Health Prognosis of Bearings Based on Transferable Autoregressive Recurrent Adaptation with Few-shot Learning

Jichao Zhuang[1], Minping Jia[1, *], Cheng-Geng Huang[2], Michael Beer[3,4,5], and Ke Feng[6, *]

1 School of Mechanical Engineering, Southeast University, Nanjing 211189, China
2 School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China
3 Institute for Risk and Reliability, Leibniz University Hannover, Hannover, Germany
4 Institute for Risk and Uncertainty, University of Liverpool, Liverpool, UK
5 International Joint Research Center for Resilient Infrastructure & International Joint Research Center for Engineering Reliability and Stochastic Mechanics, Tongji University, Shanghai 200092, China
6 School of Mechanical Engineering, Xi'an Jiaotong University, Xian 710049, China

*Corresponding author: M. Jia (mpjia@seu.edu.cn) and K. Feng (ke.feng@outlook.com.au).

Abstract

Data-driven prognostic and health management technologies are instrumental in accurately monitoring the health of mechanical systems. However, the availability of few-shot source data under varying operating conditions limits their ability to predict health. Also, the global feature extraction process is susceptible to temporal semantic loss, resulting in reduced generalization of extracted degradation features. To address these challenges, a transferable autoregressive recurrent adaptation method is proposed for bearing health prognosis. In the enhancement of few-shot data, a novel sample generation module with attribute-assisted learning, combined with adversarial generation, is introduced to mine data that better matches the source sample distribution. Additionally, a deep autoregressive recurrent model is designed, incorporating a statistical mode to consider the degradation processes more comprehensively. To complement the semantic loss, a semantic attention module is developed, embedded into the basic model of meta learning. To validate the effectiveness of this approach, extensive bearing prognostics are conducted across six tasks. The results demonstrate the clear advantages of this proposed method in bearing prognosis, especially when dealing with limited bearing data.

Keyword

Autoregressive regression; Remaining useful life; Adversarial augmentation; Meta learning; Semantic attention mechanism

1 Introduction

Rotating machinery is widely applied across various industries, including aviation, aerospace, automotive, and others, such as engines and wind turbines. However, over extended periods of operation, faults in these systems become inevitable [1] and can even result in catastrophic losses [2]. As an integral component of Prognostics and Health Management (PHM) [3], the health monitoring of rotating machinery plays a vital role in accurately diagnosing the current condition of the machine,

ensuring safe and reliable continuous operation [4]. Rolling bearings serve as critical components in rotating machinery, and the presence of faults within them can directly lead to significant equipment accidents. The degradation of normal bearings is a gradual process, starting from the initial anomaly and progressing through various degradation stages until the final failure occurs [5]. Real-time and precise prediction of bearing health trends, along with early failure warnings, holds paramount importance in guaranteeing the long-term safe operation of rotating machinery.

In industrial settings, bearings often contend with diverse loads, speeds, and complex environmental factors [6]. Collecting run-to-failure vibration data from bearings under varying operating conditions can be time-consuming. In essence, the actual Remaining Useful Life (RUL) prediction of bearings is typically hindered by the limited availability of few-shot data.

1.1 *Challenges of Limited Data Prediction*

For the issue of distribution discrepancies in the few-shot data under different operating conditions, interestingly, the Domain Adaptation (DA) approach provides a new solution for its study. Huang et al. [7] developed a novel network framework with the DA module and implemented transfer fault prediction between machines with different structures, measurement settings, and operating conditions. Wang's work [1] has brought inspiration from a wide range of scholars, and it is a job worth recognizing. Zhu et al. [8] combined an implicit Markov model and a multilayer perceptron and generalized it to another operating condition. Mao et al. [9] designed a transfer learning module to adjust the target features, significantly improving the model's generalization. Most DA-based methods attempt to mitigate domain bias by learning a domain-invariant representation within the global via a statistical metric. In addition, Costa et al. [10] used a domain adversarial neural network to learn domain-invariant features that can provide more reliable RUL predictions under datasets with different operating conditions and failure modes. Chen et al. [11] proposed a deep convolutional generative adversarial network and employed it to set thresholds to monitor the health status of wind turbine generators. Ragab et al. [12] improved the accuracy of RUL regression by considering target-specific mutual information in domain adversarial adaptation. Most studies on adversarial strategies have focused on fault diagnosis [13], aiming to train feature extractors to deceive domain discriminators and generate domain-invariant features. Although existing methods have achieved significant prediction results, using DA only to find target transitions similar to the source domain fails to guarantee optimal transmission performance in few-shot data scenarios. Implementing the methods described above relies heavily on having ample samples from both the source and target domains.

In the industry, only a limited number of samples from the source domain are accessible. This limitation arises because an abundance of labeled samples would lead to substantial economic losses and require a significant investment in manpower and resources, rendering it impractical. Moreover, training deep models necessitates a substantial volume of raw data support. As a result, the scarcity of source data in a few-shot context places constraints on the cross-domain prognostication capabilities of the deep model.

1.2 *RUL Prediction with Limited Data*

Recently, meta learning [14] and Generative Adversarial Networks (GANs) [15] have become attractive learning methods to handle fault diagnosis and prognosis in few-shot sample scenarios. Unfortunately, a substantial portion of research in the

field of meta learning has been primarily directed towards fault identification [16]. Our previous work utilized adversarial learning [17] and domain generalization [18] to achieve the generation of samples and mining of latent domains, successfully applying to bearing RUL prediction. Long et al. [19] used meta learning to train a meta learner across a large number of randomly generated meta tasks, quickly generalizing it to target fault diagnosis tasks containing only a few-shot labeled samples. Ren et al. [20] presented a training framework based on meta learning for learning domain-invariant strategies in fault prediction under unknown operating conditions. Notably, Xu et al. [21] emphasized the challenges in acquiring high-quality samples in real-world applications and underscored the significance of developing few-shot learning-based models to broaden their applicability in engineering. Zhang et al. [22] proposed a prior knowledge-enhanced self-supervised feature learning framework for few-shot diagnosis. However, these methods are only developed for few-shot sample scenarios suitable for effective global feature extraction, but ignore local attributes in the raw data, leading to feature semantics being lost. In particular, it is more evident in few-shot samples. Thus, the motivation for this work is to utilize few-shot learning to address remaining life predictions with limited data under different operating conditions.

To address these challenges, a transferable autoregressive recurrent adaptation framework is proposed for bearing prognosis in few-shot source data scenarios. Specifically, a deep autoregressive recurrent model is constructed based on the statistical mode for extracting more effective degradation features. Then, a new sample adversarial augmentation module with attribute-assisted learning is developed to generate data that more closely matches the source sample distribution. A meta learner with complementary semantic loss is learned by embedding a semantic attention module to accomplish the prognostic task with a small number of samples. Finally, the approach is validated by acquiring signals from accelerated bearing degradation tests. The main contributions of this paper are as follows.

(1) A transferable autoregressive recurrent adaptation framework is proposed to realize data augmentation and semantic information learning, thus accomplishing the prognosis task of bearings under few-shot data.

(2) A data augmentation scheme embedded with attribute-assisted learning is developed to ensure that the new data matches the source sample distribution better.

(3) A semantic attention module is designed to construct more robust meta learning by embedding it into the basic model and ensuring that each subtask learns semantic information.

The rest of the article is organized as follows. The second section presents the related works. Next, the proposed TARA framework and the detailed methodology are described in Section 3. The case study is reported in Section 4. Finally, Section 5 concludes.

2 Related Works

2.1 *Data Augmentation for Adversarial Learning*

Generative adversarial networks (GANs) [23] are proposed for generating data using adversarial training and game strategy, in which the generator G and discriminator D are trained separately by a max-min alternating optimization strategy,

as shown in Fig. 1. GAN aims to generate pseudo samples with similar distributions to real samples through a mutual adversarial process between the generator and the discriminator. Specifically, the generator attempts to capture the latent distribution of the real data and generate fake samples that can deceive the discriminator. The training objective of the discriminator is to determine whether the input sample is real or fake. The objective function of GAN can be defined as follows.

$$\min_{G}\ \max_{D}(D,G)=\mathbb{E}_{x\sim P_{data}}\left[\log D(x)\right]+\mathbb{E}_{z\sim Q(z)}\left[1-D(G(z))\right] \tag{1}$$

where P_{data} denotes the distribution of the training data, z refers to the training random noise, and $Q(z)$ is the prior distribution of the noise. Minimizing the loss of the generator can generate more realistic pseudo samples, while maximizing the loss of the discriminator can enable pseudo samples to be further identified. The new samples constructed by adversarial generation can further expand the sample space, which is one of the main strategies for few-shot learning [16].

The majority of methods employ an adversarial generation strategy to create a substantial number of auxiliary samples from a limited set of raw few-shot samples, thereby achieving the RUL prediction task through the integration of the generated data. Ren et al. [24] a dual multi-scale generative adversarial network that incorporates feature fusion to maintain the similarity between generated samples and real samples while enhancing the diversity of the generated samples. Ding et al. [25] proposed an adversarial out-domain augmentation framework to generate pseudo-domains, thus increasing the diversity of available samples. Thus, training the generator in an adversarial manner and generating pseudo-domains by maximizing the domain discrepancies of the potential representations is an effective method for data enhancement. Lu et al. [26] stated that the use of available time-series degradation data to generate synthetic data can enhance the predictor's learning performance, thus improving the RUL prediction accuracy. These methods employ adversarial generation to accomplish data augmentation, but ignore the attribute links between the source samples and the generated samples, which is necessary for the augmentation of the samples. The similarity between the new and source samples is not measured, resulting in attribute links not being considered. This can lead to adversarial generation of new samples that may be negatively informative, which is detrimental to the training of the model. It may lead to adversarial generation of new samples with negative information, which is detrimental to the training of the model.

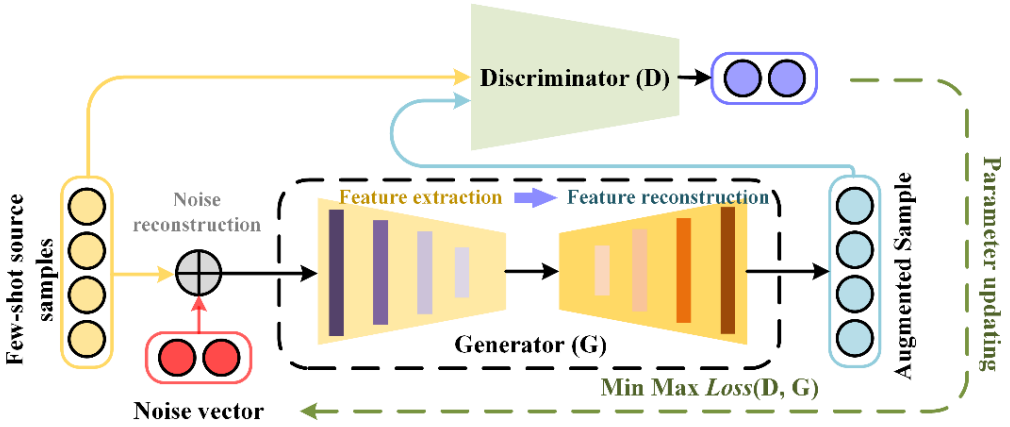


Fig. 1 Data augmentation for adversarial generation

2.2 Meta Learning

The lack of run-to-failure bearing data has been one of the challenges in developing and practically implementing robust bearing prediction models [26]. Thus, few-shot learning has been attempting to give a more significant advantage to diagnosis and prognosis in this scenario [27]. This paper is expected to generalize quickly to new tasks with trained agents and a priori knowledge of a limited test sample. The general description is as follows.

$$\min_{\theta} = \sum_{(x_i, y_i) \in D_{train}} \mathcal{L}oss\left(g\left(x_i; \theta\right), y_i\right) \tag{2}$$

where θ is a few-shot learning model and $(x_i, y_i) \in D_{train}$ denotes training samples and labels. Fortunately, meta learning provides a practical learning framework to deal with the challenges of few-shot learning [28]. Unlike traditional methods, meta learning is a flexible framework, aiming to learn a priori experiences from multiple related tasks and rely on previously acquired learning parameters to improve learning performance on the target task, as shown in Fig. 2. Overall, the learning objectives of meta learning can be defined as follows.

$$\mathcal{L}oss\left(\theta^*\right) = \sum_{m=1}^{M} \mathcal{L}oss^m\left(\theta^m\right) \tag{3}$$

$$\min_{\theta} \mathcal{L}oss\left(\mathcal{L}oss \sim p\left(\mathcal{L}oss\right); \theta\right) \tag{4}$$

where θ^* is the optimal learning parameter of the meta learner, θ^m is the learning parameter of each learning task, an $p\left(\mathcal{L}oss\right)$ denotes the different distributions of each task. It can be seen from Eq. (3) that meta learning demonstrates the ability to "learn to learn" meta-knowledge θ^*. It ensures that applying the model to tasks with different distributions is more generalizable and adaptable. In Fig. 2, the optimal parameters θ^* obtained by meta learning can be fine-tuned in each task and successfully reach their optimal training mode, which can significantly avoid local optimization and retraining of the model. It is worth noting that θ^* in meta learning is learned across multiple tasks during the learning process, and the optimal θ^* minimizes the loss of new tasks.

