Unsupervised Bearing Fault Diagnosis via a Multi-Layer Subdomain Adaptation Network

Nguyen Duc Thuan, Nguyen Thi Hue, Hoang Si Hong*
School of Electrical and Electronic Engineering, Hanoi University of Science and Technology, Hanoi, Vietnam

***Abstract*—Bearings play a crucial role in the functioning of rotating machinery, making it essential to monitor their condition for maintaining system stability and dependability. In recent years, intelligent diagnostic techniques for bearing issues have made significant progress due to advancements in artificial intelligence. These methods rely heavily on data, requiring data collection and labeling to develop the learning model, which is often highly challenging and nearly infeasible in industrial settings. As a result, a domain adaptation-based transfer learning approach has been suggested. This approach aims to minimize the difference between the distribution of accessible data and the unlabeled real-world data, enabling the model trained on public data to function effectively with actual data. In this paper, we introduce a sophisticated subdomain adaptation technique for cross-machine bearing fault diagnosis using vibration, termed multi-layer subdomain adaptation. Verification experiments were conducted, and the findings indicate that the proposed approach offers relatively high accuracy up to 97.47% and excellent transferability. Comparative experiments revealed that the proposed method is a superior technique for bearing fault diagnosis and slightly outperforms other methods (3-5%) in both predictive and noise-ignore capabilities. Comprehensive validation experiments were conducted using the HUST dataset.**

Keywords—Bearing fault; fault diagnosis; domain adaptation; transfer learning

I. Introduction

Bearings are an essential component in rotating machinery. Bearing-related failures account for up to 50% of total machine failures [1]. Precise detection of rolling element bearing faults is crucial for ensuring the reliable operation of rotating machinery. This is because any failure in the bearing could directly impact the functioning of the entire machine. Luckily, modern machine learning techniques have helped make significant progress in data-driven approaches to bearing fault diagnosis [1]. As a result, traditional methods that rely heavily on expert knowledge are no longer required. This has attracted considerable attention and is an area of extensive research [2].

The traditional data-driven approaches for fault diagnosis have been successful in achieving accurate results. However, this is only possible when sufficiently labeled samples or data are available (vibration, acoustic emission, current, etc.). To guarantee high accuracy in fault diagnosis, the testing data must match or is similar in probability distribution to the dataset used for training [3]. This is important because the fault diagnosis models need to be carefully trained and may be difficult to apply directly to different machines or operating conditions.

Transfer learning is a promising tool to overcome the limitations of traditional data-driven approaches for fault diagnosis. It involves transferring knowledge from one task to another, and one commonly utilized technique is domain adaptation [3]. To extract good feature representation across domains, various methods have been proposed, such as the deep adaptive network (DAN) introduced by Long et al. [4] and the hybrid distance-guided adversarial network (HDAN) proposed by Han et al. [5]. However, current domain adaptation methods are limited to diagnosing faults within the same equipment under varying conditions, making it difficult to obtain specific data to train the expected fault diagnosis model. Therefore, transfer fault diagnosis across different machines has become increasingly important. Song et al. [6] introduced a retraining strategy-based domain adaptation network, while Guo et al. [7] utilized a one-dimensional generation adversarial network (ML1-D-GAN) to jointly train generated and real damage data, enabling sufficient labeled data to overcome the limitation of existing models. Recently, Feng et al. [8] presented a domain adversarial similarity-based meta-learning network (DASMN) and Li et al. [9] developed an optimal ensemble deep transfer network (OEDTN) that utilized maximum mean discrepancy (MMD) with different kernels to construct multiple diverse DTNs.

Although transfer fault diagnosis across machines has been achieved with existing methods, there are still weaknesses in terms of fault diagnosis accuracy. The reason is that most of the mentioned methods are based on reducing the discrepancy in feature distribution. They may only adjust the overall distribution since the criteria function only accounts for the statistical parameters of the entire domain and not for each individual class/subdomain. Thus, to overcome this limitation and enhance the transfer fault diagnosis performance across different machines, we introduce a novel multi-layer adaptation network based on LMMD [10] criteria over layers in this article. The methodology is in Section II and experiments are in Section III. The main contributions of this paper are:

- We propose a novel multi-layer subdomain adaptation method that adjusts the domain distribution in each layer of the shared feature extraction module.
- We evaluate the performance of the proposed method on HUST bearing dataset and do comparative experiments to verify its ability to improve fault diagnosis accuracy across different bearings/machines.

*Corresponding Author.

II. Methodology

A. Problem Description

In this section, we start by discussing the issue of bearing fault diagnosis across machines. We begin by assuming that there is a rolling bearing monitoring dataset labeled from one (source) machine $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$, which we call the source domain data. The dataset contains N_s samples with labels y_i^s which belong to the labeled space Y_s. The samples x_i^s belong to the sample space X^s and are governed by the marginal probability distribution $P_s(X^s)$. We also have another rolling bearing monitoring dataset without labels from another (target) machine, called the target domain data $D_t = \{(x_i^t)\}_{i=1}^{N_t}$, which contains N_t samples and all samples are governed by the marginal probability distribution $P_t(X^t)$. Since the two datasets come from different machines, their marginal probability distributions are different, and we have $P_s \neq P_t$.

