A Semi-supervised Fault Diagnosis Method Based on Graph Convolution for Few-shot Fault Diagnosis

Yuyan Li[1], Tiantian Wang[2], and Jingsong Xie[3]

[1,3] *College of Traffic and Transportation Engineering, Central South University, Changsha, China*

liyuyan@csu.edu.cnu

jingsongxie@csu.edu.cn

[2] *College of Mechanical and Vehicle Engineering, Hunan University, Changsha, China*
wangtt@hnu.edu.cn

ABSTRACT

In practical bearing fault diagnosis, labeled fault data are difficult to obtain, and limited samples will lead to training overfitting. To address the above problems, a semi-supervised fault diagnosis method based on graph convolution is proposed. Firstly, the KNN graph construction method based on Euclidean distance (ED-KNN) is used to achieve label propagation. Then, a graph convolutional network framework based on dot product attention mechanism (GPGAT) was constructed to enhance the weights of high similarity nodes and diagnose bearing faults. The proposed method is validated on a public bearing dataset. The results show that the proposed method can make full use of very few labeled samples for fault diagnosis. Compared with other state-of-the-art methods, the proposed method achieves better diagnosis performance.

1. INTRODUCTION

Rotating machinery plays a crucial role in manufacturing, industrial robotics, transportation, and other fields. Bearings, as vital components of rotating machinery, may lead to significant economic losses if they fail. Bearings generate vast amounts of data during operation, and how to extract useful information from this data has become a hot topic in bearing fault diagnosis research in recent years n (Zhang et al., 2023). Intelligent fault diagnosis is an automated reasoning process based on data-driven approaches. In recent years, various deep learning models have been successfully applied to intelligent fault diagnosis (Jiao et al., 2020). However, their effective training relies on a large amount of labeled data, which is quite challenging in practical fault diagnosis (Yang et al., 2023). In engineering, labeling and

screening data are time-consuming tasks, making it essential to study high-precision bearing fault diagnosis methods under extremely scarce labeled samples.

Semi-supervised learning can leverage a small number of labeled samples to learn the information contained in the vast majority of unlabeled samples. In recent years, it has been widely studied in intelligent fault diagnosis in mechanical systems. Ding et al. (2023) trained multiple GANs to eliminate abnormal cases, thereby enhancing the performance of small-sample fault diagnosis in a semi-supervised manner. Yu et al. (2020) investigated a data augmentation method based on consistency regularization, which achieved fault diagnosis of bearings in cases where labeled samples are limited. Zhang et al. (2019) proposed an Active Semi-Supervised Learning GAN (ASSL-GAN), which minimizes the loss function through alternate updates to achieve higher accuracy. These methods can to some extent address the challenge of insufficient labeled samples in fault diagnosis tasks.

In recent years, with the flourishing development of Graph Neural Networks (GNNs) (Scarselli, F. et al., 2019), graph-based semi-supervised algorithms have gradually become a research hotspot. A graph based semi supervised learning algorithm propagates labeled data labels to unlabeled data by constructing a graph. The following paper provides a similar method implementation. Xie et al. (2022) utilized multi-scale graph convolution to aggregate multi-scale information of labeled samples and introduced an attention mechanism to form a new adaptive feature fusion layer. They proposed the Semi-supervised Multi-Scale Attention Graph Convolutional Network (MSA-GCN) for fault diagnosis and achieved satisfactory results. Kavianpour et al. (2022) addressed the issues of insufficient labeling of fault diagnosis data, changing operating conditions, and data loss in practical applications by aligning subdomains of the same class. They proposed a semi-supervised method based

on Autoregressive Moving Average (ARMA) filter graph convolution, adversarial adaptation, and Multi-layer Multi-kernel Local Maximum Mean Discrepancy (MK-LMMD). The above literature demonstrates the unique advantages of Graph Neural Networks in semi-supervised learning. However, this research still faces challenges such as high labeling rates and model instability under extreme labeling conditions, and there are still some shortcomings in feature mining for low-labeled samples, limiting its practical application.

In response to the scenario of fault diagnosis with extremely few labeled data, this paper proposes a network that combines Euclidean distance-based KNN graph (ED-KNN) with dot product graph attention mechanism (DPGAT). By utilizing Euclidean distance to measure the distance between labeled and unlabeled samples, an accurate KNN graph is obtained. Then, the dot product attention mechanism is used to further increase the weights of neighboring nodes with high similarity, in order to learn the optimal representation of the graph. The proposed method is experimented on a publicly available bearing dataset. The results demonstrate that the proposed method achieves high-precision classification of unlabeled data with minimal training on extremely few samples, indicating its significant engineering application value.