Using meta learning as a few-shot learning tool, a lot of research works have applied meta learning to the PHM field [29]. The core idea of meta learning is to obtain the initial parameters of a model by gradient descent in meta training, and apply them to obtain the best performance by updating the existing parameters through several gradient computations when applied to unknown tasks with limited data. However, these methods ignore localized attributes in the raw data, resulting in feature semantic loss. It is especially obvious in few-shot samples.

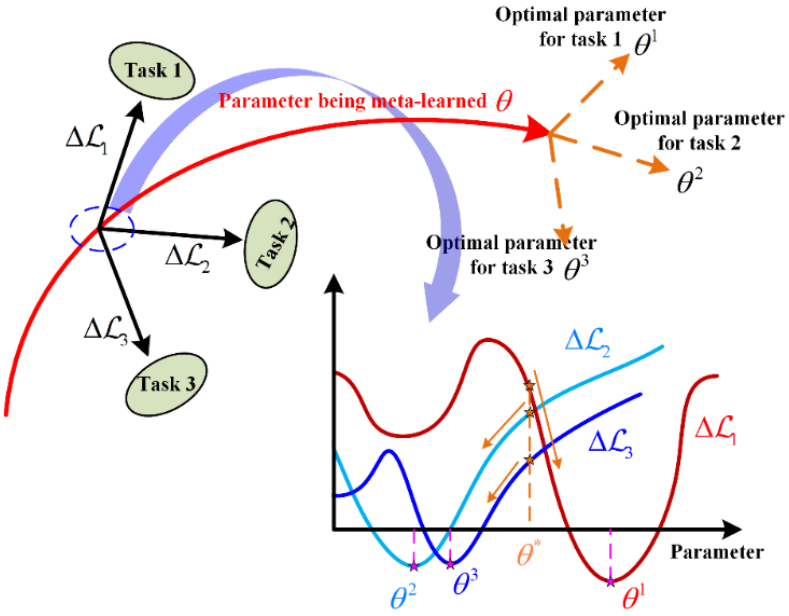


Fig. 2 Meta learning process

3 Methodology

3.1 *Overview*

For the bearing RUL prediction under few-shot source data, a transferable autoregressive recurrent adaptation (TARA) framework is proposed in this paper, as shown in Fig. 3. The main steps of TARA include deep autoregressive recurrent modeling, data augmentation, semantic attention module, and meta learning prognosis. First, a deep autoregressive recurrent model based on statistical mode is constructed to enhance the model's focus on the degradation process. Second, a new sample augmentation module with attribute-assisted learning is constructed based on the adversarial generation scheme to mine data that more closely matches the distribution of source samples. Then, a semantic attention mechanism is developed to supplement the global learning of semantic information. Also, the RUL prognosis for few-shot samples is accomplished by constructing a meta learning process. The flowchart of the proposed method is presented in Fig. 4.

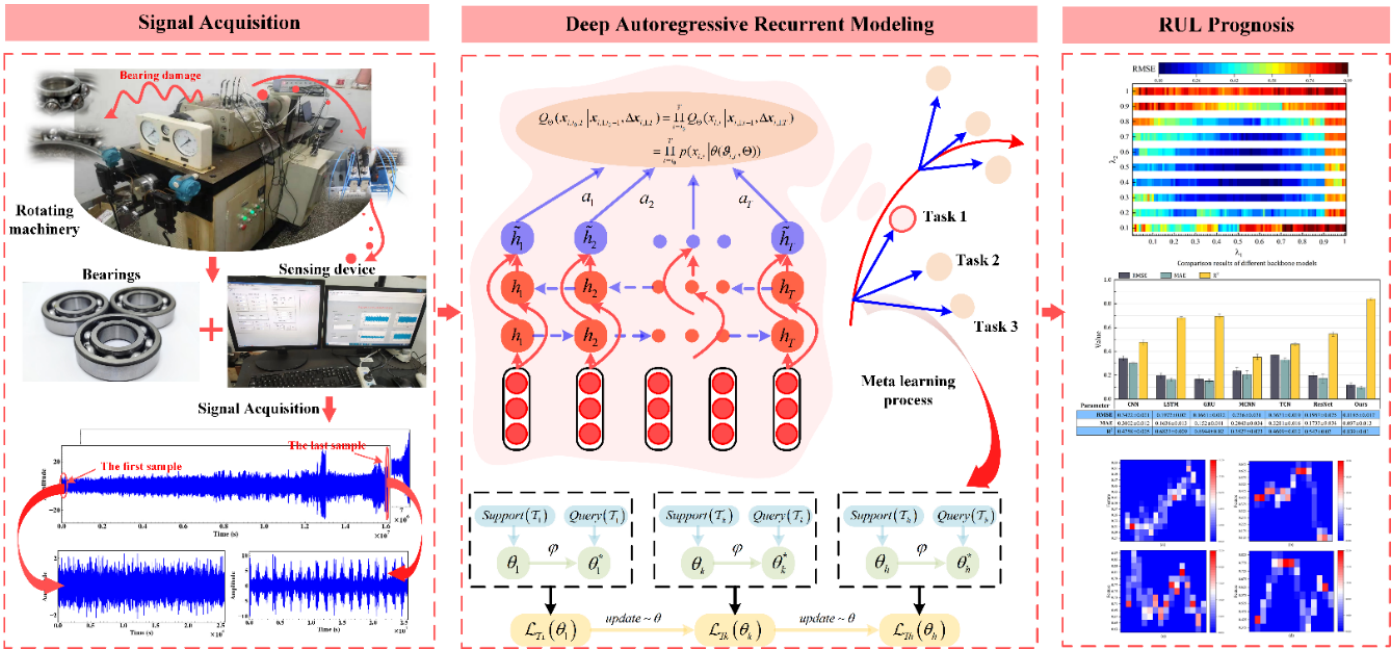


Fig. 3 Transferable autoregressive recurrent adaptation framework

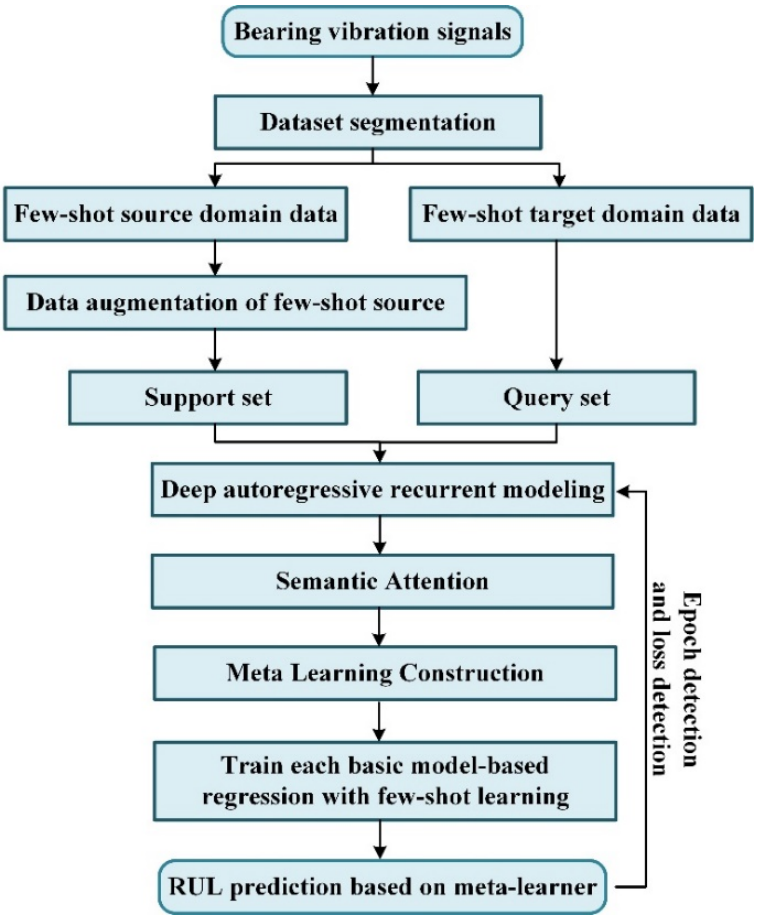


Fig.4 Flowchart of the proposed methodology

3.2 *Deep Autoregressive Recurrent Modeling*

Classical deep models attempting to learn jointly from multiple time series usually encounter effective degradation features extracted that are not remarkable. In the run-to-failure bearing signal, most of the degradation information has a severe deviation in the distribution of the percentage of different stages, leading to a slightly poorer effect of the effective degradation features extracted. Thus, a deep autoregressive recurrent model incorporating a statistical mode is designed, as shown in Fig. 5.

Given the sequence value $x_{i,t}$, $[x_{i,1},\ldots,x_{i,t_0-2},x_{i,t_0-1}]=\boldsymbol{x}_{i,1:t_0-1}$, of a time series i at the moment t and the deep autoregressive recurrent model distribution $Q_\Theta(\boldsymbol{x}_{i,t_0:T}\left|\boldsymbol{x}_{i,1:t_0-1},\Delta\boldsymbol{x}_{i,1:T}\right.)$, the conditional probability distribution for each future time series $[x_{i,t_0},x_{i,t_0+1},\ldots,x_{i,T}]=\boldsymbol{x}_{i,t_0:T}$ is established and expressed as follows.

$$P(\boldsymbol{x}_{i,t_0:T}\left|\boldsymbol{x}_{i,1:t_0-1},\Delta\boldsymbol{x}_{i,1:T}\right.) \tag{5}$$

where t_0 denotes the time of prediction $x_{i,t}$ and $\Delta\boldsymbol{x}_{i,1:T}$ represents the difference between the current moment $x_{i,t}$ and the previous moment $x_{i,t-1}$. In this paper, the likelihood function is employed to represent the composition of the model distribution, which is described as follows.

$$Q_\Theta(\boldsymbol{x}_{i,t_0:T}\left|\boldsymbol{x}_{i,1:t_0-1},\Delta\boldsymbol{x}_{i,1:T}\right.)=\prod_{t=t_0}^{T}Q_\Theta(x_{i,t}\left|\boldsymbol{x}_{i,1:t-1},\Delta\boldsymbol{x}_{i,1:T}\right.)$$
$$=\prod_{t=t_0}^{T}p(x_{i,t}\left|\theta(\boldsymbol{\vartheta}_{i,t},\Theta)\right.) \tag{6}$$

where $p(x_{i,t}\left|\theta(\boldsymbol{\vartheta}_{i,t})\right.)$ is the likelihood function, $\theta(\boldsymbol{\vartheta}_{i,t},\Theta)$ is the parameter of the deep autoregressive recurrent model, and $\boldsymbol{\vartheta}_{i,t}$ is the model parameterization expression, $\boldsymbol{\vartheta}_{i,t}=\mathcal{H}(\boldsymbol{\vartheta}_{i,t\text{-}1},x_{i,t-1},\Delta\boldsymbol{x}_{i,t},\Theta)$. Note that the value $x_{i,t-1}$ at moment $t\text{-}1$, $\Delta\boldsymbol{x}_{i,t}$, and output $\boldsymbol{\vartheta}_{i,t-1}$ will be used as inputs at moment t. Thus, the iterative computation of the moment $t=1,\ldots,t_0-1$ is performed to obtain its output $\boldsymbol{\vartheta}_{i,t_0-1}$ and the corresponding model distribution $\tilde{x}_{i,t_0:T}\sim p(\cdot\left|\theta(\tilde{\boldsymbol{\vartheta}}_{i,t},\Theta)\right.)$. The predicted value $\tilde{x}_{i,t_0-1}$ and quartile for future time periods are estimated by the likelihood function.

