Engineered Science
DOI: https://dx.doi.org/10.30919/es1793

Transnosis: Transformer-Variation-based Bearing Compound Fault Diagnosis with Zero-shot Learning and Domain Adaptation

Qi Liu,[1,2,*,#] Wenjing Liu,[1,#] Chao Peng,[1] Yuming Fan,[2,3] Biao Wang,[4] Hang Zhang,[5,*] Ergude Bao[1,*] and Jiqiang Liu[6,*]

Abstract

Bearing compound fault diagnosis is an important problem, and signal data from the target bearing are collected and inputted into deep learning model for diagnosis. One challenge of this problem is, to initially train the deep learning model, it is difficult or impossible to obtain data of compound faults, and the training data are usually single fault data. Another challenge is, the training data are usually obtained in working conditions different from the target bearing. To solve the bearing compound fault diagnosis problem addressing the two challenges, we introduce Transnosis, a Transformer-variation-based bearing compound fault diagnosis method with zero-shot learning and domain adaptation. Transnosis is a combination of three solid and time-tested Transformer variations: Cross-domain Transformer (CDTrans), Query2Label and Shifted Patch Tokenization/Locality Self-Attention (SPT/LSA) with dedicated designs. In the design of Transnosis, we propose weight-sharing triple-attention decoders to incorporate domain adaptation in decoders of combined CDTrans and Query2Label, multidimensional embedding approach to make signal data enhancement motivated by the SPT technique, and also splitting-for-voting approach for further optimization. We test Transnosis with BJTU-RAO datasets on six tasks, and experimental results demonstrate Transnosis can accurately diagnose compound faults by effectively distinguishing features of different faults and clustering features of the same fault from different working conditions.

Keywords: Compound fault diagnosis; Transformer variation; Zero-shot learning; Domain adaptation.

Received: 29 May 2025; Revised: 16 September 2025; Accepted:12 October 2025.
Article type: Research article.

1. Introduction

Bearings are important components in various mechanical systems, facilitating rotational movement of axles, motors, turbines and so on. Fault of bearings may lead to unexpected downtime, costly repairs or even accidents of the whole system. Consequently, fault diagnosis when a fault occurs is critical for maintaining the system integrity. On the other hand, deep learning has been demonstrated highly accurate and thus widely used in various fields,[1,2] so to achieve effective bearing fault diagnosis, bearing's vibration, acceleration, sound or temperature signals are collected from sensors, and deep learning methods are applied on the signal data.

Traditional fault diagnosis using deep learning usually deals with single fault identification; however, in practice, bearings can experience compound faults. The compound fault is multiple single faults occurring in one bearing, so the signals are obscured with overlapping patterns, complicating the diagnosis process.[3-9] (i) A challenge to diagnosing the compound faults is insufficient training data. In industrial practice, it is usually not easy to collect large amounts of signal data for bearings due to cost, time or equipment limitations. As a result, the collected training data are usually composed of single faults instead of compound faults. (ii) Another challenge is target bearing's unknown working condition. Bearings may operate under various conditions, including changes in load, speed and temperature. These variations can significantly impact the obtained sensor signals, so that deep learning models trained with signal data from source bearing

[1] *School of Software Engineering, Beijing Jiaotong University, No. 3 Shangyuan Residence, Haidian District, Beijing, 100044, China*
[2] *CRRC Academy, 9th Floor Building 5 Nord Center II, Fengtai Science and Technology Park, Fengtai District, Beijing, 100070, China*
[3] *School of Automation and Intelligence, Beijing Jiaotong University, No. 3 Shangyuan Residence, Haidian District, Beijing, 100044, China*
[4] *Collaborative Innovation Center of Railway Traffic Safety, Beijing Jiaotong University, No. 3 Shangyuan Residence, Haidian District, Beijing, 100044, China*

may not generate well for an arbitrary target bearing.

In recent three years, to address challenge (i), a few zero-shot learning models have been studied trained with single fault data for compound fault diagnosis. Xu *et al*. propose a zero-shot fault semantics learning model which constructs semantic vectors from single fault signals, and calculates Euclidean distance between the constructed semantic vectors and signal features to diagnose compound faults.[10,11] Based on this, they optimize the semantic vectors combining manually constructed and learned semantics, and also optimize the extracted signal features with adaptive edge center loss.[12] They also further optimize the semantic vectors, extracted signal features and classification module with a fresh fault semantic constructing approach, deep residual contraction network and adaptive smoothing approach, respectively.[13] Tang *et al*. propose to detect fault feature regions and then use deep belief network with discrimination terms to diagnose compound faults.[14] Gao *et al*. propose to use adaptive multi-strategy cuckoo search algorithm to optimize important parameters of maximum correlated kurtosis deconvolution (MCKD), and then use the MCKD and convolutional neural network (CNN) to denoise signal data and diagnose compound faults, respectively.[15] Li *et al*. propose a wavelet capsule network with back tracking technique and compound fault decoupling to provide interpretable fault diagnosis results.[16] Hu *et al*. propose an evidential neural network with novel evidence prediction function, penalty terms and Fourier transform based data augmentation to diagnose compound faults and also additional unknown faults.[17] To address challenge (ii), though many transfer learning models have been studied for single fault diagnosis,[18–25] there have been not so many such models for compound fault diagnosis. In addition, though some models are free of compound fault data during training,[26,27] others do need the data and thus do not achieve zero-shot learning.[28,29] Huang *et al*. propose a transferable capsule network to decouple compound fault for diagnosis, and they use domain adversarial learning for domain adaptation.[26,27] Zhang *et al*. propose primary and prototype correction auxiliary classifiers with Frobenius norm of cross correlation matrices to diagnose compound faults, and they use adaptive weighting of sample features for domain adaptation.[28] Wang *et al*. propose a domain reinforcement learning feature adaptation model with correlation alignment to diagnose compound faults, and they also use domain adversarial learning for domain adaptation.[29]

On the other hand, the Transformer model was introduced in 2017 for natural language processing.[30] Because of its innovative architecture especially the self-attention mechanism, Transformer has revolutionized the artificial intelligence field, being improved and tailored in not only natural language processing, but also computer vision, spatiotemporal data mining, signal processing, *etc*. Until recently, many variations of Transformer have been proposed, and the most famous ones include DeepSeek,[31] Generative Pre-trained Transformer (GPT),[32] Bidirectional Encoder Representations from Transformer (BERT),[33] Contrastive Language-Image Pre-Training (CLIP),[34] and Vision Transformer (ViT).[35] Although not directly designed to solve the bearing compound fault diagnosis problem or any of the two challenges, some of the variations aim at highly related problems/challenges and their solutions are valuable for reference. Query2Label is a Transformer variation to solve multi-label image classification problem (269 cites by May 2025).[36] It can be trained with single-label images for the multi-label classification, so could be utilized for bearing compound fault diagnosis trained with single fault data, and thus to address challenge (i). Cross-domain Transformer (CDTrans) is a Transformer variation to address source-target image domain adaptation problem (309 cites by May 2025).[37] It could be customized to achieve domain adaptation for bearing signal data, and thus to address challenge (ii). In addition, Shifted Patch Tokenization (SPT) and Locality Self-Attention (LSA) are techniques in Transformer to alleviate training with lack-of-feature image data problem (294 cites by May 2025).[38] They could be referenced for bearing signal data enhancement, and thus to further address challenge (i). Nevertheless, to customize and combine these Transformer variations for bearing compound fault diagnosis, a few issues exist and proper designs are required. (a) Further domain adaptation is needed to accommodate additional label embeddings inputted into the combined Query2Label and CDTrans. To combine the Query2Label with CDTrans, the CDTrans will be inputted with both training and compound fault data to encode them for domain adaptation, and the Query2Label will be inputted with label embeddings and decode both CDTrans' encodings and the embeddings for classification. Due to the additional label embeddings inputted, further domain adaptation is needed in Query2Label. (b) If the SPT technique is directly applied, the signal data will be

[5] *Institute of Engineering Thermophysics, Chinese Academy of Sciences, No. 11 Beisihuanxi Road, Haidian District, Beijing, 100190, China*

[6] *School of Cyberspace Science and Technology, Beijing Jiaotong University, No. 3 Shangyuan Residence, Haidian District, Beijing, 100044, China*

**Email:* lq@crrc.tech (Q. Liu); zhanghang@iet.cn (H. Zhang); baoe@bjtu.edu.cn (E. Bao); jqliu@bjtu.edu.cn (J. Liu)

[#] These authors contributed equally to this work.

simply split into more pieces with a sliding window. This is an ordinary signal processing approach not effective to achieve data enhancement.

