Proceedings of the 14th Asian Control Conference
July 5-8, 2024, Dalian, China

TAMTL: A Novel Meta-Transfer Learning Approach for Fault Diagnosis of Rotating Machinery

1st Yuheng Wu
School of Automation Science and Engineering
Xi'an Jiaotong University
Xi'an 710049, China
wuyuheng_0401@stu.xjtu.edu.cn

2nd Qingyu Yang*
School of Automation Science and Engineering, SKLMSE Laboratory
Xi'an Jiaotong University
Xi'an 710049, China
yangqingyu@mail.xjtu.edu.cn

3rd Donghe Li
School of Automation Science and Engineering
Xi'an Jiaotong University
Xi'an 710049, China
lidonghe2020@xjtu.edu.cn

4th Pengtao Song
School of Automation Science and Engineering
Xi'an Jiaotong University
Xi'an 710049, China
songpengtao@stu.xjtu.edu.cn

***Abstract*—This paper proposes a novel fault diagnosis scheme for rotating machinery based on meta-transfer learning and test augmentation. The model-agnostic meta-learning (MAML) framework is applied to the fault diagnosis problem by dividing the training and testing tasks according to different operating conditions, which allows the user to arbitrarily select the appropriate basic model according to the task requirements. Then, an additional pre-training phase based on meta-transfer learning is designed to improve the comprehensive performance, and a testing stage is introduced to evaluate the generalization performance of the hyperparameters of fine-tuned model. Experimental results on the CWRU dataset demonstrate that the proposed scheme can achieve high accuracy, stability, and efficiency in fault recognition under cross-condition scenarios.**

***Index Terms*—Fault diagnosis, rotating machinery, meta-transfer learning, test augmentation, MAML**

I. Introduction

Resource-intensive industries such as coal and steel are currently undergoing a transformation that focuses on improving efficiency while removing outdated capacity, which is also driving advances in intelligent automation solutions for the transport industry. As the power source of logistics transportation and distribution system (LTDS), the operation data of the motor contains a wealth of equipment fault information and is the key to develop intelligent diagnostic system in the field of modern LTDS. However, a significant challenge in industry upgrading is devising effective fault diagnosis algorithms amidst a scarcity of fault samples. Traditional fault diagnosis algorithms for rotating machinery typically involve an initial step of feature extraction from data, followed by classification using a classifier, in which case, the diagnostic accuracy is heavily reliant on the compatibility among the feature extraction algorithm, the classifier, and the original dataset [1], [2]. With the evolution of information technology in industrial devices, some statistical algorithms have been successively applied to feature extraction, including short-time Fourier transform (STFT) [3], wavelet transform [4], empirical mode decomposition (EMD) [5], and Hilbert-Huang transform (HHT) [6]. Successively, some robust machine learning algorithms, such as k-nearest neighbors (KNN), support vector machines (SVM) [6], and random forests [7], are employed to identify specific fault classifications.

With the further development of computer hardware and arithmetic power, 'end-to-end' deep learning algorithms have been progressively applied to the field of fault diagnosis, which can not only maintain a high recognition accuracy, but also address the issues of low generalizability and high expertise requirements associated with traditional diagnosis algorithms. Some milestone results can be seen in [8]. In recent years, benefiting from the vigorous development of deep learning algorithms, how to effectively apply fault diagnosis schemes to engineering practice has become a tricky challenge. Against this backdrop, several researches have been carried out successively to improve the generalization ability of diagnosis algorithms and address the prevalent issue of insufficient valid data. For example, Hasan *et al.* [9] achieved more than 90% accuracy in cross-condition fault identification by combining convolutional neural networks (CNNs) and transfer learning. In [10], a novel scheme based on metric learning is proposed for the intelligent diagnosis of rotating machinery, and the feasibility of the sample less learning in engineering practice is verified. Li *et al.* [11] proposed a model-agnostic meta-learning based fault diagnosis method, which converts raw signals under different operating conditions into time-

This work is supported by Shaanxi Logistics Group – Logistics Science and Innovation Integration Development Research Center of Xi'an Jiaotong University under Grant SKH2023005.

frequency images, and then randomly samples them based on the meta-learning architecture. On this basis, a novel strategy of hierarchical recursive meta-learning for data reconstruction is designed in [12], where the CWRU dataset is divided into 92 working conditions for experimental study, providing a new solution to deal with the common problems of fewer samples in real engineering.