In this paper, $\mathcal{H}$ is set up as a 4-layer Bidirectional Long Short-Term Memory (BiLSTM) network and Θ denotes the BiLSTM parameters. Specifically, BiLSTM can make predictions by utilizing the previous and subsequent information in the sequence through multiple successive feedforward layers. The constructed transformation function models temporal dependencies in time series by combining forward LSTM and backward LSTM, obtaining more comprehensive contextual information and thereby capturing bidirectional semantic dependencies. The structure of a deep autoregressive recurrent model consists of an input layer, a convolutional layer, a BiLSTM layer, a likelihood estimation layer, and an output layer, where a 1-dimensional convolutional layer is employed to extract features from the data $x_{i,t}$ and $\Delta\boldsymbol{x}_{i,1:T}$. In addition, the convolved features are fused with the raw data using the residual structure, and the result is utilized as an input to the 4-layer BiLSTM. In the likelihood estimation layer, the output of the BiLSTM network is used to compute its mean and variance and parameterize the model distribution. The error between the predicted $\hat{y}_i$ and true value y_i is calculated.

$$\mathcal{L}_{mse}=\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i-y_i)^2 \tag{7}$$

where n is the number of samples.

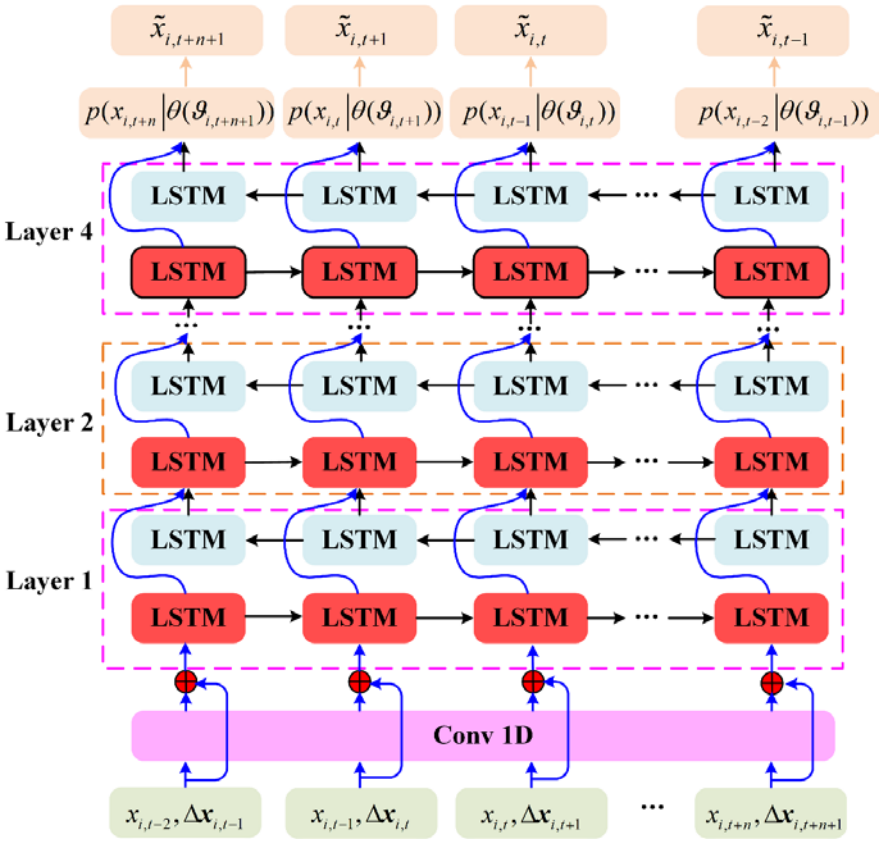


Fig. 5 Deep autoregressive recurrent network

3.3 *Data Augmentation of Few-shot Source*

Although meta learning [30] can utilize the learning of multiple basic models to cognize more meta-knowledge within multiple tasks, a small number of samples still constrains the feature learning of each basic model. Thus, a data augmentation scheme embedded with attribute-assisted learning is proposed for solving the few-shot source sample problem to ensure that the new data generated is more consistent with the source sample distribution, as shown in Fig. 6. The scheme consists of three main steps, including attribute assisted extraction, adversarial generation, and attribute semantic metrics. Specifically, attribute-assisted vectors are obtained by attribute-assisted extraction and are employed as information fusion in an adversarial generation to generate new samples with more similarity. Also, the attribute-assisted vectors are utilized as semantic metrics of the features of the new samples to encourage the model to update its own parameters within each gradient descent, thus meeting the needs of the adversarial generation goal.

Given source samples $\boldsymbol{x}_s$ and source labels y_s, we aim to learn a generator G: $\boldsymbol{e} \times \boldsymbol{z} \to \boldsymbol{x}_n$, where random embeddings $\boldsymbol{e}_s^i \in \mathcal{E}$ and Gaussian noise $\boldsymbol{z} \in Z$ are taken as input. In attribute-assisted extraction, the source sample is extracted as attribute-assisted vectors $\mathcal{M}_s^i$ in feature extractor 1, where $i \in \{1, 2\}$. It is represented as follows.

$$\mathcal{M}_s^i = \boldsymbol{E}\left(\boldsymbol{x}_s \cdot \boldsymbol{e}_s^i\right) \tag{8}$$

The generator G is a downsampling-up sampling structure. Random noise $\boldsymbol{e}_s^3 \times \boldsymbol{z}$ is fed into the encoder for feature coding and intermediate vectors $\mathcal{F}$ are obtained. The attribute-assisted vectors are embedded into the intermediate coding

constructs, which are described as follows.

$$\mathcal{M}_s^3 = \left(\mathcal{F} \oplus \mathcal{M}_s^1\right) \oplus \left(\mathcal{F} \oplus \mathcal{M}_s^2\right) \tag{9}$$

where $\mathcal{M}_s^3$ is the vector constructed by attribute-assisted learning, and new samples $\boldsymbol{x}_n$ are reconstructed in the decoder. Specifically, the downsampling convolution is employed to extract key features from the samples, and then the samples are reconstructed using the upsampling transposed convolution.

$$\boldsymbol{x}_n = G(\mathcal{M}_s^i, \boldsymbol{e}_s^3, \boldsymbol{z}, y_s, \tilde{y}_n) = \begin{cases} convT(\mathcal{M}_s^i, conv(\boldsymbol{e}_s^3, \boldsymbol{z})) \\ \tilde{y}_n^i = y_s^i + i \end{cases} \tag{10}$$

where $\tilde{y}_n$ is a pseudo-label promoted by an indicator in combination with a source label. The generated new sample and source sample are utilized as inputs to the discriminator *D*. Also, a max-min alternating adversarial training strategy is used to optimize *G* and *D*. The adversarial loss can be described as follows.

$$\mathcal{L}_{DA} = \min_G \max_D \left(\mathcal{L}_G = d\left(E_2\left(\boldsymbol{x}_n\right), \mathcal{M}_s^i\right), \mathcal{L}_D = \sigma\left(\boldsymbol{x}_n, \boldsymbol{x}_s\right)\right) \tag{11}$$

where $\mathcal{L}_G$ refers to the generator loss, $\mathcal{L}_D$ is the discriminator loss, $E_2\left(*\right)$ denotes the feature extractor 2, $\sigma\left(*\right)$ is the cross-entropy function, and $d\left(*\right)$ is the attribute semantic metric function. This paper uses Maximum Mean Discrepancy (MMD) [30] as a metric between new sample features and attribute-assisted vectors. The general description of MMD is as follows.

$$d_{mmd}(H^S, H^T) = \left\| \frac{1}{n_S} \sum_{i=1}^{n_S} \phi(H^S) - \frac{1}{n_T} \sum_{j=1}^{n_T} \phi(H^T) \right\|_{\mathcal{H}}^2 \tag{12}$$

where H^S and H^T are the output for the source and target data, respectively, $\|\cdot\|_{\mathcal{H}}$ is a reproducing kernel Hilbert space, and the kernel map $\phi(\cdot)$ is defined as a combination of kernel tricks $\{k\}$, $k(H^S, H^T) = (\phi(H^S), \phi(H^T)) \to \mathcal{H}$. Extending it to attribute semantic metric can be transformed into Eq. 13.

$$d(\mathcal{A}, \mathcal{M}_s^i) = \sum_{i=1}^{N} \left\| \frac{1}{n} \left(\sum_{i=1}^{n} \phi(\mathcal{A}) - \sum_{j=1}^{n} \phi(\mathcal{M}_s^i) \right) \right\|_{\mathcal{H}}^2 \tag{13}$$

where $\mathcal{A}$ denotes the new sample features and $N \in \{1, 2\}$. Defining the kernel function as a Gaussian kernel, Eq. 13 can be converted as follows.

$$d(\mathcal{A}, \mathcal{M}_s^i) = \sum_{i=1}^{N} \left(\frac{1}{n^2} \sum_{t=1}^{n} \sum_{r=1}^{n} k(\mathcal{A}_t, \mathcal{A}_r) + \frac{1}{n^2} \sum_{t=1}^{n} \sum_{r=1}^{n} k(\mathcal{M}_{st}^i, \mathcal{M}_{sr}^i) - \frac{2}{n^2} \sum_{t=1}^{n} \sum_{r=1}^{n} k(\mathcal{A}_t, \mathcal{M}_{sr}^i) \right) \tag{14}$$

