Intelligence & Robotics

Research Article

Open Access

Check for updates

Adaptive variational empirical mode decomposition aware intelligent data-driven modeling for complex industrial processes

Yujun Chen[1], Xiuli Zhu[1,2], Peng Wang[3], Kuangrong Hao[4], Kairui Sheng[1]

[1]Department of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China.
[2]Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai 200240, China.
[3]Department of Automation, Shanghai Jiaotong University, Shanghai 200240, China.
[4]Department of Information Sciences and Technology, Donghua University, Shanghai 201620, China.

Correspondence to: Dr. Xiuli Zhu, Department of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Yangpu District Jungong Road, Shanghai 200093, China. E-mail: xiulizhu@usst.edu.cn

Abstract

Due to the strong noise, high dimensionality and time-varying characteristics of industrial process data, data-driven modeling faces challenges in feature extraction and model interpretability. To address these issues, this paper proposes a new prediction model based on adaptive variational empirical mode decomposition-guided (AVEMDG) graph convolutional networks (GCNs). First, each sensor signal is decomposed into high-frequency and low-frequency features using empirical mode decomposition (EMD) to effectively capture multi-band information. Second, the weights of these features are adaptively updated through variational inference (VI) combined with Bayesian reasoning to handle the importance and uncertainty of features. Next, the GCN is used to model the spatiotemporal dependencies in the sensor network and is trained using the reweighted feature data. Last, the proposed method is applied to the prediction of the melt viscosity index (MVI), a key performance indicator (KPI) of the actual polyester fiber polymerization process. Ablation study and comparative experiment are conducted to evaluate the contribution of each component and the generality of the proposed model. Experimental results show that this method can effectively improve the model prediction accuracy, thereby enhancing the interpretability of the soft sensor model and providing guidance for the production of industrial processes.

Keywords: Data-driven modeling, soft sensor, industrial process, feature extraction, deep learning

1. INTRODUCTION

In modern industry, the prediction and real-time monitoring of key performance indicators (KPIs) are essential for improving production efficiency and ensuring stable industrial operations[1–3]. However, data collected during production often contains significant noise due to the complexity of physical and chemical properties and harsh operating conditions, leading to constraints by mass and energy conservation laws and complex spatiotemporal correlations, irregular sampling intervals, and varied sampling frequencies. Additionally, the high dimensionality, strong correlations, and redundancy in the data make KPI monitoring and prediction even more challenging[4]. Although the widespread adoption of distributed control systems (DCS) has enabled vast data collection[5–7], many KPIs remain difficult to measure or record directly with existing sensors. In cases where measurement is possible, environmental conditions, complex analysis requirements, and high implementation costs limit feasibility. To address these challenges, soft sensing technology has emerged as a practical solution[8], finding extensive application in industrial processes for reliable KPI estimation under difficult conditions.

Soft sensing refers to methods and algorithms that estimate or predict certain physical quantities or product quality in industrial processes based on existing measurement data and knowledge. Soft sensors differ from physical sensors in that they rely on computer software based systems or embedded systems for their implementation, rather than using direct physical measurements[9]. Soft sensing technology can be divided into two categories: mechanism-driven and data-driven models[10]. The mechanism-driven approach uses prior knowledge such as first principles, reaction kinetics, and mass balance to establish quantitative relationships between process variables and KPIs[11]. For example, Åström *et al.* derived the dynamic model of a natural circulation drum boiler by first principles[12]. Lampinen *et al.* proposed a kinetic reaction model for simulating the direct leaching process of zinc sulfide concentrate with the help of chemical kinetics knowledge[13]. However, these mechanism-driven approaches require a detailed and precise understanding of the underlying mechanisms, which is often difficult to obtain for increasingly complex processes[11]. In this case, since data-driven models are built entirely on data and do not rely on prior knowledge, data-driven methods are usually more suitable for KPI prediction[14,15].

With the rapid development of computer and information technology, feature extraction, as a key technology in data-driven modeling of industrial processes, has been significantly improved in this context. Feature extraction methods such as partial least squares (PLS)[16], independent component analysis (ICA)[17], slow feature analysis (SFA)[18] and canonical correlation analysis (CCA)[19] have been continuously expanded and improved[20]. Among them, the SFA has received widespread attention in the industry because the principle behind it is consistent with the nature of industrial processes; that is, the real process dynamics often change slowly, while noise often changes rapidly[21]. SFA can extract time-dependent latent features from process data and has been successfully used for process monitoring and modeling. Various soft sensors have been designed and developed to predict important variables that are not measured in the process[22–25].