2. RELATED THEORIES

2.1. Graph neural networks

Unlike convolutional neural networks (CNNs), graph neural networks (GNNs) are a class of learning models based on graph-structured data. They can define graph convolutions based on the connections between nodes in non-Euclidean space. The difference between CNNs and GNNs in terms of convolution can be intuitively illustrated as shown in Figure 1. The involved graph structure can be simplified as follows:

$$G = (H, A) \tag{1}$$

Here, $H = \{h_1, h_2, \cdots, h_n\} \in R^{n \times d}$ represents the set of nodes; n is the number of nodes; d is the dimensionality of the input node features; $A \in R^{n \times n}$ represents the adjacency matrix representing the connections between nodes. The graph convolutional layer updates node features by aggregating neighboring node features. Typically, given the input graph G, the convolutional layer outputs a new set of node features $H' = \{h_1', h_2', \cdots, h_n'\} \in R^{n \times d'}$ with dimension d'. The graph convolutional layer can be represented as:

$$h_i' = \Gamma(h_i, Y(\{h_j \mid j \in N_i\})) \tag{2}$$

Among them, N_i is the number of neighboring nodes of node hi; $\Gamma(\cdot)$ represents nonlinear layers; $Y(\cdot)$ represents a certain node aggregation pattern.

2.2. Semi-supervised Learning with GNN

Graph-based semi-supervised learning typically involves establishing explicit relationships between labeled data and a large amount of unlabeled data using a graph structure, where data points are represented as vertices and the similarity between points is represented as edges. The constructed graph is then inputted into a graph neural network to obtain feature-level representations of the graph and its nodes. These graph-level or node-level features are then fed into a classifier for classification and fault diagnosis. This process leverages the graph structure to effectively utilize both labeled and unlabeled data for semi-supervised fault diagnosis.

3. PROPOSED METHOD

3.1. ED-KNN

Graphs can represent the similarity relationships between samples. Initially, the time-domain vibration signals collected from bearings are segmented via multiple sampling. Subsequently, these segments are transformed into frequency-domain signals using Fast Fourier Transform (FFT). The KNN graph is constructed by assessing the adjacency between labeled and unlabeled samples using Euclidean distance. The distance metric formula utilizing Euclidean distance is:

$$dis(x_i, y_i) = \left(\sum_{l=1}^{d} |x_i - y_i|^2 \right)^{\frac{1}{2}} \tag{3}$$

Where x_i represents the feature of the central node and y_i represents the neighboring node of x_i. For a certain central node $x' \in x_i$, the distance values between it and other neighboring nodes are arranged in ascending order:

$$D = \{dis_1(x', y_1), \ldots, dis_n(x', y_n)\}, (dis_1 < \ldots < dis_n) \tag{4}$$

The neighboring nodes of node x' are selected through k-nearest neighbors, denoted as:

$$\text{Top} - \text{k} = \{x_1', x_2', \ldots, x_k'\} \tag{5}$$

Top-k represents the set of k-nearest neighbors of x_i, where k is the number of nearest neighbors. Through experiments, it has been found that when k is set to 5, the quality of the constructed graph is satisfactory. By constructing the ED-KNN graph, each time k unlabeled data points are assigned pseudo-labels. This step establishes an intrinsic graph structure connection between labeled and unlabeled observed data, which can be regarded as a form of label propagation process.

When applying KNN nearest neighbor search, the sample set consists of all the samples from the bearings in that sample set. The connecting nodes are selected based on the proximity

determined by distances, where the top K nearest neighbors are chosen.