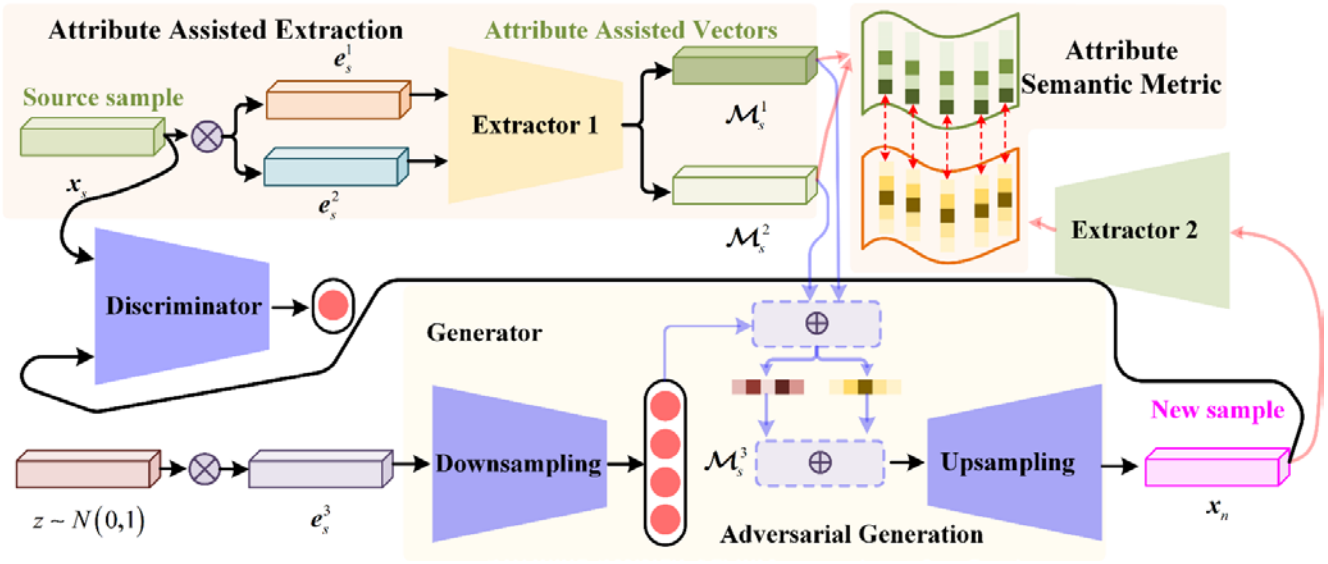


Fig. 6 Data augmentation module

3.4 *Meta Learning Prognosis*

3.4.1 *Semantic Attention*

The semantic attention module aims to enhance the representation of each feature by exploiting the sequential relationships within each sample. Convolutional features $conv(\mathcal{J})$ and degradation features $\mathcal{J}$ are adopted as inputs to generate the correlation attention map 1 in a softmax function, thus multiplying it with $\mathcal{J}$ to obtain the semantic attention embedding. Considering the correlation between the sequences, the representation of the sequences is enhanced, as shown in Fig. 7(a). The correlation attention map 1 is computed through convolutional, matrix multiplication, and softmax layers.

$$Map_1 = \frac{e^{\mathcal{K}(\mathcal{J}_i, covn(\mathcal{J}_i))}}{\sum_{i=1}^{N} e^{\mathcal{K}(\mathcal{J}_i, covn(\mathcal{J}_i))}} \tag{15}$$

where $\mathcal{J}_i$ denotes the *i*-th sequence value and $\mathcal{K}(\mathcal{J}_i, covn(\mathcal{J}_i))$ is the unnormalized relation function. The attention semantic embedding can be expressed as follows.

$$\tilde{\mathcal{J}} = Map_1 \otimes \mathcal{J} \tag{16}$$

Attention relations between sequences are established through preliminary augmented representations. Then, an augmented representation between the preliminary semantic attention embedding $\tilde{\mathcal{J}}$ and the convolutional features $conv(\mathcal{J})$ is obtained in the second layer of attention embedding, as shown in Fig. 7(b). The correlation attention map 2 can be defined as follows.

$$Map_2 = \frac{e^{\mathcal{K}(\mathrm{conv}(\tilde{\mathcal{J}}_i), covn(\mathcal{J}_i))}}{\sum_{i=1}^{N} e^{\mathcal{K}(\mathrm{conv}(\tilde{\mathcal{J}}_i), covn(\mathcal{J}_i))}} \tag{17}$$

$$\hat{\mathcal{J}} = conv\left(\left(Map_2 \otimes conv(\tilde{\mathcal{J}})\right) \oplus conv(\mathcal{J})\right) \tag{18}$$

where $\hat{\mathcal{J}}$ denotes the augmented representation. From Eq. 25, it can be seen that the augmented attention representation fuses

the information between the convolutional degradation features and the raw degradation features. The deep autoregressive recurrent model is employed in meta learning as the basic model. Also the semantic attention module is embedded into the basic model and placed at locations between each layer. The ablation analysis of the attention module at different locations is demonstrated in Subsection 4.2.3.

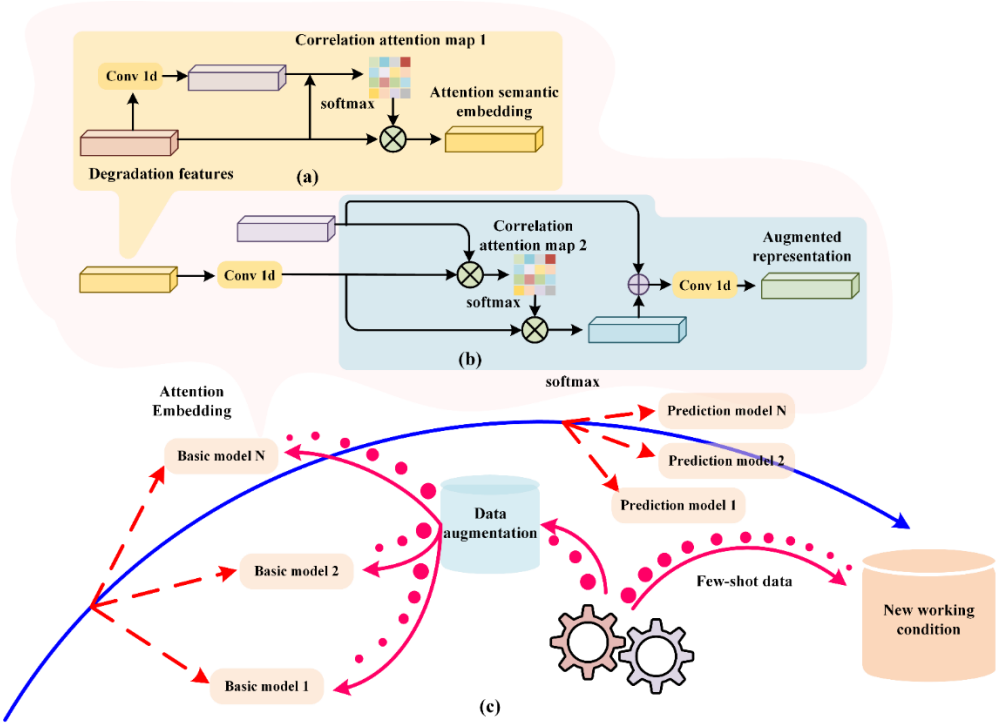


Fig. 7 Semantic attention mechanism embedding

3.4.2 *Meta Learning Construction*

During the meta training process, the model is based on multiple prediction tasks for known operating conditions, acquires a priori knowledge by optimizing the initialization parameters, and then uses the learned knowledge to achieve fast and accurate few-shot bearing RUL prediction under new operating conditions. It can improve the generalization ability of all deep neural network learning models, enabling deep learning models to be quickly and accurately applied to new tasks. In the meta learning framework, *h* basic models are employed for training, as shown in Fig. 8. The meta objective is to find the best model parameters across multiple tasks by minimizing the total loss.

The basic model for a given task can be defined as $\mathcal{T}_\theta$, where θ is the trainable parameter provided by the meta learner in the base model. In meta training, the target loss is minimized for each basic model.

$$\min_{\theta} \mathcal{L}_{\mathcal{T}_i}(\mathcal{T}_\theta) \tag{19}$$

where $\mathcal{L}_{\mathcal{T}_i}$ denotes the loss function of the basic model in the *i*-th task $\mathcal{T}_i$. After one iteration on task $\mathcal{T}_i$, the trainable parameters updated by $\mathcal{T}_{i+1}$ can be obtained as follows.

$$\theta_i^N = \theta_i - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_i}\left(\mathcal{T}_{\theta_i^{N-1}}\right) \tag{20}$$

where $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}\left(\mathcal{T}_{\theta_i}\right)$ is the gradient of the loss function in the task $\mathcal{T}_i$, α is the learning rate of the basic model, and θ_i^N

denotes the *N*-th training parameter. The basic model is applied to supervised RUL regression and the regression task loss using mean square error can be defined as follows.

$$\mathcal{L}_{\mathcal{T}_i}\left(\mathcal{T}_{\theta_i}\right)=\sum_{x,y\in\mathcal{T}_i}\left\|\mathcal{T}_{\theta_i}\left(x\right)-y\right\|^2 \tag{21}$$

where $\mathcal{T}_{\theta_i}\left(x\right)$ denotes the RUL prediction result of the *i*-th model. Feedback $\nabla_{\theta_i}\mathcal{L}_{\mathcal{T}_i}\left(\mathcal{T}_{\theta_i}\right)$ to the meta learner, the objective of the meta learner can be defined as follows.

$$\min_{\theta}\sum_{\mathcal{T}_i}\mathcal{L}_{\mathcal{T}_i}\left(\mathcal{T}_{\theta_i}\right) \tag{22}$$

The loss function gradient can be employed to update the parameters θ of the meta learner. It can be defined as follows.

$$\theta\leftarrow\theta-\beta\sum_{\mathcal{T}_i}\nabla_{\theta}\mathcal{L}_{\mathcal{T}_i}\left(\mathcal{T}_{\theta_i}\right) \tag{23}$$

where β denotes the learning rate of the meta learner. In each task learning, the support and query sets are used as inputs to each basic model. The key parameters of the proposed method are reported in Table 1.

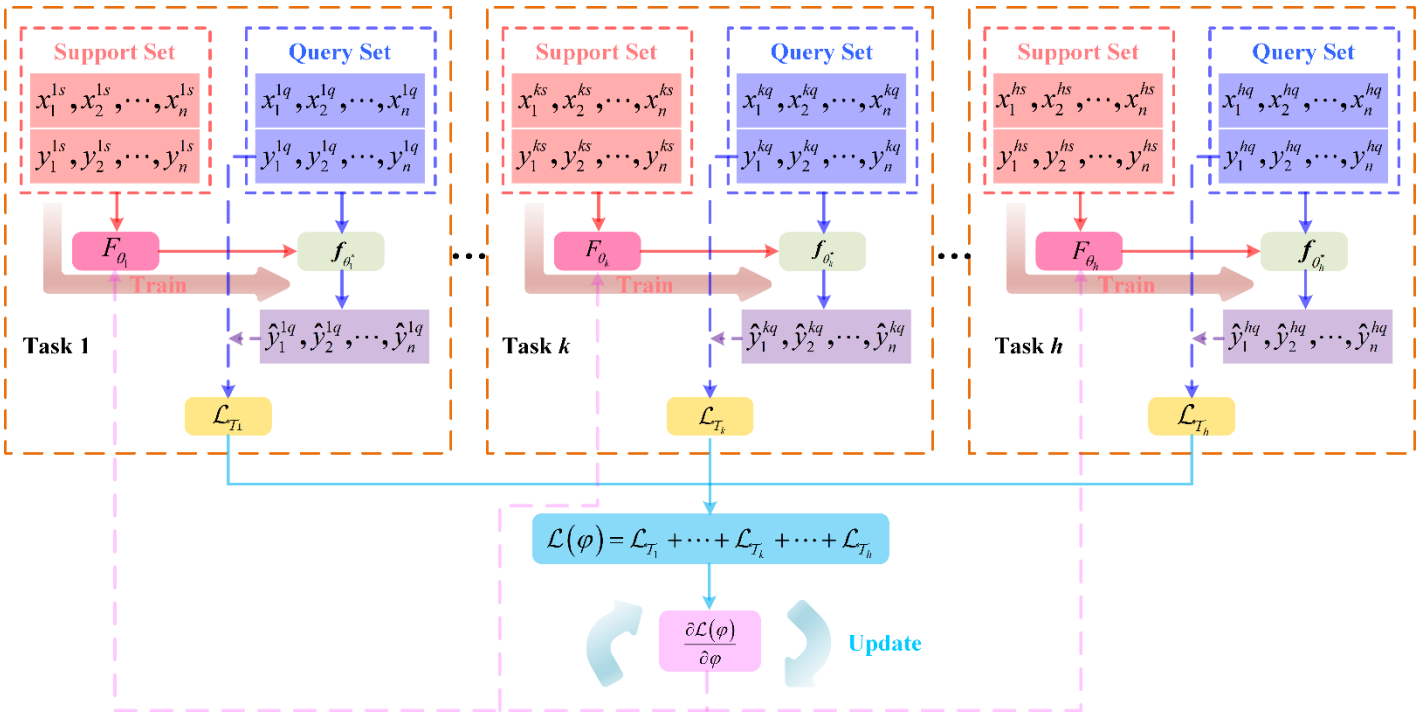


Fig. 8 Meta learning construction for RUL prediction

TABLE 1 Description of the parameters of TARA

Parameters	Value	Parameters	Value
Max_epoch	1000	Learning rate	0.001
Optimizer	Adam	λ_1	0.7
α	0.001	λ_2	0.4
$\mathcal{M}_s^1$	Linear (1024, 100)	$\mathcal{M}_s^2$	Linear (1024, 100)
$\mathcal{M}_s^3$	Linear (1024, 100)	β	0.01