The main focus of this paper is on the transfer fault diagnosis of rolling bearings across different machines i.e., from the source to the target domain. Traditional data-driven methods rely solely on the source domain data with labels to train a classification neural network, which is a nonlinear mapping between the sample space X^s and the labeled space Y^s [11]. However, directly using the established nonlinear mapping/network to recognize the health status of unlabeled samples from the target domain will yield low fault diagnosis accuracy, as the two domains have different data distributions [12]. Thus, to improve the fault diagnosis accuracy, it is crucial to train the fault diagnosis model using not just labeled data from the source domain but also unlabeled data from the target domain. This presents the challenge of learning domain-invariant features by minimizing the data distribution discrepancy between the source and target domain data. As a result, knowledge obtained from one machine can be used for fault diagnosis in another machine.

B. Proposed Method

In this study, a multi-layer subdomain adaptation model is developed to transfer fault diagnosis between different bearings. As shown in Fig. 1, our goal is to train a feature extractor that can accurately diagnose bearing faults for data in the target domain. The training process involves iteratively calculating the objective functions and updating the model parameters using the backpropagation algorithm. Afterwards, we obtain a trained model that can predict the label of new data in the target domain.

The training process is described as follows: the training data consists of labeled source data (x^s, y^s) and unlabeled target data (x^t). Different domains use the same feature extractor, consisting of three one-dimensional convolutional layers and two fully connected layers (see Table I). The output of the feature extractor is the predicted label for the input data, used to calculate the objective functions based on the training objective. The training objective is to classify faults and minimize the distance of probability distribution between the outputs of different domains. Therefore, there are two objective functions: the objective function for classification (L_{cls}) and the objective function for adaptation (L_{ada}). The classification objective function is calculated based on the true label and predicted one of the data in the source domain. The adaptation objective function is calculated by sum of the distribution discrepancies (LMMD) between hidden features from different domains. This is a new and important aspect of this method, instead of relying solely on the distribution distance of features in the last layer. The important concept of computing the distribution discrepancy will be clarified in Section II C.

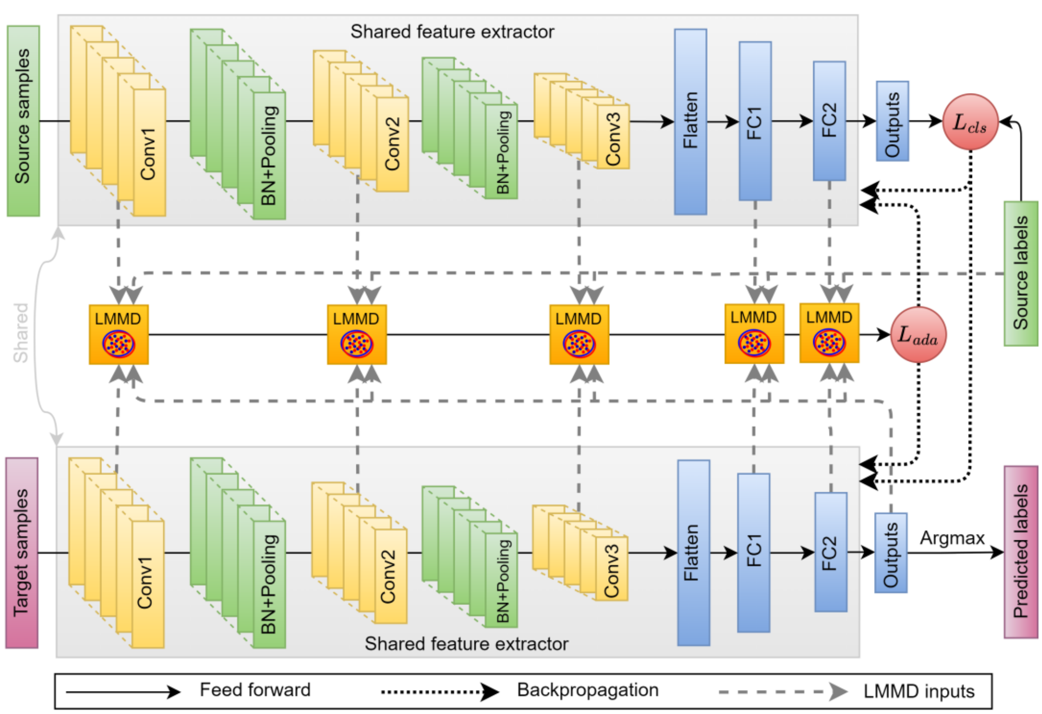

Fig. 1. Overview of the proposed method.