ED-KNN graph construction utilizes the similarity of feature vectors among samples to establish connections between them. By exploiting the joint dependencies between labels, label information is propagated along these connections, enabling the assignment of pseudo-labels to unlabeled samples. This approach facilitates a more thorough exploration of limited label information, thereby augmenting the model's capacity to learn from label information. The intuitive workflow is depicted in Fig. 1. Among them, when constructing the training set, the ED-KNN graph is constructed with labeled data as the central node and unlabeled data as neighboring nodes. Due to the large number of unlabeled data, there will be unlabeled data that has not been assigned and will not participate in model training. Meanwhile, the same sample may also be repeatedly labeled and participate in the construction of graphs with different central nodes. It is worth noting that since we use the entire graph for training, Neighboring nodes output features through weighted output. Therefore, nodes that are repeatedly labeled will not affect training, as they will be assigned different node weights in different graphs. When constructing the test set, all unlabeled data points in the test set are sequentially used to calculate the Euclidean distance from all other samples, and the top K-nearest samples are selected as neighboring nodes. Therefore, the number of constructed graphs is the same as the number of samples in the test set. Labeled data is only provided during the training phase, while in the testing phase, there is no availability of labeled data. When performing convolution calculations after constructing a graph, it is necessary to ensure that the number of linked nodes is consistent, otherwise graph convolution calculations will be very difficult. Therefore, we construct a KNN graph based on the top 5 nearest neighboring nodes. In fact, calculating the distance between nodes, selecting nodes through counting, and selecting nodes through threshold are similar.
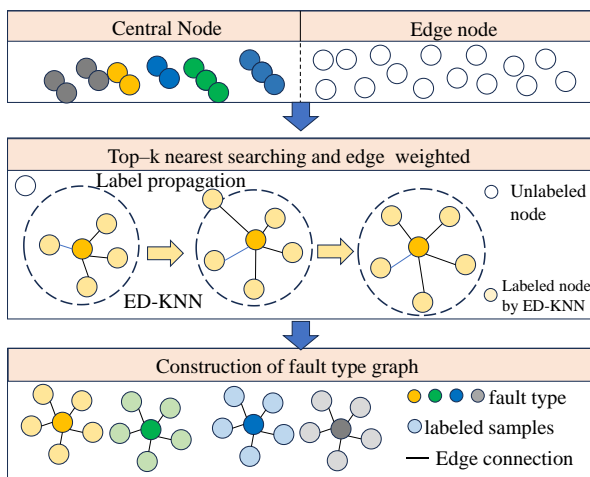

Figure 1. The process of ED-KNN.

3.2. DPGAT Diagnosis Framework

This paper proposes utilizing the dot-product attention mechanism to better learn graph representations by computing the weights of neighboring nodes' influence on the central node. Let $h = \{h_1, h_2, \ldots, h_N\}, h_i \in \mathbb{R}^F$ denote the input features of nodes, where N and F represent the number of nodes and the feature dimension, respectively. The output features of nodes are denoted as $h' = \{h_1', h_2', \ldots, h_N'\}, h_i' \in \mathbb{R}^{F'}$. $W \in \mathbb{R}^{F' \times F}$ represents the weight matrix of linear transformations applied at each node. Finally, softmax normalization is applied, followed by Leaky ReLU to introduce non-linearity. The output features of nodes are obtained using the following equation:

$$\mathbf{h}_i' = \sum_{j \in N_i} \alpha_{ij} \mathbf{W} \mathbf{h}_j \tag{6}$$

α_{ij} signifies the attention coefficient from neighboring node j to central node i, reflecting the significance of node j with respect to node i. α_{ij} is derived through SoftMax normalization of the attention parameter e_{ij} for each edge. The expression for the attention coefficient of node pair (i, j) is given by:

$$\alpha_{ij} = \text{softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik})} \tag{7}$$

The attention parameter e_{ij} is obtained through the outer product attention mechanism, which originates from node representation learning (Kim, D., & Oh, A. , 2022). The outer product of a node with its transpose can be regarded as its attention score. The mathematical expression for the outer product attention mechanism is:

$$e_{ij} = (\mathbf{W} h_i)^{\text{T}} \cdot \mathbf{W} h_j \tag{8}$$

Plugging it into Eq. (7) enables us to obtain the attention coefficients for each pair of nodes.

$$\alpha_{ij} = \text{softmax}(e_{ij}) \tag{9}$$

The features outputted by DPGAT are inputted into a fully connected (FC) layer to obtain the predicted label set. The prediction process can be represented as:

$$z_i = FC(\mathbf{h}_i') \tag{10}$$

The loss function of the DPGAT is:

$$loss_{Class} = -\frac{1}{I} \sum_i^I \sum_t^T y_i^{(t)} \ln(z_i^{(t)}) \tag{11}$$

Here, I denotes the label index; T stands for the number of classes; $y_i^{(t)}$ represents the t-dimensional value of the true labels; and $z_i^{(t)}$ signifies the t-dimensional value of the predicted label z_i.

3.3. The Overall Procedure

The overall framework of the proposed method is shown in Fig. 2, with specific explanations as follows.

