10-1-2023

Remaining useful life prediction of bearings with attention-awared graph convolutional network

Yupeng Wei
San Jose State University, yupeng.wei@sjsu.edu

Dazhong Wu
University of Central Florida

Recommended Citation

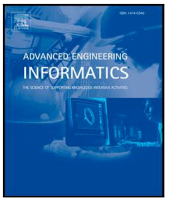
Yupeng Wei and Dazhong Wu. "Remaining useful life prediction of bearings with attention-awared graph convolutional network" *Advanced Engineering Informatics* (2023). https://doi.org/10.1016/j.aei.2023.102143

Advanced Engineering Informatics 58 (2023) 102143

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei



Full length article

Remaining useful life prediction of bearings with attention-awared graph convolutional network

Yupeng Wei [a,*], Dazhong Wu [b]

[a] Department of Industrial and Systems Engineering, San Jose State University, San Jose, 95192, CA, USA
[b] Department of Mechanical and Aerospace Engineering, University of Central Florida, Orlando, 32816, FL, USA

ARTICLE INFO

Keywords:
Bearing
Remaining useful life
Graph convolutional network
Attention mechanism
Deep learning

ABSTRACT

Graph Convolutional Networks (GCNs) have recently been used to predict the remaining useful life (RUL) of bearings due to its effectiveness in revealing correlations in condition monitoring data. However, traditional GCNs use a single graph only, either a temporal-correlated graph or a feature-correlated graph without considering both temporal and feature correlations of condition monitoring data. Additionally, traditional GCNs rely heavily on pre-defined graphs to aggregate correlated features. However, the topology of these pre-defined graphs may vary depending on a pre-defined threshold for cosine similarity or covariance which might affect prediction accuracy and robustness. To address these issues, we introduce a spectral graph convolutional operation that can handle both temporal-correlated and feature-correlated graphs, which allows one to consider both the temporal and feature correlations simultaneously. Moreover, we introduce a self-attention mechanism to construct the temporal-correlated and feature-correlated graphs automatically without defining a threshold. Such a mechanism allows the predictive model to learn graphs automatically during training so that the prediction accuracy and robustness can be significantly improved. The proposed method is demonstrated on two bearing datasets, and the experimental results have shown that it outperforms both traditional GCNs and other deep-learning methods in predicting RUL of bearings.

1. Introduction

A bearing is a machine component that constrains relative motion to only its desired motion and reduces friction between moving parts [1,2]. Bearings play a critical role in numerous fields, such as the aerospace industry, industrial machinery, agriculture equipment, and so on. For example, bearings are used in wind turbines to support the rotating blades and ensure smooth and efficient operation [3]; and bearings are also used in aircraft and satellites to support rotating parts and provide smooth motion in high-speed and demanding environments [4]. Like all other machine components, the performance of bearings degrades over time, also known as bearing degradation [5]. Bearing degradation can occur due to a variety of factors, including wear and tear, corrosion, contamination, improper installation, and overloading [6]. Bearing degradation can have negative effects on a system or machine in which it is used, including reduced efficiency, reduced lifespan, downtime, and even safety hazards [7,8]. To avoid or alleviate the negative effects of bearing degradation, it is critical to acknowledge the health condition as well as predict the remaining useful life (RUL) of bearings so that predictive maintenance can be performed in a timely manner to restore the degraded bearing to its proper performance [9].

Over the past decade, machine learning has been widely used to predict the RUL of bearings. The machine learning methods that are used for RUL prediction of bearings can be broadly classified into two categories: classical machine learning methods and deep learning methods. The classical machine learning methods include, but are not limited to, ensemble learning [10], support vector regression [11], the Gaussian process [12], fuzzy logic [13], and so on. For example, Shi et al. [14] employed an ensemble learning method to estimate the RUL of rolling bearings and explored the influence of various base learners and features on the prediction accuracy. Experiments have demonstrated that increasing the diversity of base learners and features in ensemble learning leads to a significant improvement in prediction accuracy. Kumar et al. [15] utilized a Gaussian process regression model to predict the RUL of bearings, and the prediction process was assisted by a health index that was developed using the Kullback–Leibler divergence constraint. The numerical results have demonstrated that the learned health index can effectively infer the health condition of bearings, and the Gaussian process regression model provides

* Corresponding author.
E-mail address: yupeng.wei@sjsu.edu (Y. Wei).

https://doi.org/10.1016/j.aei.2023.102143
Received 17 May 2023; Received in revised form 27 July 2023; Accepted 9 August 2023
Available online 4 September 2023

a highly accurate prediction of RUL. Huang et al. [16] proposed a Takagi–Sugeno fuzzy logic approach to predict the RUL of bearings. The approach employed maximum likelihood estimation for parameter estimation and numerical experiments have demonstrated its ability to predict the RUL, even with limited data.

Classical machine learning algorithms are not effective in handling complex problems that involve non-linear relationships or large amounts of data [17]. To address this issue, deep learning methods have been increasingly used for RUL predictions of bearings as they are able to learn hierarchical representations from complex and unstructured data, thereby improving prediction accuracy. The most popular deep learning methods include convolutional neural network (CNN) [18,19], long short-term memory (LSTM) [20], gated recurrent units (GRU) [21,22], recurrent neural network (RNN) [23], and some generative models such as variational autoencoder (VAE) [24,25] and generative adversarial network (GAN) [26]. For example, Yao et al. [27] combined a 1D CNN with an RNN for the purpose of predicting the RUL of bearings. The 1D CNN was utilized to gather temporal information from condition monitoring signals, while the RNN was utilized to process this information. To prevent overfitting, a global maximum pooling layer was utilized in place of a traditional fully connected layer. Ma and Mao [28] introduced a Convolutional LSTM network for estimating the RUL of rotating bearings. The LSTM network was designed to incorporate convolutional operations in both the input-to-state and state-to-state transitions, allowing it to take advantage of the strengths of LSTM while also incorporating time–frequency features. The results of their experiments demonstrate that the proposed Convolutional LSTM network surpasses both traditional LSTM and CNN models. Aside from the previously mentioned deep learning methods, generative models are also gaining popularity for generating various features and health indices. These features and indices can improve the accuracy of predicting the RUL of a bearing. As an example, Suh et al. [29] utilized a GAN to produce multiscale features for predicting the RUL of bearings. They utilized a U-Net architecture to capture sequence patterns in 1-D vibration signals. The results showed that the proposed approach effectively extracts features, resulting in improved prediction accuracy. Jing et al. [30] presented a VAE for predicting the RUL of bearings. This VAE is designed with a transformer backbone to capture the temporal correlations in condition monitoring data.

Although the deep learning methods mentioned above have been demonstrated to be effective in predicting the RUL of bearings, they are not effective in revealing the correlation of condition monitoring data [31,32]. This correlation can be utilized to identify and combine condition monitoring data with high affinity or similarity to improve model robustness and accuracy [33]. To address this issue, undirected graphs have been increasingly used to reveal such data correlations, where a graph vertex represents a data vector and the edges represent the similarity or affinity of condition monitoring data [34,35]. Graph Convolutional Networks (GCNs) are very effective in dealing with these undirected graphs because GCNs can consider the topological architecture of undirected graphs. GCNs predict RUL by using multiple repeated spectral graph convolution layers. Each layer performs two key operations: first, it aggregates similar data based on a pre-defined graph, and second, it projects the aggregated data into a higher-dimensional space to enhance the feature representation. To the best of our knowledge, GCNs have been used in the field of prognostics health management (PHM) to deal with two types of graphs: temporal-correlated graphs and feature-correlated graphs. The temporal-correlated graph is an undirected graph built in the time domain, where each vertex of the graph represents multiple features at a particular moment. GCNs that deal with temporal-correlated graphs can aggregate multiple features at distinct times, enabling them to consider temporal correlation in RUL predictions [36]. The feature-correlated graph is an undirected graph built in the feature channel, where each vertex of the graph represents a single time-series feature. GCNs that handle feature-correlated graphs can aggregate similar features so that feature correlations are considered in RUL predictions [33,37].

In summary, although traditional GCNs are effective in predicting the RUL of bearings, there are two issues that need to be addressed to improve their accuracy and robustness. Firstly, existing GCNs only use a single graph, either a temporal-correlated or feature-correlated graph, to predict RUL. This means that they can only consider either the temporal or feature correlation of the condition monitoring data. To the best of our knowledge, very few studies have been conducted to develop GCNs that can handle both temporal-correlated and feature-correlated graphs. Secondly, traditional GCNs rely heavily on pre-defined graphs to aggregate correlated features. However, the topology of these pre-defined graphs may vary depending on a pre-defined threshold for cosine similarity or covariance which might affect prediction accuracy and robustness. To address these two issues, we introduce a spectral graph convolutional operation that can handle both temporal-correlated and feature-correlated graphs, allowing us to consider both temporal and feature correlations of the condition monitoring data simultaneously. Additionally, we introduce a self-attention mechanism to construct the temporal-correlated and feature-correlated graphs automatically without defining a threshold. Such a mechanism allows the predictive model to learn graphs automatically during training, so that the prediction accuracy and robustness can be largely improved. Furthermore, we use another attention-based selection method to select the most important features generated by the proposed attention-awared graph convolutional operation. This method uses a multi-head attention mechanism to identify the most important features and dynamically weigh their contributions to the RUL prediction, thereby improving the accuracy and robustness of the model. The primary contributions of this work can be summarized as follows:

- A spectral graph convolutional operation that can handle both temporal-correlated and feature-correlated graphs is developed to consider both temporal and feature correlations of condition monitoring data simultaneously.
- A self-attention mechanism is introduced to construct graphs automatically during training. By constructing more accurate graphs, we can further improve the prediction accuracy and robustness.

The remaining sections of this paper are organized in the following manner. Section 2 introduces the proposed attention-awared graph convolutional network. Section 3 and Section 4 provide two case studies to demonstrate the effectiveness of the proposed method. Finally, in Section 5, a summary of the work is given, along with a discussion of future work.

2. Attention-awared graph convolutional network

In this section, we present the attention-awared graph convolutional network. First, we introduce the spectral graph convolutional operation that can be used to handle the topology of both temporal-correlated and feature-correlated graphs, followed by building graphs with the self-attention mechanism.

2.1. Spectral graph convolutional operation for temporal and feature-correlated graphs

The spectral graph convolutional operation for both graphs starts with initializing the temporal-correlated and feature-correlated graphs. To build the initial graphs, we extract features from the bearings' condition monitoring data. The extracted features are then sampled using a moving window of length S and a stride of 1. The ωth sampled features for bearing i is denoted as $\mathbf{X}_{i,\omega} \in \mathbb{R}^{S\times F}$, where S denotes the size of the moving window and the time length of the sampled features, and F represents the number of extracted features. To initialize

the temporal-correlated graph and feature-correlated graph, the cosine similarity matrix $\mathbf{C}_{i,\omega} \in \mathbb{R}^{S\times S}$ and the covariance matrix $\mathbf{V}_{i,\omega} \in \mathbb{R}^{F\times F}$, respectively, should be calculated from $\mathbf{X}_{i,\omega}$. The matrix elements $c_{s,s'}^{(i,\omega)}$ of $\mathbf{C}_{i,\omega}$ can be obtained by Eq. (1), where $\mathbf{x}_s^{(i,\omega)}$ denotes the sth column vector of the ωth sampled extracted features for bearing unit i; $\|\cdot\|$ is the l2-norm of a vector.

$$c_{s,s'}^{(i,\omega)} = \left(\mathbf{x}_s^{(i,\omega)} \cdot \mathbf{x}_{s'}^{(i,\omega)}\right) / \left(\|\mathbf{x}_s^{(i,\omega)}\| \|\mathbf{x}_{s'}^{(i,\omega)}\|\right), \quad s = 1, \ldots, S \tag{1}$$

The matrix elements $v_{f,f'}^{(i,\omega)}$ of $\mathbf{V}_{i,\omega}$ can be obtained by Eq. (2), where $\mathbf{x}_f^{(i,\omega)}$ denotes the fth row vector of the ωth sampled extracted features for bearing unit i; $\mathbb{E}[\cdot]$ is an expectation of a vector.

$$v_{f,f'}^{(i,\omega)} = \mathbb{E}\left[\left(\mathbf{x}_f^{(i,\omega)} - \mathbb{E}\left[\mathbf{x}_f^{(i,\omega)}\right]\right)\left(\mathbf{x}_{f'}^{(i,\omega)} - \mathbb{E}\left[\mathbf{x}_{f'}^{(i,\omega)}\right]\right)^T\right], \quad f = 1, \ldots, F \tag{2}$$

With respect to the cosine similarity matrix $\mathbf{C}_{i,\omega}$, a function is applied to such a matrix to derive the corresponding graph adjacency matrix $\mathbf{A}_t \in \mathbb{R}^{S\times S}$, and such a function can be written as Eq. (3), where $a_{s,s'}$ is the element of the graph adjacency matrix $\mathbf{A}_t$ and φ is the threshold that can be used to determine the value of such a matrix. If $a_{s,s'} = 1$, there is an edge between time node s and time node s' in the generated temporal-correlated graph $\mathbb{G}_t$. If $a_{s,s'} = 0$, there is no edge between time node s and time node s' in $\mathbb{G}_t$. If there is an edge between two time nodes in $\mathbb{G}_t$, it means that these two time nodes are temporally correlated, and if there is no edge, they are not temporally correlated.

$$a_{s,s'} = \begin{cases} 1 & c_{s,s'}^{(i,\omega)} \geq \varphi, \\ 0 & c_{s,s'}^{(i,\omega)} < \varphi. \end{cases} \tag{3}$$

With respect to the covariance matrix $\mathbf{V}_{i,\omega}$, the corresponding graph adjacency matrix $\mathbf{A}_f \in \mathbb{R}^{F\times F}$ can be obtained based upon the covariance of two row vectors in the feature matrix $\mathbf{X}_{i,\omega}$. The element $a_{f,f'}$ of the graph adjacency matrix $\mathbf{A}_f$ is set to one if the covariance of two features is positive, and to zero otherwise. If $a_{f,f'} = 1$, there is an edge between feature nodes f and f' in the feature-correlated graph $\mathbb{G}_f$, meaning that these two features are correlated. If $a_{f,f'} = 0$, there is no edge between feature nodes f and f' in $\mathbb{G}_f$, meaning that these two features are not correlated.