In this paper, we focus on the two challenges (i)-(ii) about bearing compound fault diagnosis, solve the two issues (a)-(b) to utilize a few solid and time-tested Transformer variations, and propose Transnosis: a Transformer-variation-based compound fault diagnosis method with zero-shot learning and domain adaptation. To combine the Query2Label and CDTrans in Transnosis, we propose weight-sharing triple-attention decoders in each decoding layer. Motivated by the SPT technique, we propose a signal multidimensional embedding approach to generate more signal features. In addition, for further optimization, we propose a signal splitting-for-voting approach to split raw target domain signal data into a few segments and vote for classification. With these approaches, the modified Query2Label in Transnosis can be trained with single fault data to diagnose compound fault, the CDTrans can achieve domain adaptation between the training data and compound fault data, and the customized SPT/LSA can achieve data enhancement in different levels.

Table 1 summarizes the correspondence between the challenges, Transformer variations, issues to use the variations and approaches in Transnosis as discussed above, and Fig. 1 presents techniques used in Transnosis. In summary, we have the following contributions.

(1) We design Transnosis from a few solid and time-tested Transformer variations for bearing compound fault diagnosis. It can be trained with single fault data from source domain to diagnose compound faults from target domain.

(2) We propose weight-sharing triple-attention decoders, signal multidimensional embedding approach and splitting-for-voting approach to utilize the existing Transformer variations.

(3) Experimental results demonstrate Transnosis is effective and can outperform the current existing methods.

2. Method

2.1 Overview

In training phase, the Transnosis architecture has two inputs: source domain signal data and target domain signal data, and has three outputs: classified fault types from source domain signals, from target domain signals and from both.

(1) The signal data are inputted into a multi-dimensional embedding layer to generate embeddings with enhancement. This layer is based on the multidimensional embedding approach.

(2) For encoding, the source and target domain embedding data are inputted into several layers each containing three weight-sharing Transformer encoders to generate source, target and source-target domain encodings, respectively. These encoders use LSA or cross-attention for domain adaptation.

(3) For decoding, initial label embeddings each corresponding to a fault type are inputted into several layers each containing three Transformer decoders for source, target and source-target domain decodings, respectively. These decoders are the weight-sharing triple-attention decoders. They use LSA between labels, cross-attention between labels and signal encodings for classification, and additional LSA or cross-attention for further domain adaptation.

In classification phase, Transnosis is inputted with target domain signal data and it outputs the classified fault types. Only the Transformer encoders and decoders for target domain signal data are used in this process.

To achieve effective domain adaptation and obtain accurate classification result, the raw target domain signal data are split into segments for both training and classification.

Table 1: Correspondence between challenges, Transformer variations, issues to use the variations, and approaches in Transnosis.

Challenges	Transformer variations	Issues to use the variations	Approaches
Unknown working condition	CDTrans	How to incorporate domain adaptation in decoders of combined CDTrans and Query2Label	Weight-sharing triple-attention decoders
	Query2Label		
Insufficient training data	SPT/LSA	How to make signal data enhancement	Multidimensional embedding approach
Additional optimization			Splitting-for-voting approach

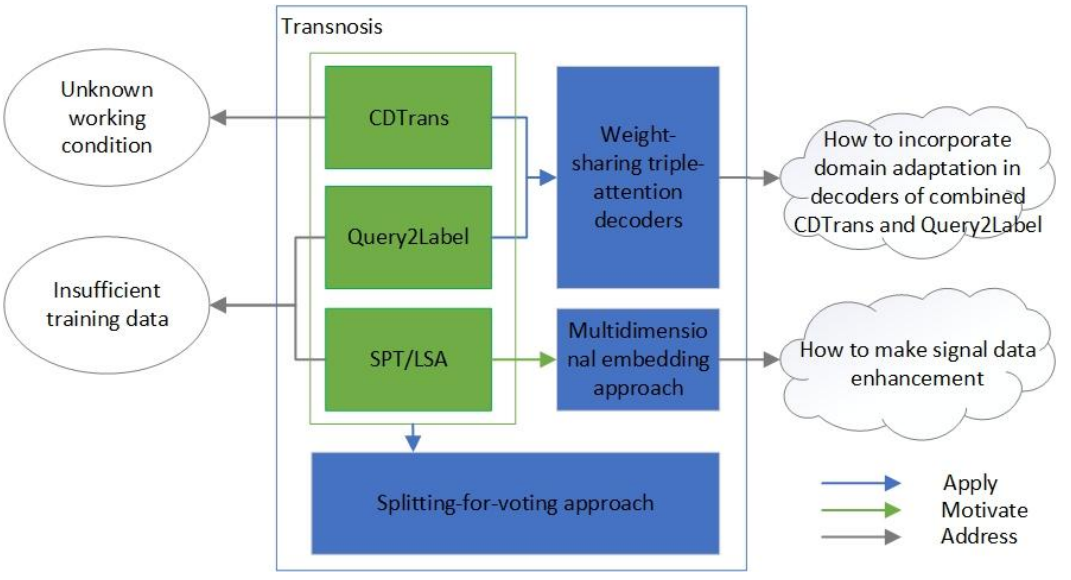

Fig. 1: Techniques used in Transnosis.

