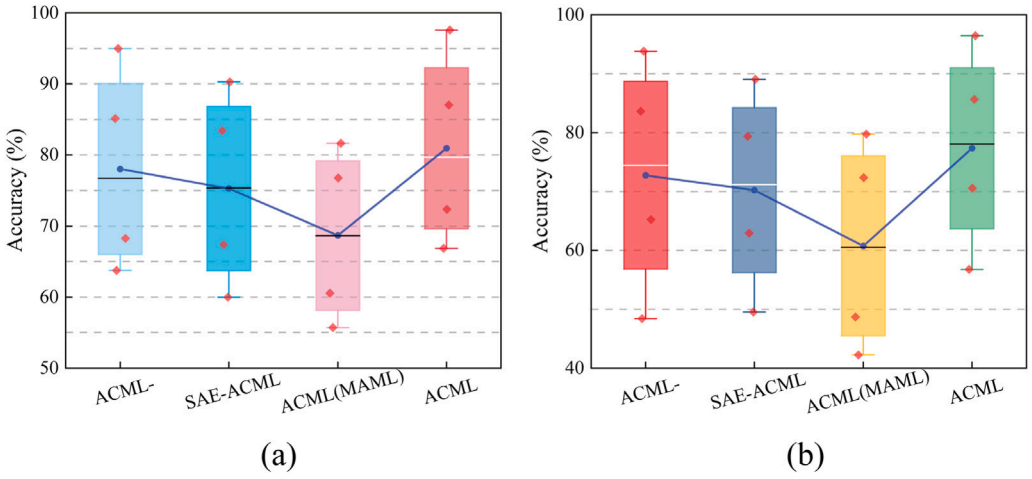


Fig. 15. Accuracy distribution for different methods under two datasets. (a) Accuracy distribution in 5-way scenarios under two datasets, (b) Accuracy distribution in 10-way scenarios under two datasets.

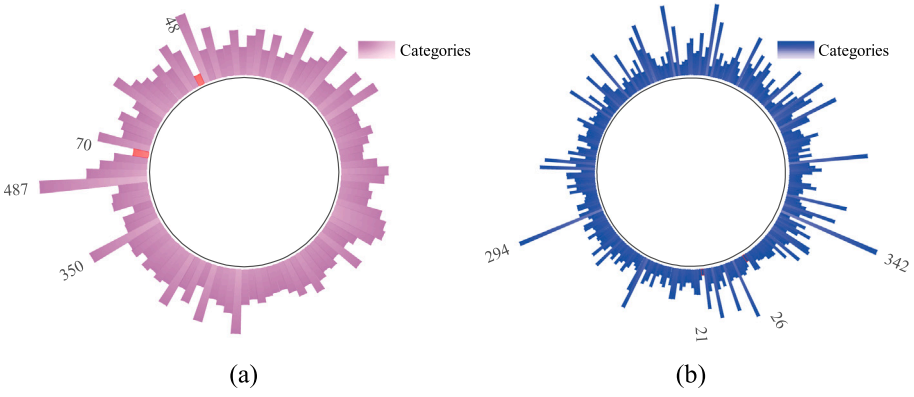


Fig. 16. Number of samples per cluster based on the cluster generator. (a) Number of samples per cluster in the CWRU dataset, (b) Number of samples per cluster in the rotor dataset.