1) Signal Acquisition: Collect vibration signals from bearings using sensors on the experimental platform.

2) Graph Construction: Divide the collected vibration signals into multiple independent samples and split them into training and testing sets. The training set comprises a small amount of labeled data and real collected data, while the testing set consists only of unlabeled data. Utilize the ED-KNN method to obtain the KNN graph.

3) Model Training: Construct a feature extraction network based on DPGAT. Obtain output features through Eq. (6). Input the training set sequentially into two DP-GAT layers and two FC layers, and obtain the predicted label set through Eq. (10). Then compute the loss using Eq. (11).

4) Model Testing: Feed the unlabeled testing set into the trained model to obtain diagnostic results and compare them with other semi-supervised fault diagnosis methods based on common GNNs.
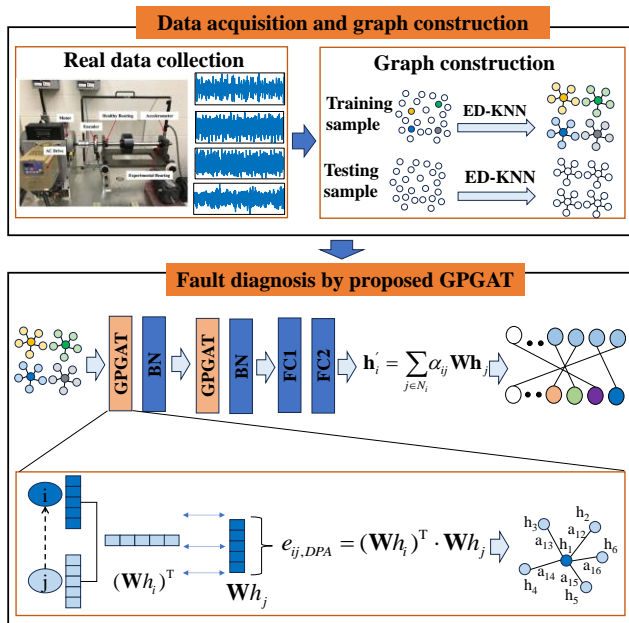

Figure 2. The framework of proposed method.

4. VALIDATION OF PERFORMANCE

The effectiveness of the proposed method was validated through two semi-supervised fault diagnosis instances. In Case 1, the dataset from Case Western Reserve University (Smith et al., 2015) was utilized, while in Case 2, the dataset from the University of Ottawa (UofO) (Huang & Baddour, 2018) was employed. To demonstrate the superiority of the proposed approach, it was compared with seven widely

researched graph neural networks, including Basic GAT (Veličković et al., 2018), DGAT, Graph Transformer (Shi et al., 2021), GraphConv (Morris et al., 2019), ChebConv (Defferrard et al., 2016), GraphSage (Hamilton et al., 2017), and GEN (Li et al., 2016). The above methods are only for graph convolutional models and do not involve a semi supervised learning process. We put it into the semi supervision framework proposed in this paper (using ED-KNN construction diagram) to verify the progressiveness of the proposed GPGAT.

4.1. Case 1: CWRU Dataset

The CWRU dataset was tested using SKF 6205 drive-end bearings. The sampling frequency of the accelerometer was 48 kHz. The bearing loads were categorized as 0HP, 1HP, 2HP, and 3HP, with corresponding speeds of 1797rpm, 1772rpm, 1750rpm, and 1730rpm, respectively. The health conditions of the bearings included four forms: Inner Race Fault (IF), Rolling Element Fault (ReF), Outer Race Fault (OF), and Normal Condition (NC). For each health condition's vibration signal, a sampling length of 1024 and the same sampling interval are used to ensure that there is no repetition between the data, resulting in 400 samples. These 400 samples were then randomly divided into training and testing samples at a ratio of 1:1. Verify the effectiveness of the proposed method through accuracy validation on the test set

Table 1. Description of the CWRU dataset.

Fault type	Speed(rpm)	Labeled samples and labeled rate	Train	Test
OF IF ReF NC	1730	4×1(0.25%)	4× 200	4× 200

4.2. Case 2: UofO Dataset

The dataset originates from the SpectraQuest Mechanical Fault Simulator at the University of Ottawa. Two ER16K ball bearings were installed to support the rotating shaft, which could be replaced with bearings in different health states. Accelerometers (ICP accelerometer, model 623C01) were placed on the experimental bearing housing for vibration data collection, while an incremental encoder (model EPC-775) measured the shaft speed. The signal sampling frequency was 200 kHz, and each experiment lasted for 10 seconds, including both acceleration and deceleration processes. For Case 2, vibration signals from bearings in four different states, including three types of faults and normal condition, were selected. The length of each sample was 4096 sampling points.