TABLE I. SPECIFICATION OF THE FEATURE EXTRACTOR

No.	Layer	Filter size	Output size	Activation function	BN/Pooling
0	Input	-	(4096, 1)	-	-
1	Conv1	(15, 1, 16)	(4096, 4)	ReLU	BN+Pooling
2	Conv2	(5, 16, 16)	(1024, 8)	ReLU	BN+Pooling
3	Conv3	(3, 32, 16)	(256, 16)	ReLU	-
-	Flatten	-	(4096, 1)	-	-
4	FC1	(4096, 512)	(512, 1)	ReLU	-
5	FC2	(512, 64)	(64, 1)	ReLU	-
6	Output	(64, 4)	(4, 1)	Softmax	-

BN: Batch Normalization

C. Loss Function

The objective of the model is fault classification and domain adaptation based on minimizing the distribution distance. For the classification objective function, it relies on the well-known cross-entropy function:

$$L_{cls} = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) \tag{1}$$

where n is the number of classes, y_i is the ground-truth label, and $\hat{y}_i$ is the softmax probability for the i-th class. As for the adaptation objective function, LMMD is an MMD-based criterion for measuring distribution discrepancy [10]. The MMD formula for calculating the distribution distance is described as follows:

$$MMD(P_s, P_t) \triangleq \left\| \boldsymbol{E}_{P_s}[\Phi(x^s)] - \boldsymbol{E}_{P_t}[\Phi(x^t)] \right\|_H^2 \tag{2}$$

where P_s, P_t denote the distribution of source and target domain; $\boldsymbol{E}$ denotes the expectation; H denotes the reproducing kernel Hilbert space; Φ: $X^{s,l}, X^{t,l} \rightarrow H$. In practice, an estimate of the MMD compares the square distance between the empirical kernel mean embeddings as:

$$\begin{aligned} MMD(P_s, P_t) \approx & \frac{1}{N_s^2} \sum_{i=1}^{N_s} \sum_{j=1}^{N_s} k\left(x_i^s, x_j^s\right) \\ & + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} k\left(x_i^t, x_j^t\right) \end{aligned} \tag{3}$$

where N_S, N_t stand for the number of samples in source and target domain; the kernel k means $k(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle$. The MMD-based techniques primarily emphasized the alignment of overall distributions while disregarding the connections between subdomains within the same category. It is crucial to consider the relationships between these relevant subdomains and align their distributions between the source and target domains. To achieve this, we considered the Local Maximum Mean Discrepancy (LMMD) method in (4). In (4), the letter c denotes the class label. Eq. (4) is then estimated as (5). In (5), $\omega_i^{sc}, \omega_j^{tc}$ stand for the weight of x_i^s, x_j^t belonging to class c, are computed as (6). In (6), y_{ic} is the c-th entry of the output vector y_i with input x_i.

$$LMMD(P_s, P_t) \triangleq \boldsymbol{E}_c \left\| \boldsymbol{E}_{P_s^c}[\Phi(x^s)] - \boldsymbol{E}_{P_t^c}[\Phi(x^t)] \right\|_H^2 \tag{4}$$

$$LMMD \approx \frac{1}{C} \sum_{c=1}^{C} \left\| \sum_{i=1}^{N_s} \omega_i^{sc} \Phi(x_i^s) - \sum_{j=1}^{t} \omega_j^{tc} \Phi\left(x_j^t\right) \right\|_H^2 \tag{5}$$

$$\omega_i^c = \frac{y_{ic}}{\sum_{j=1}^{N} y_{jc}} \tag{6}$$

Afterward, the adaptation loss L_{ada} is computed as (7). It must be noted that for the convolution layer, the feature maps are flatted before calculating the LMMD. It means that layer number 3 and the next flatten layer utilize the same LMMD.

$$L_{ada} = \sum_{i=1}^{5} LMMD \;\; at \; layer \; i \tag{7}$$

Finally, the overall objective function in (8) is the sum of the two aforementioned objective functions. With this, the model can maintain its classification ability while also adjusting the embedding features to a common distribution.

$$L = L_{cls} + L_{ada} \tag{8}$$

III. EXPERIMENTS

A. HUST bearing Dataset

The verification experiments were conducted exclusively on the HUST bearing dataset as in our previous work [1]. What makes this dataset especially advantageous is that it contains fault signals from five bearings across different types of defects and working conditions. The data acquisition system is shown in Fig. 2. The data acquisition system includes a 1-HP induction motor, an accelerometer of PCB352C33 and a measurement module with torque and velocity sensors.

Because of this, we can assess the performance of our proposed method for various domain adaptation tasks across different bearings or machines. For the purpose of test analysis, we selected bearings of types 6205, 6206, and 6207 with a no-load shaft speed. Each type of bearing includes four health conditions: normal (N), inner race fault (I), outer race fault (O), and ball fault (B). The faults were generated using the wire-cutting method, which creates cracks with a size of 0.2 mm. The accelerometer captured the vibration signals at a sampling rate of 51,200 samples per second for 10 seconds. To augment training/test data, the raw vibration signal was truncated into segments with a length of 4096 with 75% overlap. Then we obtain 496 segments/class/bearing with each segment as an input of the model.

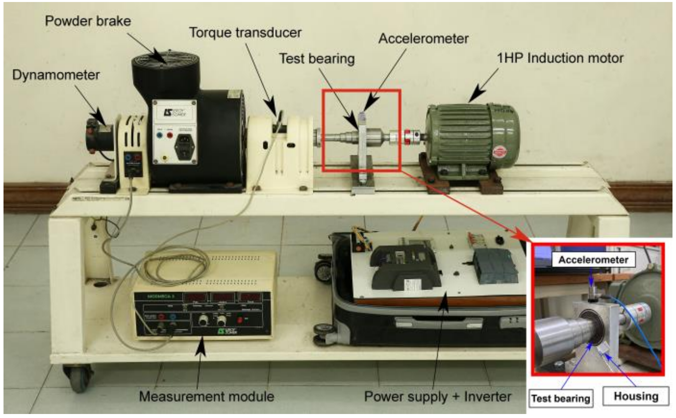

Fig. 2. HUST bearing data acquisition system [1].

