Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 08, August-2024

Bearing Fault Diagnosis Based on Tagcn-Transformer

Yuan Li
Tianjin University of Technology and Education,
School of Mechanical Engineering.
Tianjin, China

***Abstract*—To address the issue that traditional deep learning fault diagnosis models rely on a large number of faulty samples for training, while it is often difficult to collect data from equipment in a faulty or failure state in practical engineering applications, a TAGCN-Transformer-based bearing fault diagnosis method is proposed. First, TAGCN aggregates the neighborhood information of each node and uses graph pooling to update the features of each node. Second, Transformer captures the feature information of each graph data, constructing feature vectors that describe both the local and global characteristics of bearing faults. Finally, a classifier is used for intelligent identification of bearing fault types. Experimental results show that the method maintains a high accuracy in bearing fault identification even with limited samples, proving to be an effective approach for bearing fault feature extraction and pattern recognition**

***Keywords*—Rolling Bearing, Fault Diagnosis, Graph Neural Network, Transformer**

INTRODUCTION

Intelligent fault diagnosis, through the automatic extraction of deep features from signals, intelligently assesses the health status of equipment. It has become a crucial method for ensuring the safe operation and healthy service of machinery under big data conditions[1]. Traditional deep learning models, such as Convolutional Neural Networks (CNNs) [2]and Recurrent Neural Networks (RNNs)[3], can only handle regular data within Euclidean spaces, such as 2D grid images and 1D time series data. They are not suited for processing irregular data in non-Euclidean spaces, such as molecular graphs, traffic networks, and social networks. Additionally, traditional deep learning models only utilize the feature information of sample data itself, neglecting the intrinsic coupling relationships between sample data.

Graph Neural Networks (GNNs) have demonstrated exceptional performance on non-Euclidean data, such as in r social network analysis [4], protein design[5], and drug development[6]. They are capable of thoroughly exploring relationships between nodes for feature extraction, which presents new opportunities for the development of rolling bearing fault diagnosis. Xiao Lin et al. proposed a Bearing Fault Detection (GNNBFD) method based on GNNs. This method first constructs a graph using the similarity between samples; then, it inputs the constructed graph into a GNN for feature mapping, followed by fault detection using the mapped samples in a base detector[7]. Zhang Dingcheng combined graph convolution operators, graph coarsening methods, and graph pooling operations; a Deep Graph Convolution Network (DGCN) based on graph theory was used for acoustic-based fault diagnosis of rolling bearings[8]. Zhang Zhewang et al. proposed a Granger Causality-based Bearing Fault Detection GNN method (GCT-GNN), which effectively improved the model's robustness to noise[9]. Zhang Yin et al. introduced a rolling bearing fault diagnosis method based on Graph Convolutional Networks, optimizing the model using first-order ChebNet to effectively handle sample imbalance issues[10]. Although GNNs perform well with non-Euclidean data, their performance is influenced by their message-passing strategies. This can result in limitations such as a restricted number of layers, limited representational learning capacity, and issues like over-smoothing and over-compression[11].

Transformers can effectively capture the coupling information between graph nodes and enhance the representational capability of neural networks. Yang Zhuohong et al. proposed a signal transformer neural network (SiT) based on a pure attention mechanism for bearing fault diagnosis, which improves feature selection capabilities[12]. Tang Xinyu et al. introduced a wavelet-transform-based Vision Transformer (ViT) model, leveraging its powerful image classification abilities to enhance fault diagnosis[13]. Hou Yandong et al. proposed a multi-feature parallel fusion model called Diagnosisformer, based on attention mechanisms, for better fault feature representation in rolling bearings[14]. However, existing Transformer methods only encode the positional relationships between nodes, rather than explicitly encoding the structural relationships between nodes, and thus cannot effectively identify structural similarities or represent structural coupling relationships between nodes.

Based on this, this paper proposes a TAGCN-Transformer method. This approach combines the strengths of Transformers in aggregating long-range contextual information with the advantages of TAGNNs in capturing structural information of graphs, effectively capturing both local and global features of samples. Experimental results in rolling bearing fault diagnosis demonstrate that the method can accurately classify bearing faults even with a limited number of samples.