4 Case Experiment

4.1 *Dataset Description*

The run-to-failure test data of the bearings are obtained from the accelerated bearing degradation test on the ABLT platform, as shown in Fig. 9. Measurements of run-to-failure vibration and temperature can be obtained utilizing temperature sensors and PCB acceleration sensors. Three accelerometers are employed to measure the vibration information, and the specific test parameters and RUL prediction tasks are shown in Table 2. In run-to-failure experiments, the test stops when the root mean square of the vibration signal reaches a threshold. The platform enables testing of general-purpose rolling bearings closer to industrial applications and includes a control module, a hydraulic cylinder drive loading device module, a bearing test module, an oil pressure regulation module, and a data acquisition module. The bearings are lubricated by lubricating oil during the run-to-failure test. To quantify the performance, two evaluation metrics are employed, including root mean square error (RMSE), mean absolute error (MAE), and R-square (R^2) [31].

$$\mathrm{RMSE}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i-y_i\right)^2} \tag{23}$$

$$\mathrm{MAE}=\frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i-y_i\right| \tag{24}$$

$$\mathrm{R}^2=1-\frac{\sum_{i=1}^{n}\left(\hat{y}_i-y_i\right)^2}{\sum_{i=1}^{n}\left(\hat{y}_i-\bar{y}\right)^2} \tag{25}$$

$$\bar{y}=\frac{1}{n}\sum_{i=1}^{n}y_i \tag{26}$$

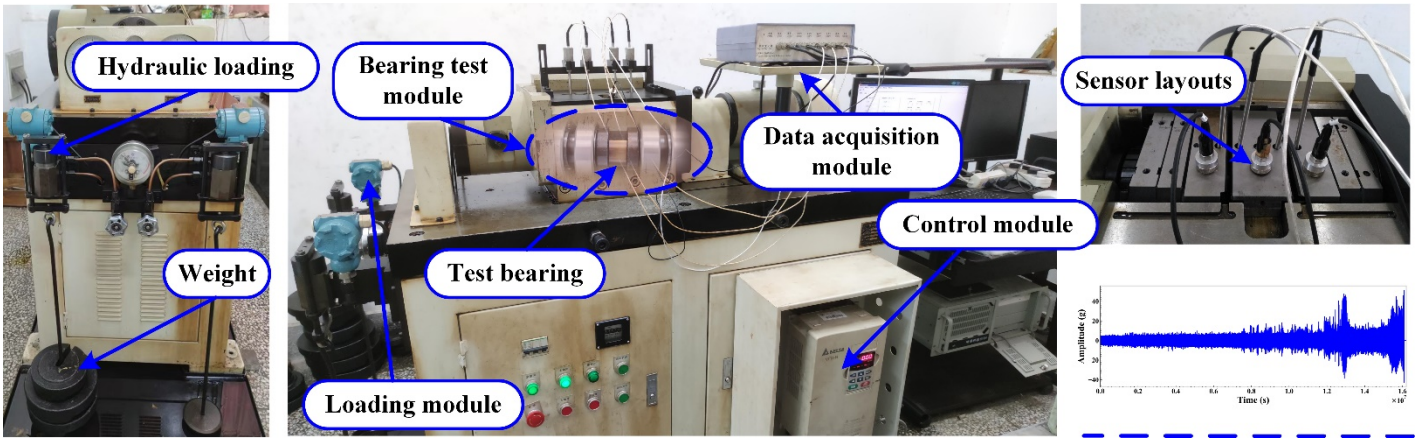


Fig. 9 Experimental test rig

TABLE 2 Test parameters and RUL prediction tasks

	Operating condition A	Operating condition B
Bearing type	6308 rolling bearings	
Load (N), Speed (rpm)	15000, 3000	17500, 3000
Data saving interval (s)	30	30
Proportion of data (%)	20, 30, 50	20, 30, 50
Threshold for shutdown	20	45
Sampling frequency (Hz)	25600	25600

Bearing	Bearing 1-1, Bearing 1-2, Bearing 1-3		Bearing 2-1, Bearing 2-2, Bearing 2-3
A → B	Task 1	Support Set: Bearing 1-1	Query Set: Bearing 2-1
	Task 2	Support Set: Bearing 1-2	Query Set: Bearing 2-1
	Task 3	Support Set: Bearing 1-3	Query Set: Bearing 2-1
	Meta-testing Set	Bearing 2-2, Bearing 2-3	
B → A	Task 4	Support Set: Bearing 2-1	Query Set: Bearing 1-1
	Task 5	Support Set: Bearing 2-2	Query Set: Bearing 1-1
	Task 6	Support Set: Bearing 2-3	Query Set: Bearing 1-1
	Meta-testing Set	Bearing 1-2, Bearing 1-3	

4.2 *Ablation Study*

4.2.1 *Different Sensitivity Parameters*

The different sensitive parameter settings of the model exhibit different effects on the prediction performance. In addition, combinations of these parameters can cause different sensitivity behaviors to RUL predictions. The prediction effects of different sensitive parameter combinations on the model are discussed in this subsection, and different parameter combinations are employed for the training and prediction of TARA. Note that these different parameters are subjected to a global exploration by the grid search algorithm, thus finding a set of best-fit parameter combinations. The RUL prediction results of different parameter combinations are shown in Fig. 10. It is evident that when λ_1 and λ_2 are set too large, the stability of the model can be significantly affected, resulting in catastrophic prediction results. In the appropriate range $\left(\lambda_1 \in [0.3, 0.8], \lambda_2 \in [0.3, 0.7]\right)$, the model can accurately predict the RUL of the bearing and obtain a low prediction error. It is clear from the results that the prediction performance of the model becomes more and more robust in the combined effect as λ_1 and λ_2 increases within the appropriate range. Obviously, TARA exhibits the best combined performance at $\lambda_1 = 0.7, \lambda_2 = 0.4$.

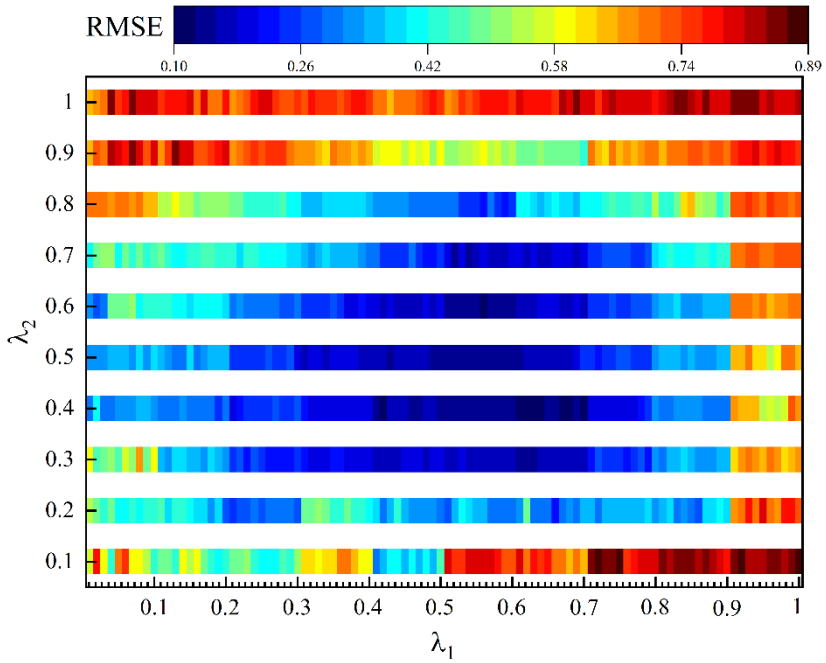
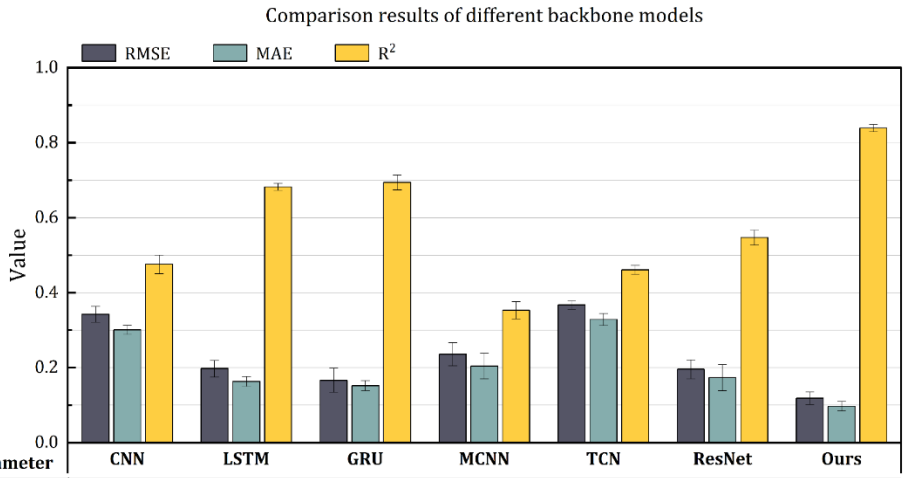


Fig. 10 Comparison results of different sensitivity parameters

4.2.2 *Different Backbone Models*

The deep autoregressive recurrent model is able to effectively extract the degradation features from the time series. To verify its effectiveness, a comparative analysis of six backbone networks is performed, including Convolutional Neural Network (CNN), Gate Recurrent Unit (GRU), Multiscale Convolutional Network (MCNN) [32], LSTM, Temporal Convolutional Network (TCN), and Residual Network (ResNet). Note that 1D convolutional operations are employed for CNN, MCNN, and ResNet. These models are adopted to replace the backbone network of TARA. TARA with different backbone networks is retrained and tested to perform RUL prediction. The comparison results of the different backbone models are shown in Fig. 11. Note that these results are generated in A → B (Bearing 2-2). Evidently, the prediction ability obtained by the proposed model far exceeds that of the other six methods and has significant advantages. In contrast, CNN, TCN, ResNet, and MCNN appear incapable of the prediction task in scenarios with few-shot samples. For recurrent networks, the prediction of GRU is stronger than LSTM but still has a significant gap with the proposed model. A comparative analysis of different backbone networks shows that the proposed model is competitive in handling time series data in a few-shot sample scenarios. It can be attributed to the fact that the statistical mode is embedded into the deep model, thus having a more sensitive focus on temporal information.