Table 2. Description of the UofO dataset.

Fault type	Speed(rpm)	Labeled samples and labeled rate	Train	Test
OF				
IF	846~1428	4×20(5%)	4×200	4× 200
ReF				
NC				

Table 3. The test accuracy on the Case 1 and Case 2.

Method	Case1	Case2
GPGAT(Proposed)	**98.67%**	**97.38%**
GAT	95.57%	94.87%
DGAT	96.17%	90.38%
Graph Transformer	90.58%	85.00%
GraphConv	94.17%	93.63%
ChebConv	95.14%	94.87%
SAGE	95.83%	87.17%
GEN	91.75**%**	91.37%

4.3. Experimental Results

To validate the superiority of the proposed construction method, the GPGAT was compared with seven others advanced GNN methods, and the average diagnostic accuracy is shown in Tables 3. In Case 1, the proposed GPGAT achieved a classification accuracy of 98.67%, which is 2.5% higher than the other best-performing method DGAT. In Case 2, the proposed GPGAT achieved a classification accuracy of 97.38%, which is 2.71% higher than the other best-performing methods GAT and ChebConv. The GPGAT proposed in this paper achieved better diagnostic accuracy compared to other graph convolution methods on both datasets, validating the effectiveness of the proposed approach.

To further demonstrate the diagnostic performance of the proposed method, we visualize the confusion matrix for Case 2, as shown in Fig. 3. Each health state has 200 test samples. The horizontal axis represents the predicted labels, while the vertical axis represents the true labels, where 0-3 denote the four health states OF, IF, ReF, and NC listed in Table 2. It can be observed that for the multi-class classification task, the proposed method GPGAT exhibits the best diagnostic performance.
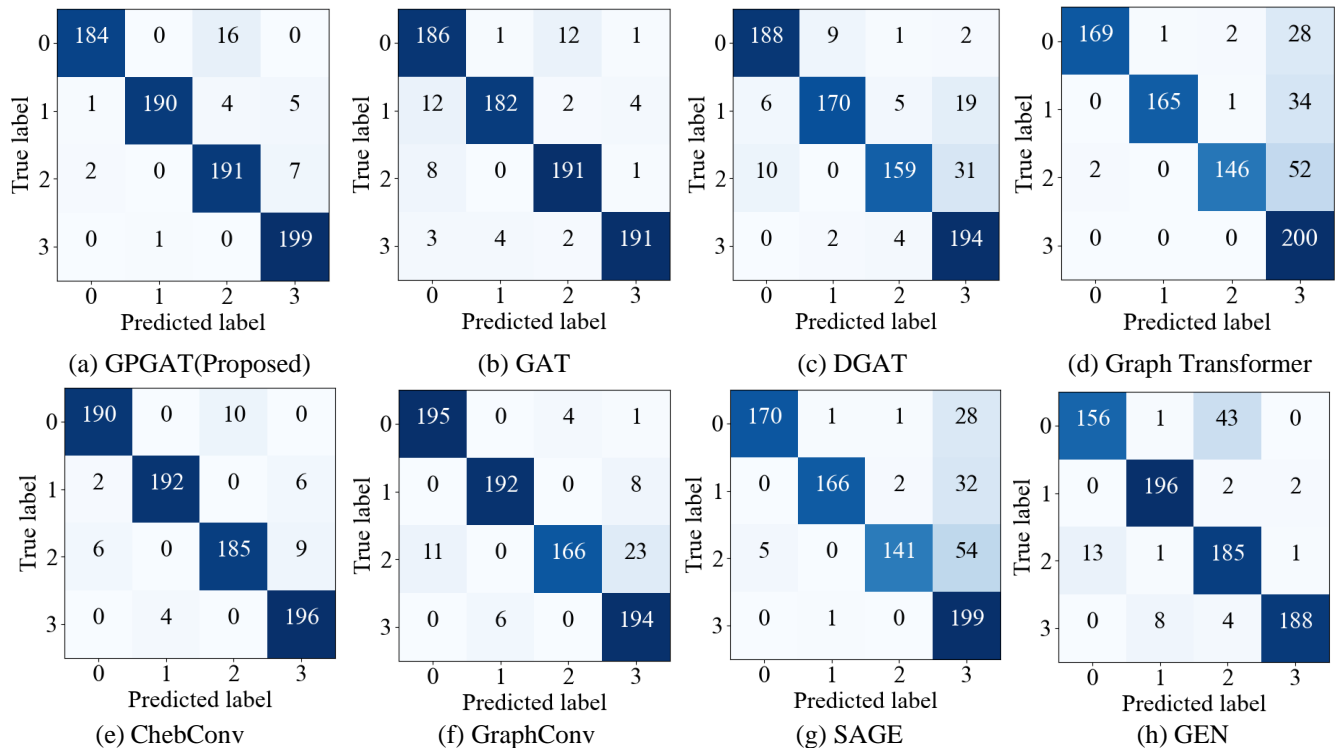