I. BASIC THEORY

A. Graph Neural Networks

Graph Neural Networks (GNNs) are a series of neural network models based on graphs, similar to CNNs. To aggregate data features, GNNs use a convolution process. The difference between GNNs and CNNs is that GNNs perform convolution on graphs, while CNNs handle discrete convolution in Euclidean space. The computational complexity of standard convolution is determined by the number and size of convolutional kernels. Here, we provide an introduction to the general framework of GNNs. Given a graph$G$, a node Vin the graph updates its hidden state h_v^{t+1} based on its previous stateh_v^t and messages m_v^{t+1} from its neighbors.

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) \qquad (1)$$

Where，

$$m_v^{t+1} = \sum_{u \in N(v|G)} M_t(h_v^t, h_u^t, e_{vu}) \qquad (2)$$

M_t is the message passing function, $U_t$is the GNN update function, and $N(v|G)$represents the set of neighboring nodes of node Vin the graph.

B. Graph Neural Networks

The attention mechanism was initially applied in the field of image processing. It emulates human attention by rapidly scanning the entire image to identify key areas of focus, thereby allocating more attention resources to capture detailed information about these focal points. Essentially, attention uses weights to represent the importance of target information and computes the attention value by performing a weighted sum of the target value and the weight. The attention mechanism model is shown in Figure 1.

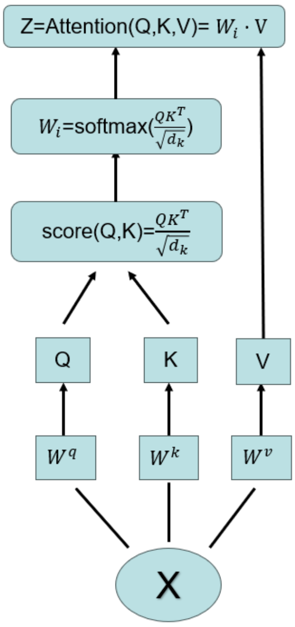

Z=Attention(Q,K,V)= $W_i \cdot$ V
W_i=softmax($\frac{QK^T}{\sqrt{d_k}}$)
score(Q,K)=$\frac{QK^T}{\sqrt{d_k}}$
Q
K
V
W^q
W^k
W^v
X

Fig. 1. Self-attention mechanism architecture diagram

I.I TAGNN-TRANSFORMER MODEL PRINCIPLES

A. TAGNN Model

In the GNN architecture, how to aggregate information from neighboring nodes into the node feature representation is crucial. TAGCN is a significant improvement over GCN. TAGCN explores a universal K-local filter, where K is retained as a hyperparameter. The K convolutional kernels in TAGCN have receptive fields ranging from 1 to K and are used for graph convolution in the vertex domain. The calculation process is shown in formula (3).

$$H^{(l+1)} = \sigma(G^{(l)}H^{(l)} + b\mathrm{I}) \qquad (3)$$

Here, $G^{(l)}$ is the graph convolution kernel of the graph convolution layer, with the kernel definition provided in formula (4). $H^{(l)}$ represents the node feature matrix, bIdenotes the bias term, and $H^{(l+1)}$ represents the new node feature matrix obtained after multiple graph convolution operations.

$$G^{(l)} = \sum_{k=0}^{K} g_k^{(l)} \hat{A}_k \qquad (4)$$

Here, $g_k^{(l)}$ represents the polynomial coefficients of the convolution layer l, and is the normalized adjacency matrix. Compared to GCN, it retains the hyperparameters.
After K rounds of message passing, the features of the original graph data are concatenated with the features from each layer after graph convolution:

$$H_{concat} = [H_1 \| H_2 \| \cdots \| H_K] \qquad (5)$$

Finally, we obtain the final node $\hat{A}$ feature matrixh_w in the graph pooling layer：

$$h_w = h_{G_w^K} = R(\{\mathrm{H}_{concat} \mid v \in G_w^K) \qquad (6)$$

In the equation, R represents the pooling layer.The node feature update process is illustrated in Figure 2.