The adaptation tasks are defined as S5-6, S5-7, S6-5, S6-7, S7-5, S7-6, M5, M6, and M7. Each task name includes 2 parts: the prefix (S: single-source | M: multiple-source), and the suffix to supplement information for the prefix (5: 6205 | 6: 6206 | 7: 6207). In detail, for single-source tasks, the suffix denotes the source bearing and the target one e.g., 5-6 means that the source bearing is 6205 and the target one is 6206. As for multiple-source tasks, the suffix number denotes the target bearing, while the source is the remaining two bearings. For each task, 80% source data and 80% target data are used to train the network, while the remaining 20% source data is used to validate and 20% target data is used to test/evaluate.

B. Experiments Setup

In this section, we describe the experimental setup. The first experiment is to evaluate the effectiveness of the proposed method in transferring knowledge to adaptive tasks defined in Section III A. The second experiment is an extension of the first experiment with added multi-level noise components in the training/testing data. The purpose of the second experiment is to assess the proposed method's performance in the presence of noise, which is common in real-world scenarios. Finally, the third experiment aims to compare the proposed method's performance with other methods to demonstrate its superiority in adaptability. All models were trained/evaluated on the same HUST bearing dataset to ensure fairness in comparison. The metrics for model evaluation are (overall) accuracy (9), precision (1), sensitivity (11), F1-score (12), confusion matrix, and t-SNE visualization to observe the feature distribution of source and target domains [13].

$$Accuracy = \frac{No.of\ correct\ predictions}{No.of\ all\ samples} \tag{9}$$

$$Precision = \frac{No.of\ true\ positive\ predictions}{No.of\ all\ positive\ predictions} \tag{10}$$

$$Sensitivity = \frac{No.of\ true\ positive\ predictions}{No.of\ all\ positive\ samples} \tag{11}$$

$$F1 - score = 2 \times \frac{Precision \times Sensitivity}{Precision + Sensitivity} \tag{12}$$

The verification experiments were conducted on a computer with the following specifications: an Intel i7 12700F CPU, 16 GB of RAM, and a 24GB Nvidia GeForce GTX 3090 GPU. These experiments were implemented utilizing the PyTorch framework from the source https://github.com/ZhaoZhibin/UDTL [14]. For the training process, the hyperparameters were configured as follows: the number of epochs was set to 100, the batch size at 64, the learning rate to 0.001, the momentum at 0.9, the optimizer was Adam, and the weight decay was 0.00001. To provide a reliable measure of accuracy, each model was re-trained 10 times, and the results were reported as mean and standard deviation values.

IV. Results and Discussions

In the first experiment, we evaluated the performance of the proposed method across all tasks. Table II shows the overall accuracy, precision, sensitivity, and F1-score of the model with the test data in the target domain. It can be seen that the overall accuracy of the single-source tasks is quite good, ranging from 86% to 96%. Specifically, task S6-5 had the lowest accuracy of 86.62%, while task S6-7 had the highest accuracy of 96.46%. Furthermore, the inversely related tasks (e.g. S5-6 and S6-5) had similar accuracy. This reveals that the result of transfer learning depends on the relationship between domains and not on which domain is the source and which is the target.

Regarding other metrics for single-source tasks, we can see that precision, sensitivity, and F1-score are relatively consistent for each task. The magnitude of these metrics varies by no more than 1%. This is achieved by the class balance in the test data set.

TABLE II. Transfer Fault Diagnosis Results with the Proposed Method

Task	Overall accuracy (%)	Precision (%)	Sensitivity (%)	F1-score (%)
S5-6	88.38	89.43	88.38	88.90
S5-7	92.93	93.78	92.93	93.35
S6-5	86.62	87.91	86.62	87.26
S6-7	96.46	96.60	96.46	96.53
S7-5	89.14	90.27	89.14	89.70
S7-6	94.44	94.64	94.44	94.54
M5	92.42	92.75	92.42	92.59
M6	95.71	95.87	95.71	95.79
M7	97.47	97.54	97.47	97.51

In Table II, for multi-source tasks, it is easy to see that their performance is significantly improved compared to the single-source tasks. Task M5 achieved 92.42% accuracy, higher than Sx-5 tasks due to more contribution from the source domain. Tasks M6 and M7 both scored above 95% on all criteria. This superiority is achieved through an increase in source data, which can compensate for each other's deficiencies to help the model learn more effectively. From this, we can conclude that enhancing the source domain data will improve the model's diagnostic capabilities. Additionally, we can observe a trend where the accuracy of multi-source tasks depends on the shared target domain of the single-source tasks. This means that the Mx predictive ability will be a function of the Ma-x abilities.