Next, the spectral graph convolutional operation for both graphs should be employed to handle the topology of two graphs $\mathbb{G}_t$ and $\mathbb{G}_f$, where the spectral graph convolutional operation includes the spectral graph convolutional operation performed on both temporal-correlated graph $\mathbb{G}_t$ and feature-correlated graph $\mathbb{G}_f$. Eq. (4) shows the spectral graph convolutional operation for both graphs, where g_t denotes the graph filter from the temporal-correlated graph $\mathbb{G}_t$; g_f denotes the graph filter provided by the generated feature-correlated graph $\mathbb{G}_f$; $\mathscr{F}$ denotes the graph Fourier transform, and $\mathscr{F}^{-1}$ denotes the inverse graph Fourier transform;

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \left[\mathscr{F}^{-1}\left(\mathscr{F}(g_t) \odot \mathscr{F}(\mathbf{X}_{i,\omega})\right),\ \mathscr{F}^{-1}\left(\mathscr{F}(g_f) \odot \mathscr{F}(\mathbf{X}_{i,\omega}^T)\right)\right] \tag{4}$$

In order to manage Eq. (4) more effectively, the spectral graph convolutional operation performed on both $\mathbb{G}_t$ and $\mathbb{G}_f$ can be rewritten in a format of eigendecomposition of the temporal Laplacian matrix $\mathbf{L}_t$ and feature Laplacian matrix $\mathbf{L}_f$, which can be written as Eq. (5),

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \left[\mathbf{Q}_t\left(\mathbf{Q}_t^T g_t \odot \mathbf{Q}_t^T \mathbf{X}_{i,\omega}\right),\ \mathbf{Q}_f\left(\mathbf{Q}_f^T g_f \odot \left(\mathbf{X}_{i,\omega}\mathbf{Q}_f\right)^T\right)\right] \tag{5}$$

Eq. (5) can also be expressed as Eq. (6), where $\mathbf{Q}_t$ denotes the eigenvector of $\mathbf{L}_t$; θ_t represents the parameters in the graph filter g_t; Λ_t denotes the vector of eigenvalues of $\mathbf{L}_t$; $\mathbf{Q}_f$ denotes the eigenvector of $\mathbf{L}_f$, θ_f represents the parameters in g_f, and Λ_f denotes the array of eigenvalues of $\mathbf{L}_f$.

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \left[\mathbf{Q}_t g_{t,\theta_t}\left(\Lambda_t\right)\mathbf{Q}_t^T\mathbf{X}_{i,\omega},\ \mathbf{Q}_f g_{f,\theta_f}\left(\Lambda_f\right)\mathbf{Q}_f^T\mathbf{X}_{i,\omega}^T\right] \tag{6}$$

The temporal Laplacian matrix $\mathbf{L}_t$ is defined as shown in Eq. (7), where $\mathbb{I}_t \in \mathbb{R}^{S\times S}$ is the identity matrix and $\mathbb{D}_t$ is a diagonal matrix whose diagonal entries are the degrees of each node in the temporal correlated graph $\mathbb{G}_t$.

$$\mathbf{L}_t = \mathbb{I}_t - \mathbb{D}_t^{-1/2}\mathbf{A}_t\mathbb{D}_t^{-1/2} \tag{7}$$

Since the temporal Laplacian matrix $\mathbf{L}_t$ is a real symmetric matrix, it can be decomposed into its eigenvectors and eigenvalues as shown in Eq. (8).

$$\mathbf{L}_t := \mathbf{Q}_t\Lambda_t\mathbf{Q}_t^{-1} = \mathbf{Q}_t\Lambda_t\mathbf{Q}_t^T \tag{8}$$

The feature Laplacian matrix $\mathbf{L}_f$ is defined as shown in Eq. (9), where $\mathbb{I}_f \in \mathbb{R}^{F\times F}$ is the identity matrix and $\mathbb{D}_f$ is a diagonal matrix whose diagonal entries are the degrees of each node in the feature-correlated graph $\mathbb{G}_f$.

$$\mathbf{L}_f = \mathbb{I}_f - \mathbb{D}_f^{-1/2}\mathbf{A}_f\mathbb{D}_f^{-1/2} \tag{9}$$

Likewise, Eq. (9) can be decomposed into its eigenvectors and eigenvalues as shown in Eq. (10).

$$\mathbf{L}_f := \mathbf{Q}_f\Lambda_f\mathbf{Q}_f^{-1} = \mathbf{Q}_f\Lambda_f\mathbf{Q}_f^T \tag{10}$$

Then, Eq. (11) can be derived by substituting Eqs. (8) and (10) into Eq. (6).

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \left[g_{t,\theta_t}\left(\mathbf{L}_t\right)\mathbf{X}_{i,\omega},\ g_{f,\theta_f}\left(\mathbf{L}_f\right)\mathbf{X}_{i,\omega}^T\right] \tag{11}$$

Due to the high computational cost of solving Eq. (11), it is widely accepted to use an approximation technique that involves the first-order Chebyshev polynomials for the spectral convolutional operation [38]. Eq. (12) presents the formulation of the 1st-order Chebyshev polynomials approximation of Eq. (11). Here, $O = 1$ denotes the first order approximation, $\tilde{\mathbf{L}}_t$ represents the scaled temporal Laplacian matrix, $\mathscr{C}_o(\cdot)$ is the oth order Chebyshev polynomials, and $\tilde{\mathbf{L}}_f$ represents the scaled feature Laplacian matrix.

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \left[\sum_{o=1}^{O}\theta_{t,o}\mathscr{C}_o\left(\tilde{\mathbf{L}}_t\right)\mathbf{X}_{i,\omega},\ \sum_{o=1}^{O}\theta_{f,o}\mathscr{C}_o\left(\tilde{\mathbf{L}}_f\right)\mathbf{X}_{i,\omega}^T\right] \tag{12}$$

Eq. (12) can also be expanded as Eq. (13)

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \left[\left(\theta_t\left(\mathbb{D}_t^{-1/2}\mathbf{A}_t\mathbb{D}_t^{-1/2} + \mathbb{I}_t\right)\right)\mathbf{X}_{i,\omega},\ \left(\theta_f\left(\mathbb{D}_f^{-1/2}\mathbf{A}_f\mathbb{D}_f^{-1/2} + \mathbb{I}_f\right)\right)\mathbf{X}_{i,\omega}^T\right] \tag{13}$$

Next, we set $\tilde{\mathbf{A}}_t$ equals $\mathbf{A}_t + \mathbb{I}_t$ and $\tilde{\mathbf{A}}_f$ equals $\mathbf{A}_f + \mathbb{I}_f$ and substitute $\tilde{\mathbf{A}}_t$ and $\tilde{\mathbf{A}}_f$ into Eq. (13), Eq. (14) can be derived.

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \left[\hat{\mathbf{A}}_t\mathbf{X}_{i,\omega}\Theta_t,\ \hat{\mathbf{A}}_f\mathbf{X}_{i,\omega}^T\Theta_f\right] \tag{14}$$

In Eq. (14), $\Theta_t \in \mathbb{R}^{F\times F'}$ is the matrix format of the parameters in the graph filter g_t, and $\Theta_f \in \mathbb{R}^{S\times S'}$ is the matrix format of the parameters in the graph filter g_f. $\hat{\mathbf{A}}_t$ is the scaled temporal adjacency matrix, which is expressed as Eq. (15), here $\tilde{\mathbb{D}}_t$ denotes the diagonal matrix whose diagonal entries are the degrees of $\tilde{\mathbf{A}}_t$, and $\tilde{\mathbf{A}}_t$ can be written as $\mathbf{A}_t + \mathbb{I}_t$.

$$\hat{\mathbf{A}}_t = \tilde{\mathbb{D}}_t^{-1/2}\tilde{\mathbf{A}}_t\tilde{\mathbb{D}}_t^{-1/2} \tag{15}$$

In addition, $\hat{\mathbf{A}}_f$ is the scaled feature adjacency matrix, which is expressed as Eq. (16), here $\tilde{\mathbb{D}}_f$ denotes the diagonal matrix whose diagonal entries are the degrees of $\tilde{\mathbf{A}}_f$, and $\tilde{\mathbf{A}}_f$ can be written as $\mathbf{A}_f + \mathbb{I}_f$.

$$\hat{\mathbf{A}}_f = \tilde{\mathbb{D}}_f^{-1/2}\tilde{\mathbf{A}}_f\tilde{\mathbb{D}}_f^{-1/2} \tag{16}$$

To enhance the effectiveness of the spectral graph convolutional operation for both graphs, a bias weighted vector and an activation function are introduced into Eq. (14). The resulting equation is denoted as

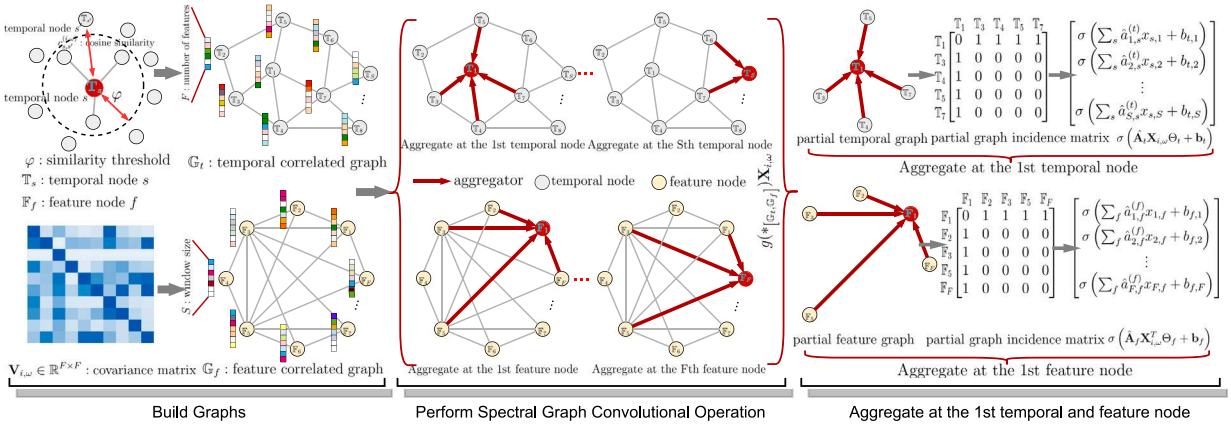


Fig. 1. Framework of the proposed spectral graph convolutional operation for both temporal-correlated and feature-correlated graphs, including the graphs construction and the spectral graph convolutional operation.

Eq. (17), where $\mathbf{b}_t$ denotes the temporal bias weighted vector, $\mathbf{b}_f$ denotes the feature bias weighted vector, and σ represents the activation function.

$$g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \sigma\left[\hat{\mathbf{A}}_t\mathbf{X}_{i,\omega}\Theta_t + \mathbf{b}_t,\ \hat{\mathbf{A}}_f\mathbf{X}_{i,\omega}^T\Theta_f + \mathbf{b}_f\right] \tag{17}$$

Fig. 1 displays the proposed spectral graph convolutional operation for both temporal-correlated and feature-correlated graphs. The operation begins with constructing both temporal-correlated and feature-correlated graphs. The temporal-correlated graph $\mathbb{G}_t$ is constructed using cosine similarity, while the feature-correlated graph $\mathbb{G}_f$ is constructed using covariance. The spectral graph convolutional operation for both graphs is then applied to exploit the topology of the constructed graphs. Mathematically, these two operations can be represented as $g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega}$. In such an operation, both temporal and feature information are aggregated at each corresponding node based on its neighbors.