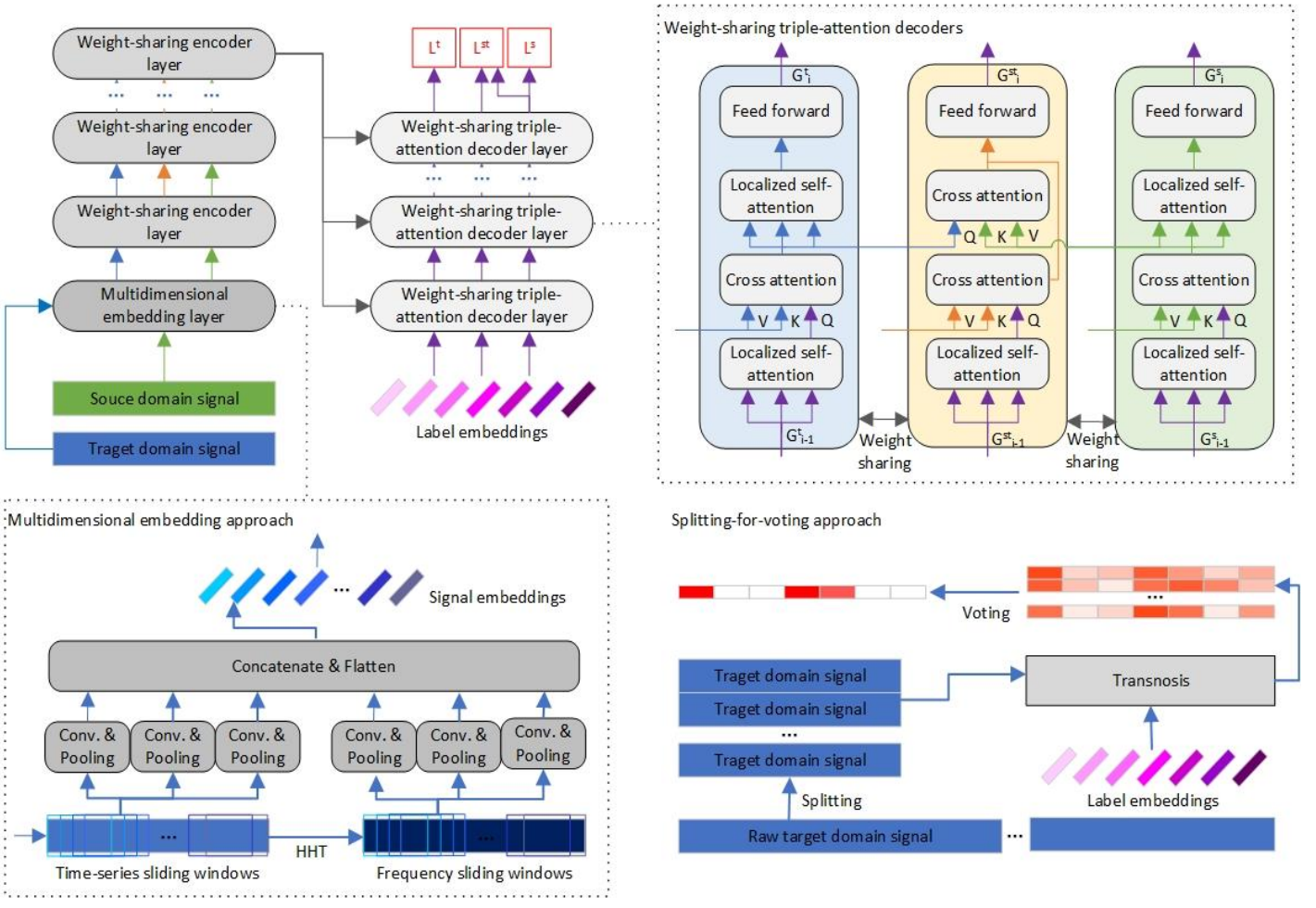

Fig. 2: Illustration of the Transnosis architecture.

Classification results from the segments are jointly considered using a voting strategy to obtain final fault type. This is the splitting-for-voting approach. The Transnosis architecture is illustrated in Fig. 2. Detailed explanations of data flow in these steps are explained in sections below.

2.2 Data enhancement with multidimensional embedding approach

The inputted signal data $X=\{X_i|1\leq i\leq n\}$ contains n sampling points, and each point X_i has d^s dimensions such as tri-axial acceleration. The X is processed with a sliding window of size

w and step length p to generate $n'=\frac{(n-w)}{p}$ tokens $T^s=\{T_i^s|1\le i\le n'\}$. Each token T_i^s is essentially a $w\times d^s$ matrix, and is processed with a large scale Convolutional Neural Network (CNN) to generate an embedding L_i^s using Eq. (1).

$$L_i^s = maxPool(cov(T_i^s, k_l), w_l) \quad (1)$$

where k_l is a relatively large convolutional kernel size and w_l is a relatively large pooling window size. Additionally, T_i^s is processed with a moderate scale CNN to generate embedding M_i^s by replacing the k_l and w_l with moderate kernel size k_m and window size w_m, respectively. T_i^s is also processed with a small scale CNN to generate embedding S_i^s with small kernel size k_s and window size w_s, respectively.

Besides, the X is processed with Hilbert-Huang Transform (HHT) to generate frequency data $Y=\{Y_i|1\le i\le n\}$ with Y_i of d^f dimensions. The Y is processed with the same sliding window as above to generate tokens $T^f=\{T_i^f|1\le i\le n'\}$. Each token T_i^f is also processed with the large, moderate and small scale CNNs to generate embeddings L_i^f, M_i^f and S_i^f. Finally, the embeddings L_i^s, M_i^s, S_i^s, L_i^f, M_i^f and S_i^f are concatenated using Eq. (2) to form the ith embedding E_i for encoding.

$$E_i = flatten([L_i^s, M_i^s, S_i^s, L_i^f, M_i^f, S_i^f]) \quad (2)$$

Therefore, the inputted signal data X is converted to embedding data $E=\{E_i|1\le i\le n'\}$. With this embedding, pseudo label for target domain data can be generated as discussed in Supporting information.

2.3 Cross-domain encoding for adaptation

For encoding of the embedding data, there are N layers and in layer i there are three weight-sharing transformer branches H_i^s, H_i^t and H_i^{st} ($1\le i\le N$). The H_i^s includes a LSA module for source domain embedding data E^s; the H_i^t also includes a LSA module for target domain embedding data E^t; the H_i^{st} includes a cross-attention module for both E^s and E^t.

More specifically, input of the H_i^s is output of the previous layer E_{i-1}^s, and output is E_i^s; input of the H_i^t is output of the previous layer E_{i-1}^t, and output is E_i^t; input of the H_i^{st} is output of the previous layer E_i^{st}, as well as E_{i-1}^s and E_{i-1}^t, and output is E_i^{st}. When i=1, $E_0^s=E^s$; $E_0^t=E^t$; E_0^{st} is randomly initiated. The LSA module in the H_i^s processes E_{i-1}^s as the query, key and value with FFN module using Eqs. (3) and (4).

$$E'^s_i = multiHeadLSA(E_{i-1}^s, E_{i-1}^s, E_{i-1}^s) \quad (3)$$

$$E_i^s = FFN(E'^s_i) \quad (4)$$

The LSA module in the H_i^t is identical to the H_i^s. The cross-attention module in the H_i^{st} processes E_{i-1}^s as the query and E_{i-1}^t as the key and value with FFN module using Eqs. (5) and (6).

$$E'^{st}_i = multiHeadCross(E_{i-1}^s, E_{i-1}^t, E_{i-1}^t) \quad (5)$$

$$E_i^{st} = FFN(E'^{st}_i) \quad (6)$$

The encoding data from the last layer are thus E_N^s, E_N^t and E_N^{st}. Note that for simplicity of explanation, the addition and normalization modules following the LSA module and FFN module are not presented.

2.4 Data-label joint decoding with weight-sharing triple-attention decoders

For decoding of the final encoding data, similar to the encoding layers, there are M layers and in layer i there are also three weight-sharing transformer branches G_i^s, G_i^t and G_i^{st} ($1\le i\le M$). Differently, the G_i^s includes a LSA module for label embedding data E^l, a cross-attention module for the E^l and E_N^s, and a second LSA module for output of the cross-attention module; the G_i^t includes a LSA module for the E^l, a cross-attention module for the E^l and E_N^t, and a second LSA module for output of the cross-attention module; the G_i^{st} includes a LSA module for the E^l, a cross-attention module for the E^l and E_N^{st}, and a second cross-attention module for outputs of the cross-attention modules from G_i^s and G_i^t.

More specifically, inputs of the G_i^s are output of the previous layer D_{i-1}^s and the E_N^s, and output is D_i^s; inputs of the G_i^t are output of the previous layer D_{i-1}^t and the E_N^t, and output is D_i^t; inputs of the G_i^{st} are output of the previous layer D_{i-1}^{st} and the E_N^t, as well as outputs of G_i^s and G_i^t's cross-attention modules D''^s_i and D''^t_i, respectively, output is D_i^{st}. When i=1, $D_0^s=D_0^t=D_0^{st}=E^l=\{E_k^l|1\le k\le m\}$, where E_j^l is randomly initiated embedding for the kth class label. The LSA module in the G_i^s processes D_{i-1}^s as the query, key and value using Eq. (7).

$$D'^s_i = multiHeadLSA(D_{i-1}^s, D_{i-1}^s, D_{i-1}^s) \quad (7)$$

Then the cross-attention module processes D'^s_i as the query and E_N^s as key and value using Eq. (8).

$$D''^s_i = multiHeadCross(D'^s_i, E_N^s, E_N^s) \quad (8)$$

Then the second LSA module processes D''^s_i as the query, key and value with FFN module using Eqs. (9) and (10).

$$D'''^s_i = multiHeadLSA(D''^s_i, D''^s_i, D''^s_i) \quad (9)$$

$$D_i^s = FFN(D'''^s_i) \quad (10)$$

The LSA module, cross-attention module and the second LSA module in G_i^t are identical to those in G_i^s. The LSA module and cross-attention module in G_i^{st} are identical to those in G_i^s, and the second cross-attention module processes the D''^s_i as the query and D''^t_i as key and value with FFN module using Eqs. (11) and (12).

$$D'''^{st}_i = MultiHeadCross(D''^s_i, D''^t_i, D''^t_i) \quad (11)$$

$$D_i^{st} = FFN(D'''^{st}_i + D''^{st}_i) \quad (12)$$

The decoding data from the last layers are thus D_N^s, D_N^t and D_N^{st}. With the decoding data, probabilities of source domain data fitting the m label classes $p^s=\{p_k^s|1\le k\le m\}$ can be calculated using Eq. (13).

$$p^s = sigmoid(WD_N^s + b) \quad (13)$$

The probabilities of target domain data and source and target domain data can be calculated using identical equations. Loss functions of the three Transformer branches are cross-entropy loss, cross-entropy loss and distillation loss, respectively.