Although many feature extraction methods are available, challenges still exist for data-driven modeling due to imperfect feature extraction in complex industrial processes. Reigosa *et al.* proposed an active method for island detection based on high-frequency signal voltage injection[26]. The advantage of this method is that the adverse effects caused by the injection of high-frequency voltage can be almost ignored, but the effects caused by low-frequency signals are ignored. Zhang *et al.* introduced the SFA method into non-stationary process monitoring, which can extract features that change slowly over time in process data[27]. However, they ignored high-frequency features and did not consider their impact. Wang *et al.* used ensemble empirical mode decomposition (EEMD) to divide the input signal into high- and low-frequency sequences and improved the prediction accuracy[28]. However, it is noted that the feature fusion process did not account for the differential contributions that high- and low-frequency sequences might offer to the overall predictive performance.

Additional challenges arise from the complexity of deep learning architectures, coupled with their black-box nature, which hinders the interpretability of the information processing within these networks[29,30]. Deep learning models employ a cascade of non-linear transformations across multiple layers to distill sophisticated representations from data, resulting in highly complex internal structures. This complexity, coupled with the challenges of adapting feature importance in dynamic industrial settings characterized by intricate spatiotemporal dependencies, often leads to a shortfall in capturing the nuanced relationships inherent in such environments. Consequently, the interpretability of these models is compromised, which in turn hampers their ability to provide actionable insights for practical industrial applications. This makes it difficult for engineers and decision-makers to understand how the model processes data and makes predictions, limiting its application in industrial processes[31–33].

Numerous studies have emerged on the interpretability of deep learning. Wang *et al.* proposed the concept of graph to describe the parameter relationships between and within layers[34]. By changing the data format, the parameters of the deep network are analyzed, which increases the interpretability of the deep network from another perspective. Xie and Grossman developed a graph convolutional neural network framework that is interpretable and can learn material properties directly from the connections between atoms in crystals, thereby providing a universal and interpretable representation of crystalline materials[35].

To solve the above problems, this paper proposes a graph convolutional network (GCN) model guided by adaptive variational empirical mode decomposition (AVEMDG-GCN) for data-driven modeling in complex industrial processes. The main contributions of this paper are presented below.

(1) A novel interpretable data-driven modeling method for adaptive variational empirical mode decomposition (EMD) perception is proposed. By introducing dynamic optimization weights, adaptive dynamic balance of high- and low-frequency components based on signal features is achieved, significantly improving the flexibility and accuracy of signal decomposition and reconstruction, providing a more adaptable solution for modeling complex industrial processes.

(2) In the evidence lower bound (ELBO) optimization framework, a dynamic weight allocation mechanism was innovatively introduced to achieve adaptive dynamic adjustment based on the optimization state between the likelihood term and Kullback-Leibler (KL) divergence term, effectively balancing the reconstruction accuracy and regularization strength in the modeling process, significantly enhancing the robustness and convergence stability of the model optimization.

(3) Based on a dataset of polyester polymerization processes in real industrial scenarios, the proposed AVEMDG-GCN model was comprehensively validated. The experimental results showed that the model can demonstrate excellent performance in complex dynamic behavior modeling and feature extraction tasks, further demonstrating the practical engineering applicability and reliability of the method.

The rest of this study is organized as follows. Section 2 introduces the EMD of time series signals and the GCN model. Section 3 introduces the AVEMDG-GCN model and the variational inference (VI) adaptive weight update. Section 4 introduces the experimental setup and preprocessing of the polyester fiber dataset, and then analyzes the experimental results in detail. Finally, Section 5 comprehensively summarizes the research content and proposes the current limitations and future research directions.