2.2. Building graphs with the self-attention mechanism

Temporal-correlated and feature-correlated graphs that are pre-constructed based on cosine similarity and covariance have been used to build predictive models [31,37], as shown in Eqs. (1) to (3). However, the topology of these pre-constructed graphs may vary depending on a pre-defined threshold φ for cosine similarity or covariance. The pre-constructed graphs significantly affect the accuracy and robustness of the predictive model. To address this issue, we introduce a self-attention mechanism to construct the temporal-correlated and feature-correlated graphs automatically without defining a threshold. Such a mechanism allows the predictive model to learn graphs automatically during training, so that the prediction accuracy and robustness can be largely improved. The main idea of building attention-awared graphs using the self-attention mechanism is constructing the temporal attention adjacency matrix $\hat{\mathbb{A}}_t$ and the feature attention adjacency matrix $\hat{\mathbb{A}}_f$ that can perform the same functions as $\hat{\mathbf{A}}_t$ and $\hat{\mathbf{A}}_f$. There are two primary reasons why we build the graphs using the self-attention mechanism. Firstly, it generates an attention matrix that represents the importance level of the condition monitoring data. The data aggregation process can be performed based on the coefficients in the attention matrix instead of using the coefficients in the traditional scaled adjacency matrix, which ensures that the most relevant and important data can be aggregated accordingly. Second, it generates comparable $\hat{\mathbf{A}}_t$ and $\hat{\mathbf{A}}_f$ values, ranging from 0 to 1, that prevent severe gradient explosion and vanishing issues when handling the automatically generated graphs. Further details about attention mechanism can be found in [39,40].

The temporal attention adjacency matrix $\hat{\mathbb{A}}_t$ generated by the self-attention mechanism can be mathematically represented as Eq. (18), where SoftMax denotes the normalized exponential activation function that can be used to generate attention value from 0 to 1; Tanh refers to the hyperbolic tangent activation function; both $\mathbb{W}_{t,2} \in \mathbb{R}^{S\times D_t}$ and $\mathbb{W}_{t,1} \in \mathbb{R}^{D_t\times F}$ denotes the trainable matrices used in the attention mechanism.

$$\hat{\mathbb{A}}_t = \text{SoftMax}\left(\mathbb{W}_{t,2}\cdot\text{Tanh}\left(\mathbb{W}_{t,1}\cdot\mathbf{X}_{i,\omega}^T\right)\right) \tag{18}$$

Likewise, the feature attention adjacency matrix $\hat{\mathbb{A}}_f$ generated by the self-attention mechanism can be mathematically represented as Eq. (19), where both $\mathbb{W}_{f,2} \in \mathbb{R}^{F\times D_f}$ and $\mathbb{W}_{f,1} \in \mathbb{R}^{D_f\times S}$ denotes the trainable matrices used in the attention mechanism.

$$\hat{\mathbb{A}}_f = \text{SoftMax}\left(\mathbb{W}_{f,2}\cdot\text{Tanh}\left(\mathbb{W}_{f,1}\cdot\mathbf{X}_{i,\omega}\right)\right) \tag{19}$$

Next, substitute Eqs. (18) and (19) into Eq. (17) with replacing $\hat{\mathbf{A}}_t$ and $\hat{\mathbf{A}}_f$ by $\hat{\mathbb{A}}_t$ and $\hat{\mathbb{A}}_f$, respectively, we can obtain Eq. (20),

$$\sigma\Big[\text{SoftMax}\left(\mathbb{W}_{t,2}\cdot\text{Tanh}\left(\mathbb{W}_{t,1}\cdot\mathbf{X}_{i,\omega}^T\right)\right)\cdot\mathbf{X}_{i,\omega}\Theta_t + \mathbf{b}_t,\ \text{SoftMax}\left(\mathbb{W}_{f,2}\cdot\text{Tanh}\left(\mathbb{W}_{f,1}\cdot\mathbf{X}_{i,\omega}\right)\right)\mathbf{X}_{i,\omega}^T\Theta_f + \mathbf{b}_f\Big] \tag{20}$$

Fig. 2 shows the details about how we construct the temporal attention adjacency matrix $\hat{\mathbb{A}}_t$. First, the sampled data $\mathbf{X}_{i,\omega}$ is projected into a higher dimensional space using the first parameter matrix $\mathbb{W}_{t,1}$ and then element-wise applied with a Tanh activation function. Second, the resulting tensor is projected further using the second parameter matrix $\mathbb{W}_{t,2}$ and then passed through a SoftMax activation function to obtain a probability distribution over the temporal nodes. This probability distribution is used to construct the temporal attention adjacency matrix $\hat{\mathbb{A}}_t$ for the temporal correlated graph $\mathbb{G}_t$. With the obtained adjacency matrix, a spectral graph convolutional operation performed on $\mathbb{G}_t$ is then applied to the sampled condition monitoring data. Such a operation aggregates information from the temporal nodes and their neighboring nodes in the graph to capture the temporal correlations of condition monitoring data.

Likewise, the construction process of the feature attention adjacency matrix $\hat{\mathbb{A}}_f$ is similar. First, the sampled data $\mathbf{X}_{i,\omega}$ is projected into a higher dimensional space using $\mathbb{W}_{f,1}$ and then element-wise applied with a Tanh activation function. Second, the resulting tensor is projected further using $\mathbb{W}_{f,2}$ and then passed through a SoftMax activation function to obtain a probability distribution over the temporal nodes. This probability distribution is used to construct the feature attention adjacency matrix $\hat{\mathbb{A}}_f$ for the feature-correlated graph $\mathbb{G}_f$. Next, the spectral graph convolutional operation performed on $\mathbb{G}_f$ is applied to the sampled condition monitoring data. The resulting tensors of the proposed attention-awared spectral graph convolutional operation can be mathematically summarized as Eq. (21), where $\mathbb{C}_{i,\omega}$ denotes

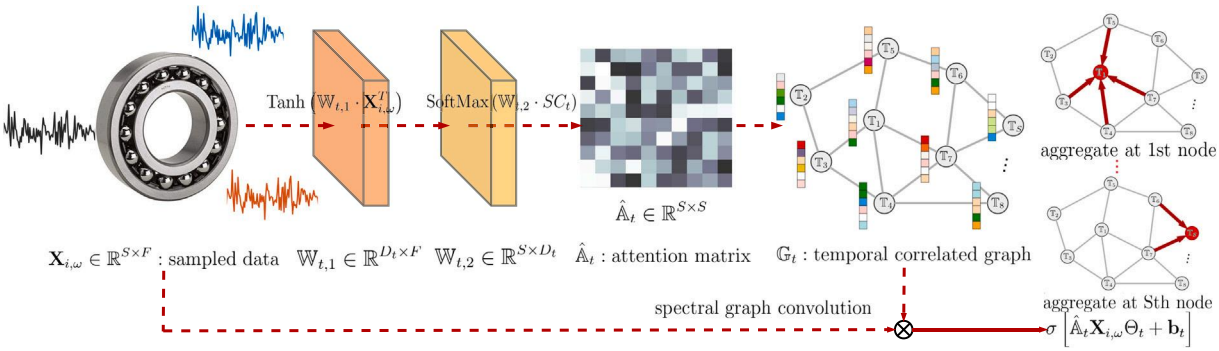


Fig. 2. The framework of generating the temporal attention adjacency matrix $\hat{\mathbb{A}}_t$.

the resulting tensor for bearing unit i and ωth sampled condition monitoring data.

$$\mathbb{C}_{i,\omega} := g\left(*_{[\mathbb{G}_t,\mathbb{G}_f]}\right)\mathbf{X}_{i,\omega} = \sigma\left[\hat{\mathbb{A}}_t\mathbf{X}_{i,\omega}\Theta_t + \mathbf{b}_t,\ \hat{\mathbb{A}}_f\mathbf{X}_{i,\omega}^T\Theta_f + \mathbf{b}_f\right] \tag{21}$$

Performing the attention-awared spectral graph convolutional operation on both graphs $\mathbb{G}_t$ and $\mathbb{G}_f$ using the same sampled data $\mathbf{X}_{i,\omega}$ can result in redundant and overlapping information. Therefore, directly utilizing the concatenated matrix $\mathbb{C}_{i,\omega}$ for RUL predictions of bearings may lead to degraded prediction performance. To address this issue, we implement the multi-head attention mechanism to select the most pertinent information from the concatenated matrix $\mathbb{C}_{i,\omega}$ for predictions. The advantages of the multi-head attention mechanism is two-fold: First, similar to the self-attention mechanism mentioned above, it can select the most relevant information from the condition monitoring data to make predictions; Second, it can project $\mathbb{C}_{i,\omega}$ into a higher dimensional space, allowing for reduction of redundant and overlapping information in a high dimensional space. In the multi-head attention mechanism, the concatenated matrix $\mathbb{C}_{i,\omega}$ is transformed by multiple parallel linear projections to create multiple representations of the input, which is expressed as Eq. (22).

$$\left(\mathbb{Q}_{i,\omega}^{(h)}, \mathbb{K}_{i,\omega}^{(h)}, \mathbb{V}_{i,\omega}^{(h)}\right) = \mathbb{C}_{i,\omega} \cdot \left(\mathbb{W}_Q^{(h)}, \mathbb{W}_K^{(h)}, \mathbb{W}_V^{(h)}\right) \tag{22}$$

In Eq. (22), $\mathbb{W}_Q^{(h)}$, $\mathbb{W}_K^{(h)}$, and $\mathbb{W}_V^{(h)}$ respectively refers to the weight matrices in the hth head to obtain the corresponding query $\mathbb{Q}_{i,\omega}^{(h)}$, key $\mathbb{K}_{i,\omega}^{(h)}$, and higher dimensional value $\mathbb{V}_{i,\omega}^{(h)}$. Next, these multiple representations of the input are used to obtain the final output $\mathbf{O}_{i,\omega}^{(h)}$ for bearing unit i, head h, and ωth sampled feature, which can be mathematically represented as Eq. (23), where d refers to the dimension of the weight matrices.

$$\mathbf{O}_{i,\omega}^{(h)} = \text{SoftMax}\left(\mathbb{Q}_{i,\omega}^{(h)} \cdot \mathbb{K}_{i,\omega}^{(h)T}/\sqrt{d}\right)\mathbb{V}_{i,\omega}^{(h)} \tag{23}$$

Next, the $\mathbf{O}_{i,\omega}^{(h)}$ in each head of the multi-head attention mechanism is concatenated, which can be written as Eq. (24), where H denotes the number of heads and $\bigcup$ denotes the concatenation operation.

$$\mathbf{O}_{i,\omega} = \bigcup_{h=1}^{H}\left\{\text{SoftMax}\left(\mathbb{Q}_{i,\omega}^{(h)} \cdot \mathbb{K}_{i,\omega}^{(h)T}/\sqrt{d}\right)\mathbb{V}_{i,\omega}^{(h)}\right\} \tag{24}$$

Finally, the $\mathbf{O}_{i,\omega}$ is flatten and transferred into a fully connected (FC) layer for final predictions, which can be represented as Eq. (25), where $\mathbb{W}$ is the kernel matrix in the FC layer and $\mathbf{b}$ is the bias vector in the FC layer, and $\hat{y}_{i,\omega}$ is the estimated RUL of bearing i at the time $\omega + S - 1$.

$$\hat{y}_{i,\omega} = \sigma\left(\mathbb{W} \cdot \text{Flatten}\left(\mathbf{O}_{i,\omega}\right) + \mathbf{b}\right) \tag{25}$$

The training loss ℓ for all training units can be written as Eq. (26), where the l_2 norm regularization terms are added for the parameter matrix Θ_t in the graph filter g_t and the parameter matrix Θ_f in the graph filter g_f to avoid overfitting.

$$\ell = \frac{1}{\sum_{i=1}^{n}\Omega_i}\sum_{i=1}^{n}\sum_{\omega=1}^{\Omega_i}(y_{i,\omega} - \hat{y}_{i,\omega})^2 + \lambda\left(\|\Theta_t\|_2^2 + \|\Theta_t\|_2^2\right) \tag{26}$$

In Eq. (26), λ represents the penalty hyperparameter for the l_2 norm regularization terms. Ω_i denotes the amount of windows for bearing i, while $y_{i,\omega}$ represents the true RUL of bearing unit i at time $\omega + S - 1$. Here, n refers to the number of bearing units. To train the proposed model, we adopt the Adam optimizer with a learning rate of α. In every iteration of training, we update the multi-head attention mechanism parameters, followed by the spectral graph convolutional operation parameters, and conclude by updating the parameters in the graph construction model that uses the self-attention mechanism.