2.5 Training and classification with splitting-for-voting approach

In the beginning of Transnosis, the inputted raw target domain signal data X are split into g segments $X=\{X^i|1\le i\le g\}$ for both training and classification. For the i th segment, its classification result is $p^i=\{p_k^i|1\le k\le m\}$. Combing the g segments' results, voting result of the k th label class is obtained using Eq. (14).

$$p_k = \frac{\sum_i p_k^i}{g} > h \quad (14)$$

where h is a voting threshold value.

With this splitting-for-voting approach, training of Transnosis includes three steps. First, source domain signal data with labels are used to train the multidimensional embedding layer. Second, the target domain signal data are inputted to the trained multidimensional embedding layer to generate pseudo labels. Third, the source domain signal data with labels and target domain signal data with pseudo labels are used to train the Transnosis model. Steps two and three iterate until convergence. The Transnosis training process is presented in Fig. 3.

3. Experiments

3.1 Dataset description

To test Transnosis, we use datasets of left axle box component from the BJTU-RAO data packet.[39] The complete data packet is composed of datasets from four components in a subway train bogie simulation platform: transaction motor, gearbox, left axle box and right axle box. Bearing type of the left axle box is HRB 352213, and monitoring signals include tri-axial acceleration and sound, and are collected with a sampling frequency 64kHz. Fault types of the left axle box include normal condition (NC), inner race fault (IF), outer race fault

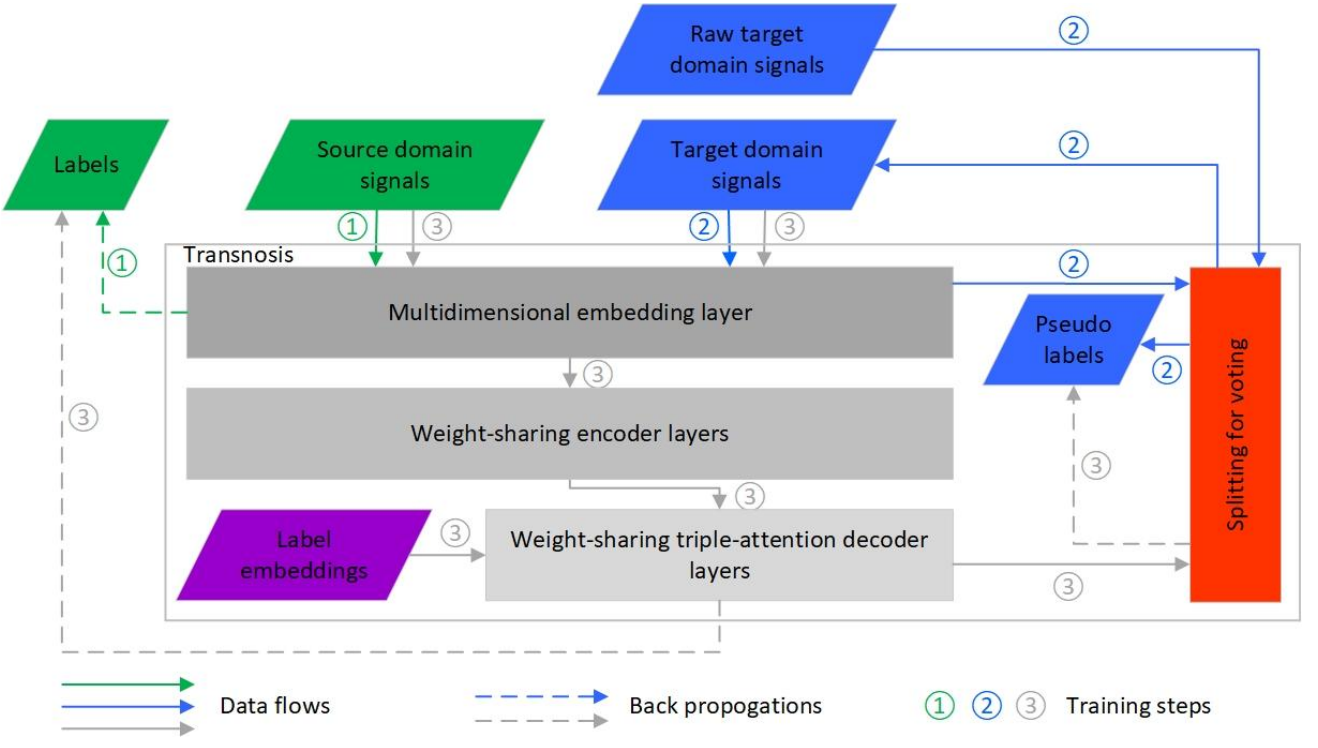

Fig. 3: Flowchart of Transnosis training.

(OF), and a combination of IF and OF forming compound fault (CF). Working conditions of the left axle box as well as the whole platform vary in motor speed from 20Hz to 60Hz and lateral load from -10kN to 10kN. A picture of the platform is presented in Fig. S1.[39]

We obtain four datasets A, B, C and D in working conditions motor speed/lateral load 20Hz/0kN, 60Hz/0kN, 20Hz/10kN and 60Hz/10kN, respectively. Each dataset contains samples of the NC, IF, OF and CF. We use the NC, IF and OF samples in one dataset for training and then all the NC, IF, OF and CF samples in another dataset for testing, so we have six tasks: training using dataset A and testing using B (A→B), training using A and testing using C (A→C), training using B and testing using C (B→C), training using A and testing using D (A→D), training using B and testing using D (B→D), training using C and testing using D (C→D). Details of the datasets for training and testing are listed in Table 2, signals of each fault in dataset B are presented in Fig. 4, and tasks in the experiments are listed in Table 3.

Table 2: Details of the datasets for training and testing.

Dataset	Working condition	Fault type for training	Fault type for testing	Number of samples
A	20Hz/0kN	NC, IF, OF	-	999
B	60Hz/0kN	NC, IF, OF	NC, IF, OF, CF	999
C	20Hz/10kN	NC, IF, OF	NC, IF, OF, CF	999
D	60Hz/10kN	NC, IF, OF	NC, IF, OF, CF	999