Fig. 3 illustrates the confusion matrices corresponding to the considered tasks. Unlike overall accuracy, the confusion matrix provides a clearer explanation of the accuracy for each class, with the value in each cell being the number of instances. At a glance, we can see that the highest accuracy is concentrated on classes I and O in all tasks. We believe this is due to the clear defect patterns in these two classes, making them easier to identify for the model. In the dataset, defects related to class B are difficult to predict accurately and are often confused with faults in class I. This phenomenon is seen from 6205-related tasks i.e., the fault characteristic of class I and B of bearing 6205 may be hard to distinguish (e.g., the fault frequency). For the case of the healthy bearing, tasks M6 and M7 achieve almost perfect accuracy, while tasks M5, S6-5, S7-5, S5-6, and S5-7 show worse accuracy. It can be speculated that there are issues with the N data for bearing 6205 (e.g., a small crack may exist). However, this is not as serious as misclassifying failures as non-failures.

Fig. 4 visualizes the distribution of features in the final layer of the neural network in the Descartes coordinate system using a visualization method called t-distributed stochastic neighbor embedding (t-SNE), which is a dimensionality reduction algorithm. We observe that the mispredictions in the classes occur due to the mismatch between the source and target class distributions of the data. To address this issue, some studies have proposed labeling some of the training data in the target domain, which can be further explored in [15].

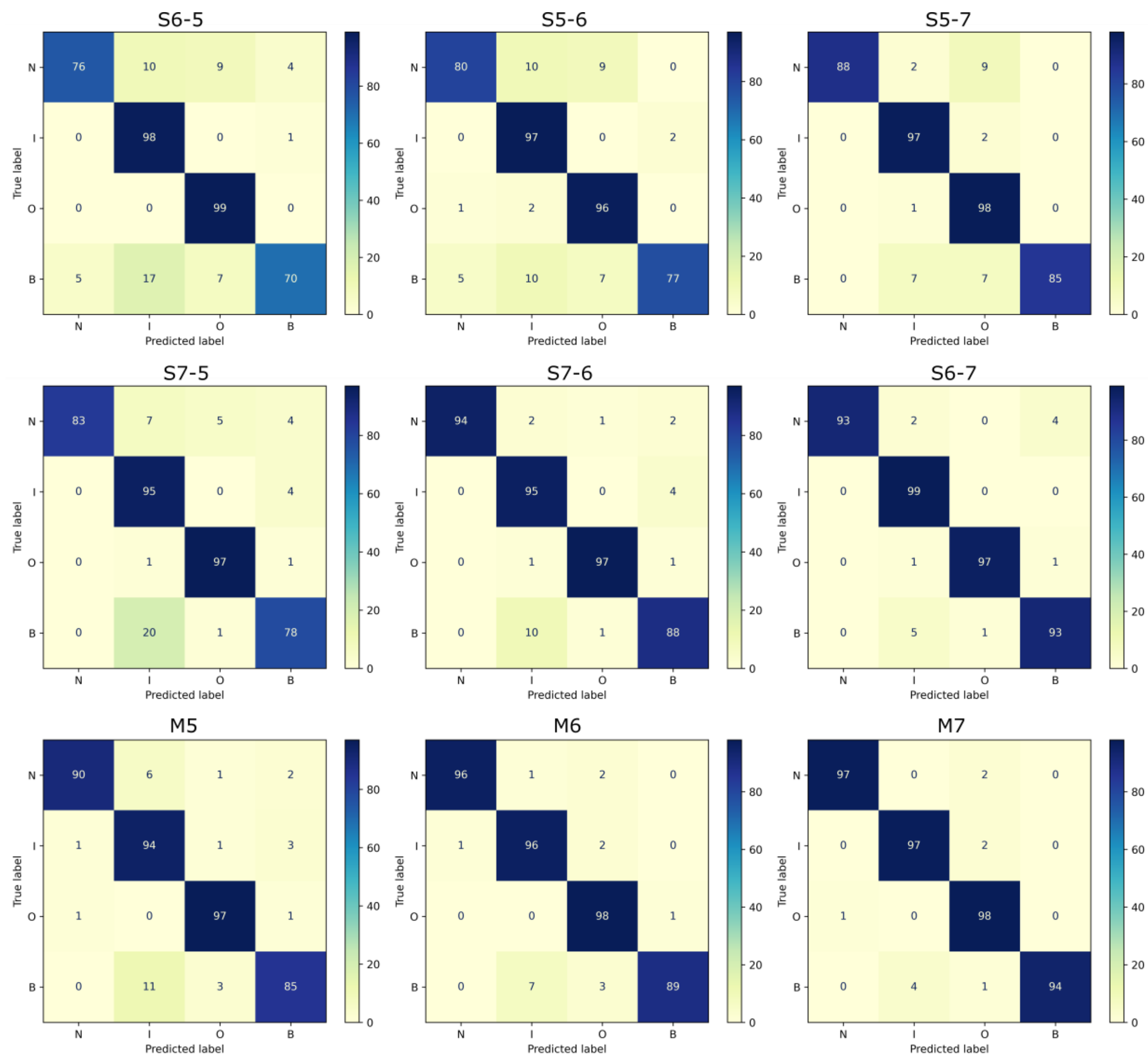

Fig. 3. Confusion matrices for all tasks.