2.3. Computational framework for RUL predictions for bearings

Fig. 3 shows the computation framework of the proposed attention-awared spectral graph convolutional operation, which includes feature extraction and sampling, building graphs with self-attention mechanism, spectral graph convolutional operation for temporal-correlated and feature-correlated graphs, and multi-head attention-based RUL predictions for bearings. First of all, features in the time and frequency domains are extracted from condition monitoring data collected from bearings, and these extracted features are sampled using a sliding window of size S. The extracted features in the time domain consist of basic statistical features, while the features in the frequency domain are extracted using the fast Fourier transform. Further details regarding the extracted features in both the time and frequency domains can be found in Section 3.2. The resulting feature matrices $\mathbf{X}_{i,\omega} \in \mathbb{R}^{S\times F}$ are fed into a self-attention mechanism to obtain temporal and feature attention adjacency matrices, denoted as $\hat{\mathbb{A}}_t \in \mathbb{R}^{S\times S}$ and $\hat{\mathbb{A}}_f \in \mathbb{R}^{F\times F}$, respectively. These matrices are used to generate temporal and feature correlated graphs $\mathbb{G}_t$ and $\mathbb{G}_f$, respectively. The spectral graph convolutional operation is then used to process these graphs and obtain a tensor $\mathbb{C}_{i,\omega}$. This tensor is fed into a multi-head attention mechanism to project it into a higher dimensional space and select the most useful information. The resulting tensor of each head are concatenated as $\mathbf{O}_{i,\omega}$, then flattened and transferred into a fully connected (FC) layer for RUL estimations.

The attention-awared graph convolutional network usually stacks multiple attention-awared spectral graph convolutional layers. Table 1 displays the pseudo-code used for training the proposed predictive model with multiple layers, where L represents the number of layers and H represents the number of heads. A total of F features are extracted from both time and frequency domains. These features are then sampled using a sliding window of size S to obtain $\mathbf{X}_{i,\omega} \in \mathbb{R}^{S\times F}$ for all bearing units i and windows ω. In each layer of the forward propagation process, the attention-awared spectral graph convolution layer l generates the temporal attention matrix and the feature attention matrix. The corresponding temporal-correlated and feature correlated-graphs are then constructed, and the final matrix $\mathbb{C}_{i,\omega}$ is obtained. The multi-head attention mechanism is then used to obtain $\mathbf{O}_{i,\omega}^{(h)}$ in each head, which are concatenated into a single matrix $\mathbf{O}_{i,\omega}$. Next, $\mathbf{O}_{i,\omega}$ is flattened and fed into a FC layer to obtain the predicted RUL $\hat{y}_{i,\omega}$ for all i and ω. The

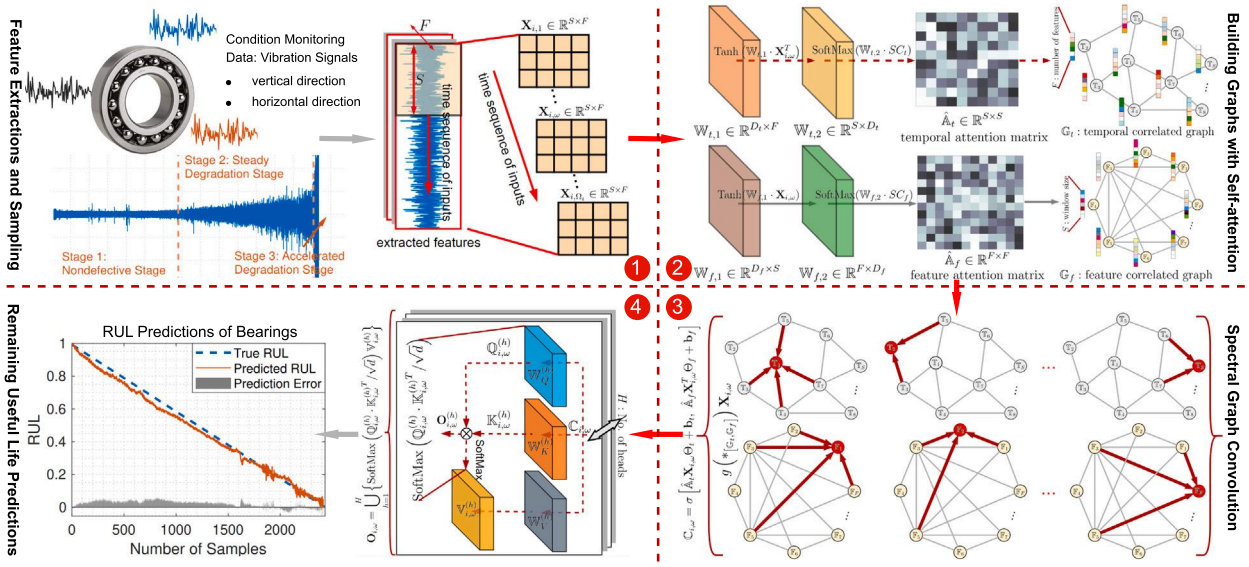


Fig. 3. The computation framework of the proposed attention-awared graph convolutional operation for temporal-correlated and feature-correlated graphs, including feature extraction and sampling, building graphs with self-attention mechanism, spectral graph convolutional operation for both graphs, and multi-head attention-based RUL predictions for bearings.

Table 1
The pseudo-code for training the proposed attention-awared graph convolutional network with multiple layers.

1. Extract features in time and frequency domain, where number of features is denoted as F
2. Sample the extracted features with a sliding window of size S to obtain $\mathbf{X}_{i,\omega} \in \mathbb{R}^{S\times F}$ for all i and ω
3. **For** iteration=1, …, I, **do**
   - 3.1. Initialize $\mathbf{H}_{i,\omega}^{(l)} = \mathbf{H}_{i,\omega}^{(f)} = \mathbf{X}_{i,\omega}$ for all bearing unit i and window ω, learning rate α
   - 3.2. **For** layer l=1, …, L, **do**
     - Generate temporal attention matrix $\hat{\mathbb{A}}_t^{(l)}$ in layer $l \Leftarrow \text{SoftMax}\left(\mathbb{W}_{t,2}^{(l)} \cdot \text{Tanh}\left(\mathbb{W}_{t,1}^{(l)} \cdot \text{transpose}\left(\mathbf{H}_{i,\omega}^{(l)}\right)\right)\right)$
     - Based on the temporal attention matrix, temporal correlated graph $\mathbb{G}_t^{(l)}$ in layer l is constructed
     - Generate feature attention matrix $\hat{\mathbb{A}}_f^{(l)}$ in layer $l \Leftarrow \text{SoftMax}\left(\mathbb{W}_{f,2}^{(l)} \cdot \text{Tanh}\left(\mathbb{W}_{f,1}^{(l)} \cdot \left(\mathbf{H}_{i,\omega}^{(f)}\right)\right)\right)$
     - Based on the feature attention matrix, feature correlated graph $\mathbb{G}_f^{(l)}$ in layer l is constructed
     - Reset $\mathbf{H}_{i,\omega}^{(t)} \Leftarrow \sigma\left(\hat{\mathbb{A}}_t^{(l)} \cdot \mathbf{H}_{i,\omega}^{(l)}\Theta_t^{(l)} + \mathbf{b}_t^{(l)}\right)$ and reset $\mathbf{H}_{i,\omega}^{(f)} \Leftarrow \sigma\left(\hat{\mathbb{A}}_f^{(l)} \cdot \text{transpose}(\mathbf{H}_{i,\omega}^{(f)})\Theta_f^{(l)} + \mathbf{b}_f^{(l)}\right)$
   - 3.3. **End** iteration and concatenate to obtain $\mathbb{C}_{i,\omega}$
   - 3.4. **For** head h=1, …, H, **do**
     - Obtain $\mathbb{Q}_{i,\omega}^{(h)}, \mathbb{K}_{i,\omega}^{(h)}, \mathbb{V}_{i,\omega}^{(h)} \Leftarrow \mathbb{C}_{i,\omega} \cdot \left(\mathbb{W}_Q^{(h)}, \mathbb{W}_K^{(h)}, \mathbb{W}_V^{(h)}\right)$, and output $\mathbf{O}_{i,\omega}^{(h)} \Leftarrow \text{SoftMax}\left(\mathbb{Q}_{i,\omega}^{(h)} \cdot \mathbb{K}_{i,\omega}^{(h)T}/\sqrt{d}\right)\mathbb{V}_{i,\omega}^{(h)}$
   - 3.5. **End** iteration and concatenate to obtain $\mathbf{O}_{i,\omega} \Leftarrow \bigcup_{h=1}^{H}\left\{\text{SoftMax}\left(\mathbb{Q}_{i,\omega}^{(h)} \cdot \mathbb{K}_{i,\omega}^{(h)T}/\sqrt{d}\right)\mathbb{V}_{i,\omega}^{(h)}\right\}$
   - 3.6. Flatten and feed $\mathbf{O}_{i,\omega}$ into a fully connected layer to obtain the predicted RUL $\hat{y}_{i,\omega} \forall i, \omega$, and calculate loss ℓ
   - 3.7. For all head h, update $\mathbb{Q}_{i,\omega}^{(h)}, \mathbb{K}_{i,\omega}^{(h)}, \mathbb{V}_{i,\omega}^{(h)} \Leftarrow \mathbb{Q}_{i,\omega}^{(h)} - \alpha\left(\partial\ell/\partial\mathbb{Q}_{i,\omega}^{(h)}\right), \mathbb{K}_{i,\omega}^{(h)} - \alpha\left(\partial\ell/\partial\mathbb{K}_{i,\omega}^{(h)}\right), \mathbb{V}_{i,\omega}^{(h)} - \alpha\left(\partial\ell/\partial\mathbb{V}_{i,\omega}^{(h)}\right)$
   - 3.8. **For** layer $l = L$, …,1, **do**
     - Update $\Theta_t^{(l)}, \Theta_f^{(l)}, \mathbf{b}_t^{(l)}, \mathbf{b}_f^{(l)} \Leftarrow \Theta_t^{(l)} - \alpha\left(\partial\ell/\partial\Theta_t^{(l)}\right), \Theta_f^{(l)} - \alpha\left(\partial\ell/\partial\Theta_f^{(l)}\right), \mathbf{b}_t^{(l)} - \alpha\left(\partial\ell/\partial\mathbf{b}_t^{(l)}\right), \mathbf{b}_f^{(l)} - \alpha\left(\partial\ell/\partial\mathbf{b}_f^{(l)}\right)$
     - Update $\mathbb{W}_{t,1}^{(l)}, \mathbb{W}_{t,2}^{(l)} \Leftarrow \mathbb{W}_{t,1}^{(l)} - \alpha\left(\partial\ell/\partial\mathbb{W}_{t,1}^{(l)}\right), \mathbb{W}_{t,2}^{(l)} - \alpha\left(\partial\ell/\partial\mathbb{W}_{t,2}^{(l)}\right)$
     - Update $\mathbb{W}_{f,1}^{(l)}, \mathbb{W}_{f,2}^{(l)} \Leftarrow \mathbb{W}_{f,1}^{(l)} - \alpha\left(\partial\ell/\partial\mathbb{W}_{f,1}^{(l)}\right), \mathbb{W}_{f,2}^{(l)} - \alpha\left(\partial\ell/\partial\mathbb{W}_{f,2}^{(l)}\right)$
   - 3.9. **End** iteration and return updated parameters
4. **End** and **Return** the trained predictive model

training loss ℓ is calculated using this predicted RUL and the ground truth of RUL. Finally, the gradient descent method is used to update the parameters in each head of the multi-head attention mechanism and the parameters in each attention-awared graph convolutional layer.

3. Case study I: IEEE PHM bearing dataset

3.1. Dataset description

The proposed method's effectiveness was evaluated using the IEEE PHM 2012 Bearing dataset, which was collected using the FEMTO-ST Institute's PRONOSTIA platform [41]. This platform is specifically designed to accelerate rolling bearings' wear and tear, enabling the detection of faults within hours. The experimental setup consisted of a gearbox attached to a rotating motor, a pneumatic jack, and a regulator that controls pressure using digital electro-pneumatic technology, which are used to manage the speed and load-up pressure of the bearings. Fig. 4 illustrates the platform, along with normal and deteriorated bearings. To prevent damage, the run-to-failure experiments were halted if the bearings exceeded an acceleration rate of 20 g-forces. This dataset was collected under three different conditions, and Table 2 shows the operational conditions used in the IEEE PHM bearing dataset, as well as the bearing indices associated with different operating conditions. To thoroughly evaluate the performance of the proposed method on all bearing units, we conducted k-fold cross-validation on each set of bearings. Specifically, we employed 7-fold cross-validation for IEEE Bearing1_1 through IEEE Bearing1_7. This allowed us to demonstrate the accuracy of the proposed method in predicting the RUL of bearings under stable operating conditions, with each fold containing one bearing unit. Additionally, to showcase the proposed method's capability in predicting RUL accurately across different operating conditions, we performed 5-fold cross-validation on bearings operated under the second and third operating conditions. The distribution of bearings in each fold was as follows: the first

Table 2
The operational conditions used in the IEEE PHM bearing dataset, as well as the bearing indices associated with different operating conditions.