The aforementioned research showcases the broad application prospects and potential value of few-shot learning algorithms based on deep learning in the field of fault diagnosis. However, certain challenges still exist in practical applications, such as the requirement for abundant source domain data to train transfer learning-based diagnosis algorithms before transferring to the target domain, and the stringent similarity between source and target domain. Instead, the meta-learning based fault diagnosis scheme is more in line with the needs of engineering practice as it can overcome the effects of unbalanced and insufficient data. However, most existing meta-learning based results often focuses unilaterally on increasing the number of training tasks [12], ignoring cross-condition issues in engineering practice. Moreover, these studies tend to over-rely on the original MAML framework and overlook a key limitation: lack of validation sets that are important in data analysis, which may affect the generalization ability of the model and lead to overfitting of the final training results when solving cross-condition diagnosis problems. Motivated by the above statements, this paper proposes a novel MAML-based scheme named Test-augmented meta-transfer learning (TAMTL) for fault diagnosis of rotating machinery. The main contributions are as follows.

- A MAML-based fault diagnosis algorithm is proposed, and the training and testing tasks are divided according to different conditions, which enhances the interpretability of the framework and improves its effectiveness in engineering applications.
- A pre-training step using meta-migration learning is designed and added to the original model, which increases the training speed and improves the stability of convergence process during model training. Moreover, the proposed scheme incorporates the advantages of transfer learning and meta-learning to mine class-separated and domain-invariant features.
- An additional testing stage is designed to assess the generalization performance of the hyperparameters of fine-tuned model. Experimental results on the CWRU dataset demonstrate that the proposed scheme can achieve high accuracy, stability, and efficiency in fault recognition under cross-condition scenarios.

II. Preliminaries

The MAML algorithm is one of the few-shot learning algorithms proposed by Chelsea Finn *et al.* [13] in 2017, and its core goal aims to quickly adapt to new tasks by relying on a small amount of labelled data to train the model, with the key advantage of being model-agnostic and can be used with various deep learning models. The framework of the original MAML model is shown in Fig. 1.

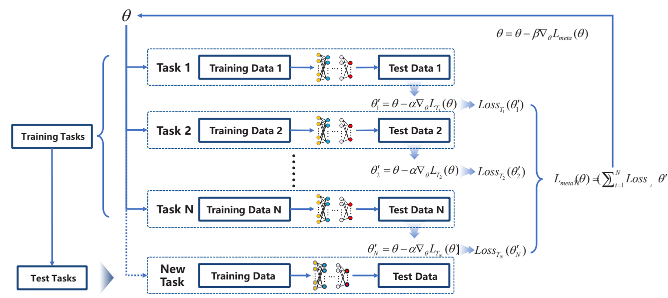

Fig. 1. The original architecture of MAML.

As a typical meta-learning approach, MAML aims to train a set of effective model parameters to replace the randomly initialized model parameters commonly used in traditional model training. These parameters should allow rapid adaptation to new tasks with minimal data updates and training iterations. In summary, MAML mainly consists of a training task and a testing task. Specifically, the role of the training task is to train the model to obtain common initial parameters applicable to a variety of tasks, while the testing task is used to evaluate the generalizability of the model on new tasks. In addition, the implementation process of the training task can be divided into two parts: inner loop and outer loop. The parameter updates and loss calculations performed in each task constitute the inner loop, while the direct updating of the shared initial parameters for all tasks constitutes the outer loop. The detailed training process is as follows.

Firstly, initialize the model parameters θ. Then, for each task T_i, the same network structure is employed for a few rounds of training and parameter updates to obtain N sets of task-specific parameter θ_i' and loss function $Loss_{T_i}(\theta_i')$, which are given by

$$\theta_i' = \theta - \alpha \nabla_\theta L_{T_i}(\theta) \quad (1)$$

$$Loss_{T_i}(\theta_i') = -\frac{1}{D_t^i} \sum_{(x_j,y_j)\in D_t^i} \sum\nolimits_{c=1}^{C} y_{j,c} \log(p_\theta(c|x_j)) \quad (2)$$

$$Loss_{T_i}(\theta_i') = -\frac{1}{D_t^i} \sum\nolimits_{(x_j,y_j)\in D_t^i} L(f_{\theta'_i}(x_j), y_j) \quad (3)$$

where α is the learning rate, $Loss_{T_i}$ is the loss function for each task, D_t^i is the number of samples in the training set of the i-th task, and $L(f_{\theta'_i}(x_j), y_j)$ denotes the cost of the model on a single sample x_j and label y_j under parameter .