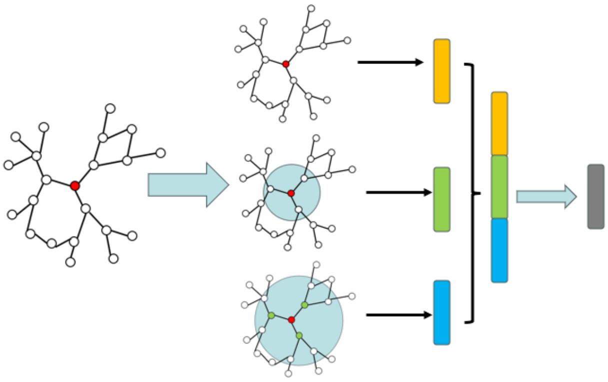


Fig. 2. Node feature update flowchart

B. Overall Network Architecture

This paper proposes a TAGNN-Transformer bearing fault diagnosis method using a graph Transformer, effectively addressing the issue of insufficient structural exploration and feature extraction between fault vibration signal nodes. The overall architecture of the proposed rolling bearing fault diagnosis method is shown in Figure 3. First, overlapping sampling is applied to the collected raw bearing fault data. The distance between the fault feature representation nodes is then computed, and the k nearest fault feature representation nodes (excluding itself) are selected to construct the adjacency matrix, obtaining the feature representation and adjacency matrix of the fault nodes, which serve as inputs to the fault diagnosis model. In the TAGNN-Transformer, the feature matrix and adjacency matrix of the fault nodes are first aggregated through the TAGNN module to gather neighborhood information and concatenated. The graph pooling layer then obtains the node feature matrix h_w. After batch normalization and fully connected layers, the new central node is passed to the Transformer neural network for classification by the MLP classifier, achieving rolling bearing fault diagnosis.

III. EXPERIMENTAL ANALYSIS

A. Introduction to Datasets and Parameter Settings

To validate the effectiveness of the proposed method, we first use the widely recognized CWRU dataset from Case Western Reserve University [15] for model training, tuning, and validation. The bearing fault test rig primarily includes an induction motor, a torque sensor, and a dynamometer, as shown in Figure 4. The data consists of vibration signals from the drive-end bearing at a speed of 1797 RPM, sampled at 12 kHz, with the bearing model being SKF6205. A description of the CWRU dataset is provided in Table 1.

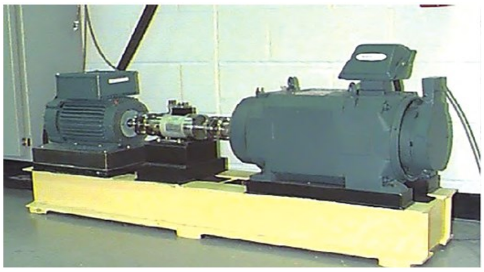


Fig. 3. Experimental setup platform

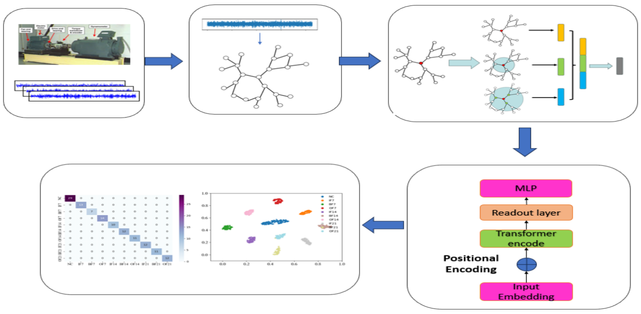


Fig. 3. TAGNN-Transformer fault diagnosis model

TABLE I. EXPERIMENTAL DATASET PARTITIONING

Diameter	Label	Damage Location	Number
0.007	**1**	Rolling Element	**10000**
	2	Inner Race	**10000**
	3	Outer Race	**10000**
0.014	**4**	Rolling Element	**10000**
	5	Inner Race	**10000**
	6	Outer Race	**10000**
0.028	**7**	Rolling Element	**10000**
	8	Inner Race	**10000**
	9	Outer Race	**10000**
	0	Normal	**10000**

Experimental hardware environment: CPU is i7-11800H, GPU is NVIDIA GeForce RTX 3060, memory is DDR4 16GB, CUDA version is 11.8. The development language used is Python, the development tool is PyCharm, and the development environment is Pytorch2.0.0 + cu118. The parameters set for the experiments are shown in Table 3.