Condition	Angular velocity (rpm)	Radial force (kN)	Twisting force (N m)	Bearing indices
Condition 1	1800	4.0	1.326	IEEE Bearing1_1 to IEEE Bearing1_7
Condition 2	1650	4.2	1.447	IEEE Bearing2_1 to IEEE Bearing2_7
Condition 3	1500	5.0	1.591	IEEE Bearing3_1 to IEEE Bearing3_3

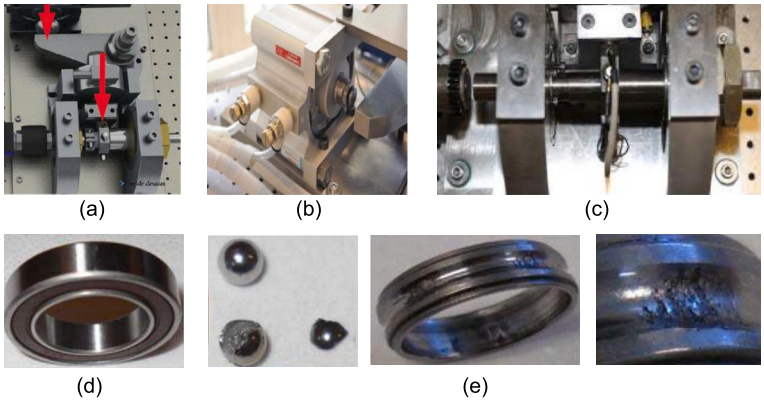


Fig. 4. (a) Force transmission; (b) Pneumatic jack; (c) Shaft support bearing; (d) Normal bearings; (e) Degraded bearings [41].

fold included IEEE Bearing2_1 and IEEE Bearing2_6; the second fold included IEEE Bearing2_2 and IEEE Bearing2_7; the third fold included IEEE Bearing2_3 and IEEE Bearing3_1; the fourth fold included IEEE Bearing2_4 and IEEE Bearing3_2; and the final fold included IEEE Bearing2_5 and IEEE Bearing3_3.

3.2. Detection of the degradation phase and features extraction

The approach of detecting different phases of degradation has been widely accepted as a means of improving the prediction performance of RUL for rotating bearings [14,42]. To identify these phases, we used an abrupt change point detection method, which is commonly employed for this purpose [43,44]. Fig. 5 illustrates the vibration signals for various bearings and the results of our phase detection. It can be observed that the number of detected phases differs among the bearing units. For instance, IEEE Bearing1_3 has three detected phases, including a non-defective phase, a steady degradation phase, and an accelerated degradation phase. In contrast, IEEE Bearing1_2 and IEEE Bearing2_7 only exhibit two phases, comprising a non-defective phase and an accelerated degradation phase. As some bearings do not manifest a steady degradation phase, we trained one predictive model to estimate the RUL for both the non-defective and steady degradation phases, and another predictive model to estimate the RUL for the accelerated degradation phase. More specifically, during the training phase, the condition monitoring data collected from the bearing units was utilized to apply the abrupt change point detection method, which aimed to identify any significant changes in the data. Based on these detected change points, the condition monitoring data was divided into two parts. The first part was used to train one predictive model to estimate the RUL for both non-defective and steady degradation phases. The second part, on the other hand, was used to train a separate predictive model to estimate the RUL for the accelerated degradation phase. If only one change point was detected, the condition monitoring data before that change point was considered as the first part, while the data after the change point was considered as the second part. However, if two or more change points were detected, the condition monitoring data before the second detected change point served as the first part, and the data after the second change point constituted the second part. Moreover, during the training phase, the abrupt change point detection method also recorded the maximum increasing rate of RMS in the first part of the data. This recorded rate was later used to classify the degradation phases to which a test bearing unit belongs. In the test phase, the condition monitoring data collected from the bearing units used for testing was continuously fed into the abrupt change point detection method in real time. Depending on the detection results, the data were processed accordingly. If no change point was detected in the current condition monitoring data, it was fed into the first trained predictive model for RUL predictions. Similarly, if change points were detected, but the increasing rate of RMS was lower than the maximum rate obtained during the training phase, the data was still fed into the first predictive model for RUL predictions. However, if change points were detected, and the increasing rate of RMS exceeded the maximum rate from the training phase, the data was fed into the second trained predictive model for RUL predictions. More specific and similar degradation stage detection methods can also be found in [14].

Next, we extracted 20 features from the signals collected in both the horizontal and vertical directions. These features comprised 12 in the time-domain and 8 in the frequency-domain. In the time-domain, the 12 features included basic statistical measures such as maximum, minimum, average, standard deviation, root mean square, kurtosis, skewness, peak-to-peak value, variance, entropy, standard deviation of inverse sine, and standard deviation of inverse tangent. In the frequency-domain, the 8 features were extracted using fast Fourier transform. These features included mean frequency, median frequency, band power, occupied bandwidth, power bandwidth, maximum power spectral density, maximum amplitude, and frequency of maximum amplitude. These specific features were chosen because their effectiveness has been demonstrated in predicting the RUL of bearings [14,36]. To enhance the monotonicity of the extracted features, we employed a cumulative sum function, which has been previously proven to be effective [36,45]. The cumulative sum function can be represented as Eq. (27), where $x_{i,\omega,f}$ denotes the fth extracted feature from the ωth sampling window for bearing i, and $C_{x_{i,\Omega,f}}$ denotes the fth cumulative feature for the ωth sampling window and bearing i.

$$C_{x_{i,\Omega,f}} = \sum_{\omega=1}^{\Omega} x_{i,\omega,f} \Big/ \left| \sum_{\omega=1}^{\Omega} x_{i,\omega,f} \right|^{1/2} \tag{27}$$

In summary, 20 features were extracted from both the time and frequency domains, separately for both the horizontal and vertical directions, resulting in a total of 40 features. Additionally, a cumulative function was applied to each of the extracted features, resulting in another 40 cumulative features with improved monotonicity. Therefore, a total of 80 features were used for predicting the RUL of bearings.

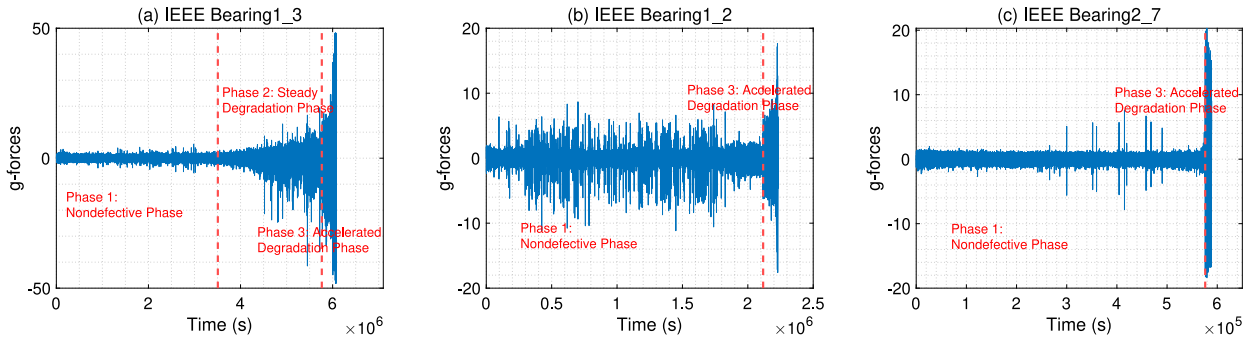


Fig. 5. The vibration signals for IEEE Bearing1_2, IEEE Bearing1_3, and IEEE Bearing2_7, and the results of the phase detection.

3.3. Hyperparameters

The proposed predictive model involves various hyperparameters, including the batch size, learning rate (α), sliding window size (S), number of extracted features (F), dimensions (D_t and D_f) of trainable matrices $\mathbb{W}_{t,1}$, $\mathbb{W}_{t,2}$, $\mathbb{W}_{f,1}$, and $\mathbb{W}_{f,2}$ for generating temporal and feature attention matrices, size of projection matrices (Θ_t and Θ_f), number of attention heads (H), dimension of weight matrices of the multi-head attention mechanism (d), activation function (σ), and number of attention-awared graph convolutional layer. To optimize the prediction performance and reduce computational cost simultaneously, we set the batch size to 20, the number of features to 80, and the learning rate to 1×10^{-4}. The dimensions of trainable matrices $\mathbb{W}_{t,1}$, $\mathbb{W}_{t,2}$, $\mathbb{W}_{f,1}$, and $\mathbb{W}_{f,2}$ for generating temporal and feature attention matrices were set to $D_t = D_f = 100$, and the sizes of the projection matrices Θ_t and Θ_f were set to 100. We used one attention head ($H = 1$) and the dimension of weight matrices of the multi-head attention mechanism was set to $d = 10$. In addition to the hyperparameters mentioned earlier, the window size (S) also significantly impacts both prediction accuracy and computational time. The window size determines the historical condition monitoring data collected over a specific period, which is used for predicting the RUL of bearings at different time points. A larger window size may improve prediction performance but comes with a considerable increase in computational cost. Conversely, a smaller window size can reduce computational cost but may result in a decrease in prediction performance. To determine the most suitable window size, we employed the grid search method and explored values ranging from 5 to 200. The results indicated that increasing the window size from 5 to 20 resulted in improved prediction performance. However, further increasing the window size from 20 to 200 did not lead to a significant change in prediction performance. Based on these findings, we aimed to optimize prediction performance while minimizing computational costs, and thus, we set the window size to 20. Additionally, we used the Rectified Linear Unit (ReLU) activation function for the attention-awared graph convolutional layers and a linear activation function for the FC layer, and we performed one temporal spectral graph convolution operation and one feature spectral graph convolution operation in this case study. In addition, we applied an l_2 norm penalty of 0.1 to the proposed predictive model to mitigate the risk of overfitting.

3.4. Temporal attention and feature attention graphs

Fig. 6(a) and (b) respectively show the temporal attention adjacency matrix $\hat{\mathbb{A}}_t$ and the feature attention adjacency matrix $\hat{\mathbb{A}}_f$ generated by the proposed attention-awared graph convolutional operation. Fig. 6(c) and (d) respectively show the scaled temporal adjacency matrix $\hat{\mathbf{A}}_t$ and the scaled feature adjacency matrix $\hat{\mathbf{A}}_f$ generated by the cosine similarity and covariance, which is a traditional approach reported in the literature. Based on the figures, we observe that the automatically generated temporal and feature attention adjacency matrices $\hat{\mathbb{A}}_t$ and $\hat{\mathbb{A}}_f$ differ from the scaled temporal and feature adjacency matrices $\hat{\mathbf{A}}_t$ and $\hat{\mathbf{A}}_f$, respectively. These differences can be observed in two ways. Firstly, the temporal and feature attention adjacency matrices exhibit more sparsity than the scaled temporal and feature adjacency matrices. Secondly, the temporal and feature attention adjacency matrices are asymmetric, while the scaled temporal and feature adjacency matrices are symmetric. The increased sparsity of the attention matrices indicates that the attention mechanism prioritizes only the most relevant features for making predictions, leading to more accurate and efficient predictions. The asymmetric property of the attention matrices enables imbalanced aggregation between adjacent nodes in both the temporal-correlated and feature-correlated graphs, allowing each node to automatically aggregate the most critical portion of condition monitoring data and filter out noise, resulting in improved prediction accuracy.

3.5. RUL prediction results

Note that we rescale the RUL of bearings to a range of 0 to 1 for comparison purposes, as the majority of studies that used this dataset reported in the literature also adopted this rescaling method. Fig. 7 shows the RUL prediction results for some of the bearing units. The proposed attention-awared graph convolutional network is capable of predicting the RUL with high accuracy for certain bearing units. For IEEE Bearing1_5 and Bearing1_6, the predicted RUL trajectories are in a good agreement with the ground truth of RUL. For some bearing units, some relatively large deviations between the predicted RUL and the ground truth are observed, however, the predictive model is still able to accurately track the overall trend. There are two potential causes of prediction errors and fluctuations. Firstly, bearing degradation is influenced by multiple stressors, which can all impact the rate of degradation. These stressors are often challenging to account for, resulting in differences between the predicted and actual RUL. Secondly, the condition monitoring data utilized to train the RUL prediction model may include some degree of noise or variability. Minor modifications to the input data can result in slightly different output predictions, leading to fluctuations in the RUL predictions.