2. PRELIMINARIES

In this section, we provide the necessary background on two key techniques used in our proposed method: EMD and GCNs. EMD is a signal processing technique that allows for adaptive decomposition of non-linear
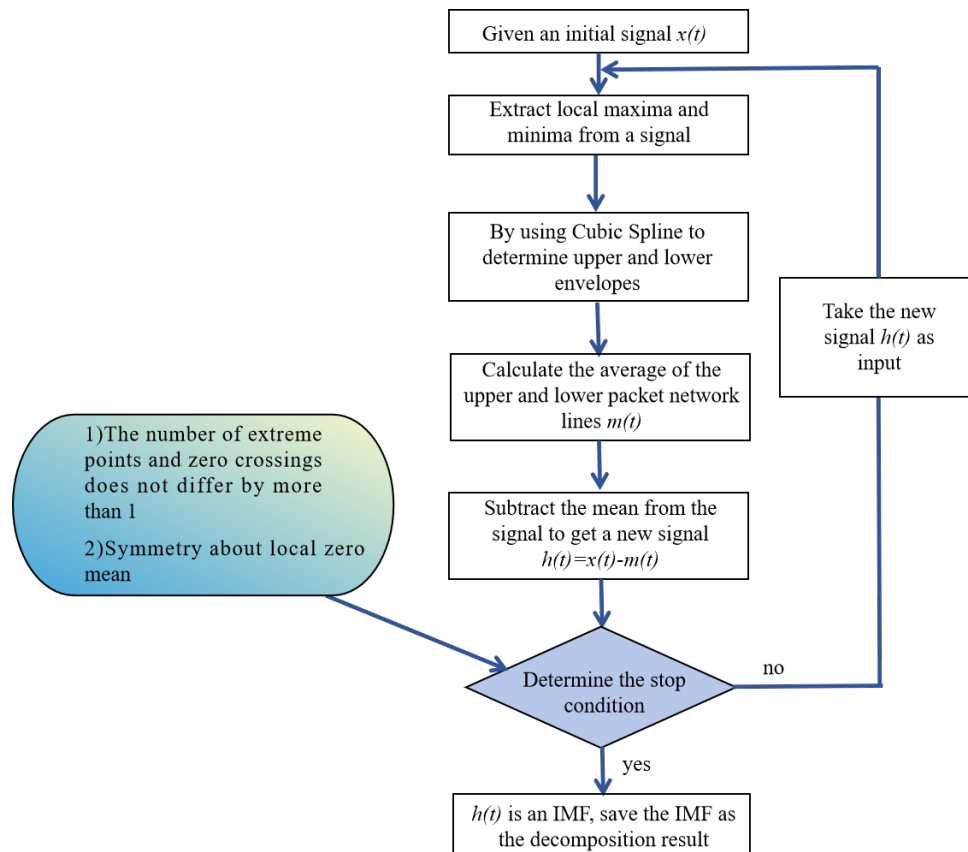
Figure 1. EMD process. EMD: Empirical mode decomposition.

and non-stationary signals, enabling the extraction of meaningful frequency components. This decomposition serves as the foundation for our adaptive frequency optimization by isolating high and low-frequency components of sensor signals. Additionally, GCNs offer a powerful approach for learning from graph-structured data, which we leverage to capture dependencies across sensor samples in a graph-based framework.

2.1 EMD

EMD is a pioneering time-frequency analysis technique renowned for its adaptive time-frequency localization capabilities [36,37]. Unlike traditional methods that rely on fixed basis functions, EMD is entirely data-driven and operates directly on the signal, making it particularly effective for analyzing non-linear and non-stationary time series. The primary purpose of EMD is to decompose a complex signal into a finite set of oscillatory components, known as intrinsic mode functions (IMFs), along with a residual trend. Each IMF represents a simple oscillatory mode of the original signal and must satisfy two key conditions: (1) the number of extrema and zero-crossings must either be equal or differ by at most one; and (2) the IMF must exhibit local symmetry around a zero mean. The decomposition process, as illustrated in Figure 1, is achieved through an iterative sifting procedure, where local extrema are systematically identified and used to generate envelopes that isolate individual IMFs. By adapting to the unique characteristics of the signal, EMD enables the extraction of dynamic features across varying frequency bands, providing a detailed representation of the signal's intrinsic oscillatory behavior. This makes EMD a powerful and versatile tool for a wide range of applications, including signal processing, fault diagnosis, and feature extraction [38].

(1) Input the original signal $x(t)$. This is the time series data that will be subjected to the decomposition process.

(2) The maximum points (local maxima) of the signal are identified and connected using cubic spline interpolation to form the upper envelope. This step ensures a smooth boundary encompassing the peaks of the signal.

(3) Similarly, the minimum points (local minima) of the signal are identified and connected using cubic spline interpolation to form the lower envelope. Together with the upper envelope, this defines the signal's local oscillatory bounds.

(4) Calculate the mean of the upper and lower envelopes to determine the average envelope $m_1(t)$. This mean envelope represents the local trend of the signal within its oscillatory range.

(5) By subtracting the mean envelope $m_1(t)$ from the original signal $x(t)$, the intermediate signal $h(t)$ is obtained. This intermediate signal is a candidate for being classified as an IMF.