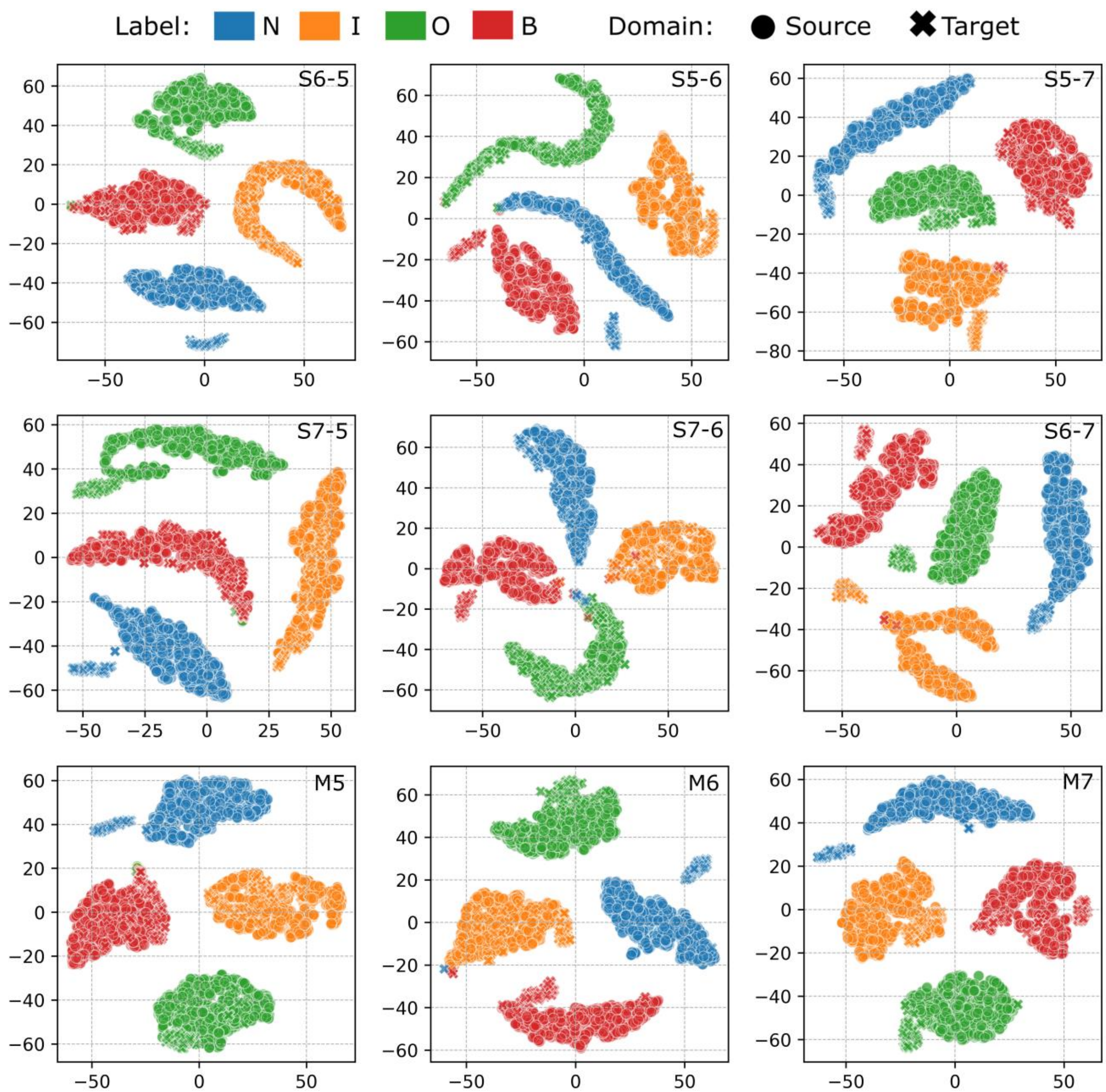

Fig. 4. t-SNE visualization of the last features in all tasks.

In the second experiment, we examined the performance of our proposed method in the presence of noise in the data. We added Gaussian white noise with a mean of 0 and a standard deviation of 1 to the training and test data. To vary the intensity of the noise, we scaled the amplitude by different coefficients ranging from 0 to 0.9 with a step of 0.3. In the third experiment, we compared our proposed method with other popular domain adaptation methods, including JMMD [16], MKMMD [17], CORAL [18], DANN [19], and CDAN [20], in the presence of noise. The effect of noise was also included to provide a comprehensive comparison. The results of the experiments demonstrated the effectiveness of our proposed method in the presence of noise. Our method outperformed the other popular domain adaptation methods, particularly as the intensity of the noise increased. The results are presented in Fig. 5.

In Fig. 5, as the level of noise increases, all examined methods experience a decrease in performance. Multi-source tasks show a decrease from 92-97% to 89-93%, while single-source tasks experience a decrease from 86-96% to 81-92%. Notably, while the other methods experience a significant accuracy drop of up to 10%, our proposed method only experiences a slight decrease of around 5% for all tasks. This indicates that our method is less affected by noise, promising stability and high reliability. Regarding the correlation between the methods, it is truly difficult to distinguish because they differ slightly in their predictive capabilities.