In order to evaluate the efficacy of our proposed attention-awared graph convolutional network for temporal-correlated and feature-correlated graphs (AGCN-TF), we conducted an ablation study against other methods listed in Table 3. In Table 3, AGCN-TF is the name we use to refer to our proposed model; AGCN-T is the attention-awared GCN for temporal-correlated graphs without using the self-attention based feature-correlated graph; AGCN-F is the attention-awared GCN for feature-correlated graphs without using the self-attention based temporal-correlated graph; GCN-TF is the GCN with using the traditional temporal-correlated and feature-correlated graphs; GCN-T is a GCN with only using the traditional temporal-correlated graph; and GCN-F is a GCN with only using the traditional feature-correlated graph.

Table 4 summarizes the RMSE of RUL predictions for all bearings in the IEEE PHM dataset, employing the methods listed in Table 3.

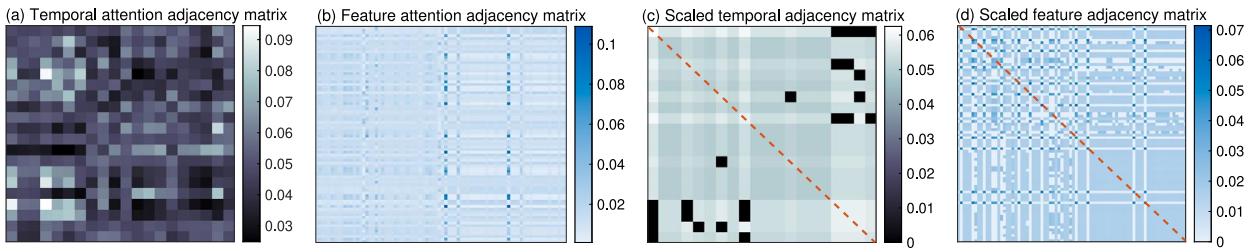


Fig. 6. (a) The temporal attention adjacency matrix $\hat{\mathbb{A}}_t$, (b) the feature adjacency attention matrix $\hat{\mathbb{A}}_f$, (c) the scaled temporal adjacency matrix $\hat{\mathbf{A}}_t$, (d) the scaled feature adjacency matrix $\hat{\mathbf{A}}_f$.

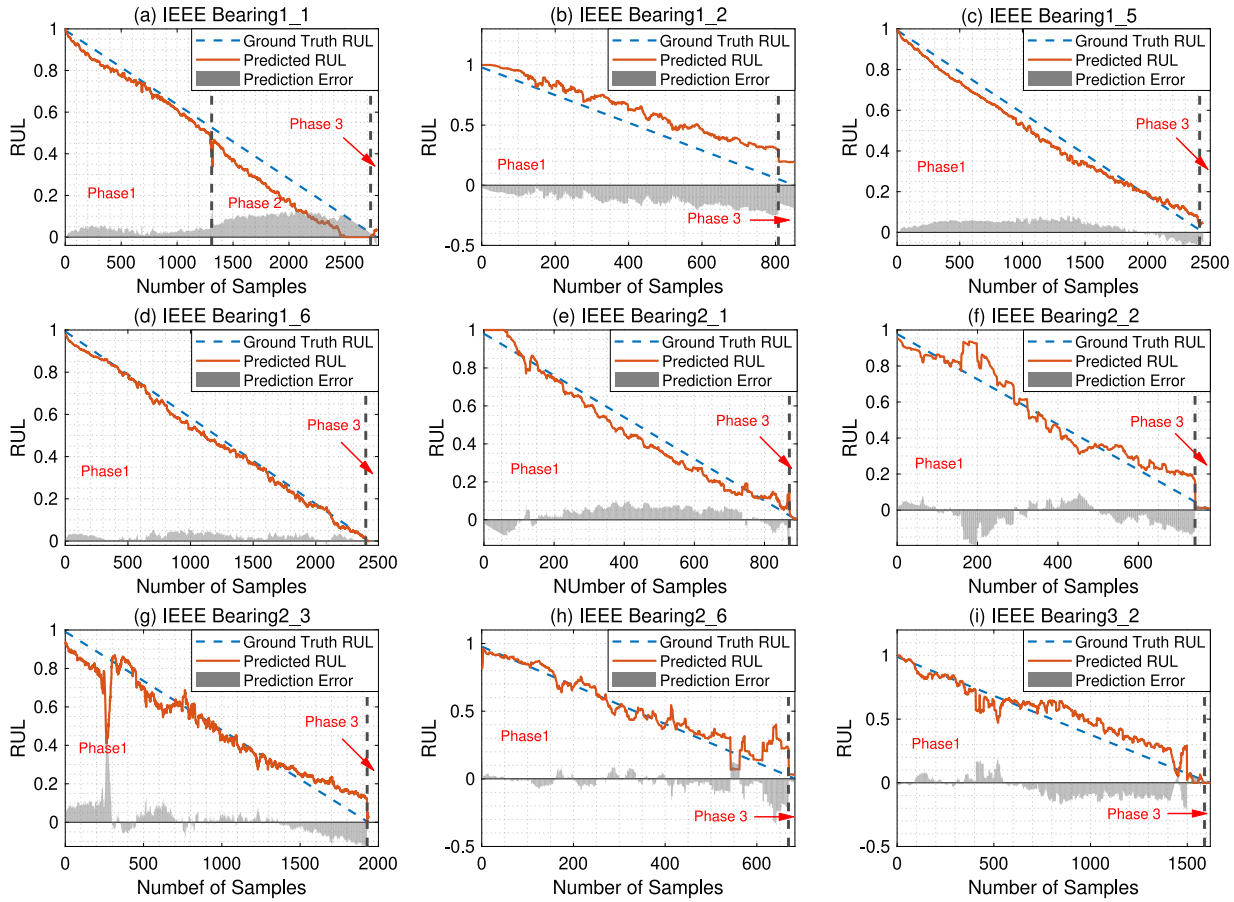


Fig. 7. The RUL prediction results for a selection of bearing units from the IEEE PHM bearing dataset.

Table 3
Abbreviations and descriptions of the proposed method and other comparative methods.

Abbreviations	Description
AGCN-TF	Attention-awared GCN for both graphs (proposed)
AGCN-T	Attention-awared GCN for temporal-correlated graphs
AGCN-F	Attention-awared GCN for feature-correlated graph
GCN-TF	GCN with only using the traditional temporal-correlated graph
GCN-T	GCN with only using the traditional temporal-correlated graph
GCN-F	GCN with only using the traditional feature-correlated graph

The proposed attention-awared graph convolutional network for both graphs exhibits superior performance compared to other methods listed in Table 3. For example, in the case of IEEE Bearing2_2, the proposed method attains an RMSE of 0.072 for RUL prediction, whereas the other methods shows RMSE values that varies between 0.101 to 0.147. Furthermore, for bearings operating under three different conditions, the proposed method displays an average prediction RMSE of 0.122, while the other methods demonstrates an average prediction RMSE ranging from 0.123 to 0.156.

To evaluate the effectiveness of the proposed approach, the proposed AGCN-TF model was compared with various deep learning techniques that have been reported in the literature in recent years. The performance of the proposed AGCN-TF, as well as the methods listed in Table 3 and other deep learning techniques, was evaluated by calculating the average prediction RMSE for bearings under three different operating conditions, as presented in Table 5. The deep learning techniques reflected in Table 5 include C-LSTM (convolutional LSTM), CNN (convolutional neural network), DAN (deep adversarial network), GAN (generative adversarial network), and TGRU (transferable bidirectional GRU). The results indicate that the proposed AGCN-TF model outperforms both the methods listed in Table 3 and other deep learning techniques reported in the literature, regardless of the operating conditions. For instance, when considering bearings operating under condition 1, the average prediction RMSE of the proposed model is

Table 4
The RMSE of RUL predictions for all bearings in the IEEE PHM bearing dataset.

Operating condition	Bearing index	AGCN-TF	AGCN-T	AGCN-F	GCN-TF	GCN-T	GCN-F
Condition 1	IEEE Bearing1_1	0.073	0.086	0.120	0.088	0.067	0.096
	IEEE Bearing1_2	0.141	0.172	0.146	0.205	0.141	0.132
	IEEE Bearing1_3	0.062	0.080	0.131	0.080	0.076	0.036
	IEEE Bearing1_4	0.141	0.203	0.108	0.150	0.230	0.092
	IEEE Bearing1_5	0.048	0.058	0.079	0.041	0.061	0.052
	IEEE Bearing1_6	0.026	0.045	0.028	0.018	0.037	0.038
	IEEE Bearing1_7	0.116	0.124	0.117	0.124	0.126	0.131
Condition 2	IEEE Bearing2_1	0.055	0.062	0.079	0.040	0.031	0.091
	IEEE Bearing2_2	0.072	0.101	0.147	0.109	0.132	0.123
	IEEE Bearing2_3	0.076	0.071	0.207	0.073	0.084	0.117
	IEEE Bearing2_4	0.257	0.272	0.201	0.218	0.245	0.142
	IEEE Bearing2_5	0.246	0.280	0.092	0.272	0.246	0.245
	IEEE Bearing2_6	0.079	0.103	0.218	0.072	0.048	0.076
	IEEE Bearing2_7	0.253	0.263	0.273	0.275	0.248	0.246
Condition 3	IEEE Bearing3_1	0.184	0.246	0.165	0.365	0.340	0.286
	IEEE Bearing3_2	0.089	0.177	0.298	0.113	0.160	0.106
	IEEE Bearing3_3	0.148	0.163	0.250	0.126	0.137	0.085
	Average	**0.122**	0.147	0.156	0.139	0.142	0.123

Table 5
The average prediction RMSE of the proposed AGCN-TF and other deep learning methods reported in the literature.

Condition	AGCN-TF	AGCN-T	AGCN-F	GCN-TF	CLSTM [46]	CNN [47]	DAN [48]	GAN [29]	TGRU [49]
1	**0.087**	0.110	0.104	0.101	0.159	0.189	0.206	0.105	0.230
2	**0.148**	0.165	0.174	0.151	0.173	0.260	0.206	0.187	0.170
3	**0.140**	0.195	0.238	0.202	0.152	0.290	0.366	–	0.150

Table 6
Details about the operating conditions used in the XJTU-SY bearing dataset.

Condition	Angular velocity (rpm)	Radial force (kN)	Bearing index
Condition 1	2100	12	XJTU-SY Bearing1_1 to XJTU-SY Bearing1_5
Condition 2	2250	11	XJTU-SY Bearing2_1 to XJTU-SY Bearing2_5
Condition 3	2400	10	XJTU-SY Bearing3_1 to XJTU-SY Bearing3_5

0.087, whereas the average prediction RMSE of other techniques ranges from 0.105 to 0.230.

4. Case study II: XJTU-SY bearing dataset

4.1. Data description

The efficacy of the proposed method was verified in this case study using the XJTU-SY bearing dataset, which was jointly acquired by Xi'an Jiaotong University and Changxing Sumyoung Technology Company [50]. The dataset consists of vibration signals collected in two orientations from 15 LDK UER204 bearings, operating under three distinct conditions. To collect the condition monitoring data, two identical accelerometers were placed 90 degrees apart on the housing. The experiments ceased if the bearing acceleration exceeded 20 g-forces, similar to the protocol followed in the IEEE PHM bearing dataset. Further details on the experimental setup can be found in [50]. Fig. 8 shows the platform used to collect the dataset, including normal and degraded bearings with various types of failures. Table 6 provides information on the operating conditions. Similarly, in order to comprehensively evaluate the performance of the proposed method on all bearing units, we conducted k-fold cross-validation on bearings operated under both conditions 1 and 2. For bearings in each operating condition, a 5-fold cross-validation was performed, with each fold containing one bearing unit.

4.2. Detection of the degradation phase and features extraction

In this case study, we applied the identical approach described in Section 3.2 to detect various degradation phases. Fig. 9 displays the vibration signals of different bearings and the results of the phase detection. It is evident that the number of detected phases varies among different bearing units. For example, XJTU-SY Bearing1_1 and XJTU-SY Bearing2_5 have three detected phases, which include a non-defective phase, a steady degradation phase, and an accelerated degradation phase. On the other hand, XJTU-SY Bearing1_4 only exhibits two phases, comprising a non-defective phase and an accelerated degradation phase. As some bearings do not display a steady degradation phase, we trained one predictive model to estimate the RUL for both the non-defective and steady degradation phases, and another predictive model to estimate the RUL for the accelerated degradation phase. Similar to Section 3.2, 80 identical features, including cumulative features, were extracted and used for RUL predictions of bearings in this case study.