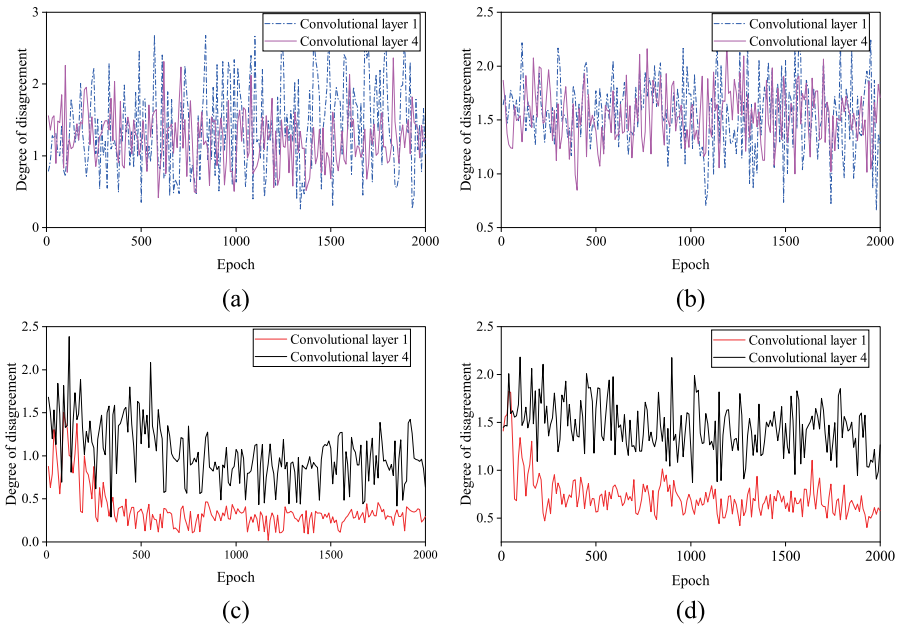


Fig. 17. Experimental results on the degree of divergence in different datasets. (a) Degree of task disagreement based on MAML in the CWRU dataset, (b) Degree of task disagreement based on MAML in the rotor dataset, (c) Degree of task disagreement based on meta-learner in the CWRU dataset, (d) Degree of task disagreement based on meta-learner in the rotor dataset.

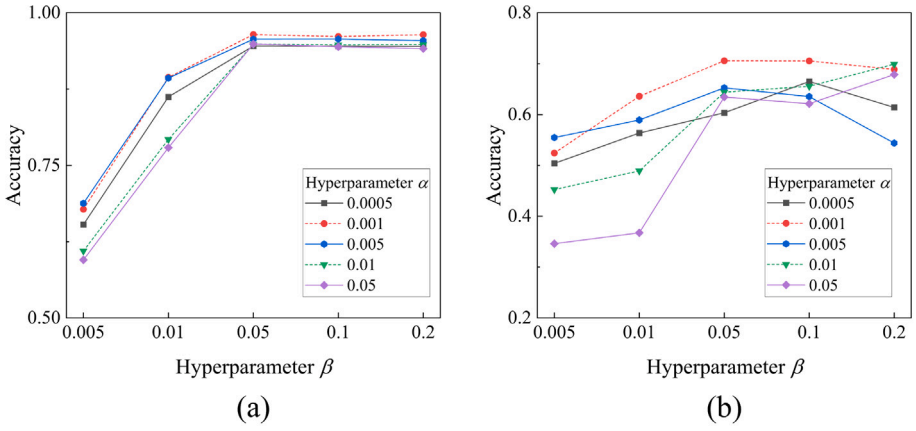


Fig. 18. Ablation study of hyperparameters α and β. (a) CWRU dataset, (b) Rotor dataset.

unlabeled datasets. The method is divided into three modules for fault diagnosis: feature generator, cluster generator and meta-learner. In the absence of any supervised information, high abstraction features are collected to enable better embedding of the data in the cluster generator, which ensures the classification accuracy of the model at the source. During the meta-learning process, category consistency is achieved by splitting the raw dataset in the task. It is argued that forcing the gradients to be averaged in MAML leads to conflicts between tasks, so a gradient update mechanism is proposed that strengthens the gradient agreement of MAML by adding weights to the tasks in the embedding parameter-independent task encoder. The performance of the present method is validated with two different datasets. The experimental results show that ACML can effectively transfer the prior knowledge obtained by the meta-learner to the new task and achieve high classification accuracy in the few-shot problem. ACML automatically generates suitable unsupervised task distributions, allowing it to outperform some supervised methods.

As industry moves towards smart manufacturing, it becomes increasingly important to realize real-time unsupervised fault diagnosis. Future work can further focus on enhanced feature representation techniques, real-time fault diagnosis, domain adaptation and scaling of large-scale datasets. Enhancing the performance of fault diagnosis frameworks with limited labeled data by combining meta-learning with more advanced contrastive learning techniques is a key direction. In addition, the development of algorithms that can efficiently handle real-time data streams will make unsupervised diagnosis systems more practical. In cross-domain applications, combining transfer learning and domain adaptive techniques can improve the generalization ability of the model. And with the popularity of IIOT, research on how to integrate ACML with IIOT for distributed monitoring and diagnosis will bring more application opportunities for smart manufacturing and predictive maintenance.

CRediT authorship contribution statement

Zhiqian Zhao: Writing – original draft, Software, Methodology, Conceptualization. **Yinghou Jiao:** Writing – review & editing, Methodology, Investigation. **Yeyin Xu:** Writing – review & editing, Resources, Data curation. **Zhaobo Chen:** Writing – review & editing, Methodology, Data curation. **Enrico Zio:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors are very grateful to the support provided by National Natural Science Foundation of China (Grant No. 11972131, 12072089 and 12102319), the Fundamental Research Funds for the Central Universities, China (Grant No. xzy012021004) and the China Scholarship Council (No. 202306120153).