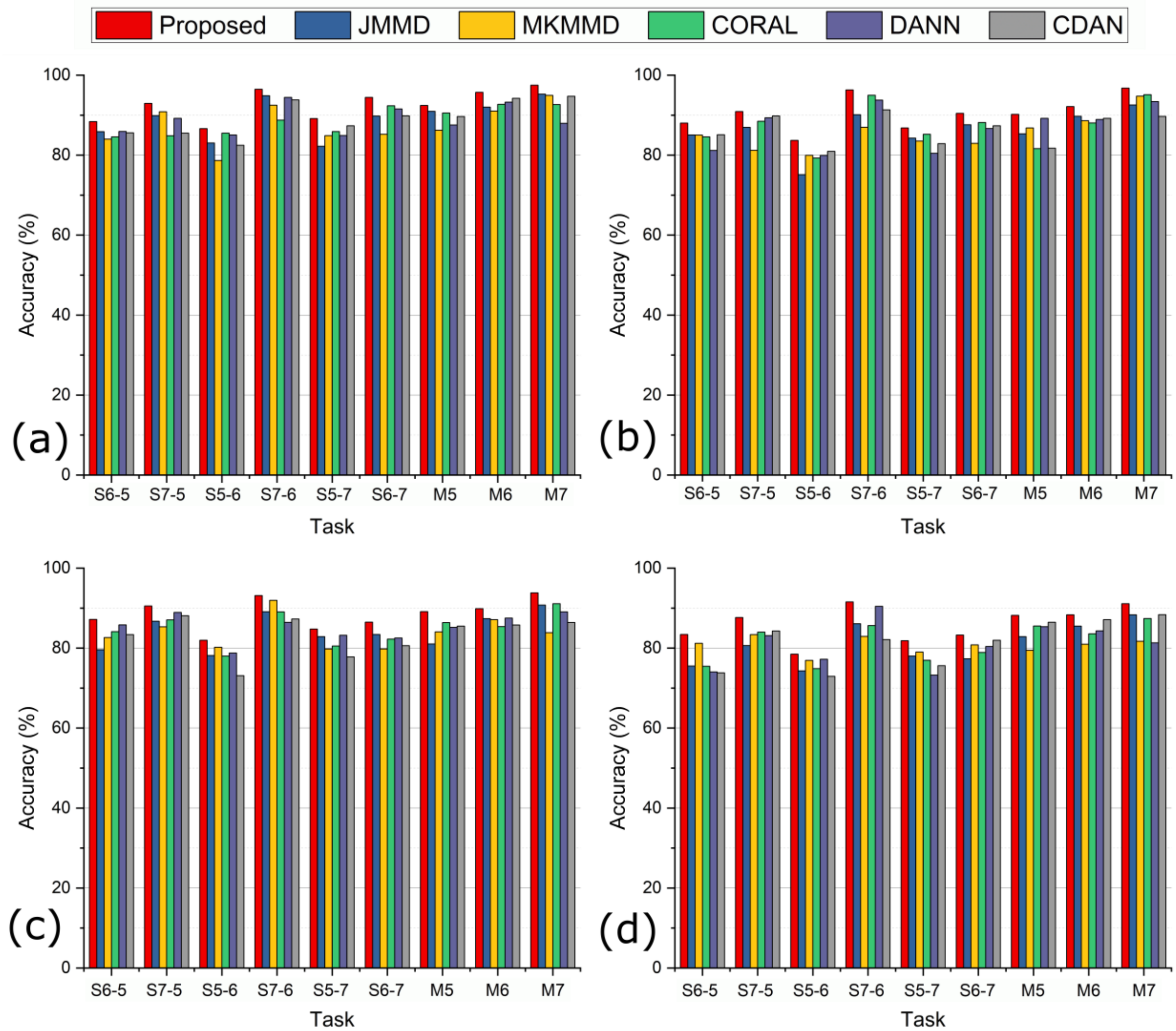

Fig. 5. The performance of different methods under different levels of noise. Noise levels are: (a) 0; (b) 0.3; (c) 0.6; (d) 0.9.

Nonetheless, we still recognize the effectiveness of our proposed method because, in all cases, it slightly outperforms the other methods. Our success can be attributed to taking into account the varying distributions between classes (subdomains), which sets us apart from other approaches. Furthermore, our approach is reinforced by a thorough examination of all hidden layers, an aspect that has been overlooked by many previous studies. Going forward, we aim to explore algorithms that improve adaptability in scenarios where distributions cannot be homogenized.

V. Conclusion

This study introduces a new method based on transfer for fault diagnosis in bearings across various machines, named weighted multi-layer subdomain adaptation. Due to the weakness of traditional metrics as MMD for feature alignment between different domains, we inspired by LMMD to develop a new model architecture for the task of domain adaptation. This method is validated using HUST bearing dataset for nine transfer fault diagnosis tasks where labeling of target domain data is not required. Verification experiments were conducted, and the findings indicate that the proposed approach offers relatively high accuracy up to 97.47% and excellent transferability. Comparative experiments revealed that the proposed method is a superior technique for bearing fault diagnosis and slightly outperforms other methods (3-5%) in both predictive and noise-ignore capabilities.

Acknowledgment

This research is funded by Hanoi University of Science and Technology (HUST) under project number: T2022-PC-006.

References

[1] N. D. Thuan and H. S. Hong, “HUST bearing: a practical dataset for ball bearing fault diagnosis,” Feb. 2023, doi: 10.48550/arXiv.2302.12533.