Then, integrating the cost of N tasks yields the loss function of the outer loop as

$$L_{meta}(\theta) = \sum\nolimits_{i=1}^{N} L_{T_i}(\theta'_i) \quad (4)$$

Subsequently, the initial parameter θ is updated by

$$\theta = \theta - \beta \nabla_\theta L_{meta}(\theta) \quad (5)$$

where β is the learning rate of the outer loop.

The above process is repeated for parameter iteration until the termination condition is met and the resulting optimized parameters are used for the test task. Then, after a few steps of fine-tuning, the final model parameters derived from θ can be used to evaluate generalization ability.

However, the following challenges still exist in applying MAML framework to fault diagnosis problems.

- MAML was originally developed for datasets such as Omniglot and Mini-Imagenet, which have numerous classes with few samples per class. In contrast, fault diagnosis datasets are typically characterized by continuous-time vibration signals with fewer categories. Previous attempts to use MAML to fault diagnosis often failed to ensure a clear separation between training and testing data, thus hindering its cross-condition generalization and reducing its practical utility.
- The original MAML framework relies on fine-tuning test tasks to evaluate model performance. However, few-shot training commonly leads to overfitting, and assessing model performance based solely on test task accuracy is not an effective measure of the generalizability of initial parameters. Thus, additional evaluation methods are needed to address this issue.
- MAML requires training on multiple tasks, which consumes a lot of time and computing resources. For real-world applications, it is critical to improve the speed of model training and convergence efficiency while maintaining fault identification accuracy.

Based on the above discussion, in the subsequent section, we propose an improved diagnosis scheme to address the issues of dataset applicability, evaluation methodology and computational efficiency.

III. Design of The TAMTL Architecture

This section presents the design process of the proposed TAMTL, which mainly involves three stages: model adaptation, pre-training initial parameters, and test augmentation.

A. Model Adaptation

The embedded CNN algorithm in the original MAML architecture is primarily designed for processing two-dimensional image data and is not suitable for the common one-dimensional continuous-time domain data in the fault diagnosis field. Therefore, when adjusting the shape of the input data in the original architecture, the basic model also needs to be tuned. In this paper, the WDCNN proposed in [8] is used as a basic model for the MAML framework. This algorithm utilizes wide convolution kernels in the first convolutional layer to better extract features and improve noise resistance, making it suitable for directly processing one-dimensional time domain data. The specific model structure is shown in Fig. 2, which includes 5 convolutional layers, 5 pooling layers, and 1 fully connected layer, followed by softmax for fault recognition classification.

Additionally, the dataset is divided into training tasks and testing tasks based on different operating conditions, ensuring

Number	Network Layer	Kernel Size/Stride	Kernel Number	Output Size	Padding
1	Convolution 1	64×1/16×1	16	128×16	Y
2	Pooling 1	2×1/2×1	16	64×16	N
3	Convolution 2	3×1/1×1	32	64×32	Y
4	Pooling 2	2×1/2×1	32	32×32	N
5	Convolution 1	3×1/1×1	64	32×64	Y
6	Pooling 1	2×1/2×1	64	16×64	N
7	Convolution 1	3×1/1×1	64	16×64	Y
8	Pooling 1	2×1/2×1	64	8×64	N
9	Convolution 1	3×1/1×1	64	6×64	N
10	Pooling 1	2×1/2×1	64	3×64	N
11	Full Connection	100	1	100×1	
12	Softmax	10	1	10	

Fig. 2. The network parameters of WDCNN.

that the training tasks handle data from a single operating condition and focus on identifying fault types specific to that condition. The testing tasks utilize data from completely different operating conditions, ensuring the rigor and effectiveness of the testing process. This treatment moves away from the random selection of training tasks used in traditional MAML architectures, enhances the interpretability of the model, and clearly indicates the focus of the research: how to use limited data from multiple operating conditions to improve the ability of the inherent model to diagnose faults under new conditions.

Specifically, in the original MAML framework, the setting is N-way, K-shot, where the model needs to learn to identify N different categories with only K samples. However, in the adjusted framework, since the load conditions of the rotating machinery remain consistent within each training task, and the types and quantities of faults to be classified are consistent, N is fixed as the number of fault types for each operating condition in the revised framework and is no longer a tunable hyperparameter. Thus, we have:

$$D^i_{train} = N \times K \tag{6}$$

where the size of training set for task T_i is equal to the number of categories multiplied by the number of samples per category, and N is a fixed constant.