Data availability

Data will be made available on request.

References

Aliramezani, M., Koch, C.R., Shahbakhti, M., 2022. Modeling, diagnostics, optimization, and control of internal combustion engines via modern machine learning techniques: A review and future directions. Prog. Energy Combust. Sci. 88, 100967.

Anowar, F., Sadaoui, S., Selim, B., 2021. Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne). Comp. Sci. Rev. 40, 100378.

Baik, S., Choi, M., Choi, J., Kim, H., Lee, K.M., 2024. Learning to learn task-adaptive hyperparameters for few-shot learning. IEEE Trans. Pattern Anal. Mach. Intell. 46 (3), 1441–1454.

Baik, S., Hong, S., Lee, K.M., 2020. Learning to forget for meta-learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2379–2387.

Bertugli, A., Vincenzi, S., Calderara, S., Passerini, A., 2020. Few-shot unsupervised continual learning through meta-examples. arXiv preprint arXiv:2009.08107.

Cha, Y.-J., Ali, R., Lewis, J., 2024. Deep learning-based structural health monitoring. Autom. Constr. 161, 105328. http://dx.doi.org/10.1016/j.autcon.2024.105328.

Cha, Y.-J., Choi, W., 2017. Deep learning-based crack damage detection using convolutional neural networks. Comput.-Aided Civ. Infrastruct. Eng. 32 (5), 361–378.

Cha, Y.-J., Choi, W., Suh, G., Mahmoudkhani, S., Büyüköztürk, O., 2018. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. Comput.-Aided Civ. Infrastruct. Eng. 33 (9), 731–747.

Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning. PMLR, pp. 1597–1607.

Chen, Y., Zhang, D., Zhang, H., Wang, Q.-G., 2022. IEEE Trans. Ind. Electron. 69 (12), 13462–13472. http://dx.doi.org/10.1109/TIE.2022.3144572,

Choe, S., Mehta, S.V., Ahn, H., Neiswanger, W., Xie, P., Strubell, E., Xing, E., 2024. Making scalable meta learning practical. Adv. Neural Inf. Process. Syst. 36.

Ding, P., Jia, M., 2021. Mechatronics equipment performance degradation assessment using limited and unlabeled data. IEEE Trans. Ind. Inform. 18 (4), 2374–2385.

Ding, Y., Zhuang, J., Ding, P., Jia, M., 2022. Self-supervised pretraining via contrast learning for intelligent incipient fault detection of bearings. Reliab. Eng. Syst. Saf. 218, 108126.

Dong, X., Shen, J., Shao, L., 2022. Rethinking clustering-based pseudo-labeling for unsupervised meta-learning. In: European Conference on Computer Vision. Springer, pp. 169–186.

Entezami, A., Sarmadi, H., Behkamal, B., 2023. Long-term health monitoring of concrete and steel bridges under large and missing data by unsupervised meta learning. Eng. Struct. 279, 115616. http://dx.doi.org/10.1016/j.engstruct.2023.115616.

Feng, Y., Chen, J., Xie, J., Zhang, T., Lv, H., Pan, T., 2022a. Meta-learning as a promising approach for few-shot cross-domain fault diagnosis: Algorithms, applications, and prospects. Knowl.-Based Syst. 235, 107646.

Feng, Y., Chen, J., Yang, Z., Song, X., Chang, Y., He, S., Xu, E., Zhou, Z., 2021. Similarity-based meta-learning network with adversarial domain adaptation for cross-domain fault identification. Knowl.-Based Syst. 217, 106829.

Feng, Y., Chen, J., Zhang, T., He, S., Xu, E., Zhou, Z., 2022b. Semi-supervised meta-learning networks with squeeze-and-excitation attention for few-shot fault diagnosis. ISA Trans. 120, 383–401.

Feng, K., Ji, J., Zhang, Y., Ni, Q., Liu, Z., Beer, M., 2023. Digital twin-driven intelligent assessment of gear surface degradation. Mech. Syst. Signal Process. 186, 109896. http://dx.doi.org/10.1016/j.ymssp.2022.109896.

Finn, C., Abbeel, P., Levine, S., 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning. PMLR, pp. 1126–1135.

He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2020. Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9729–9738.

Hsu, K., Levine, S., Finn, C., 2018. Unsupervised learning via meta-learning. arXiv preprint arXiv:1810.02334.

Hu, X., Li, Y., Jia, L., Qiu, M., 2021a. A novel two-stage unsupervised fault recognition framework combining feature extraction and fuzzy clustering for collaborative aiot. IEEE Trans. Ind. Inform. 18 (2), 1291–1300.