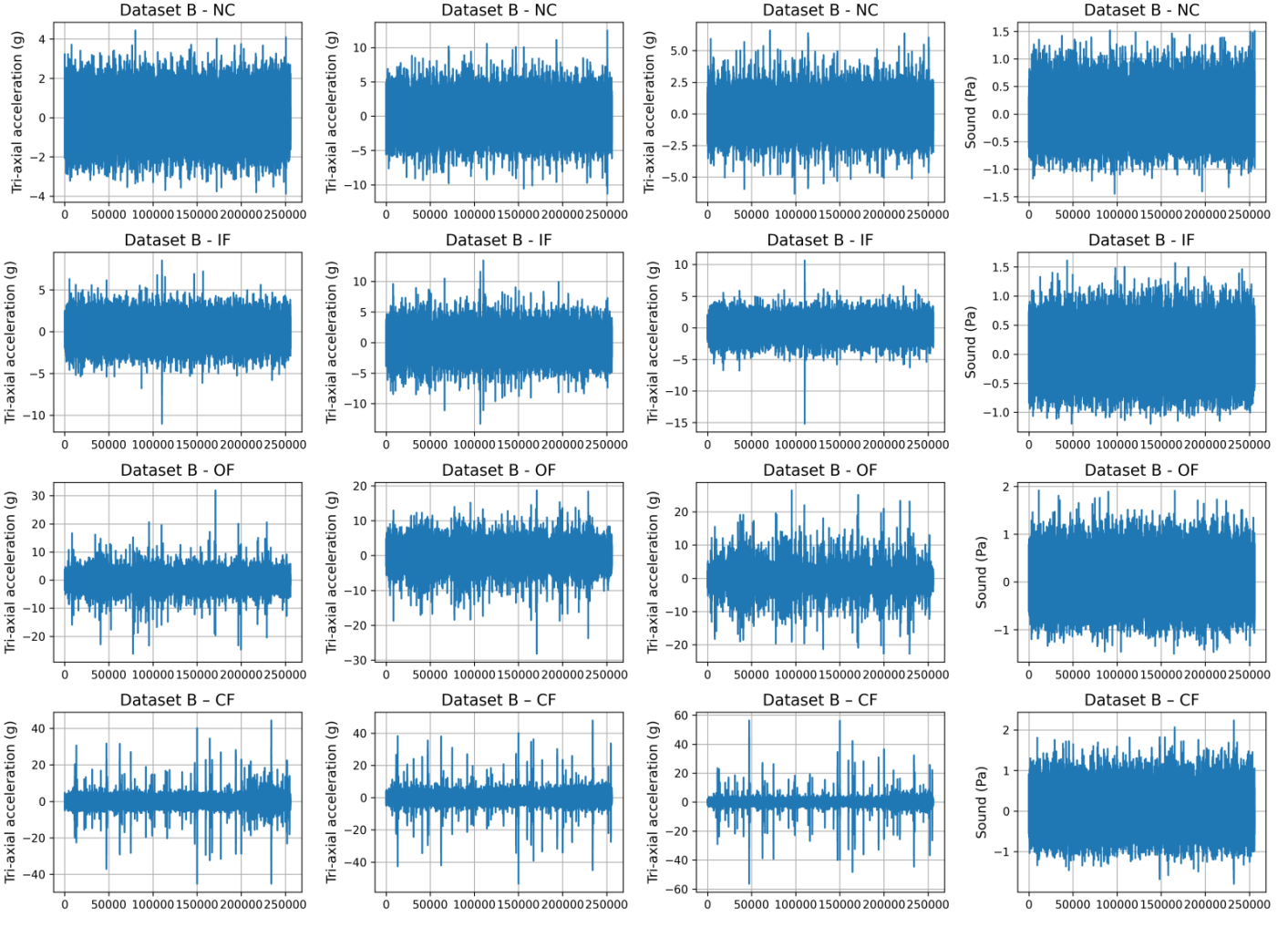

Fig. 4: Signals of each fault in dataset B.

Table 3: Task in the experiments.

Task	Source domain	Target domain
A→B	A	B
A→C	A	C
B→C	B	C
A→D	A	D
B→D	B	D
C→D	C	D

3.2 Hyperparameters of Transnosis

Based on the method description in section 3, Transnosis' hyperparameters are determined by preliminary testing and the Transformer related publications. In the preprocessing phase, the sliding window size w and step length p are 128 and 64, respectively. The large, moderate and small convolutional kernel sizes k_l , k_m and k_s in CCN are 7, 5 and 3, respectively, and the corresponding pooling window sizes w_l , w_m and w_s are 4, 3 and 2, respectively. The dimension of frequency data d^f is 60. In the en/decoding and classification phase, the numbers of Transformer encoder layers N and decoder layers M are both 2. The number of segments g for raw target domain signal data is 4, and the voting threshold value h is 0.65. Transnosis is implemented using Pytorch and trained using Adam optimizer with a learning rate of 0.00001 and 50 epochs. The hyperparameters and their values are listed in Table 4.

Table 4: Hyperparameters and values in Transnosis.

Hyperparameter	Value
w	128
p	64
k_l	7
k_m	5
k_s	3
w_l	4
w_m	3
w_s	2
d^f	60
N	2
M	2
g	4
h	0.65

3.3 Compound fault diagnosis results on various tasks

We run Transnosis on the tasks described above, and for each task, we shuffle the samples in testing dataset and input them into Transnosis for diagnosis. Table 5 and Table S1 list the diagnosis accuracy of each fault. It can be seen that Transnosis exhibits high accuracy for all the fault types. Compared with NC, IF and OF, the diagnosis accuracy of CF is a little lower, since no CF sample is used for training.

In addition, Fig. 5 presents probabilities fitting each fault for all testing samples. 0-25%, 25-50%, 50-75% and 75-100% of the samples have fault types NC, IF, OF and CF, respectively. It can be seen that for the NC, IF and OF samples, Transnosis classifies them as NC, IF and OF with high probabilities above the threshold value, respectively. For the CF samples, Transnosis classifies them as both IF and OF with high probabilities above the threshold value, and thus CF. Only a few CF samples are misclassified with the probabilities of either IF or OF drops below the threshold value.

3.4 Comparison with existing methods

3.4.1 Compared methods

As introduced in section 1, there are limited methods with both zero-shot learning and domain adaptation. We compare Transnosis with a few representative methods deep decoupling convolutional neural network (DDCNN),[40] wavelet capsule network (WavCapsNet),[16] transferable capsule network (TCN),[26] and domain adversarial capsule network (DACN).[27]

Table 5: Diagnosis accuracies of each fault on various tasks.

Task	NC	IF	OF	CF
A→B	100.00%	91.97%	99.60%	93.17%
A→C	100/00%	91.16%	100.00%	85.54%
A→D	100.00%	96.79%	100.00%	94.78%
B→C	99.20%	98.80%	99.20%	98.39%
B→D	100.00%	100.00%	100.00%	96.39%
C→D	94.38%	99.60%	97.99%	97.59%