(6) If the intermediate signal $h(t)$ satisfies the IMF conditions - specifically, having the number of extrema and zero-crossings equal or differing by at most one, and symmetry around a zero mean - it is designated as the first IMF. Otherwise, the intermediate signal is treated as a new input, and steps (2)-(5) are repeated to refine the mean envelope and generate a new intermediate signal $h(t)$. This iterative sifting process continues until the intermediate signal satisfies the conditions for obtaining the first IMF, denoted as imf_1.

(7) After obtaining imf_1, subtract it from the original signal $x(t)$ to generate a new signal. This new signal represents the residual after the first IMF has been extracted. Steps (2)-(6) are then repeated to obtain subsequent IMFs, one at a time, through iterative sifting. The process continues until no further IMFs can be generated, leaving a final residual $R(t)$ that contains the remaining non-oscillatory trend. The original signal can thus be expressed as the sum of all extracted IMFs and the residual, as given in

$$x(t) = \sum_{i=1}^{n} imf_i + R(t) \tag{1}$$

2.2 GCNs

Sensor signals in industrial processes usually have complex dependencies, which are usually not regular grids or simple Euclidean distance relationships. Based on this, we use graph structures to describe the correlation between sensors and capture the spatial topological characteristics of industrial systems. For sensor data mapping, we use a K-nearest neighbor (KNN)[39] approach to build adjacency relationships based on the feature vector of each sensor to reflect the similarity between sensors.

In our model, the data of each sensor node represents a graph signal, and the basic purpose of the GCN is to extract spatial features in the graph structure. The nodes in the GCN graph correspond to various sensors in the sensor network, and the edges in the graph represent the physical connections or functional associations between sensors, such as spatial proximity or coupling relationships between process variables. When designing GCN graphs, we construct adjacency matrices by analyzing the topology of the sensor network and the correlation between variables, so that the graph structure can reflect the physical meaning of the sensor network. The graph convolution layer is the core component of GCN[40], which captures local spatial relationships by aggregating neighborhood information to each node.

For a given graph $G = (V, E)$, the Laplacian matrix $L = D - A$ is essential for studying the structural properties of the graph, where D is the degree matrix, and A is the adjacency matrix constructed based on the KNN algorithm. The degree matrix D is diagonal, with each diagonal entry representing the degree of the corresponding node, while the adjacency matrix A encodes the connections between nodes. The Laplacian

matrix L captures key properties of the graph, such as connectivity and spectral characteristics. Since L is a real symmetric matrix, it can be diagonalized using eigendecomposition, as given in

$$L = U\Lambda U^T = \begin{bmatrix} u_1 & u_2 & \ldots & u_n \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix} \begin{bmatrix} u_1^T \\ u_2^T \\ \vdots \\ u_n^T \end{bmatrix} \tag{2}$$

Here, λ_i denotes the eigenvalues of the Laplacian matrix, representing the frequencies of the graph, and $\{u_i\}$ are the corresponding eigenvectors, which serve as the Fourier bases of the graph. For a particular graph signal $x \in \mathbb{R}^{n \times d}$, the Fourier transform, defined as $\hat{x} = U^T x$, projects the signal onto the graph spectral domain. The inverse transform $x = U\hat{x}$ reconstructs the original signal from its spectral representation. This approach, based on the graph Fourier transform, provides an effective way to analyze and capture correlations in graph-structured data, such as sensor networks or social graphs.

However, direct computation using Laplacian's eigendecomposition can be computationally expensive for large graphs. To address this, we apply a first-order Chebyshev polynomial approximation to the spectral filter and adopt a simplified first-order GCN model. The interlayer propagation rule for the GCN is defined in

$$X^{l+1} = \sigma \left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X^l W \right) \tag{3}$$

Here, $\tilde{A} = A + I$ is the adjacency matrix with added self-loops, where I is the identity matrix. The degree matrix $\tilde{D}$ is computed as $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$, ensuring proper normalization. W is the weight matrix to be learned during training, and X^l and X^{l+1} represent the graph signal features at layers l and $l + 1$, respectively. The activation function σ introduces non-linearity, enabling the model to capture complex patterns in the graph data. This propagation rule balances computational efficiency and representational power, making it well-suited for graph-based learning tasks.

By constructing an adjacency matrix based on KNN, our model can capture the similarity and dependency relationships among sensors in industrial processes, thereby extracting meaningful spatial features within complex industrial systems. This approach provides an effective means for graph-structured modeling of diverse sensor data.