Hu, Y., Liu, R., Li, X., Chen, D., Hu, Q., 2021b. Task-sequencing meta learning for intelligent few-shot fault diagnosis with limited data. IEEE Trans. Ind. Inform. 18 (6), 3894–3904.

Huang, D., Shen, L., Yu, Z., Zheng, Z., Huang, M., Ma, Q., 2022. Efficient time series anomaly detection by multiresolution self-supervised discriminative network. Neurocomputing 491, 261–272.

Huisman, M., Plaat, A., van Rijn, J.N., 2023. Understanding transfer learning and gradient-based meta-learning techniques. Mach. Learn. 1–20.

Jang, H., Lee, H., Shin, J., 2023. Unsupervised meta-learning via few-shot pseudo-supervised contrastive learning. arXiv preprint arXiv:2303.00996.

Jia, J., Feng, X., Yu, H., 2024. Few-shot classification via efficient meta-learning with hybrid optimization. Eng. Appl. Artif. Intell. 127, 107296. http://dx.doi.org/10.1016/j.engappai.2023.107296.

Lao, Z., He, D., Sun, H., He, Y., Lai, Z., Shan, S., Chen, Y., 2024. Few-shot fault diagnosis of switch machine based on data fusion and balanced regularized prototypical network. Eng. Appl. Artif. Intell. 135, 108847. http://dx.doi.org/10.1016/j.engappai.2024.108847.

Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., Nandi, A.K., 2020. Applications of machine learning to machine fault diagnosis: A review and roadmap. Mech. Syst. Signal Process. 138, 106587. http://dx.doi.org/10.1016/j.ymssp.2019.106587.

Li, X., Zhang, W., Ding, Q., Sun, J.-Q., 2020. Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation. J. Intell. Manuf. 31 (2), 433–452.

Liu, S., Chen, J., He, S., Shi, Z., Zhou, Z., 2022. Subspace network with shared representation learning for intelligent fault diagnosis of machine under speed transient conditions with few samples. ISA Trans. 128, 531–544.

Liu, K., Li, Y., 2024. Remaining useful life prediction across machines using multi-source adversarial online knowledge distillation. Eng. Appl. Artif. Intell. 130, 107726. http://dx.doi.org/10.1016/j.engappai.2023.107726.

Liu, Z., Song, Y., Tang, R., Duan, G., Tan, J., 2023. Few-shot defect recognition of metal surfaces via attention-embedding and self-supervised learning. J. Intell. Manuf. 34 (8), 3507–3521.

Mueller, P.N., 2024. Attention-enhanced conditional-diffusion-based data synthesis for data augmentation in machine fault diagnosis. Eng. Appl. Artif. Intell. 131, 107696. http://dx.doi.org/10.1016/j.engappai.2023.107696.

Ouyang, L., Che, Y., Park, C., Chen, Y., 2024. A novel active learning gaussian process modeling-based method for time-dependent reliability analysis considering mixed variables. Reliab. Eng. Syst. Saf. 244, 109916. http://dx.doi.org/10.1016/j.ress.2023.109916.

Pan, C., Shang, Z., Li, W., Liu, F., Tang, L., 2024. Bearing fault diagnosis based on high-confidence pseudo-labels and dual-view multi-adversarial sparse joint attention network under variable working conditions. Eng. Appl. Artif. Intell. 133, 108625. http://dx.doi.org/10.1016/j.engappai.2024.108625.

Peng, T., Shen, C., Sun, S., Wang, D., 2022. Fault feature extractor based on bootstrap your own latent and data augmentation algorithm for unlabeled vibration signals. IEEE Trans. Ind. Electron. 69 (9), 9547–9555.

Russell, M., Wang, P., Liu, S., Jawahir, I.S., 2024. Mixed-up experience replay for adaptive online condition monitoring. IEEE Trans. Ind. Electron. 71 (2), 1979–1986. http://dx.doi.org/10.1109/TIE.2023.3260351.

Sinaga, K.P., Yang, M.-S., 2020. Unsupervised k-means clustering algorithm. IEEE Access 8, 80716–80727.

Smith, W.A., Randall, R.B., 2015. Rolling element bearing diagnostics using the case western reserve university data: A benchmark study. Mech. Syst. Signal Process. 64, 100–131.

Su, H., Yang, X., Xiang, L., Hu, A., Xu, Y., 2022. A novel method based on deep transfer unsupervised learning network for bearing fault diagnosis under variable working condition of unequal quantity. Knowl.-Based Syst. 242, 108381.