Parameter	CNN	LSTM	GRU	MCNN	TCN	ResNet	Ours
RMSE	0.3422±0.021	0.1972±0.02	0.1661±0.032	0.236±0.031	0.3671±0.019	0.1957±0.025	0.1185±0.017
MAE	0.3002±0.012	0.1636±0.013	0.152±0.011	0.2043±0.034	0.3281±0.016	0.1733±0.034	0.097±0.013
R^2	0.4758±0.025	0.6823±0.009	0.6944±0.02	0.3527±0.023	0.4609±0.012	0.547±0.02	0.839±0.01

Fig. 11 Comparison results of different backbone models

4.2.3 *Analysis of Attention Mechanism*

The semantic attention module aims to construct a complementary channel of local semantic information in each meta learning task. To validate the effectiveness of the attention module, different locations of the attention module are performed for ablation comparisons, and the results are reported in Table 3. The 1-th attention module is placed between the first and second layers of the LSTM. Note that the attention modules are placed sequentially between the layers of the BiLSTM. How the attention mechanism affects the learning of a task by a model piqued our interest. Thus, the features output from the modules in A → B (Bearing 2-3) are visualized in Fig. 12. It can be found that the prediction results that can be obtained by the model without the attention module are relatively poor, and its feature distribution is not obvious in the front period, thus preventing

the model from fully capturing the degradation features. The model prediction can be effectively improved by embedding the 1-th layer of attention module, but the features show a tendency to be weak in the latter period. Overall, the 2-th layer of attention module achieves the best prediction results. Although the feature distribution fluctuates, it is still relatively stable, and the features are more prominent and noticeable, as shown in Fig. 12 (c). Instead, the 3-th layer attention module becomes more volatile in feature presentation, which increases in magnitude. In the specific quantitative metrics, this uncontrolled feature bias is detrimental to the model's stable prediction.

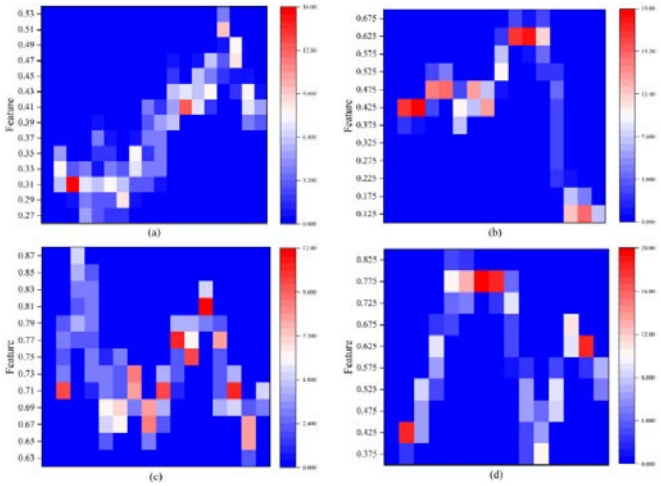


Fig. 12 Feature analysis of attention mechanism

TABLE 3 The metric results of the attention module

Attention module	A → B		
	RMSE	MAE	R^2
--Without multi-scale semantic attention	0.1212±0.019	0.1503±0.018	0.7521±0.023
--1-th layer of attention module	0.1025±0.014	0.0881±0.012	0.8523±0.011
--2-th layer of attention module	0.0823±0.011	0.0631±0.017	0.9192±0.009
--3-th layer of attention module	0.1006±0.020	0.0853±0.015	0.8613±0.017

4.3 *RUL Prediction*

The meta training of the model and RUL prognostic tests are performed in different tasks, and the RUL results for different task scenarios are shown in Fig. 13. Quantitative analysis is conducted for all results reported in Table 4. It can be clearly observed that the proposed TARA has achieved impressive results in different prognostic scenarios. Although there are substantial fluctuations in the predicted RUL curves, all amplitudes are within acceptable limits. In particular, the predicted fluctuations in Bearing 1-3 and Bearing 2-2 appear to be relatively flat later in the forecast. Satisfactorily, the predicted RUL curves consistently exhibit a clear trend and monotonicity, which are generally consistent with the actual RUL trends. Moreover, the 95% confidence bands obtained by the model always cover all the actual RULs over the full lifetime. The quantitative results show that the proposed TARA obtains commendable prediction errors in all scenarios. These errors are relatively concentrated and not divergent, which is sufficient to indicate that the prediction performance of the proposed model is stable. Particularly, it can be noticed that in the Bearing 2-2 prediction, the early period's prediction curves are unsatisfactory, and these error fluctuations are obviously difficult to match the actual degradation trend. We suspect that the early bearings are in

the break-in damage stage, where the degradation features of the signal are weak, thus affecting the discrimination of degradation feature extraction by the model and resulting in more pronounced error fluctuations.

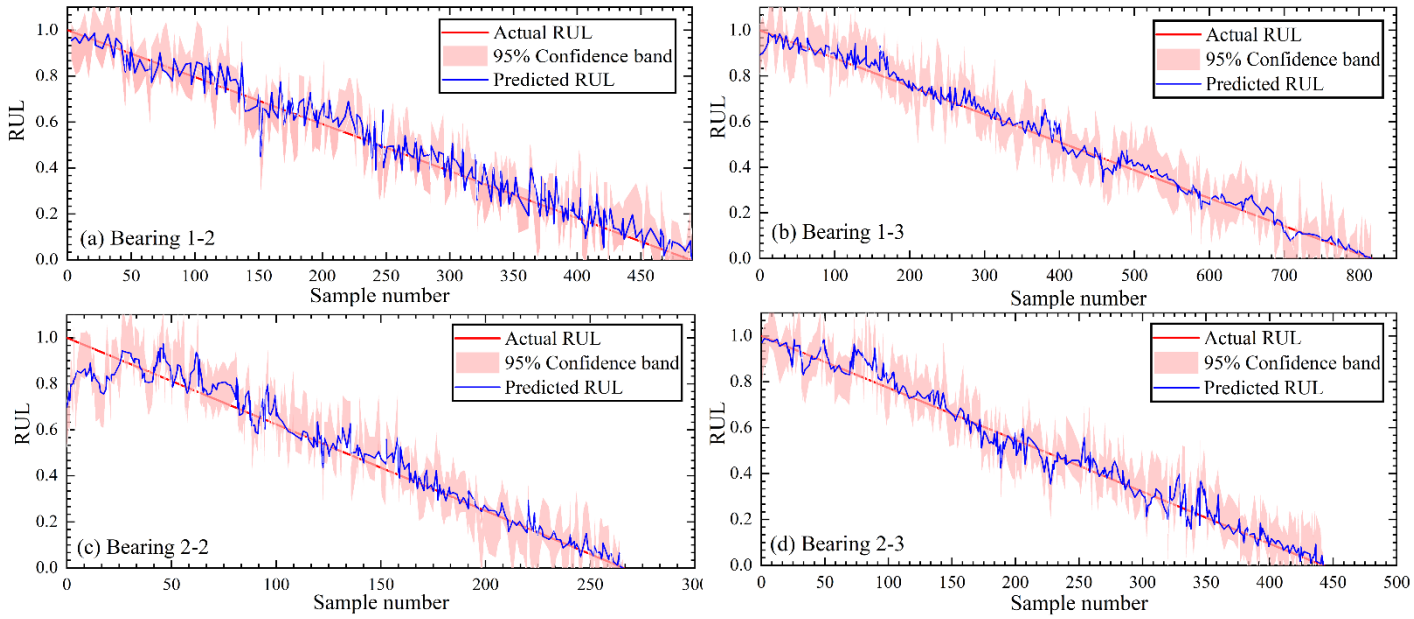


Fig. 13 Predicted results under different tasks [B → A: (a), (b); A → B: (c), (d)]

TABLE 4 The metric results of TARA

	Bearing	RMSE	MAE	R^2
A → B	Bearing 2-2	0.1185±0.017	0.0970±0.013	0.8390±0.010
	Bearing 2-3	0.0823±0.011	0.0631±0.017	0.9192±0.009
B → A	Bearing 1-2	0.1090±0.011	0.0803±0.014	0.8583±0.021
	Bearing 1-3	0.0603±0.027	0.0889±0.012	0.9058±0.016

4.4 *Comparison with State-of-the-art Methods*

Different ablation experiments can further illustrate the feasibility of the proposed TARA in self-constructed comparisons. However, it is insufficient to illustrate that the proposed method is still significantly contestable among similar methods. This subsection analyzes the proposed TARA compared to four similarly motivated and efficient methods. The four methods include an adversarial generation-based method 1 [25], two meta learning-based methods 2 [33] and 3 [28], and a domain adaptation-based method 4 [17]. Note that the descriptions of the different comparison methods are reported in Table 5, which allows further observation of the differences between them. The key parameters of these methods are reported in Table 6. The results obtained by these methods are quantified, as reported in Table 7.

TABLE 5 Description of differences in comparison methods

Methods	Description of differences
Method 1	Domain generalization method. The adversarial generation approach is utilized to generate new samples and perform RUL prediction using domain generalization for training, where the adversarial generation approach is different from our work.
Method 2	Meta learning method. A meta learning model is developed through CNN and GRU, which embeds a model of the domain adaptation learning. This is different from our approach.

Method 3	Meta learning method. A multi-head attention mechanism is employed to enhance feature capture, which is different from our approach, and we develop a semantic attention module.
Method 4	Domain adaptation method. The transfer of the source and target domains is achieved by means of adversarial learning, where the adversarial approach used is different from that of our method. This is an adversarial training of the model.

TABLE 6 Description of key parameters of different methods

Methods	Structures
Method 1	Generator (Conv1d(k = 7, s = 1, p = 3), InstanceNorm1d(), ReLU(), 2 × {Conv1d(k = 4, s = 2, p = 1), InstanceNorm1d(), ReLU()}, 2 × ResidualConnect{Conv1d(k = 3, s = 1, p = 1), InstanceNorm1d(), ReLU()}, 2 × {ConvTranspose1d(k = 4, s = 2, p = 1), InstanceNorm1d(), ReLU()}, ConvTranspose1d(k = 7, s = 1, p = 3), InstanceNorm1d(), Tanh()).Classifier (ResNet18-1d, Linear(2), Softmax()).Regressor (ResNet18-1d, Linear(1), Sigmoid())
Method 2	Block 1 (Convolution {2×2-32}, Activation function {ReLU}, Batch normalization {32}, Max pooling {1×2}). Block 2 (Convolution {2×2-64}, Activation function {ReLU}, Batch normalization {64}, Max pooling {1×2}). Block 3 (Convolution {2 × 2-64} Activation function {ReLU}, Batch normalization {64}, Max pooling {1×2}). Block 4 (Convolution {2×2-128}, Activation function {ReLU}, Batch normalization {128}, Max pooling {1×2}). Fully-connected {1×1×1152}
Method 3	G1 (Convolution (k=5, s=1), Position encoding, multi-head self-attention (), Reshape, Convolution (k=1, s=1)). G2 (Convolution (k=5, s=1), multi-head self-attention (),Reshape, Convolution (k=1, s=1)). G3 (Convolution (k=5, s=1), multi-head self-attention (), Reshape, Convolution (k=1, s=1)). Regressor (Liner (64), Liner (32), Linear(1), Sigmoid())
Method 4	Feature extractor (2 × {Dilated Causal Conv2d (k = 3, s = 1, p = 1, d = 1/2), weighetNorm (), ReLU (), Dropout (0.3), Dilate Causal Conv2d (k= 7, s = 1, p = 1, d = 1)}; 2 × Residual Connect{Conv2d (k = 3, s= 1, p = 1/2)} Maxpooling (2), Dual self-attention () 2 × {Dilated Causal Conv2d (k = 3, s = 1, p = 1, d = 1/2), weighetNorm (), ReLU (), Dropout (0.3), Dilate Causal Conv2d (k= 7, s = 1, p = 1, d = 1)}; 2 × Residual Connect{Conv2d (k = 3, s= 1, p = 1/2)}). RUL regressor(Dual self-attention (), Maxpooling (2), Liner (64), Liner (32), ReLU (), Dropout (0.3), Liner (1), Sigmoid). Discriminator (Liner (32), ReLU (), Liner (16), ReLU (), Liner (2), Softmax ())