TABLE II. EXPERIMENTAL PARAMETERS

Index	Parameters	Parameter Values
1	batch_size	32
2	Learning Rate	0.0003
3	Optimizer	Adam
4	dropout	0.2
5	global_pool	add
6	Num-heads	3
7	epochs	50

B. Analysis of Experimental Results

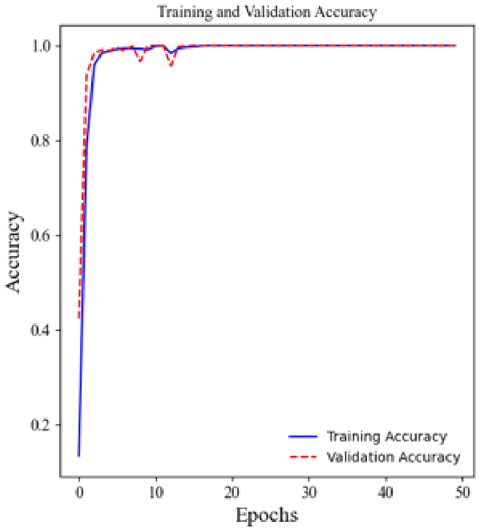


Fig. 4. Accuracy curve for CWRU dataset

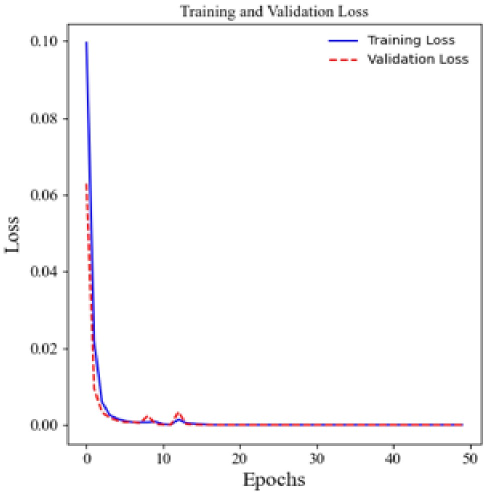
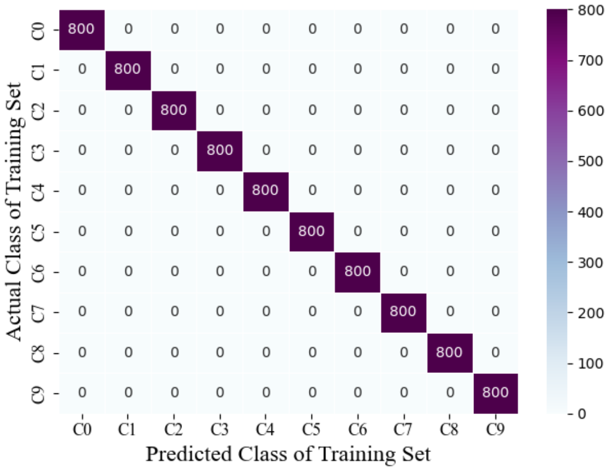


Fig. 5. Loss curve for CWRU dataset

Figure 4 shows the accuracy curves for the training set and the test set. From the accuracy curves, it can be observed that at the 6th epoch, the accuracy for the training set reaches 97%, and the accuracy for the validation set reaches 95%, indicating that the model has a fast learning rate. Figure 5 shows the loss function curves for the training set and the test set. From the loss function curves, it can be seen that at the 10th epoch, the loss function for both the training set and the validation set is already below 0.01 and gradually approaches 0 during subsequent training. After training for 100 epochs, the accuracy for the training set and the test set reaches 100% and 99.7%, respectively.