4.3. Hyperparameters

In this case study, to optimize the prediction performance and reduce computational cost simultaneously, we set the batch size to 20, the number of features to 80, the learning rate to 1×10^{-4}, and the window size to $S = 20$. The dimensions of trainable matrices $\mathbb{W}_{t,1}$, $\mathbb{W}_{t,2}$, $\mathbb{W}_{f,1}$, and $\mathbb{W}_{f,2}$ for generating temporal and feature attention matrices were set to $D_t = D_f = 100$, and the sizes of the projection matrices Θ_t and Θ_f were set to 100. We used one attention head ($H = 1$) and the dimension of weight matrices of the multi-head attention mechanism was set to $d = 10$. Additionally, we used the Rectified Linear Unit (ReLU) activation function for the attention-awared graph convolutional layers and a linear activation function for the FC layer, and we performed four temporal spectral graph convolution operations and two feature spectral graph convolution operations in this case study. In addition, we applied an l_2 norm penalty of 0.1 to the proposed predictive model to mitigate the risk of overfitting.

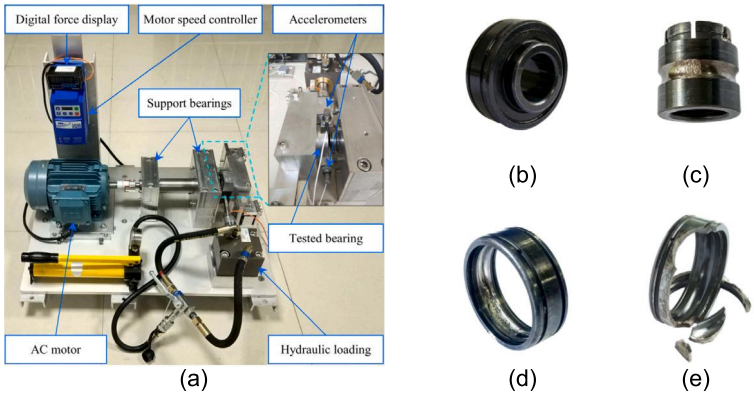


Fig. 8. (a) The experimental platform used in the XJTU-SY bearing dataset; (b) Normal bearings; (c) (d) (e) Degraded bearings [50].

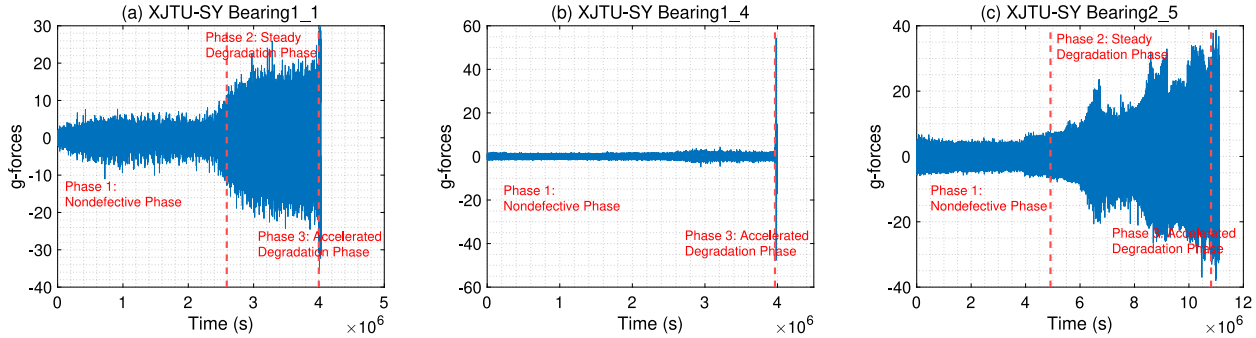


Fig. 9. The vibration signals for XJTU-SY Bearing1_1, XJTU-SY Bearing1_4, and XJTU-SY Bearing2_5, and the results of the phase detection.

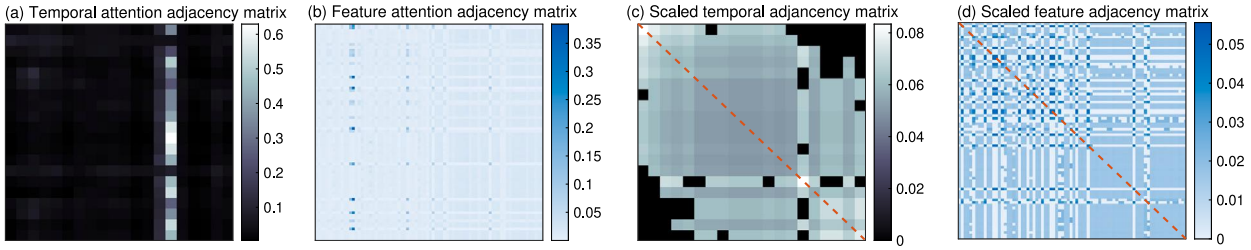


Fig. 10. (a) The temporal attention adjacency matrix $\hat{\mathbb{A}}_t$, (b) the feature adjacency attention matrix $\hat{\mathbb{A}}_f$, (c) the scaled temporal adjacency matrix $\hat{\mathbf{A}}_t$, (d) the scaled feature adjacency matrix $\hat{\mathbf{A}}_f$.

4.4. Temporal attention and feature attention graphs

Fig. 10(a) and (b) respectively show the temporal attention adjacency matrix $\hat{\mathbb{A}}_t$ and the feature attention adjacency matrix $\hat{\mathbb{A}}_f$ generated by the proposed attention-awared graph convolutional operation. Fig. 10(c) and (d) respectively show the scaled temporal adjacency matrix $\hat{\mathbf{A}}_t$ and the scaled feature adjacency matrix $\hat{\mathbf{A}}_f$ generated by the cosine similarity and covariance, which is a traditional approach reported in the literature. Based on the figures, we observe that the automatically generated temporal and feature attention adjacency matrices $\hat{\mathbb{A}}_t$ and $\hat{\mathbb{A}}_f$ differ from the scaled temporal and feature adjacency matrices $\hat{\mathbf{A}}_t$ and $\hat{\mathbf{A}}_f$, respectively. These differences can be observed in two aspects that are sparse property and symmetric property, more details about these differences and the benefit of the proposed temporal attention matrix and feature attention matrix have been discussed in Section 3.4.

4.5. RUL prediction results

Fig. 11 presents the RUL prediction results of several bearings in the XJTU-SY bearing dataset using the proposed attention-aware graph convolutional network. From the figure, it is evident that the proposed method can accurately predict the RUL of bearings. Although there are some gaps and fluctuations between the predicted and the ground truth of RUL for certain bearings, the predicted values still follow the degradation trend of the ground truth of RUL. It should be noted that the RUL prediction performance for bearings in the XJTU-SY dataset is not comparable to that of the IEEE PHM bearing dataset. There are two possible reasons for this discrepancy. Firstly, the XJTU-SY bearing dataset contains bearings with more failure modes and distinct failure locations, leading to more distinct degradation trajectories and increased stochasticity and randomness in RUL predictions. Secondly, the XJTU-SY dataset has fewer bearings compared to the IEEE PHM dataset, resulting in a smaller amount of training data being available for this case study.

Table 7 shows the prediction RMSE for XJTU-SY Bearing1_1 to XJTU-SY Bearing2_5 in the XJTU-SY dataset using the techniques listed in Table 3. The results demonstrate that the proposed attention-aware graph convolutional network outperforms the other methods in Table 3. For example, for XJTU-SY Bearing1_1, the proposed method achieved an RUL prediction RMSE of 0.076, while the other methods had RMSE values ranging from 0.083 to 0.135. Additionally, for all bearings

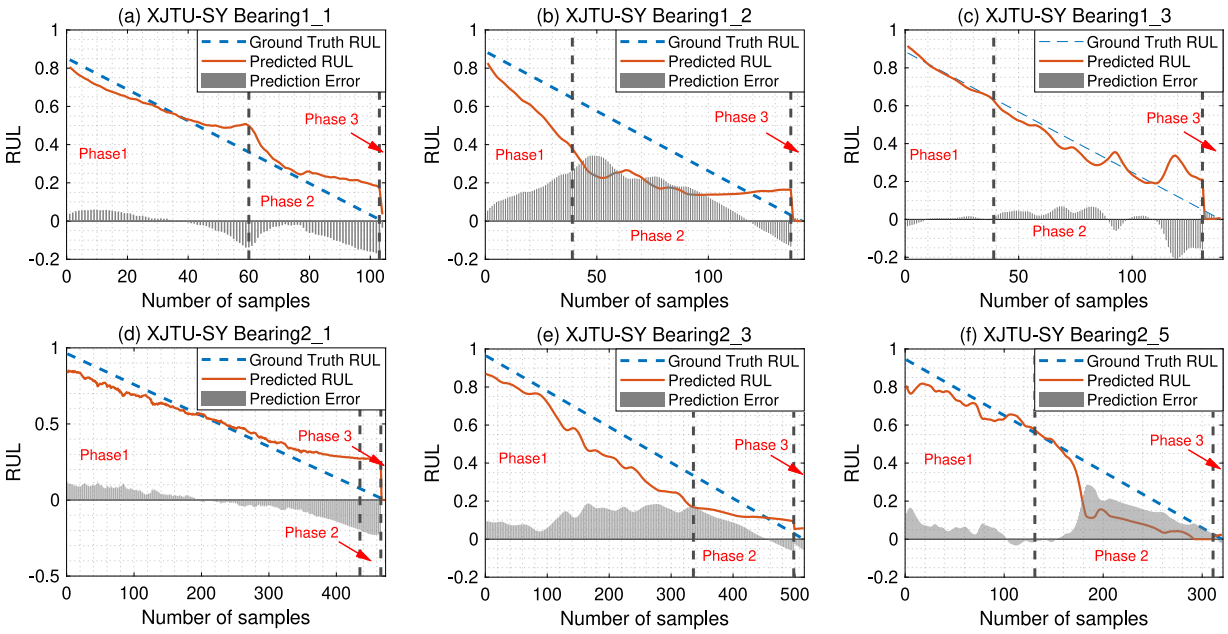


Fig. 11. The RUL prediction results for a selection of bearing units in the XJTU-SY bearing dataset.

Table 7
The RUL prediction RMSE for XJTU-SY Bearing1_1 to XJTU-SY Bearing2_5 in the XJTU-SY bearing dataset.

Operating condition	Bearing index	AGCN-TF	AGCN-T	AGCN-F	GCN-TF	GCN-T	GCN-F
Condition 1	XJTU-SY Bearing1_1	0.076	0.108	0.089	0.102	0.083	0.135
	XJTU-SY Bearing1_2	0.189	0.220	0.257	0.305	0.239	0.283
	XJTU-SY Bearing1_3	0.067	0.088	0.075	0.117	0.078	0.090
	XJTU-SY Bearing1_4	0.251	0.209	0.245	0.211	0.221	0.260
	XJTU-SY Bearing1_5	0.223	0.232	0.208	0.185	0.218	0.178
Condition 2	XJTU-SY Bearing2_1	0.097	0.184	0.110	0.121	0.189	0.169
	XJTU-SY Bearing2_2	0.180	0.261	0.202	0.189	0.253	0.167
	XJTU-SY Bearing2_3	0.120	0.149	0.115	0.126	0.137	0.131
	XJTU-SY Bearing2_4	0.272	0.336	0.311	0.322	0.339	0.307
	XJTU-SY Bearing2_5	0.117	0.149	0.092	0.092	0.110	0.078
	Average	**0.159**	0.194	0.170	0.177	0.187	0.180

Table 8
The average RMSE of the proposed A-BiGCN and other deep learning methods reported in the literature.

Condition	AGCN-TF	AGCN-T	AGCN-F	GCN-TF	DANN [48]	CNN-GRU [27]	LSTM [51]	SAGCN-SA [36]
Condition 1	**0.161**	0.171	0.175	0.184	0.297	0.191	0.264	0.166
Condition 2	**0.157**	0.216	0.166	0.170	0.240	0.184	0.346	0.213

operated under two different conditions, the proposed method had an average prediction RMSE of 0.159, compared to an average prediction RMSE ranging from 0.170 to 0.194 for the other methods.

To evaluate the effectiveness of our proposed method, we compared the proposed AGCN-TF model with various deep learning techniques reported in recent literature. Table 8 shows the average prediction RMSE for bearings operated under two different conditions using our proposed method and other deep learning methods from the literature. The deep learning methods included in the comparison were DANN (deep adversarial neural network), CNN-GRU (CNN with gated recurrent unit), LSTM (long short-term memory), and SAGCN-SA (self-adaptive GCN with self-attention mechanism). Based on the average RMSE values in Table 8, we concluded that the proposed AGCN-TF outperforms other methods in most cases, regardless of the operating conditions. For instance, when predicting the RUL of bearings operated under condition 2, our proposed model achieved an average RMSE of 0.157, while the average RMSE of other methods ranged from 0.166 to 0.346.