Then, the loss function of each task is given by

$$L_{CE}(f_\theta(x), y) = -\sum_{c=1}^{C} y_c \log(p_\theta(c|x)) \tag{7}$$

where C is the number of categories, y_c is an indicator variable that is 1 if the sample x belongs to category c, and 0 otherwise, and $p_\theta(c|x)$ is the probability predicted by the model under parameters θ.

Substituting Eq. (6) and (7) into Eq. (2) yields the following loss to calculate the cost function for tasks after determining the basic model and task partitioning

$$Loss_{T_i}(\theta'_i) = -\frac{\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{c=1}^{C} y_{n,k,c}\log(p_{\theta'_i}(c|x_{n.k}))}{NK} \tag{8}$$

where $x_{n,k}$ and $y_{n,k,c}$ represent the input and indicator variables, respectively, for the k-th sample in the n-th category.

This section implements the adjustment of the original MAML framework by defining the basic model and adapting the task partitioning to make it suitable for fault diagnosis in rotating machinery.

B. Test Augmentation

The original MAML framework uses the parameter obtained from training as the initial parameter for the testing tasks. After fine-tuning with a small number of samples in the testing tasks, the final model parameters are obtained, as shown in Eq. (5). Moreover, the parameters fine-tuned with few samples can be used to classify and recognize samples in the query set of the testing tasks as a basis for adjusting the hyperparameters and evaluating the framework performance. Although the original MAML framework avoids overfitting to some extent by cross-validating across multiple tasks, as evidenced by the fact that the model performs well on query sets across multiple tasks, the framework still suffers from logical flaws and risk of overfitting due to the lack of a separate validation set. Especially in the testing tasks, the limited data used in the fine-tuning process further amplifies the possibility of model overfitting, thus affecting its generalization ability. To address this issue, this section introduces a method of test augmentation into the adapted meta-learning architecture.

Specifically, while keeping the main structure of the adjusted meta-learning scheme unchanged, the original testing tasks are first regarded as validation tasks, and the performance of the trained model parameters on the validation tasks is served as the basis for adjusting the hyperparameters. Then, a dataset with the same operating conditions and sufficient fault samples as the validation task is then fed into the initially fine-tuned model parameters, and the recognition accuracy of the fine-tuned parameters on that dataset will be used as a true measure of the performance of the framework. The hyperparameters in the meta-learning framework mainly include the learning rate β of the outer loop and the number of update steps in the inner loop tasks. After completing the adaptive adjustment of the meta-learning framework, the number of tasks and the model structure are fixed. Therefore, the main hyperparameters that need to be adjusted are concentrated on the learning rate β of the outer loop and the number of update steps in the inner loop tasks. The calculation process of these two hyperparameters is given by

$$\begin{cases} \alpha_{opt} = \arg\min_{\alpha} L_{val}(\theta'(\alpha)) \\ Steps_{opt} = \arg\min_{\alpha} L_{val}(\theta'_{Steps}) \end{cases} \tag{9}$$

where α_{opt} is the optimized learning rate, $Steps_{opt}$ is the optimized number of update steps, $\theta'(\alpha)$ and θ'_{Steps} represent the model parameters under the corresponding conditions, $L_{val}(\theta')$ is the loss on the validation set, which is given by

$$L_{val}(\theta') = \frac{1}{|D_{val}|} \sum\nolimits_{(x,y)\in D_{val}} L(f_{\theta'}(x), y) \tag{10}$$

where D_{val} is the number of samples in the validation task, and θ' denotes the parameters obtained from fine-tuning parameter θ.

This design separates the adjustment of hyperparameters from the evaluation of the framework performance, thereby enhancing the rigor of the scheme in experiments, and providing a more intuitive and effective assessment of the generalization performance of fine-tuned model parameters.

C. Pre-training Initial Parameters

After adjustment and test augmentation, the meta-learning model still needs to undergo training on multiple tasks to obtain a good initial starting point . However, due to the introduction of the test augmentation step, training this meta-learning framework consumes more time and computing resources compared to the original MAML. In practical engineering, this often implies higher economic and time costs. Thus, in this section, a meta-transfer based pre-training step is designed to speed up the training and reduce the time of parameter convergence.