To provide a more intuitive display of the classification results, the confusion matrix is shown in Figure 6, and the t-SNE visualization results are shown in Figure 7. From Figures 6 and 7, it is evident that the method in this paper effectively classifies the 10 fault types. This is primarily due to the method's simultaneous extraction of global and local features of faults, which enhances the feature representation capability and improves the utilization of useful information, thereby boosting the model's performance.



	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
C0	800	0	0	0	0	0	0	0	0	0
C1	0	800	0	0	0	0	0	0	0	0
C2	0	0	800	0	0	0	0	0	0	0
C3	0	0	0	800	0	0	0	0	0	0
C4	0	0	0	0	800	0	0	0	0	0
C5	0	0	0	0	0	800	0	0	0	0
C6	0	0	0	0	0	0	800	0	0	0
C7	0	0	0	0	0	0	0	800	0	0
C8	0	0	0	0	0	0	0	0	800	0
C9	0	0	0	0	0	0	0	0	0	800

Fig. 6. confusion matrix

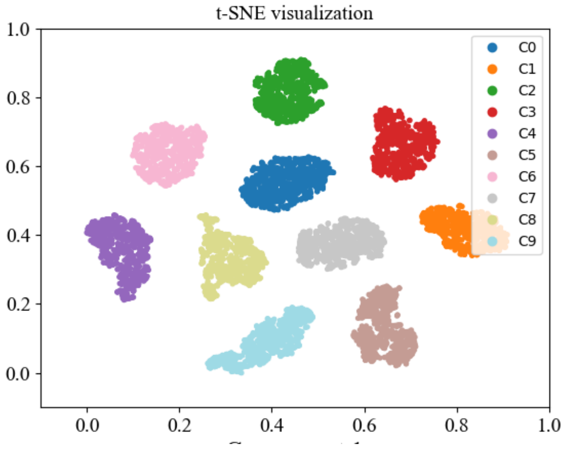
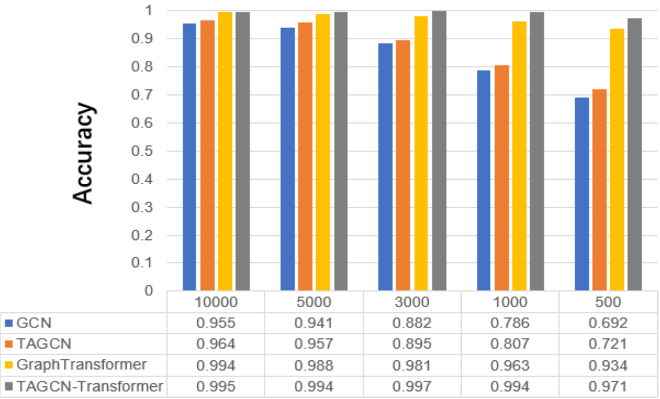


Fig. 7. the t-SNE visualization results

C. Analysis of diagnostic results under limited sample

Research on the generalization ability of bearing fault diagnosis models under varying sample sizes involves randomly sampling from the dataset and controlling the total number of samples to 1,000, 5,000, 3,000, 1,000, and 500. The samples are then divided into training, validation, and test sets in an 8:1:1 ratio.

To further validate the effectiveness of the GNN-Transformer neural network, this model is compared with several typical neural networks (GraphTransformer[16], GCN[17], and TAGCN[18]). The diagnostic results under different sample sizes are shown in Figure 8.



	10000	5000	3000	1000	500
■ GCN	0.955	0.941	0.882	0.786	0.692
■ TAGCN	0.964	0.957	0.895	0.807	0.721
■ GraphTransformer	0.994	0.988	0.981	0.963	0.934
■ TAGCN-Transformer	0.995	0.994	0.997	0.994	0.971

Fig. 8. Diagnostic results under different sample sizes

From Figure 8, it can be seen that all four models generally show a decreasing trend as the number of samples in the dataset decreases. Specifically, when the total number of fault samples is 10,000, 5,000, 3,000, and 1,000, the fault recognition accuracy of the TAGNN-Transformer diagnostic model is nearly identical. Therefore, the number of training samples can be appropriately reduced to save neural network computation time and improve pattern recognition efficiency, with minimal impact on fault recognition accuracy.