In Fig. 14, it can be found that in a few-shot sample, it is insufficient to use only domain adaptation to transfer new working conditions. In other words, a small amount of sample data fails to support the training of domain adaptation, which leads to the learning ability of the model for the target working conditions is not sufficient. Also, considering the few-shot learning strategies into modeling and training, all of these methods are able to obtain a significant improvement in prediction performance, as in methods 1, 2, and 3. Fortunately, the advantages of the proposed TARA are still significant compared to these methods. In Method 1, the data augmentation method can further be used to obtain more accurate RUL predictions by expanding the sample data. However, in the A→B scenario, the prediction results of method 1 are more catastrophic, where the

fluctuations of prediction errors in the early stages are not normally accepted. The problem of few-shot samples is solved by expanding the sample distribution through the adversarial generation strategy. Although method 1 ensures that the samples are sufficient for model training, the prediction stability of method 1 is not sufficiently guaranteed. It can be attributed to the fact that these expanded samples may be irrelevant to the target data, i.e., these learnings are not valid for the target domain. Compared to methods 2 and 3, the meta learning method can further obtain more accurate RUL predictions by learning to update the parameters in different subtasks. In method 3, the training strategy with few-shot samples is the one that can satisfy the learning of a small number of samples, where the prediction results achieved by method 3 are second only to the proposed TARA. It lets us know that the training strategy of utilizing a few-shot learning can further improve the performance of the model, which is a direction worth exploring. Overall, the prediction performance of the proposed TARA is significantly competitive and effective among similar state-of-the-art methods.

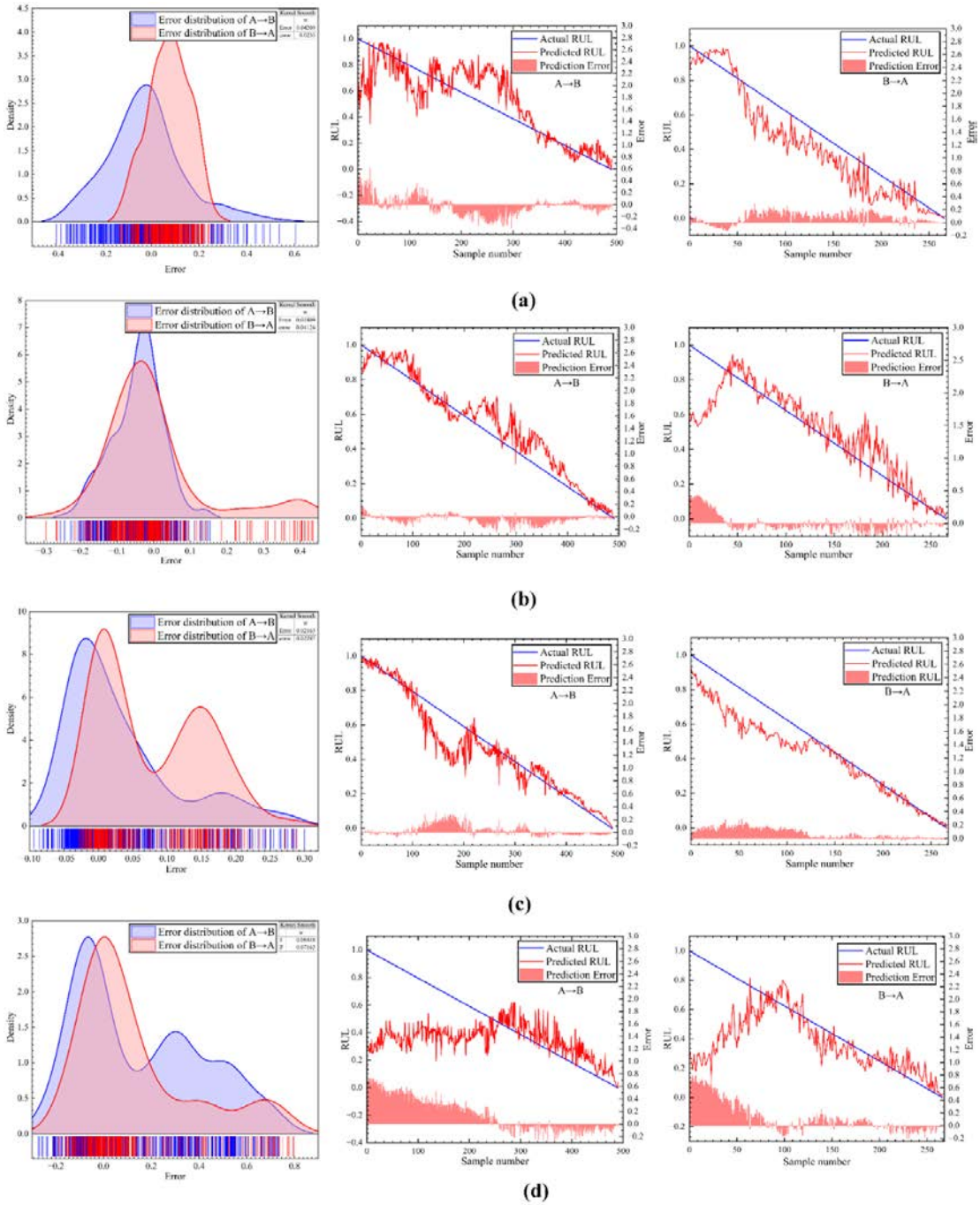


Fig. 14 Prediction results of different tasks (a. method 1, b. method 2, c. method 3, d. method 4)

TABLE 7 The metric results of different methods

Bearings	Method 1			Method 2			Method 3			Method 4		
	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2
Bearing 1-2	0.1631	0.1231	0.6832	0.1311	0.1014	0.7953	0.1458	0.1333	0.7465	0.3035	0.2312	0.3966
Bearing 1-3	0.1732	0.1620	0.6498	0.1352	0.0953	0.7853	0.1302	0.1076	0.7983	0.3493	0.3072	0.4780
Bearing 2-2	0.1503	0.1206	0.7320	0.1399	0.0947	0.7668	0.1139	0.0944	0.8453	0.2749	0.1765	0.4995
Bearing 2-3	0.1954	0.2749	0.5171	0.1502	0.1284	0.7259	0.1253	0.1065	0.8093	0.3525	0.3189	0.4723

4.5 *Additional Dataset Validation*

To further validate the generalizability of the proposed method on other data, the XJTU-SY bearing dataset is used for further validation. The bearing vibration data are obtained from the accelerated degradation experimental platform, as shown in Fig. 15. Two unidirectional accelerometers are mounted horizontally and vertically on the test bearing to collect vibration signals from operation to failure. Different operating conditions tests are executed with a sampling frequency of 25.6 kHz, a sampling interval of 1 Min, and a sampling duration of 1.28 s. Bearing testing stops when the bearing reaches 90% of its reliability for its basic rated life. The tasks in Case II are similar to those in Case I, as shown in Table 8.

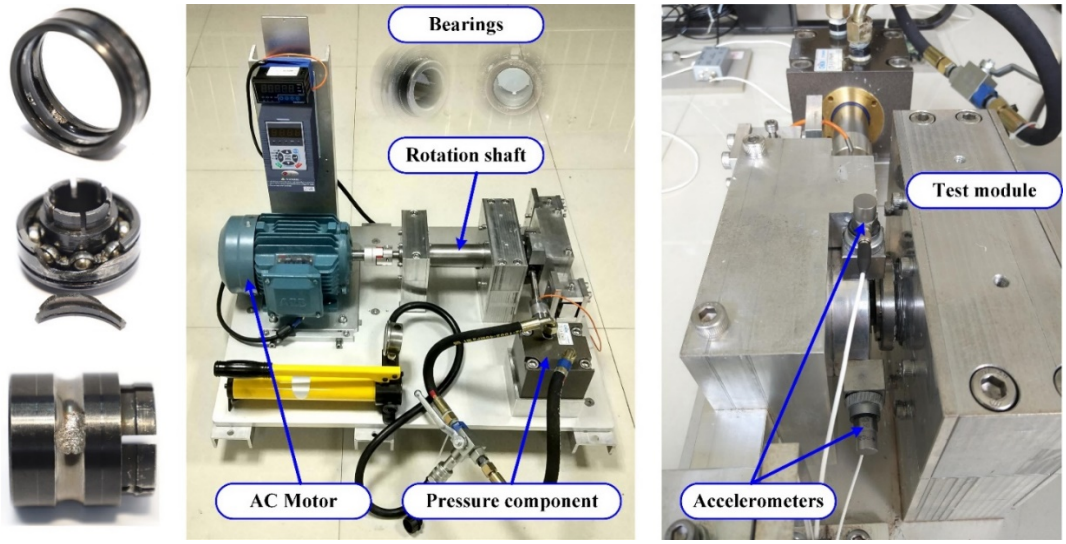


Fig. 15 Accelerated degradation experimental platform.

TABLE 8 Prediction task description for XJTU-SY dataset

Condition 1 → Condition 2	Task A	Support Set: XJTU Bearing 1-1 Query Set: XJTU Bearing 2-1
	Task B	Support Set: XJTU Bearing 1-2 Query Set: XJTU Bearing 2-1
	Task C	Support Set: XJTU Bearing 1-3 Query Set: XJTU Bearing 2-1
	Meta-testing Set	XJTU Bearing 2-2, XJTU Bearing 2-3
Condition 2 → Condition 1	Task D	Support Set: XJTU Bearing 2-1 Query Set: XJTU Bearing 1-1
	Task E	Support Set: XJTU Bearing 2-2 Query Set: XJTU Bearing 1-1
	Task F	Support Set: XJTU Bearing 2-3 Query Set: XJTU Bearing 1-1

	Meta-testing Set	XJTU Bearing 1-2, XJTU Bearing 1-3

The results of all methods are reported in Table 9. It can be found that the proposed TARA is also able to achieve satisfactory results in all tests. Compared to other methods, our method still has significant advantages. The best performance values are achieved on RMSE, MAE, and R^2. In contrast, Method 1 is still not adapted to the prediction task with limited samples. Also, it can be noticed that the results of Method 2 and Method 3 are very close. It also illustrates that meta learning does achieve few-shot learning. Method 4 predicts even worse, which also suggests that simple domain adaptation is not able to perform these prediction tasks effectively. In Method 4, the predicted results are largely skewed from the actual RUL values. Our proposed method can focus on more obvious degradation features in few-shot learning and construct significant feature representations of them in the semantic attention mechanism. In summary, the proposed method also achieves the best performance in other datasets.