5. Conclusion and future work

In summary, we introduced a spectral graph convolutional operation that can handle both temporal-correlated and feature-correlated graphs. This operation allows one to take into account temporal and feature correlations simultaneously. We also introduced a self-attention mechanism to construct the temporal-correlated and feature-correlated graphs automatically during training so that the accuracy and robustness of the predictive model were significantly improved. Furthermore, we used a multi-head attention-based selection mechanism to automatically select the most important high-dimensional features generated by the attention-aware graph convolutional operation. We demonstrated the effectiveness of the method on two publicly available bearing datasets. The experimental results showed that our method achieved an average RMSE of 0.122 and 0.159 on the two datasets, respectively. Moreover, our method outperformed traditional GCNs and other deep learning methods such as convolutional LSTM, generative adversarial network, and self-adaptive GCN. In the future, we plan to investigate the effectiveness of our method on other condition monitoring datasets.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] Jaouher Ben Ali, Brigitte Chebel-Morello, Lotfi Saidi, Simon Malinowski, Farhat Fnaiech, Accurate bearing remaining useful life prediction based on Weibull distribution and artificial neural network, Mech. Syst. Signal Process. 56 (2015) 150–172.

[2] Jerome Antoni, R.B. Randall, Differential diagnosis of gear and bearing faults, J. Vib. Acoust. 124 (2) (2002) 165–171.

[3] Henrique Dias Machado de Azevedo, Alex Maurício Araújo, Nadège Bouchonneau, A review of wind turbine bearing condition monitoring: State of the art and challenges, Renew. Sustain. Energy Rev. 56 (2016) 368–379.

[4] Michael Scharfe, Thomas Roschke, Enrico Bindl, Daniel Blonski, Design and development of a compact magnetic bearing momentum wheel for micro and small satellites, 2001.

[5] Dong Wang, Kwok-Leung Tsui, Statistical modeling of bearing degradation signals, IEEE Trans. Reliab. 66 (4) (2017) 1331–1344.

[6] Joshua J. Jacobs, Nadim J. Hallab, Anastasia K. Skipor, Robert M. Urban, Metal degradation products: a cause for concern in metal-metal bearings? Clin. Orthop. Relat. Res.® 417 (2003) 139–147.

[7] Benjamin Gould, Nicholas Demas, Robert Erck, Maria Cinta Lorenzo-Martin, Oyelayo Ajayi, Aaron Greco, The effect of electrical current on premature failures and microstructural degradation in bearing steel, Int. J. Fatigue 145 (2021) 106078.

[8] Yuning Qian, Ruqiang Yan, Shijie Hu, Bearing degradation evaluation using recurrence quantification analysis and Kalman filter, IEEE Trans. Instrum. Meas. 63 (11) (2014) 2599–2610.

[9] Jinjiang Wang, Yuanyuan Liang, Yinghao Zheng, Robert X. Gao, Fengli Zhang, An integrated fault diagnosis and prognosis approach for predictive maintenance of wind turbine bearing with limited samples, Renew. Energy 145 (2020) 642–650.

[10] Zhixiong Li, Dazhong Wu, Chao Hu, Janis Terpenny, An ensemble learning-based prognostic approach with degradation-dependent weights for remaining useful life prediction, Reliab. Eng. Syst. Saf. 184 (2019) 110–122.

[11] Mingming Yan, Xingang Wang, Bingxiang Wang, Miaoxin Chang, Isyaku Muhammad, Bearing remaining useful life prediction using support vector machine and hybrid degradation tracking model, ISA Trans. 98 (2020) 471–482.

[12] Sylvester A. Aye, P.S. Heyns, An integrated Gaussian process regression for prediction of remaining useful life of slow speed bearings based on acoustic emission, Mech. Syst. Signal Process. 84 (2017) 485–498.

[13] Marcin Witczak, Bogdan Lipiec, Marcin Mrugalski, Ralf Stetter, A fuzzy logic approach to remaining useful life estimation of ball bearings, in: Advanced, Contemporary Control: Proceedings of KKA 2020—The 20th Polish Control Conference, Łódź, Poland, 2020, Springer, 2020, pp. 1411–1423.

[14] Junchuan Shi, Tianyu Yu, Kai Goebel, Dazhong Wu, Remaining useful life prediction of bearings using ensemble learning: The impact of diversity in base learners and features, J. Comput. Inf. Sci. Eng. 21 (2) (2021).

[15] L.A. Kumaraswamidhas, S.K. Laha, et al., Bearing degradation assessment and remaining useful life estimation based on Kullback-Leibler divergence and Gaussian processes regression, Measurement 174 (2021) 108948.

[16] Fei Huang, Alexandre Sava, Kondo H. Adjallah, Zhouhang Wang, Fuzzy model identification based on mixture distribution analysis for bearings remaining useful life estimation using small training data set, Mech. Syst. Signal Process. 148 (2021) 107173.

[17] Shen Zhang, Shibo Zhang, Bingnan Wang, Thomas G. Habetler, Deep learning algorithms for bearing fault diagnostics—A comprehensive review, IEEE Access 8 (2020) 29857–29881.

[18] Mengqi Miao, Jianbo Yu, Zhihong Zhao, A sparse domain adaption network for remaining useful life prediction of rolling bearings under different working conditions, Reliab. Eng. Syst. Saf. 219 (2022) 108259.

[19] Li Jiang, Tianao Zhang, Wei Lei, Kejia Zhuang, Yibing Li, A new convolutional dual-channel Transformer network with time window concatenation for remaining useful life prediction of rolling bearings, Adv. Eng. Inform. 56 (2023) 101966.

[20] Sheng Xiang, Yi Qin, Jun Luo, Huayan Pu, Baoping Tang, Multicellular LSTM-based deep learning model for aero-engine remaining useful life prediction, Reliab. Eng. Syst. Saf. 216 (2021) 107927.

[21] Jinglong Chen, Hongjie Jing, Yuanhong Chang, Qian Liu, Gated recurrent unit based recurrent neural network for remaining useful life prediction of nonlinear deterioration process, Reliab. Eng. Syst. Saf. 185 (2019) 372–382.

[22] Jianghong Zhou, Yi Qin, Dingliang Chen, Fuqiang Liu, Quan Qian, Remaining useful life prediction of bearings by a new reinforced memory GRU network, Adv. Eng. Inform. 53 (2022) 101682.

[23] Tian Han, Jiachen Pang, Andy C.C. Tan, Remaining useful life prediction of bearing based on stacked autoencoder and recurrent neural network, J. Manuf. Syst. 61 (2021) 576–591.

[24] Yupeng Wei, Dazhong Wu, Janis Terpenny, Learning the health index of complex systems using dynamic conditional variational autoencoders, Reliab. Eng. Syst. Saf. 216 (2021) 108004.

[25] Yupeng Wei, Dazhong Wu, Janis Terpenny, Constructing robust and reliable health indices and improving the accuracy of remaining useful life prediction, J. Nondestruct. Eval. Diagn. Progn. Eng. Syst. 5 (2) (2022) 021009.

[26] Jun Zhu, Nan Chen, Changqing Shen, A new data-driven transferable remaining useful life prediction approach for bearing under different working conditions, Mech. Syst. Signal Process. 139 (2020) 106602.

[27] Dechen Yao, Boyang Li, Hengchang Liu, Jianwei Yang, Limin Jia, Remaining useful life prediction of roller bearings based on improved 1D-CNN and simple recurrent unit, Measurement 175 (2021) 109166.

[28] Meng Ma, Zhu Mao, Deep-convolution-based LSTM network for remaining useful life prediction, IEEE Trans. Ind. Inform. 17 (3) (2020) 1658–1667.

[29] Sungho Suh, Paul Lukowicz, Yong Oh Lee, Generalized multiscale feature extraction for remaining useful life prediction of bearings with generative adversarial networks, Knowl.-Based Syst. 237 (2022) 107866.

[30] Tao Jing, Pai Zheng, Liqiao Xia, Tianyuan Liu, Transformer-based hierarchical latent space VAE for interpretable remaining useful life prediction, Adv. Eng. Inform. 54 (2022) 101781.

[31] Tianfu Li, Zhibin Zhao, Chuang Sun, Ruqiang Yan, Xuefeng Chen, Multireceptive field graph convolutional networks for machine fault diagnosis, IEEE Trans. Ind. Electron. 68 (12) (2020) 12739–12749.

[32] Yong Feng, Jinglong Chen, Zijun Liu, Haixin Lv, Jun Wang, Full graph autoencoder for one-class group anomaly detection of IIoT system, IEEE Internet Things J. 9 (21) (2022) 21886–21898.

[33] Tianfu Li, Zhibin Zhao, Chuang Sun, Ruqiang Yan, Xuefeng Chen, Hierarchical attention graph convolutional network to fuse multi-sensor signals for remaining useful life prediction, Reliab. Eng. Syst. Saf. 215 (2021) 107878.

[34] Xiaoyu Yang, Ying Zheng, Yong Zhang, David Shan-Hill Wong, Weidong Yang, Bearing remaining useful life prediction based on regression shapalet and graph neural network, IEEE Trans. Instrum. Meas. 71 (2022) 1–12.

[35] Zongliang Xie, Jinglong Chen, Yong Feng, Shuilong He, Semi-supervised multiscale attention-aware graph convolution network for intelligent fault diagnosis of machine under extremely-limited labeled samples, J. Manuf. Syst. 64 (2022) 561–577.

[36] Yupeng Wei, Dazhong Wu, Janis Terpenny, Bearing remaining useful life prediction using self-adaptive graph convolutional networks with self-attention mechanism, Mech. Syst. Signal Process. 188 (2023) 110010.

[37] Yupeng Wei, Dazhong Wu, Prediction of state of health and remaining useful life of lithium-ion battery using graph convolutional network with dual attention mechanisms, Reliab. Eng. Syst. Saf. 230 (2023) 108947.

[38] David K. Hammond, Pierre Vandergheynst, Rémi Gribonval, Wavelets on graphs via spectral graph theory, Appl. Comput. Harmon. Anal. 30 (2) (2011) 129–150.

[39] Jianpeng Cheng, Li Dong, Mirella Lapata, Long short-term memory-networks for machine reading, 2016, arXiv preprint arXiv:1601.06733.

[40] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, Yoshua Bengio, A structured self-attentive sentence embedding, 2017, arXiv preprint arXiv:1703.03130.

[41] Patrick Nectoux, Rafael Gouriveau, Kamal Medjaher, Emmanuel Ramasso, Brigitte Chebel-Morello, Noureddine Zerhouni, Christophe Varnier, PRONOSTIA: An experimental platform for bearings accelerated degradation tests, in: IEEE International Conference on Prognostics and Health Management, PHM'12, IEEE Catalog Number: CPF12PHM-CDR, 2012, pp. 1–8.

[42] Yupeng Wei, Dazhong Wu, Janis Terpenny, Robust incipient fault detection of complex systems using data fusion, IEEE Trans. Instrum. Meas. 69 (12) (2020) 9526–9534.

[43] Marc Lavielle, Using penalized contrasts for the change-point problem, Signal Process. 85 (8) (2005) 1501–1510.

[44] Rebecca Killick, Paul Fearnhead, Idris A. Eckley, Optimal detection of changepoints with a linear computational cost, J. Amer. Statist. Assoc. 107 (500) (2012) 1590–1598.

[45] Kamran Javed, Rafael Gouriveau, Noureddine Zerhouni, Patrick Nectoux, A feature extraction procedure based on trigonometric functions and cumulative descriptors to enhance prognostics modeling, in: 2013 Ieee Conference on Prognostics and Health Management (Phm), IEEE, 2013, pp. 1–7.

[46] Shaoke Wan, Xiaohu Li, Yanfei Zhang, Shijie Liu, Jun Hong, Dongfeng Wang, Bearing remaining useful life prediction with convolutional long short-term memory fusion networks, Reliab. Eng. Syst. Saf. 224 (2022) 108528.

[47] Xiang Li, Wei Zhang, Qian Ding, Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction, Reliab. Eng. Syst. Saf. 182 (2019) 208–218.

[48] Xiang Li, Wei Zhang, Hui Ma, Zhong Luo, Xu Li, Data alignments in machinery remaining useful life prediction using deep adversarial neural networks, Knowl.-Based Syst. 197 (2020) 105843.

[49] Yudong Cao, Minping Jia, Peng Ding, Yifei Ding, Transfer learning for remaining useful life prediction of multi-conditions bearings based on bidirectional-GRU network, Measurement 178 (2021) 109287.

[50] Biao Wang, Yaguo Lei, Naipeng Li, Ningbo Li, A hybrid prognostics approach for estimating remaining useful life of rolling element bearings, IEEE Trans. Reliab. 69 (1) (2018) 401–412.

[51] Dongdong Zhao, Liu Feng, A two-stage machine-learning-based prognostic approach for bearing remaining useful prediction problem, IAENG Int. J. Comput. Sci. 48 (4) (2021).