Comparing TAGNN and GCN models, TAGNN achieves higher accuracy than GCN when the total number of samples is the same. In comparison between TAGNN-Transformer and GraphTransformer models, it is observed that TAGNN-Transformer exhibits less fluctuation in fault recognition accuracy than GraphTransformer when the number of samples per fault type is 1,000, 800, and 500, and has higher accuracy than GCN. This indicates that the TAGNN-Transformer model has better generalization ability than GraphTransformer. Overall, the TAGNN-Transformer model demonstrates good generalization performance even under conditions of few samples.

IV. CONCLUSION

This paper presents a TAGNN-Transformer model for rolling bearing fault diagnosis. The TAGNN-Transformer enhances the model's feature extraction capability by using the TAGNN module with K different sizes of graph convolution kernels to extract and fuse local features at various scales. It employs the Transformer to perform adaptive learning of feature information, allowing the model to focus on more important features and improve performance. Experimental results demonstrate that the proposed fault diagnosis method performs well in fault classification tasks under conditions of limited samples.

REFERENCES

[1] Lei Yaguo, Yang Bin, Yang Zhaojun. : Deep transfer diagnosis method for machinery in big data era. Journal of Mechanical Engineering 55(7):1-8(2019).

[2] LeCun Y, Bottou L, Bengio Y. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[3] Hochreiter, S. & Schmidhuber, J. Long short-term memory. Neural Computation, 9(8), 1735-1780. (1997).

[4] FAN W, MA Y, LI Q. Graph neural networks for social recommendation; proceedings of the The world wide web conference, F, 2019 [C].

[5] INGRAHAM J, GARG V, BARZILAY R. Generative models for graph-based protein design [J]. Advances in neural information processing systems, 2019, 32.

[6] GAUDELET T, DAY B, JAMASB A R. Utilizing graph machine learning within drug discovery and development [J]. Briefings in bioinformatics, 2021, 22(6): bbab159.

[7] XIAO L, YANG X, YANG X. A graph neural network-based bearing fault detection method [J]. Scientific Reports, 2023, 13(1): 5286.

[8] ZHANG D, STEWART E, ENTEZAMI M. Intelligent acoustic-based fault diagnosis of roller bearings using a deep graph convolutional network [J]. Measurement, 2020, 156: 107585.

[9] ZHANG Z, WU L. Graph neural network-based bearing fault diagnosis using Granger causality test [J]. Expert Systems with Applications, 2024, 242: 122827.

[10] ZHANG Y, LI H. Rolling Bearing Fault Diagnosis Based on Graph Convolution Neural Network; proceedings of the International Conference on Intelligent Computing, F, 2022 [C]. Springer.

[11] LI Q, HAN Z, WU X-M. Deeper insights into graph convolutional networks for semi-supervised learning; proceedings of the Proceedings of the AAAI conference on artificial intelligence, F, 2018 [C].

[12] YANG Z, CEN J, LIU X. Research on bearing fault diagnosis method based on transformer neural network [J]. Measurement Science and Technology, 2022, 33(8): 085111.

[13] TANG X, XU Z, WANG Z. A novel fault diagnosis method of rolling bearing based on integrated vision transformer model [J]. Sensors, 2022, 22(10): 3878.

[14] HOU Y, WANG J, CHEN Z. Diagnosisformer: An efficient rolling bearing fault diagnosis method based on improved Transformer [J]. Engineering Applications of Artificial Intelligence, 2023, 124: 106507.

[15] Smith, W. A., Randall, R.B.: Rolling element bearing diagnostics using the case western reserve university data: a benchmark study. Mechanical System and Signal Processing 64-65:100-131 (2015).

[16] YUN S, JEONG M, KIM R. Graph transformer networks [J]. Advances in neural information processing systems, 2019, 32.

[17] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks [J]. arXiv preprint arXiv:160902907, 2016.

[18] DU J, ZHANG S, WU G,. Topology adaptive graph convolutional networks [J]. arXiv preprint arXiv:171010370, 2017.