Figure 3. Classification confusion matrix for Case 2.

To better illustrate the feature extraction performance of the proposed method, the output feature vectors are reduced to two dimensions using T-SNE for Case 2, as shown in Fig. 4. From (a), it can be seen that the four types of features

represented by the four colors have a good degree of aggregation, and the distance between each type is relatively far, indicating that the proposed GPGAT has a good feature extraction ability. The features extracted by GPGAT exhibit higher aggregation and greater distances from each other

compared to other methods. The proposed method demonstrates better discriminative ability for all health states, and GPGAT maintains good diagnostic performance even at extremely low label rates.
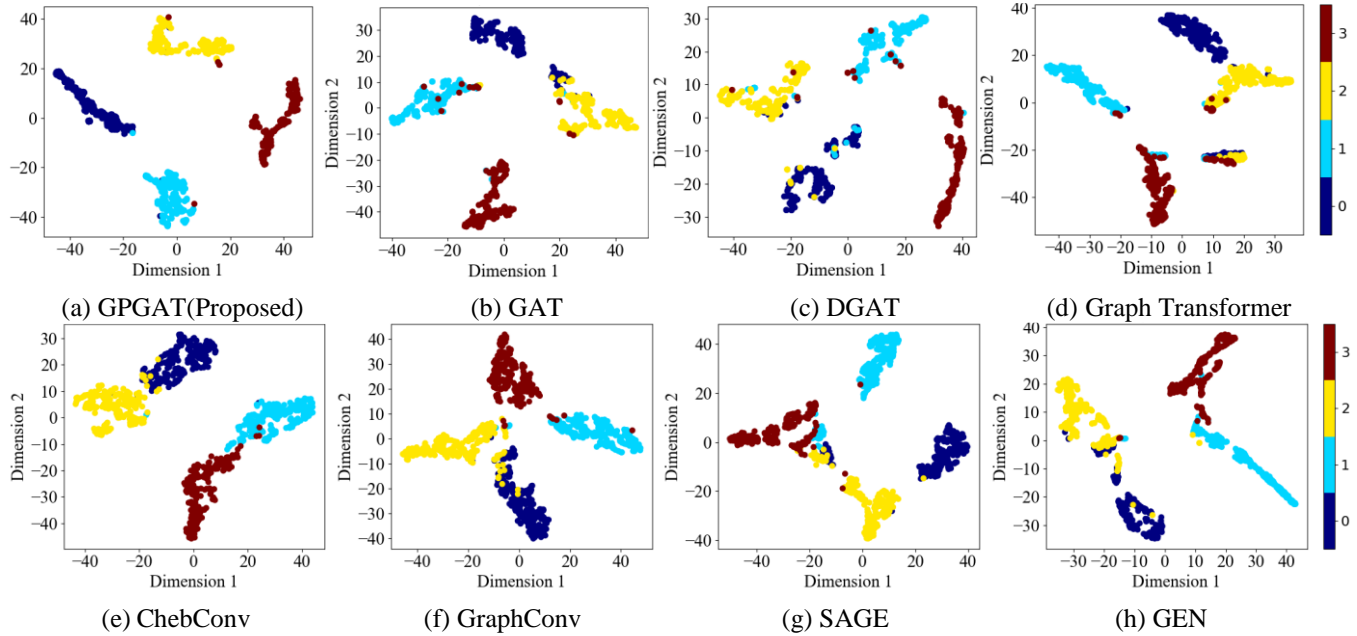

(a) GPGAT(Proposed) (b) GAT (c) DGAT (d) Graph Transformer

(e) ChebConv (f) GraphConv (g) SAGE (h) GEN

Figure 4. 2D visualization of the output features for all the methods on Case 2.