[2] N. Duc Thuan, N. Thi Hue, P. Quang Vuong, and H. Si Hong, “Intelligent Bearing Fault Diagnosis With a Lightweight Neural Network,” in 2022 11th International Conference on Control, Automation and Information Sciences (ICCAIS), Nov. 2022, pp. 261–266. doi: 10.1109/ICCAIS56082.2022.9990211.

[3] B. Yang, Y. Lei, F. Jia, and S. Xing, “An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings,” Mech. Syst. Signal Process., vol. 122, pp. 692–706, May 2019, doi: 10.1016/J.YMSSP.2018.12.051.

[4] M. Long, Y. Cao, J. Wang, and M. I. Jordan, “Learning Transferable Features with Deep Adaptation Networks,” Feb. 2015, doi: doi.org/10.5555/3045118.3045130.

[5] B. Han, X. Zhang, J. Wang, Z. An, S. Jia, and G. Zhang, “Hybrid distance-guided adversarial network for intelligent fault diagnosis under different working conditions,” Measurement, vol. 176, p. 109197, May 2021, doi: 10.1016/j.measurement.2021.109197.

[6] Y. Song, Y. Li, L. Jia, and M. Qiu, “Retraining Strategy-Based Domain Adaption Network for Intelligent Fault Diagnosis,” IEEE Trans. Ind. Informatics, vol. 16, no. 9, pp. 6163–6171, Sep. 2020, doi: 10.1109/TII.2019.2950667.

[7] Q. Guo, Y. Li, Y. Song, D. Wang, and W. Chen, “Intelligent Fault Diagnosis Method Based on Full 1-D Convolutional Generative Adversarial Network,” IEEE Trans. Ind. Informatics, vol. 16, no. 3, pp. 2044–2053, Mar. 2020, doi: 10.1109/TII.2019.2934901.

[8] Y. Feng et al., “Similarity-based meta-learning network with adversarial domain adaptation for cross-domain fault identification,” Knowledge-Based Syst., vol. 217, p. 106829, Apr. 2021, doi: 10.1016/J.KNOSYS.2021.106829.

[9] X. Li, S. Huo, and B. Xi, “Updating the resolution for 16S rRNA OTUs clustering reveals the cryptic cyanobacterial genus and species,” Ecol. Indic., vol. 117, p. 106695, Oct. 2020, doi: 10.1016/J.ECOLIND.2020.106695.

[10] Y. Zhu et al., “Deep Subdomain Adaptation Network for Image Classification,” IEEE Trans. Neural Networks Learn. Syst., vol. 32, no. 4, pp. 1713–1722, Apr. 2021, doi: 10.1109/TNNLS.2020.2988928.

[11] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, “Applications of machine learning to machine fault diagnosis: A review and roadmap,” Mech. Syst. Signal Process., vol. 138, p. 106587, Apr. 2020, doi: 10.1016/j.ymssp.2019.106587.

[12] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, “Maximum Classifier Discrepancy for Unsupervised Domain Adaptation,” Dec. 2017, doi: arXiv:1712.02560.

[13] H. M and S. M.N, “A Review on Evaluation Metrics for Data Classification Evaluations,” Int. J. Data Min. Knowl. Manag. Process, vol. 5, no. 2, pp. 01–11, Mar. 2015, doi: 10.5121/ijdkp.2015.5201.

[14] Z. Zhao et al., “Applications of Unsupervised Deep Transfer Learning to Intelligent Fault Diagnosis: A Survey and Comparative Study,” IEEE Trans. Instrum. Meas., vol. 70, pp. 1–28, 2021, doi: 10.1109/TIM.2021.3116309.

[15] K. Yu, Q. Fu, H. Ma, T. R. Lin, and X. Li, “Simulation data driven weakly supervised adversarial domain adaptation approach for intelligent cross-machine fault diagnosis,” Struct. Heal. Monit., vol. 20, no. 4, pp. 2182–2198, Jul. 2021, doi: 10.1177/1475921720980718.

[16] M. Long, H. Zhu, J. Wang, and M. I. Jordan, “Deep Transfer Learning with Joint Adaptation Networks,” in Proceedings of the 34th International Conference on Machine Learning, 2017, vol. 70, pp. 2208–2217. [Online]. Available: https://proceedings.mlr.press/v70/long17a.html.

[17] A. Gretton et al., “Optimal kernel choice for large-scale two-sample tests,” in Advances in Neural Information Processing Systems, 2012, vol. 25. [Online]. Available: https://proceedings.neurips.cc/paper/2012/file/dbe272bab69f8e13f14b405e038deb64-Paper.pdf.

[18] B. Sun and K. Saenko, “Deep CORAL: Correlation Alignment for Deep Domain Adaptation,” 2016, pp. 443–450. doi: 10.1007/978-3-319-49409-8_35.

[19] Y. Ganin et al., “Domain-Adversarial Training of Neural Networks,” May 2015, [Online]. Available: http://arxiv.org/abs/1505.07818.

[20] M. Long, Z. Cao, J. Wang, and M. I. Jordan, “Conditional Adversarial Domain Adaptation,” May 2017, [Online]. Available: http://arxiv.org/abs/1705.10667.