Sun, S., Hu, W., Liu, Y., Wang, T., Chu, F., 2023. Matching contrastive learning: An effective and intelligent method for wind turbine fault diagnosis with imbalanced scada data. Expert Syst. Appl. 223, 119891. http://dx.doi.org/10.1016/j.eswa.2023.119891.

Tang, J., Xiao, J., Chen, W., Li, X., Wei, C., Ding, X., Huang, W., 2024. A prior knowledge-enhanced self-supervised learning framework using time-frequency invariance for machinery intelligent fault diagnosis with small samples. Eng. Appl. Artif. Intell. 133, 108503. http://dx.doi.org/10.1016/j.engappai.2024.108503.

Tsialiamanis, G., Sbarufatti, C., Dervilis, N., Worden, K., 2024. On a meta-learning population-based approach to damage prognosis. Mech. Syst. Signal Process. 209, 111119. http://dx.doi.org/10.1016/j.ymssp.2024.111119.

Upadhyay, R., Phlypo, R., Saini, R., Liwicki, M., 2021. Sharing to learn and learning to share–fitting together meta-learning, multi-task learning, and transfer learning: A meta review. arXiv preprint arXiv:2111.12146.

Vettoruzzo, A., Bouguelia, M.-R., Vanschoren, J., Rognvaldsson, T., Santosh, K., 2024. Advances and challenges in meta-learning: A technical review. IEEE Trans. Pattern Anal. Mach. Intell. 1–20. http://dx.doi.org/10.1109/TPAMI.2024.3357847.

Wang, Z., Cha, Y.-J., 2021. Unsupervised deep learning approach using a deep auto-encoder with a one-class support vector machine to detect damage. Struct. Health Monit. 20 (1), 406–425. http://dx.doi.org/10.1177/1475921720934051.

Wang, H., Liu, Z., Ge, Y., Peng, D., 2022. Self-supervised signal representation learning for machinery fault diagnosis under limited annotation data. Knowl.-Based Syst. 239, 107978.

Wang, Y., Shen, L., Zhang, Y., Li, Y., Zhang, R., Yang, Y., 2023. Self-supervised health representation decomposition based on contrast learning. Reliab. Eng. Syst. Saf. 239, 109455. http://dx.doi.org/10.1016/j.ress.2023.109455.

Wang, D., Zhang, M., Xu, Y., Lu, W., Yang, J., Zhang, T., 2021. Metric-based meta-learning model for few-shot fault diagnosis under multiple limited data conditions. Mech. Syst. Signal Process. 155, 107510.

Xiao, D., Qin, C., Yu, H., Huang, Y., Liu, C., 2021. Unsupervised deep representation learning for motor fault diagnosis by mutual information maximization. J. Intell. Manuf. 32 (2), 377–391.

Yang, Z., Yang, R., Huang, M., 2021. Rolling bearing incipient fault diagnosis method based on improved transfer learning with hybrid feature extraction. Sensors 21 (23), 7894.

Zhang, T., Chen, J., Li, F., Zhang, K., Lv, H., He, S., Xu, E., 2022. Intelligent fault diagnosis of machines with small & imbalanced data: A state-of-the-art review and possible extensions. ISA Trans. 119, 152–171. http://dx.doi.org/10.1016/j.isatra.2021.02.042.

Zhang, S., Ye, F., Wang, B., Habetler, T.G., 2021. Few-shot bearing fault diagnosis based on model-agnostic meta-learning. IEEE Trans. Ind. Appl. 57 (5), 4754–4764.

Zhao, Q., Ding, Y., Lu, C., Wang, C., Ma, L., Tao, L., Ma, J., 2023a. An adaptive fault diagnosis framework under class-imbalanced conditions based on contrastive augmented deep reinforcement learning. Expert Syst. Appl. 234, 121001. http://dx.doi.org/10.1016/j.eswa.2023.121001.

Zhao, Z., Zhao, R., Wu, X., Hu, X., Che, R., Zhang, X., Jiao, Y., 2023b. A meta-learning network with anti-interference for few-shot fault diagnosis. Neurocomputing 552, 126551.

Zhao, M., Zhong, S., Fu, X., Tang, B., Pecht, M., 2019. Deep residual shrinkage networks for fault diagnosis. IEEE Trans. Ind. Inform. 16 (7), 4681–4690.

Zhao, C., Zio, E., Shen, W., 2024. Domain generalization for cross-domain fault diagnosis: An application-oriented perspective and a benchmark study. Reliab. Eng. Syst. Saf. 245, 109964. http://dx.doi.org/10.1016/j.ress.2024.109964.

Zheng, S., Zhao, J., 2020. A new unsupervised data mining method based on the stacked autoencoder for chemical process fault diagnosis. Comput. Chem. Eng. 135, 106755.