In meta-learning architectures without pre-training, the parameters θ initially input into the base model are randomly initialized random numbers. We mix data with consistent fault types under different operating conditions as simple CNN training and validation data. One or two gradient updates are performed, and the resulting model parameters are used as the initial parameters to train the TAMTL framework. The specific steps for combining similar fault types under different operating conditions are shown in Fig. 3.

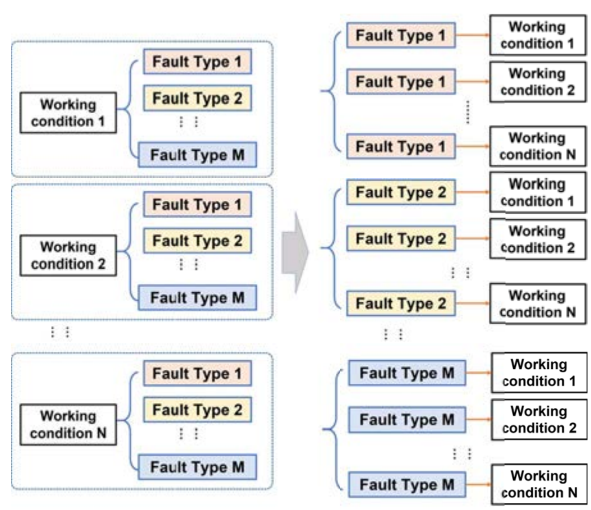

Fig. 3. Process diagram of similar types under different conditions.

As shown in Fig. 3, the original data divided into N operating conditions are reorganized, and each operating condition contains the same M fault types. Then, each identical fault type is extracted from different operating conditions to form a new dataset. In this way, the new dataset consists of M fault types. Subsequently, this composed new dataset is fed into the selected basic model for several initial iterations.

The essence of using pre-trained parameters is to replace several outer loop parameter updates in MAML with multiple single CNN parameter updates, thereby saving resources and training time. This approach ensures faster and more stable

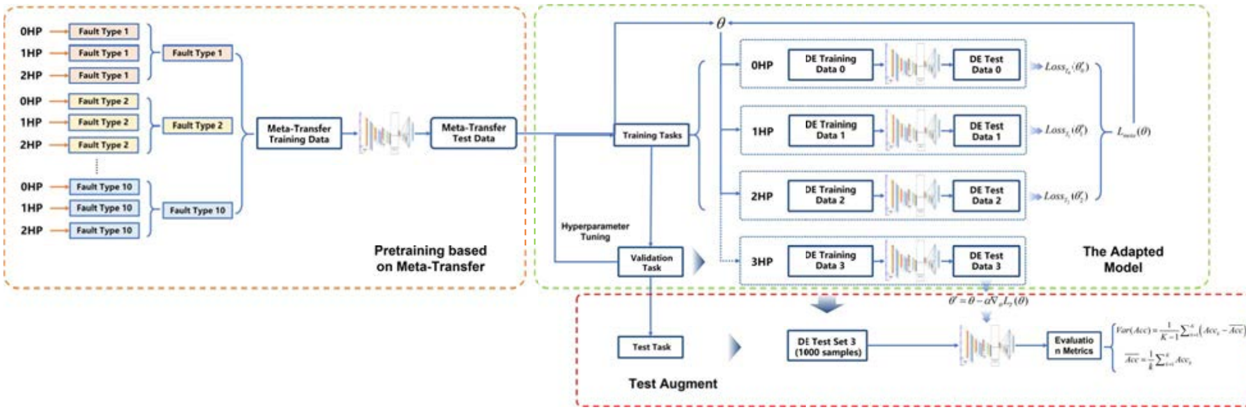

Fig. 4. Framework of the proposed TAMTL.

convergence of model parameters without increasing the number of training data or gradient updates.

IV. Numerical Experiments

This section validates the performance of the proposed TAMTL scheme with the CWRU dataset.

A. Experimental dataset

This experiment will utilize the dataset from the Case Western Reserve University (CWRU) Bearing Data Center. The CWRU dataset is an open-source standard dataset that has been widely used in the field of fault diagnosis. The device in the CWRU dataset is the SKF6205 deep groove ball bearing, and its data collection system is shown in Fig. 5.