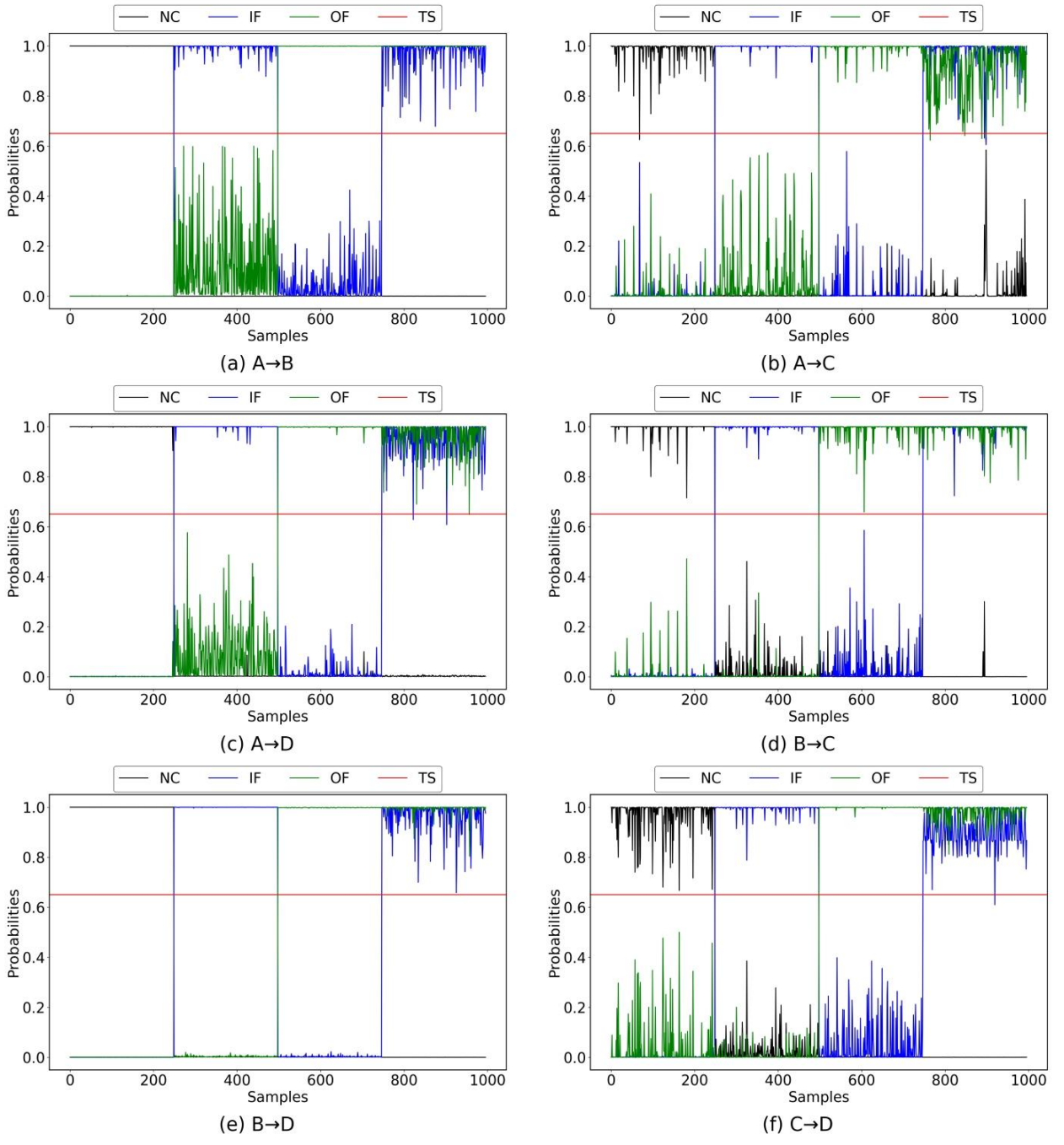

Fig. 5: Probabilities fitting each fault for all samples on tasks A→B (a), A→C (b), B→C (c), A→D (d), B→D (e) and C→D (f). TS represents the threshold value *h*.

Table 6: Characteristics of compared methods and parameter settings.

Method	Zero-shot learning	Domain adaptation	Reference
DDCNN	Yes	No	[40]
WavCapsNet	Yes	No	[16]
TCN	Yes	Yes, single domain	[26]
DACNs	Yes	Yes, single domain	[27]
DACNm	Yes	Yes, multi-domains	[27]

Table 7: Diagnosis accuracies of various methods on each task.

Method	A→B	A→C	A→D	B→C	B→D	C→D
DDCNN	86.43%	83.45%	90.89%	88.56%	86.92%	87.13%
WavCapsNet	87.05%	85.74%	91.87%	88.86%	87.25%	88.35%
TCN	89.71%	86.83%	92.94%	90.78%	89.68%	88.98%
DACNs	93.08%	91.85%	94.88%	93.67%	91.85%	92.21%
DACNm	94.89%	93.49%	96.89%	95.67%	95.45%	95.74%
Transnosis	96.18%	94.18%	97.89%	98.90%	99.10%	97.39%

All of the methods can be trained with single fault data and used for compound fault diagnosis, and the major difference is, DDCNN and WavCapsNet do no domain adaptation, TCN does single domain adaptation from source to target domain, and DACN does multiple domain adaptation among several source domains not needing source-to-target domain adaptation. We let DACN do multiple domain adaptation among the source domain and remaining two domains in each task, marking it as DACNm. For example, on task A→B, we let DACNm do multiple domain adaptation among datasets A, C and D. We also let DACN do single domain adaptation from source to target domain, marking it as DACNs. Table 6 lists characteristics of these methods and parameter settings.

3.4.2 Results

Table 7 lists diagnosis accuracies of the various methods on each task, and Fig. 6 is a corresponding radar diagram of the diagnosis accuracies. It can be seen that Transnosis is consistently more accurate than all the compared methods. DDCNN and WavCapsNet do not do domain adaptation, so their accuracies are lower than the other existing methods. TCN and DACNs do single domain adaptation, but DACNs is designed with more sophisticated techniques, so its accuracy is higher than TCN. DACNm does multiple domain adaptation using additional source domain data, so its accuracy is higher than the other existing methods. Transnosis does single domain adaptation without using additional data, but due to its effective Transformer basis, it can achieve a little higher accuracy than DACNm. Overall, these results indicate that Transnosis can achieve high fault diagnosis accuracy.

3.4.3 Visualization of feature distributions

We use t-SNE to visualize the feature distributions of the various methods on task B→D. Fig. 7 presents the feature distributions. For target domain features, Transnosis can largely distinguish them, while DDCNN, WavCapsNet, TCN,

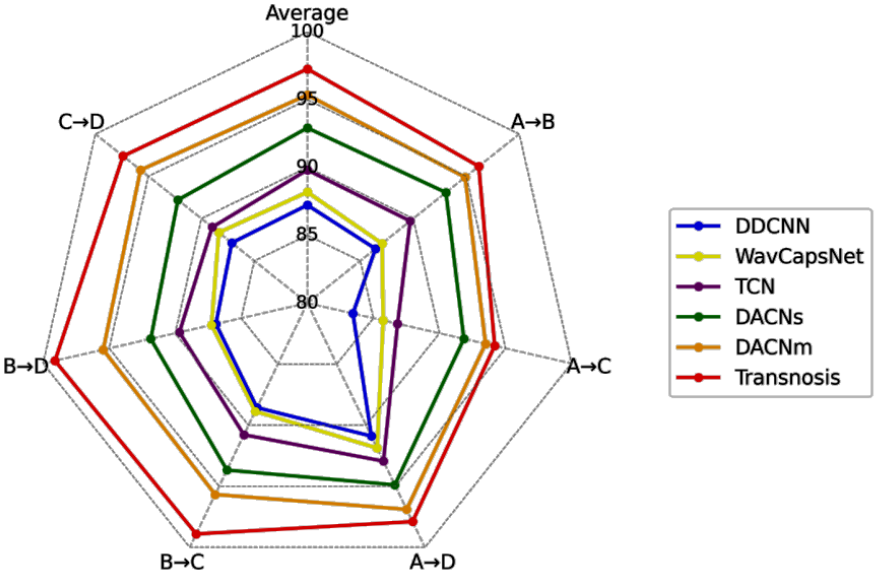

Fig. 6: Radar diagram of the diagnosis accuracies.