3. SOFT SENSOR MODELING BASED ON AVEMDG-GCN FRAMEWORK

To improve the capability of feature extraction and enhance the interpretability of data-driven modeling methods, in this paper, a novel adaptive variational EMD aware interpretable data-driven modeling method in complex industrial processes, called AVEMDG-GCN, is proposed. In this section, we introduce the design structure and implementation steps of the AVEMDG-GCN model. The overall structure of this model is shown in Figure 2, which mainly contains two components: adaptive multi-frequency weighting using VI and adaptive ELBO weighting, as outlined below.

3.1 Adaptive multi-frequency weighting using VI
Data collected from complex industrial processes often exhibit characteristics such as strong noise, dynamics, high dimensionality, and non-linearity, which pose challenges for most deep learning networks in terms of feature extraction. To address these challenges, this paper decomposes the input time series $x(t)$ into several
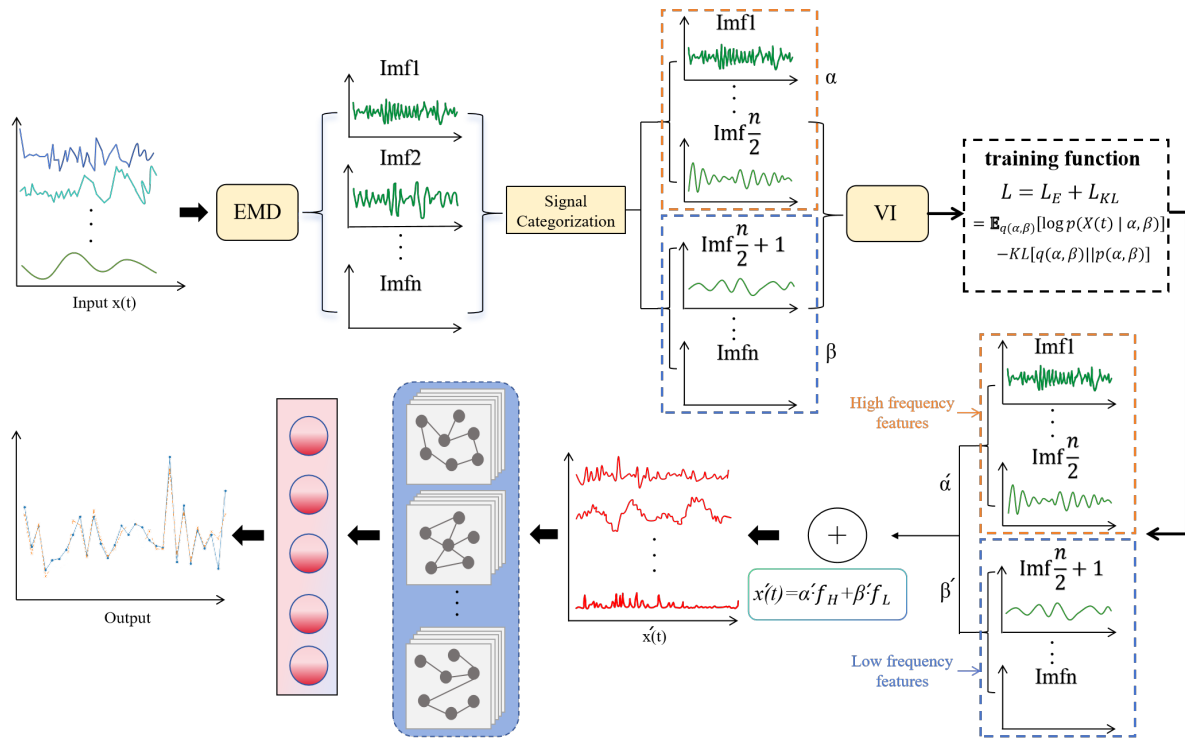
Figure 2. The proposed AVEMDG-GCN model structure. AVEMDG-GCN: Adaptive variational empirical mode decomposition-guided graph convolutional network.