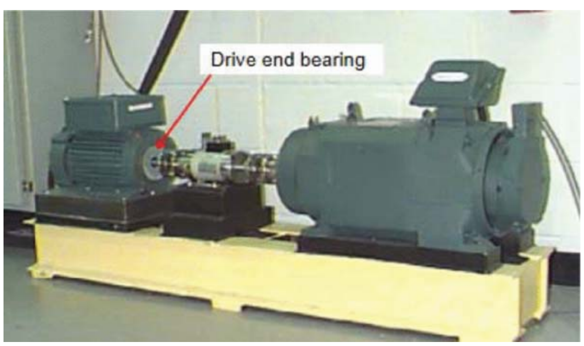

Fig. 5. The data collection system of the CWRU dataset.

In this study, the sampling frequency of the data acquisition system is 12 kHz. The dataset includes four operating conditions: 0HP, 1HP, 2HP, and 3HP. Under each operating condition, there are 10 fault scenarios, consisting of three types of defect locations (ball damage, outer race damage, inner race damage) and three sizes of damage diameters (0.007 inch, 0.014 inch, and 0.021 inch), forming nine fault states and one healthy state.

B. Experimental parameter configuration

Fig. 4 displays the TAMTL framework constructed using drive-end vibration data (DE) selected from the dataset. Three operational conditions, labeled as 0HP, 1HP, and 2HP, are utilized as training tasks, while the data from the 3HP condition is set as the testing task. Each task involves the classification of the same set of 10 fault types, with the base model adjusted to WDCNN. Before the formal training, the initial parameter θ is set to those obtained after five pre-training epochs, and the data for meta-transfer learning (pre-training) is segmented from the original training tasks to ensure the total training data remains unchanged. Finally, the performance metrics of the framework are obtained through the test augmentation part, where 1000 random samples are selected from the 3HP condition for testing. The performance metrics are given by

$$\overline{Acc} = \frac{1}{k}\sum_{k=1}^{K} Acc_k \tag{11}$$

$$Var(Acc) = \frac{1}{K-1}\sum_{k=1}^{K}\left(Acc_k - \overline{Acc}\right)^2 \tag{12}$$

where $\overline{Acc}$ and $Var(Acc)$ are the accuracy and variance of fault recognition calculated from the samples extracted from the testing task input with the parameters θ' after fine-tuning. The accuracy Acc_k of each experiment is defined as

$$Acc_k = \frac{1}{M}\sum_{i=1}^{M} y_M\left(f_k(x_i, {\theta'}_k) = y_i\right) \tag{13}$$

where f_k is the model of the k-th experiment, θ'_h denotes the corresponding parameter after fine-tuning, M stands for the number of samples in the testing set, and in this experiment, we set $M = 1000$. y_M is the indicator function. Additionally, we introduce a well-established fine-tuning-based transfer learning algorithm as the baseline algorithm.

C. Simulation Results

After setting up the framework, in each epoch, each task consists of 5 training data and 10 validation data. The outer loop learning rate is set to 1×10^{-3}. The parameters in each training task are updated 3 times, and during fine-tuning within training tasks, parameters are updated 8 times. The maximum number of iterations is set to 30. TAMTL includes a pre-training stage with 5 parameter update epochs. Therefore,

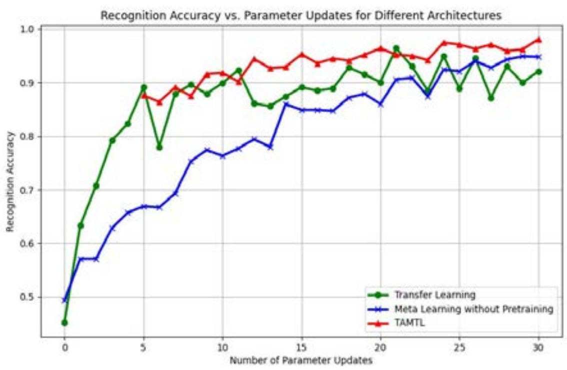

Fig. 6. Comparison of fault recognition accuracy under three learning modes.

the testing values for the TAMTL will be recorded from the classification accuracy starting from the 5th parameter update.

Fig. 6 shows the comparison of fault recognition accuracy among transfer learning, meta-learning without pre-training, and TAMTL under different parameter update epochs in a single experiment. One can that the recognition accuracy of all three schemes tends to plateau as the number of parameter updates increases, eventually converging. Under the same conditions of training data and parameter update times, transfer learning demonstrates faster adaptation to new tasks than meta-learning without pre-training. Before convergence, transfer learning often achieves higher recognition accuracy than meta-learning without pre-training. However, compared with meta-learning without pre-training, the recognition accuracy of transfer learning is slightly lower and unstable, and sometimes there is a significant "negative transfer" phenomenon. In contrast, the TAMTL framework incorporates the advantages of transfer learning and meta-learning without pre-training, achieving a balance. As shown in the figure, the test data for TAMTL is recorded from the 5th parameter update onwards. In the subsequent parameter update step, TAMTL exhibits the highest stability and convergence accuracy.