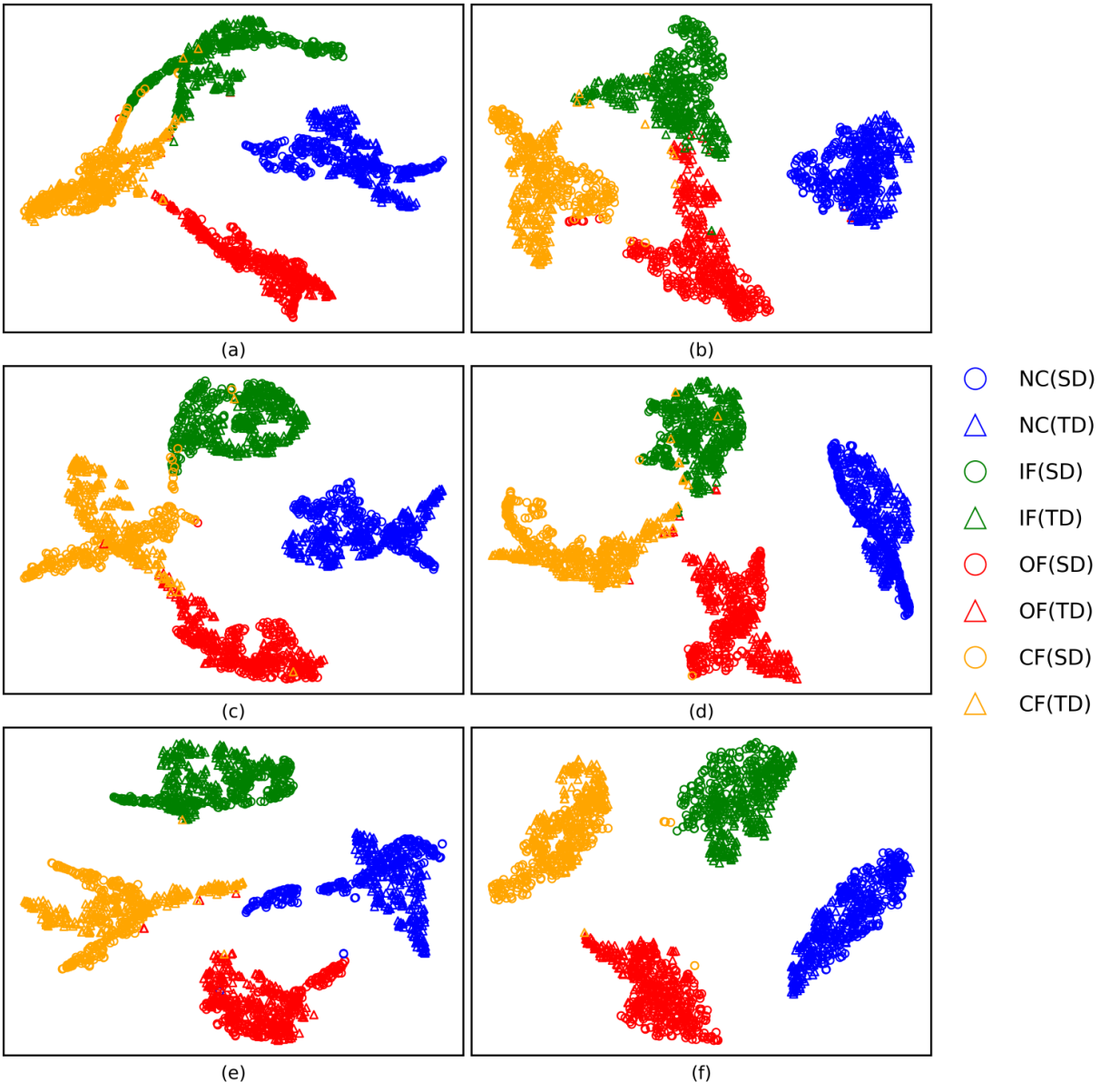

Fig. 7: Visualization of feature distributions on task B→D. (a) DDCNN, (b) WavCapsNet, (c) TCN, (d) DACNs, (e) DACNm, and (f) Transnosis.

DACNs and DACNm are not as effective as Transnosis. This is consistent with the results listed in Table 7 and Table S2. For source domain features, all the methods can effectively distinguish them, and WavCapsNet, DACNs, DACNm and Transnosis perform better than DDCNN and TCN. On the other hand, for features of the same fault from both domains, TCN, DACNs, DACNm and Transnosis can more effectively cluster them than DDCNN and WavCapsNet, since the latter do not do domain adaptation. This visualization further explains why Transnosis can achieve high fault diagnosis accuracy.

4. Conclusions

In this work, we introduce Transnosis, a combination of three solid and time-tested Transformer variations Query2Label, CDTrans and SPT/LSA with dedicated designs for bearing compound fault diagnosis. Transnosis is able to be trained with single fault data from source domain to accurately diagnose compound faults in target domain, achieving both zero-shot learning and domain adaptation. In the design of Transnosis, we propose weight-sharing triple-attention decoders to incorporate domain adaptation in decoders of combined CDTrans and Query2Label, multidimensional embedding approach to make signal data enhancement motivated by the SPT technique, and also splitting-for-voting approach for further optimization. We test Transnosis with BJTU-RAO datasets on six tasks, and experimental results demonstrate Transnosis can accurately diagnose compound faults by

effectively distinguishing features of different compound faults and clustering features of the same fault from different working conditions.

For industrial applications, computational cost and scalability of Transnosis are both concerns. The computational cost of Transnosis depends on domain adaptation in the CDTrans and weight-sharing triple-attention decoders, and it can usually be finished in a few minutes. This latency should be acceptable in most industrial applications, since unlike fault detection or remaining useful life prediction, the time to finish diagnosing an already occurred fault is usually quite relaxed. In addition, when the types of compound faults increase, Transnosis may not accurately diagnose all of them. Therefore, Transnosis is currently applicable to industrial applications with relatively simple compound faults. In the further, we will extend Transnosis to diagnose more complicated multiple compound faults, and will also investigate the possibility of real-time diagnosis. Besides these, we also plan to incorporate multiple domain adaptation in Transnosis.[27]

Acknowledgements

We acknowledge financial support from CRRC Academy.

Conflict of Interest

There is no conflict of interest.

Supporting Information

Applicable.

CRediT Statement

Qi Liu: Methodology, Writing, Supervision; **Wenjing Liu:** Software, Testing, Reviewing; **Chao Peng:** Software; **Yuming Fan:** Testing; **Biao Wang:** Reviewing; **Hang Zhang:** Methodology, Supervision; **Ergude Bao:** Methodology, Writing; **Jiqiang Liu:** Writing, Supervision.

References

[1] D. G. Takale, A. V. Dhumane, P. N. Mahalle, T. Jadhav, P. P. Gawali, A. Buchade, Optimize deep learning model for intensive care of neurological disorders patients based on facial expression, *Engineered Science*, 2024, **32**, 1364, doi: 10.30919/es1364.

[2] T. Pradeep, D. R. Kumar, N. Kumar, W. Wipulanusat, S. Keawsawasvong, J. Sunkpho, Performance evaluation and triangle diagram of deep learning models for embedment depth prediction in cantilever sheet piles, *Engineered Science*, 2024, **28**, 1082, doi: 10.30919/es1082.

[3] R. Huang, J. Xia, B. Zhang, Z. Chen, W. Li, Compound fault diagnosis for rotating machinery: state-of-the-art, challenges, and opportunities, *Journal of Dynamics, Monitoring and Diagnostics*, 2023, **2**, 13-29: 10.37965/jdmd.2023.152.

[4] Y. Jin, C. Qin, Z. Zhang, J. Tao, C. Liu, A multi-scale convolutional neural network for bearing compound fault diagnosis under various noise conditions, *Science China Technological Sciences*, 2022, **65**, 2551-2563, doi: 10.1007/s11431-022-2109-4.

[5] R. Wang, H. Jiang, K. Zhu, Y. Wang, C. Liu, A deep feature enhanced reinforcement learning method for rolling bearing fault diagnosis, *Advanced Engineering Informatics*, 2022, **54**, 101750, doi: 10.1016/j.aei.2022.101750.

[6] P. V. Shinde, R. G. Desavale, Application of dimension analysis and soft competitive tool to predict compound faults present in rotor-bearing systems, *Measurement*, 2022, **193**, 110984, doi: 10.1016/j.measurement.2022.110984.

[7] H. Pu, S. Teng, D. Xiao, L. Xu, Y. Qin, J. Luo, Compound fault diagnosis of rotating machine through label correlation modeling *via* graph convolutional neural network, *IEEE Transactions on Instrumentation and Measurement*, 2023, **73**, 3503110, doi: 10.1109/TIM.2023.3338680.

[8] W. Fan, C. Yang, C. Chen, C. He, Y. Yuan, Y. Li, Adaptive feature-oriented dictionary learning and sparse classification framework for bearing compound fault diagnosis, *IEEE Transactions on Instrumentation and Measurement*, 2024, **73**, 3518010, doi: 10.1109/TIM.2024.3383498.

[9] S. Yin, Z. Chen, Research on compound fault diagnosis of bearings using an improved DRSN-GRU dual-channel model, *IEEE Sensors Journal*, 2024, **24**, 35304-35311, doi: 10.1109/JSEN.2024.3462540.

[10] J. Xu, L. Zhou, W. Zhao, Y. Fan, X. Ding, X. Yuan, Zero-shot learning for compound fault diagnosis of bearings, *Expert Systems with Applications*, 2022, **190**, 116197, doi: 10.1016/j.eswa.2021.116197.

[11] J. Xu, S. Liang, X. Ding, R. Yan, A zero-shot fault semantics learning model for compound fault diagnosis, *Expert Systems with Applications*, 2023, **221**, 119642, doi: 10.1016/j.eswa.2023.119642.