4.4. Discussion on Labeled Rate and K-value

To validate the effectiveness of the proposed method under small sample conditions, Case 2 dataset was trained and tested with labeled samples of 4, 8, 12, 16, and 20. The experiments were repeated ten times to obtain diagnostic accuracy. As shown in Fig. 5, it can be observed that as the label rate increases from 1% to 5%, the testing accuracy continues to improve. Even with a label rate of 1%, a fault diagnosis rate of 93.75% can be achieved, while a label rate of 5% yields a fault diagnosis accuracy of 97.38%.
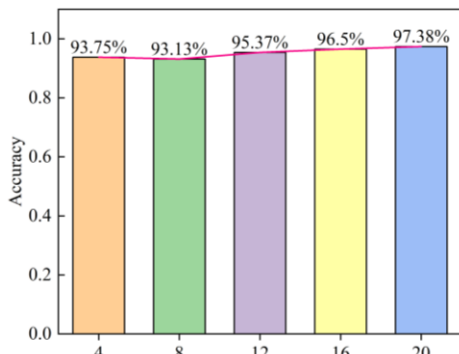

Figure 5. Accuracy of the proposed method with different low labeling rates.

Next, we discuss the influence of the k value used to construct the ED-KNN. The k value represents the number of neighboring nodes connected to each central node when creating the KNN graph. Using the proposed method, experiments were conducted on the two datasets at a label rate of 1%. Fig. 6 shows the effect of different k values on the diagnostic accuracy of the proposed model. It can be observed that on both the CWRU and UofO datasets, the diagnostic accuracy reaches its highest value when k=5, with accuracies of 100% and 93.75%, respectively. On the CWRU dataset, the change in accuracy with increasing K values is not particularly significant. This is mainly because the CWRU dataset is collected under steady-state conditions with artificially injected bearing faults, resulting in clean data with very distinct fault characteristics. Therefore, even with smaller K values, a good label propagation efficiency can be maintained. In contrast, the UofO dataset is collected under time-varying conditions, with faults occurring naturally, making fault characteristics less pronounced. Hence, an appropriate K value is required for ED-KNN. A suitable K value ensures that as much unlabeled data as possible is incorporated into the graph while minimizing graph construction errors.
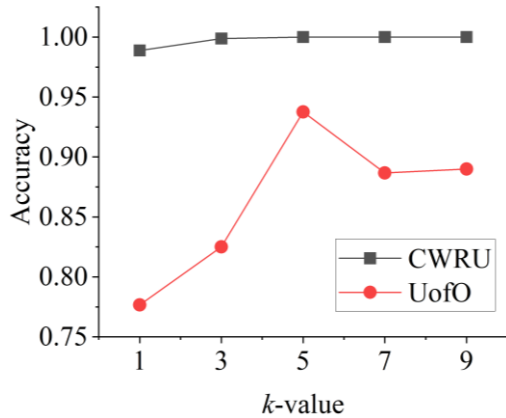
Figure 6. Diagnostic accuracy with different k values.

5. CONCLUSION

In response to the challenges faced by fault diagnosis under conditions of few samples, this paper proposes a new semi-supervised fault diagnosis method. The proposed ED-KNN calculates the Euclidean distance and sorts the distances in order to obtain the nearest neighboring nodes. It achieves label propagation from labeled data to unlabeled data. The designed GPGAT assigns different importance information to neighboring nodes through the dot product attention mechanism, further enhancing the reliability of the graph. Experimental validation was conducted on the CWRU and UofO dataset. Comparative results indicate that: (1) ED-KNN can effectively construct an undirected graph of labeled and unlabeled data, achieving label propagation. (2) The constructed GPGAT can assign different importance to neighboring nodes, thereby more accurately extracting node features and classification information from the KNN graph. (3) Compared with other state-of-the-art methods, the proposed approach can more accurately diagnose unlabeled samples under conditions of few or even extremely few samples.