Fig. 7 and Fig. 8 shows the confusion matrix and t-SNE visualization results for models saved at 0, 5, and 20 gradient update times. The first column of two figures shows the models trained with transfer learning, with gradient update frequencies of 0, 5, and 20 saved. Similarly, the second and third columns depict the models trained without pre-training and those trained with TAMTL, respectively, providing a more intuitive representation of the performance of the three architectures. Among these results, the transfer learning demonstrates faster training speed before reaching the convergence of fault diagnosis accuracy, and the meta-learning shows higher recognition accuracy and stability after convergence. The proposed TAMTL integrates the advantages of both transfer learning and meta-learning, offering not only faster training speed but also balanced training accuracy and stability.

The average accuracy and variance of fault recognition for

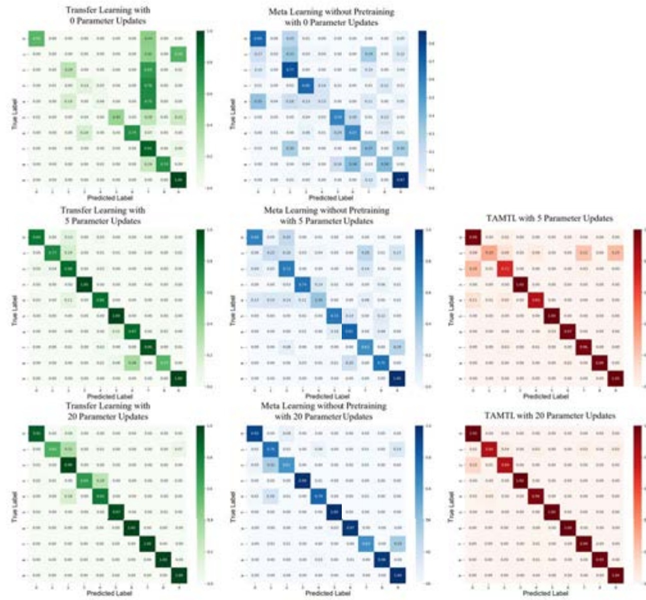

Fig. 7. Confusion matrices for the three learning modes.

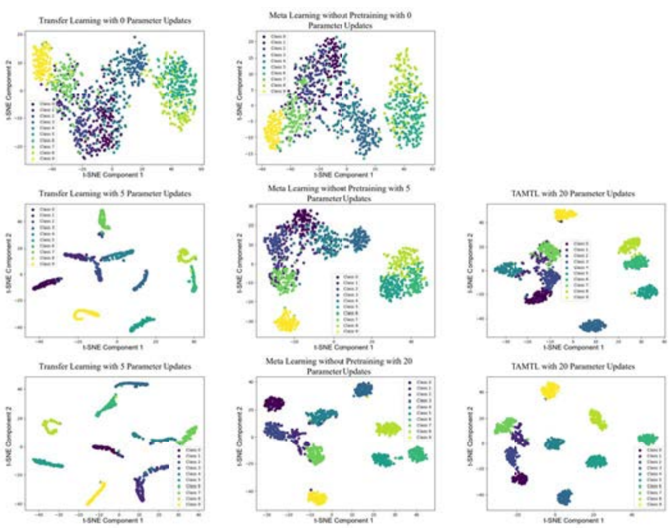

Fig. 8. t-SNE visualization results for the three modes.

the TAMTL scheme are presented numerically based on five repetitions of training with 20 parameter update iterations for each of the three architectures, as shown in Fig. 9. When all three models approach the accuracy convergence value, TAMTL exhibits the highest average accuracy and the lowest standard deviation, further confirming the previous conclusions.