IMFs through EMD as given in Equation (1). Then, we use the dichotomy method to divide the decomposed IMFs into IMFs with high-frequency characteristics and IMFs with low-frequency characteristics. At this stage, we introduce W_H and W_L as the weight coefficients of high-frequency IMF and low-frequency IMF, respectively, as given in

$$X_{re}(t) = W_H \sum_{i=1}^{N/2} IMF_i(t) + W_L \sum_{i=\frac{N}{2}+1}^{N} IMF_i(t) + R(t) \tag{4}$$

To ensure that $W_H + W_L = 1$ and $W_H, W_L > 0$, we make the following assumptions:

$$W_H = \frac{e^{\alpha}}{e^{\alpha} + e^{\beta}} \tag{5}$$

$$W_L = \frac{e^{\beta}}{e^{\alpha} + e^{\beta}} \tag{6}$$

Where α and β are two uncertain parameters. The weight form originates from the Softmax function, which can normalize parameters into probability distributions, ensuring a balanced and sensitive reflection of the importance of parameters while maintaining good mathematical properties. Given the uncertainty in the weights, we adaptively update the optimal weights using Bayes' theorem. The updated weights are determined as

$$p(\alpha, \beta | x(t)) = \frac{p(x(t) | \alpha, \beta) p(\alpha, \beta)}{p(x(t))} \tag{7}$$

Where $p(\alpha, \beta | x(t))$ is the posterior probability, which represents the probability distribution of the parameters α and β given the data $x(t)$. $p(x(t) | \alpha, \beta)$ is the likelihood function, which represents the probability of starting the data $x(t)$ given the parameters α and β; this reflects how well the data observed by the model fits the parameters. $p(\alpha, \beta)$ is the prior probability, which represents the probability distribution of parameters α and β before any factory data is available. Prior information is usually based on domain knowledge or previous

research results. $p(x(t))$ is the marginal probability, which represents the total probability of the starting data $x(t)$. It can be calculated by integrating over all possible parameter values as given in

$$p(x(t)) = \int p(x(t)|\alpha, \beta)p(\alpha, \beta)d\alpha d\beta \tag{8}$$

Since there is no obvious direct relationship between α and β, we assume that α and β are independent of each other. Then, Equation (7) can be transformed into

$$p(\alpha, \beta|x(t)) = \frac{P(X(t)|\alpha, \beta)P(\alpha)P(\beta)}{P(X(t))} \tag{9}$$

Where we assume that the prior distributions $p(\alpha)$ and $p(\beta)$ are both standard normal distributions, $P(\alpha) \sim N(0, 1), P(\beta) \sim N(0, 1)$. As can be seen from Equation (8), the marginal probability is difficult to calculate directly, making the posterior distribution very complicated, so we introduce variational distribution $q(\alpha, \beta)$ to approximate the posterior distribution $p(\alpha, \beta|x(t))$. Since α and β are independent of each other, variational distribution $q(\alpha, \beta)$ can be transformed into $q(\alpha, \beta) = q(\alpha) \cdot q(\beta)$. Since the posterior distribution $p(\alpha, \beta|x(t))$ is a normal distribution, we assume the variational distribution $q(\alpha, \beta)$ as given in[41]

$$q(\alpha, \beta) = N(\mu_\alpha, \sigma_\alpha^2) \cdot N(\mu_\beta, \sigma_\beta^2) \tag{10}$$

Where $\mu_\alpha, \sigma_\alpha^2, \mu_\beta, \sigma_\beta^2$ represent the means and variances of the normal distributions for the parameters α and β, respectively. Since the KL divergence is always non-negative and is zero only when the two distributions are exactly the same, this property makes it a reliable measure of distribution differences. When the KL divergence is small or zero, it indicates that the approximate distribution and the target distribution are similar and the optimization goal has been well achieved. Conversely, When it is large, it indicates that the difference between the distributions is significant and the model requires further optimization. Therefore, we use KL divergence to measure the difference between the $q(\alpha, \beta)$ and the $p(\alpha, \beta|x(t))$ as given in

$$\mathrm{KL}(q(\alpha, \beta) \parallel p(\alpha, \beta \mid X(t))) = \int q(\alpha, \beta) \log \frac{q(\alpha, \beta)}{p(\alpha, \beta \mid X(t))} \, d\alpha \, d\beta \tag{11}$$

Substituting Equation (7) into Equation (11) yields

$$\mathrm{KL}(q(\alpha, \beta) \parallel p(\alpha, \beta \mid X(t))) = \log p(X(t)) - \mathrm{ELBO} \tag{12}$$

Minimizing KL divergence means maximizing ELBO. ELBO is expressed as

$$\begin{aligned}
\mathrm{ELBO} &= \int q(\alpha, \beta) \log \frac{p(X(t) \mid \alpha, \beta)p(\alpha, \beta)}{q(\alpha, \beta)} \, d\alpha \, d\beta \\
&= \int q(\alpha, \beta) \log p(X(t) \mid \alpha, \beta) \, d\alpha \, d\beta + \int q(\alpha, \beta) \log \frac{p(\alpha, \beta)}{q(\alpha, \beta)} \, d\alpha \, d\beta \\
&= \mathbb{E}_{q(\alpha,\beta)} \left[\log p(X(t) \mid \alpha, \beta) \right] - \mathrm{KL}(q(\alpha, \beta) \parallel p(\alpha, \beta)).
\end{aligned} \tag{13}$$