[12] J. Xu, H. Zhang, L. Zhou, Y. Fan, Zero-shot compound fault diagnosis method based on semantic learning and discriminative features, *IEEE Transactions on Instrumentation and Measurement*, 2023, **72**, 3518313, doi: 10.1109/TIM.2023.3280503.

[13] J. Xu, H. Zhang, W. Chen, Y. Fan, X. Ding, CGASNet: a generalized zero-shot learning compound fault diagnosis approach for bearings, *IEEE Transactions on Instrumentation and Measurement*, 2024, **73**, 2513111, doi: 10.1109/TIM.2024.3373062.

[14] J. Tang, J. Wu, B. Hu, J. Liu, An intelligent diagnosis method using fault feature regions for untrained compound faults of rolling bearings, *Measurement*, 2022, **204**, 112100, doi:

10.1016/j.measurement.2022.112100.
[15] S. Gao, S. Shi, Y. Zhang, Rolling bearing compound fault diagnosis based on parameter optimization MCKD and convolutional neural network, *IEEE Transactions on Instrumentation and Measurement*, 2022, **71**, 3508108, doi: 10.1109/TIM.2022.3158379.
[16] W. Li, H. Lan, J. Chen, K. Feng, R. Huang, WavCapsNet: an interpretable intelligent compound fault diagnosis method by backward tracking, *IEEE Transactions on Instrumentation and Measurement*, 2023, **72**, 3519811, doi: 10.1109/TIM.2023.3282664.
[17] M. Hu, C. Luo, C. Wang, Z. Qiang, Compound fault recognition and diagnosis of rolling bearing in open-set-recognition setting, *Measurement*, 2025, **242**, 116132, doi: 10.1016/j.measurement.2024.116132.
[18] X. Pan, H. Chen, W. Wang, X. Su, Adversarial domain adaptation based on contrastive learning for bearings fault diagnosis, *Simulation Modelling Practice and Theory*, 2025, **139**, 103058, doi: 10.1016/j.simpat.2024.103058.
[19] Z. Du, D. Liu, L. Cui, Dynamic model-driven dictionary learning-inspired domain adaptation strategy for cross-domain bearing fault diagnosis, *Reliability Engineering & System Safety*, 2025, **258**, 110905, doi: 10.1016/j.ress.2025.110905.
[20] Q. Chang, C. Fang, W. Zhou, X. Meng, A multi-order moment matching-based unsupervised domain adaptation with application to cross-working condition fault diagnosis of rolling bearings, *Structural Health Monitoring*, 2025, **24**, 1438-1455, doi: 10.1177/14759217241262386.
[21] F. Jiang, Y. Kuang, T. Li, S. Zhang, Z. Wu, K. Feng, W. Li, Towards Enhanced Interpretability: a Mechanism-Driven domain adaptation model for bearing fault diagnosis across operating conditions, *Mechanical Systems and Signal Processing*, 2025, **225**, 112244, doi: 10.1016/j.ymssp.2024.112244.
[22] D. Liu, L. Cui, G. Wang, W. Cheng, Interpretable domain adaptation transformer: a transfer learning method for fault diagnosis of rotating machinery, *Structural Health Monitoring*, 2025, **24**, 1187-1200, doi: 10.1177/14759217241249656.
[23] L. Xiao, Y. Chen, M. Wang, H. Zhao, Q. Zhou, Multisource–multitarget partial domain adaptation for bearing fault diagnosis, *IEEE Internet of Things Journal*, 2025, **12**, 15897-15910, doi: 10.1109/JIOT.2025.3529953.
[24] X. Li, Z. Kou, C. Han, S. Huang, Deep clustering domain adaptation for fault diagnosis of rolling bearings in mining belt conveyors, *Measurement*, 2025, **248**, 116878, doi: 10.1016/j.measurement.2025.116878.
[25] N. Jia, W. Huang, C. Ding, J. Wang, Z. Zhu, Physics-informed unsupervised domain adaptation framework for cross-machine bearing fault diagnosis, *Advanced Engineering Informatics*, 2024, **62**, 102774, doi: 10.1016/j.aei.2024.102774.
[26] R. Huang, Z. Wang, J. Li, J. Chen, W. Li, A transferable capsule network for decoupling compound fault of machinery, *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, May 25-28, Dubrovnik, Croatia, IEEE, 2020, 1-6, doi: 10.1109/I2MTC43012.2020.9129078.
[27] R. Huang, J. Li, Y. Liao, J. Chen, Z. Wang, W. Li, Deep adversarial capsule network for compound fault diagnosis of machinery toward multidomain generalization task, *IEEE Transactions on Instrumentation and Measurement*, 2020, **70**, 3506311, doi: 10.1109/TIM.2020.3042300.
[28] X. Zhang, J. Wang, Z. Zhang, B. Han, H. Bao, X. Jiang, Integrated decision-making with adaptive feature weighting adversarial network for multi-target domain compound fault diagnosis of machinery, *Advanced Engineering Informatics*, 2024, **62**, 102730, doi: 10.1016/j.aei.2024.102730.
[29] Z. Wang, J. Xuan, T. Shi, Domain reinforcement feature adaptation methodology with correlation alignment for compound fault diagnosis of rolling bearing, *Expert Systems with Applications*, 2025, **262**, 125594, doi: 10.1016/j.eswa.2024.125594.
[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, *Advances in Neural Information Processing Systems*, 2017, **30**, 1-15, doi: 10.5555/3295222.3295349.
[31] Z. Deng, W. Ma, Q.-L. Han, W. Zhou, X. Zhu, S. Wen, Y. Xiang, Exploring DeepSeek: a survey on advances, applications, challenges and future directions, *IEEE/CAA Journal of Automatica Sinica*, 2025, **12**, 872-893, doi: 10.1109/JAS.2025.125498.
[32] G. Yenduri, M. Ramalingam, G. C. Selvi, Y. Supriya, G. Srivastava, P. K. R. Maddikunta, G. D. Raj, R. H. Jhaveri, B. Prabadevi, W. Wang, A. V. Vasilakos, T. R. Gadekallu, GPT (generative pre-trained transformer): a comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions, *IEEE Access*, 2024, **12**, 54608-54649, doi: 10.1109/ACCESS.2024.3389497.
[33] J. Devlin, M. Chang, K. Lee, and K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2019, **1**, 4171-4186, doi: 10.18653/v1/N19-1423.
[34] Y. Li, F. Liang, L. Zhao, Y. Cui, W. Ouyang, J. Shao, F. Yu, J. Yan, Supervision Exists Everywhere: A Data Efficient Contrastive Language-Image Pre-training Paradigm, *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022, doi: 10.48550/arXiv.2110.05208.
[35] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X.

Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, *International Conference on Learning Representations*, 2020, doi: 10.48550/arXiv.2010.11929.

[36] S. Liu, L. Zhang, X. Yang, H. Su, J. Zhu, Query2label: A simple transformer way to multi-label classification, arXiv preprint arXiv:2107.10834, 2021, doi: 10.48550/arXiv.2107.10834.

[37] T. Xu, W. Chen, W. Pichao, F. Wang, H. Li, and R. Jin, CDTrans: Cross-domain Transformer for Unsupervised Domain Adaptation, *International Conference on Learning Representations*, 2022, doi: 10.48550/arXiv.2109.06165.

[38] S. H. Lee, S. Lee, B. C. Song, Vision transformer for small-size datasets, arXiv preprint arXiv:2112.13492, 2021, doi: 10.48550/arXiv.2112.13492.

[39] Y. Qin, Y. Wang, Z. S. Li, B. Wang, A. Ding, C. Wang, Y. Qin, Y. Wang, An in-depth tutorial on BJTU-RAO bogie datasets for fault diagnosis, *IEEE Access*, 2025, **13**, 60879-60888, doi: 10.1109/ACCESS.2025.3551603.

[40] R. Huang, Y. Liao, S. Zhang, W. Li, Deep decoupling convolutional neural network for intelligent compound fault diagnosis, *IEEE Access*, 2018, **7**, 1848-1858, doi: 10.1109/ACCESS.2018.2886343