V. Conclusion

This paper proposes a novel fault diagnosis scheme for rotating machinery based on meta-transfer learning and test augmentation. It can divide training and testing tasks according to different operating conditions, and an additional pre-training and testing stage is designed to improve the comprehensive

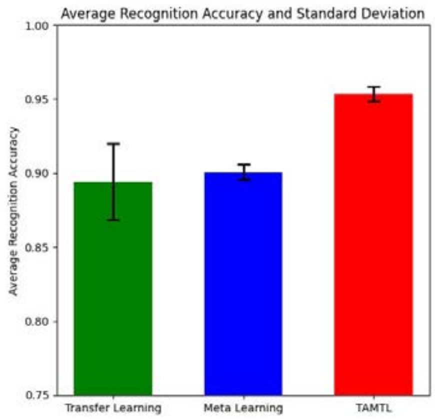

Fig. 9. The average fault recognition accuracy and standard deviation of the three architectures.

performance. The supplementary testing stage addresses the logical flaw of lacking validation tasks in the original MAML framework, enhancing the rigor of performance evaluation. Finally, experimental results on the CWRU dataset demonstrate that the proposed scheme can achieve an average fault recognition accuracy of 95% with a standard deviation of 0.0050 under conditions of limited total training samples. Compared with transfer learning and meta-learning without pre-training under the same amount of training data and number of parameter updates, TAMTL shows superior performance in diagnosis accuracy, stability, and efficiency.

REFERENCES

[1] H. F. Tao, L. Cheng, J. E. Qiu, and V. Stojanovic, "Few shot cross equipment fault diagnosis method based on parameter optimization and feature mertic," *Meas Sci Technol*, vol. 33, no. 11, Nov 1 2022.

[2] P. Song, Q. Yang, D. Li, and H. Dang, "Sparse Filtering with Joint Distribution Adaptation for Intelligent Fault Diagnosis," in *2023 42nd Chinese Control Conference*, 2023, pp. 1-5.

[3] Z. P. Feng, M. Liang, and F. L. Chu, "Recent advances in time-frequency analysis methods for machinery fault diagnosis: A review with application examples," *Mech Syst Signal Pr*, vol. 38, no. 1, pp. 165-205, Jul 5 2013.

[4] R. Q. Yan, R. X. Gao, and X. F. Chen, "Wavelets for fault diagnosis of rotary machines: A review with applications," *Signal Process*, vol. 96, pp. 1-15, Mar 2014.

[5] Y. G. Lei, J. Lin, Z. J. He, and M. J. Zuo, "A review on empirical mode decomposition in fault diagnosis of rotating machinery," *Mech Syst Signal Pr*, vol. 35, no. 1-2, pp. 108-126, Feb 2013.

[6] A. Soualhi, K. Medjaher, and N. Zerhouni, "Bearing Health Monitoring Based on Hilbert-Huang Transform, Support Vector Machine, and Regression," *Ieee T Instrum Meas*, vol. 64, no. 1, pp. 52-62, Jan 2015.

[7] M. Cerrada, G. Zurita, D. Cabrera, R. V. Sánchez, M. Artés, and C. Li, "Fault diagnosis in spur gears based on genetic algorithm and random forest," *Mech Syst Signal Pr*, vol. 70-71, pp. 87-103, Mar 2016.

[8] W. Zhang, G. L. Peng, C. H. Li, Y. H. Chen, and Z. J. Zhang, "A New Deep Learning Model for Fault Diagnosis with Good Anti-Noise and Domain Adaptation Ability on Raw Vibration Signals," *Sensors-Basel*, vol. 17, no. 2, Feb 2017.

[9] M. J. Hasan and J. M. Kim, "Bearing Fault Diagnosis under Variable Rotational Speeds Using Stockwell Transform-Based Vibration Imaging and Transfer Learning," *Appl Sci-Basel*, vol. 8, no. 12, Dec 2018.

[10] A. S. Zhang, S. B. Li, Y. X. Cui, W. L. Yang, R. Z. Dong, and J. J. Hu, "Limited Data Rolling Bearing Fault Diagnosis With Few-Shot Learning," *Ieee Access*, vol. 7, pp. 110895-110904, 2019.

[11] C. A. J. Li, S. B. Li, A. S. Zhang, Q. He, Z. H. Liao, and J. J. Hu, "Meta-learning for few-shot bearing fault diagnosis under complex working conditions," *Neurocomputing*, vol. 439, pp. 197-211, Jun 2021.

[12] H. Su, L. Xiang, A. J. Hu, Y. G. Xu, and X. Yang, "A novel method based on meta-learning for bearing fault diagnosis with small sample learning under different working conditions," *Mech Syst Signal Pr*, vol. 169, Apr 2022, Art no. 108765.

[13] C. Finn, P. Abbeel, and S. Levine, "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks," *Pr Mach Learn Res*, vol. 70, 2017.