TABLE 9 The metric results under XJTU-SY dataset

Bearings	Method 1			Method 2			Method 3			Method 4			Our		
	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2
XJTU 1-2	0.2038	0.1783	0.5138	0.1511	0.1242	0.7112	0.1507	0.1238	0.7195	0.3822	0.3433	0.4206	0.1103	0.0921	0.8640
XJTU 1-3	0.2291	0.1947	0.5005	0.1494	0.1355	0.7290	0.1439	0.1326	0.7483	0.3629	0.3072	0.4506	0.1372	0.1101	0.7598
XJTU 2-2	0.2113	0.1884	0.5198	0.1572	0.1282	0.6899	0.1528	0.1269	0.7095	0.4031	0.3884	0.3994	0.1244	0.1038	0.8114
XJTU 2-3	0.2454	0.1995	0.4937	0.1522	0.1277	0.7108	0.1533	0.1287	0.7027	0.3605	0.3402	0.4525	0.1129	0.0970	0.8499

5 Conclusion

This paper introduces the Transferable Autoregressive Recurrent Adaptation (TARA) method for predicting the remaining useful life (RUL) of bearings in situations with limited source data. The deep autoregressive recurrent model we construct incorporates a statistical mode to enhance the extraction of degradation features efficiently. Augmented learning for few-shot samples is achieved through attribute-assisted learning and adversarial generation schemes, generating new data that closely aligns with the source sample distribution. Additionally, to ensure that the global feature extraction process retains semantic information, we develop a semantic attention module and embed it into the basic model of meta learning, thereby enhancing semantic comprehension. Furthermore, we conduct numerous RUL experiments to validate the effectiveness of TARA. The experimental results demonstrate that TARA competently addresses RUL prediction in few-shot sample scenarios. Ablation studies are also included to showcase the impact of various backbone networks, sensitive parameter settings, and modules on prediction performance.

Although the proposed method can achieve the RUL prediction under different operating conditions with a limited sample, our method is still limited by physical information. Only deep virtual degradation features are not able to describe the interpretability of the model. In future research work, physical information-guided learning will be searched to accomplish the RUL prediction.

Acknowledge

The authors gratefully acknowledge the financial support of the National Natural Science Foundation of China (No. 52075095) and the China Scholarship Council. And the authors would like to appreciate the anonymous reviewers and the editor for their valuable comments.

References

[1] D. Wang, H. Cao, Y. Yang, M. Du, Dynamic modeling and vibration analysis of cracked rotor-bearing system based on rigid body element method, Mech. Syst. Signal Process. 191 (2023) 110152. https://doi.org/10.1016/j.ymssp.2023.110152.

[2] Y. Chen, M. Rao, K. Feng, G. Niu, Modified Varying Index Coefficient Autoregression Model for Representation of the Nonstationary Vibration From a Planetary Gearbox, IEEE Trans. Instrum. Meas. Y., Rao, M., Feng, K., Niu, G., 2023. Modif. Varying Index Coeff. Autoregression Model Represent. Nonstationary Vib. From a Planet. Gearbox. IEEE Trans. Instrum. Meas. . 72 (2023) 1–12. https://doi.org/10.1109/TIM.2023.3259048.

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

[4] N. Li, Y. Lei, X. Liu, T. Yan, P. Xu, Machinery Health Prognostics With Multimodel Fusion Degradation Modeling, IEEE Trans. Ind. Electron. 70 (2023) 11764–11773. https://doi.org/10.1109/TIE.2022.3231273.

[5] Q. Ni, J.C. Ji, K. Feng, Data-Driven Prognostic Scheme for Bearings Based on a Novel Health Indicator and Gated Recurrent Unit Network, IEEE Trans. Ind. Informatics. 19 (2023) 1301–1311. https://doi.org/10.1109/TII.2022.3169465.

[6] D. Wang, K.-L. Tsui, Theoretical investigation of the upper and lower bounds of a generalized dimensionless bearing health indicator, Mech. Syst. Signal Process. 98 (2018) 890–901. https://doi.org/10.1016/j.ymssp.2017.05.040.

[7] C.-G. Huang, J. Zhu, Y. Han, W. Peng, A Novel Bayesian Deep Dual Network With Unsupervised Domain Adaptation for Transfer Fault Prognosis Across Different Machines, IEEE Sens. J. 22 (2022) 7855–7867. https://doi.org/10.1109/JSEN.2021.3133622.

[8] J. Zhu, N. Chen, C. Shen, A new data-driven transferable remaining useful life prediction approach for bearing under different working conditions, Mech. Syst. Signal Process. 139 (2020) 106602. https://doi.org/10.1016/j.ymssp.2019.106602.

[9] W. Mao, J. He, M.J. Zuo, Predicting Remaining Useful Life of Rolling Bearings Based on Deep Feature Representation and Transfer Learning, IEEE Trans. Instrum. Meas. 69 (2020) 1594–1608. https://doi.org/10.1109/TIM.2019.2917735.

[10] P.R. de O. da Costa, A. Akçay, Y. Zhang, U. Kaymak, Remaining useful lifetime prediction via deep domain adaptation, Reliab. Eng. Syst. Saf. 195 (2020) 106682. https://doi.org/10.1016/j.ress.2019.106682.

[11] P. Chen, Y. Li, K. Wang, M.J. Zuo, P.S. Heyns, S. Baggeröhr, A threshold self-setting condition monitoring scheme for wind turbine generator bearings based on deep convolutional generative adversarial networks, Measurement. 167 (2021) 108234. https://doi.org/10.1016/j.measurement.2020.108234.

[12] M. Ragab, Z. Chen, M. Wu, C.S. Foo, C.K. Kwoh, R. Yan, X. Li, Contrastive Adversarial Domain Adaptation for Machine

Remaining Useful Life Prediction, IEEE Trans. Ind. Informatics. 17 (2021) 5239–5249. https://doi.org/10.1109/TII.2020.3032690.

[13] J. Jiao, M. Zhao, J. Lin, Unsupervised Adversarial Adaptation Network for Intelligent Fault Diagnosis, IEEE Trans. Ind. Electron. 67 (2020) 9904–9913. https://doi.org/10.1109/TIE.2019.2956366.

[14] J. Chen, W. Hu, D. Cao, Z. Zhang, Z. Chen, F. Blaabjerg, A Meta-Learning Method for Electric Machine Bearing Fault Diagnosis Under Varying Working Conditions With Limited Data, IEEE Trans. Ind. Informatics. 19 (2023) 2552–2564. https://doi.org/10.1109/TII.2022.3165027.

[15] P. Luo, Z. Yin, D. Yuan, F. Gao, J. Liu, An Intelligent Method for Early Motor Bearing Fault Diagnosis Based on Wasserstein Distance Generative Adversarial Networks Meta Learning, IEEE Trans. Instrum. Meas. 72 (2023) 1–11. https://doi.org/10.1109/TIM.2023.3278289.

[16] S. Zhang, F. Ye, B. Wang, T.G. Habetler, Few-Shot Bearing Fault Diagnosis Based on Model-Agnostic Meta-Learning, IEEE Trans. Ind. Appl. 57 (2021) 4754–4764. https://doi.org/10.1109/TIA.2021.3091958.

[17] J. Zhuang, M. Jia, X. Zhao, An adversarial transfer network with supervised metric for remaining useful life prediction of rolling bearing under multiple working conditions, Reliab. Eng. Syst. Saf. (2022) 108599. https://doi.org/10.1016/j.ress.2022.108599.

[18] J. Zhuang, M. Jia, Y. Ding, X. Zhao, Health Assessment of Rotating Equipment With Unseen Conditions Using Adversarial Domain Generalization Toward Self-Supervised Regularization Learning, IEEE/ASME Trans. Mechatronics. (2022) 1–11. https://doi.org/10.1109/TMECH.2022.3163289.

[19] J. Long, R. Zhang, Z. Yang, Y. Huang, Y. Liu, C. Li, Self-Adaptation Graph Attention Network via Meta-Learning for Machinery Fault Diagnosis With Few Labeled Data, IEEE Trans. Instrum. Meas. 71 (2022) 1–11. https://doi.org/10.1109/TIM.2022.3181894.

[20] L. Ren, T. Mo, X. Cheng, Meta-Learning Based Domain Generalization Framework for Fault Diagnosis With Gradient Aligning and Semantic Matching, IEEE Trans. Ind. Informatics. (2023) 1–11. https://doi.org/10.1109/TII.2023.3264111.

[21] Z. Xu, G. Tang, B. Pang, An Infrared Thermal Image Few-Shot Learning Method Based on CAPNet and Its Application to Induction Motor Fault Diagnosis, IEEE Sens. J. 22 (2022) 16440–16450. https://doi.org/10.1109/JSEN.2022.3192300.

[22] T. Zhang, J. Chen, S. He, Z. Zhou, Prior Knowledge-Augmented Self-Supervised Feature Learning for Few-Shot Intelligent Fault Diagnosis of Machines, IEEE Trans. Ind. Electron. 69 (2022) 10573–10584. https://doi.org/10.1109/TIE.2022.3140403.

[23] Z. Ren, D. Gao, Y. Zhu, Q. Ni, K. Yan, J. Hong, Generative adversarial networks driven by multi-domain information for improving the quality of generated samples in fault diagnosis, Eng. Appl. Artif. Intell. 124 (2023) 106542. https://doi.org/10.1016/j.engappai.2023.106542.

[24] Z. Ren, J. Ji, Y. Zhu, J. Hong, K. Feng, Generative adversarial network with dual multi-scale feature fusion for data

augmentation in fault diagnosis, IEEE Trans. Instrum. Meas. (2023) 1. https://doi.org/10.1109/TIM.2023.3310069.

[25] Y. Ding, M. Jia, Y. Cao, P. Ding, X. Zhao, C.-G. Lee, Domain generalization via adversarial out-domain augmentation for remaining useful life prediction of bearings under unseen conditions, Knowledge-Based Syst. 261 (2023) 110199. https://doi.org/10.1016/j.knosys.2022.110199.

[26] H. Lu, V. Barzegar, V.P. Nemani, C. Hu, S. Laflamme, A.T. Zimmerman, Joint training of a predictor network and a generative adversarial network for time series forecasting: A case study of bearing prognostics, Expert Syst. Appl. 203 (2022) 117415. https://doi.org/10.1016/j.eswa.2022.117415.

[27] Y. Feng, J. Chen, J. Xie, T. Zhang, H. Lv, T. Pan, Meta-learning as a promising approach for few-shot cross-domain fault diagnosis: Algorithms, applications, and prospects, Knowledge-Based Syst. 235 (2022) 107646. https://doi.org/10.1016/j.knosys.2021.107646.

[28] T. Pan, J. Chen, Z. Ye, A. Li, A multi-head attention network with adaptive meta-transfer learning for RUL prediction of rocket engines, Reliab. Eng. Syst. Saf. 225 (2022) 108610. https://doi.org/10.1016/j.ress.2022.108610.

[29] J. Lin, H. Shao, Z. Min, J. Luo, Y. Xiao, S. Yan, J. Zhou, Cross-domain fault diagnosis of bearing using improved semi-supervised meta-learning towards interference of out-of-distribution samples, Knowledge-Based Syst. 252 (2022) 109493. https://doi.org/10.1016/j.knosys.2022.109493.

[30] P. Tian, W. Li, Y. Gao, Consistent Meta-Regularization for Better Meta-Knowledge in Few-Shot Learning, IEEE Trans. Neural Networks Learn. Syst. 33 (2022) 7277–7288. https://doi.org/10.1109/TNNLS.2021.3084733.

[31] Y. Ding, P. Ding, M. Jia, A Novel Remaining Useful Life Prediction Method of Rolling Bearings Based on Deep Transfer Auto-Encoder, IEEE Trans. Instrum. Meas. 70 (2021) 1–12. https://doi.org/10.1109/TIM.2021.3072670.

[32] B. Wang, Y. Lei, N. Li, W. Wang, Multiscale Convolutional Attention Network for Predicting Remaining Useful Life of Machinery, IEEE Trans. Ind. Electron. 68 (2021) 7496–7504. https://doi.org/10.1109/TIE.2020.3003649.

[33] P. Ding, M. Jia, X. Zhao, Meta deep learning based rotating machinery health prognostics toward few-shot prognostics, Appl. Soft Comput. 104 (2021) 107211. https://doi.org/10.1016/j.asoc.2021.107211.