Since α and β are independent of each other, we can get

$$\mathrm{KL}(q(\alpha, \beta) \parallel p(\alpha, \beta)) = \mathrm{KL}(q(\alpha)\|p(\alpha)) + \mathrm{KL}(q(\beta)\|p(\beta)) \tag{14}$$

The final ELBO is given in

$$\mathrm{ELBO} = \mathbb{E}_{q(\alpha,\beta)} \left[\log p(X(t) \mid \alpha, \beta) \right] - (\mathrm{KL}(q(\alpha)\|p(\alpha)) + \mathrm{KL}(q(\beta)\|p(\beta))) \tag{15}$$

We assume that the likelihood function is normally distributed.

$$p(X(t)|\alpha, \beta) = N(X_{re}(t), \sigma^2) \tag{16}$$
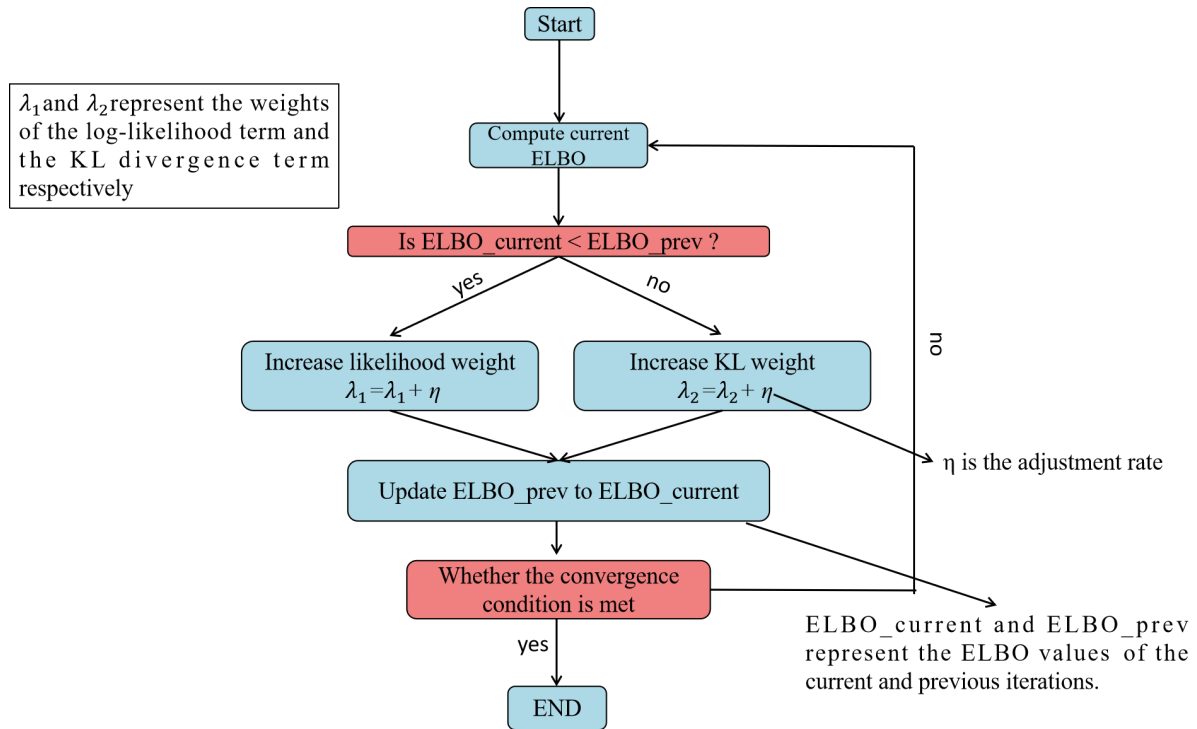
Figure 3. ELBO dynamic weight adjustment process. ELBO: Evidence lower bound.

$$\log p(X(t)|\alpha, \beta) = -\frac{1}{2}\left(\frac{||X_1(t) - X_{re}(t)||^2}{\sigma^2} + \log(2\pi\sigma^2)\right) \tag{17}$$

Since $\log(2\pi\sigma^2)$ is a constant term.

$$\log p(X(t) \mid \alpha, \beta) \propto -\frac{1}{2\sigma^2}||X(t) - X_{\text{reconstructed}}||^2 \tag{18}$$

By minimizing the error between the observed data and the model's reconstructed data, the model parameters α and β are optimized to improve the data fit.