REFERENCES

Zhang, S., Su, L., Gu, J., Li, K., Zhou, L., & Pecht, M. (2023). Rotating machinery fault detection and diagnosis based on deep domain adaptation: A survey. *Chinese Journal of Aeronautics*, 36(1), 45–74. doi: 10.1016/j.cja.2021.10.006

Jiao, J., Zhao, M., Lin, J., & Liang, K. (2020). A comprehensive review on convolutional neural network in machine fault diagnosis. *Neurocomputing*, 417, 36–63. doi: 10.1016/j.neucom.2020.07.088

Yang, X., Song, Z., King, I., & Xu, Z. (2023). A Survey on Deep Semi-Supervised Learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(9), 8934–8954. doi: 10.1109/TKDE.2022.3220219

Ding, Y., Ma, L., Ma, J., Wang, C., & Lu, C. (2019). A Generative Adversarial Network-Based Intelligent Fault Diagnosis Method for Rotating Machinery Under Small Sample Size Conditions. *IEEE Access*, 7, 149736–149749. doi: 10.1109/ACCESS.2019.2947194

Yu, K., Ma, H., Lin, T. R., & Li, X. (2020). A consistency regularization based semi-supervised learning approach for intelligent fault diagnosis of rolling bearing. *Measurement*, 165, 107987. doi: 10.1016/j.measurement.2020.107987

Zhang, X.-Y., Shi, H., Zhu, X., & Li, P. (2019). Active semi-supervised learning based on self-expressive correlation with generative adversarial networks. *Neurocomputing*, 345, 103–113. doi: 10.1016/j.neucom.2019.01.083

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1), 61–80. doi: 10.1109/TNN.2008.2005605

Xie, Z., Chen, J., Feng, Y., & He, S. (2022). Semi-supervised multi-scale attention-aware graph convolution network for intelligent fault diagnosis of machine under extremely-limited labeled samples. *Journal of Manufacturing Systems*, 64, 561–577. doi: 10.1016/j.jmsy.2022.08.007

Kavianpour, M., Ramezani, A., & Beheshti, M. T. H. (2022). A class alignment method based on graph convolution neural network for bearing fault diagnosis in presence of missing data and changing working conditions. *Measurement*, 199, 111536. doi: 10.1016/j.measurement.2022.111536

Kim, D., & Oh, A. (2022). How to Find Your Friendly Neighborhood: Graph Attention Design with Self-Supervision. arXiv. Retrieved from http://arxiv.org/abs/2204.04879

Smith, W. A., & Randall, R. B. (2015). Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study. *Mechanical Systems and Signal Processing*, 64–65, 100–131. doi: 10.1016/j.ymssp.2015.04.021

Huang, H., & Baddour, N. (2018). Bearing vibration data collected under time-varying rotational speed conditions. *Data in Brief*, 21, 1745–1749. doi: 10.1016/j.dib.2018.11.019

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018, February 4). Graph Attention Networks. arXiv. Retrieved from http://arxiv.org/abs/1710.10903

Shi, Y., Huang, Z., Feng, S., Zhong, H., Wang, W., & Sun, Y. (2021, May 9). Masked Label Prediction: Unified Message Passing Model for Semi-Supervised Classification. arXiv. Retrieved from http://arxiv.org/abs/2009.03509

Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., & Grohe, M. (2019). Weisfeiler and Leman Go Neural: Higher-Order Graph Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01), 4602–4609. doi: 10.1609/aaai.v33i01.33014602

Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. *Advances in Neural Information Processing Systems*, 29. Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2016/hash/04df4d434d481c5bb723be1b6df1ee65-Abstract.html

Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive Representation Learning on Large Graphs. *Advances in Neural Information Processing Systems*, 30. Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html

Li, G., Xiong, C., Thabet, A., & Ghanem, B. (2020, June 13). DeeperGCN: All You Need to Train Deeper GCNs. arXiv. doi: 10.48550/arXiv.2006.07739

BIOGRAPHIES

Yuyan Li received his M.Sc. degree from Kunming University of Science and Technology, Kunming, China. He is currently pursuing his D.E. degree in Traffic and Transportation Engineering at Central South University, Changsha, China. His current research interests include fault diagnosis, machine learning, vibration analysis, and pattern recognition.

Tiantian Wang received his bachelor's and Ph.D. degrees from Beihang University, Beijing, China, in 2012 and 2018, respectively. He is currently a Vice Professor at Central South University and Hunan University. His current research interests include vehicle aerodynamics and vehicle structures, especially train/tunnel aerodynamics and PHM for trains.

Jingsong Xie was born in Anren, Hunan, China, in 1989. He received his B.S. degree from the School of Mechanical Engineering, Northwestern Polytechnical University, Xi'an, China, in 2013, and his Ph.D. degree in mechanical engineering from Xi'an Jiaotong University, Xi'an, in 2018.
He joined the School of Traffic and Transportation Engineering, Central South University, Changsha, China, as a lecturer. His research interests include fault diagnosis, machine learning, vibration analysis, and crack diagnosis.