3.2 Adaptive ELBO weighting

During the optimization process, the contributions of the log-likelihood term and the KL divergence term may be unbalanced, resulting in unstable optimization and affecting the convergence speed. If the difference in the gradient scale between the two is too large, the optimizer may be more inclined to adjust one part and ignore the other, resulting in slow convergence. In addition, an overly strong KL divergence term will lead to over-regularization, making the model too simple to fit the data details; conversely, if the log-likelihood term is dominant, the model may overfit the data and affect generalization ability. Therefore, we introduce weight factors, introduce weights for log likelihood and KL divergence terms, λ_1 and λ_2, and dynamically adjust their proportions to make the contribution of the two parts of the gradient more balanced, as given in

$$\text{ELBO} = \lambda_1 \cdot \mathbb{E}_{q(\alpha,\beta)}\left[\log p(X(t) \mid \alpha, \beta)\right] - \lambda_2 \cdot (\text{KL}(q(\alpha)||p(\alpha)) + \text{KL}(q(\beta)||p(\beta))) \tag{19}$$

To achieve dynamic adjustment, we can consider adjusting these two weights according to the current optimization state (such as the rate of change of ELBO, the size of the gradient, *etc.*). We can design a strategy: if the gradient of the log-likelihood term changes greatly (which means the reconstruction error is large), the weight of λ_1 will increase; conversely, if the KL divergence term contributes more to ELBO, the weight of λ_2 will grow, as shown in Figure 3. The AVEMDG-GCN model is optimized using a loss function that combines the log-likelihood term and the KL divergence term, allowing it to capture essential features of the data while

balancing model accuracy and regularization. This loss function is defined as follows:

$$\mathcal{L} = \mathbb{E}_{q(\alpha,\beta)}\left[\log p(X \mid \alpha, \beta)\right] - \mathrm{KL}(q(\alpha,\beta) \parallel p(\alpha,\beta)) \tag{20}$$

where the log-likelihood term maximizes the model's ability to fit the observed data, capturing critical features necessary for accurate predictions. The KL divergence term acts as a regularizer, ensuring that the approximate distribution $q(\alpha,\beta)$ remains close to the prior distribution $p(\alpha,\beta)$, thus preventing overfitting and balancing high-frequency and low-frequency information. The training process is further enhanced by the Adam optimization algorithm, which iteratively updates parameters α and β. Adam uses two moving averages, m_t and v_t, for the gradient and squared gradient, respectively, to stabilize the update process:

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot \nabla \mathcal{L} \tag{21}$$

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot (\nabla \mathcal{L})^2 \tag{22}$$

where $\nabla \mathcal{L}$ represents the gradient of the loss function $\mathcal{L}$ with respect to the parameters α and β. The parameters β_1 and β_2 are typically set to 0.9 and 0.999, helping capture historical gradient information for smooth convergence. After calculating m_t and v_t, the model parameters are updated as follows:

$$\Theta \leftarrow \Theta - \eta \frac{m_t}{\sqrt{v_t} + \epsilon} \tag{23}$$

where η is the learning rate, and ϵ is a small constant (e.g., 1×10^{-8}) to prevent division by zero. This adaptive scaling method normalizes the learning rates across parameters, ensuring efficient convergence and robust learning. By combining this loss function and optimization approach, AVEMDG-GCN effectively captures complex multi-frequency and spatiotemporal dependencies within industrial processes, enhancing predictive accuracy and generalization.

The optimized weights are calculated by

$$W_H^* = \frac{e^{\mu_\alpha}}{e^{\mu_\alpha} + e^{\mu_\beta}} \tag{24}$$

$$W_L^* = \frac{e^{\mu_\beta}}{e^{\mu_\alpha} + e^{\mu_\beta}} \tag{25}$$

The final reconstructed signal $X_{\mathrm{op}}(t)$ is determined by

$$X_{\mathrm{op}}(t) = W_H^* \sum_{i=1}^{N/2} \mathrm{IMF}_i(t) + W_L^* \sum_{i=N/2+1}^{N} \mathrm{IMF}_i(t) + R(t) \tag{26}$$

3.3 Modeling procedure based on the AVEMDG-GCN

The construction of the AVEMDG-GCN-based soft sensor is summarized in the following:

Step 1: For each sensor signal, EMD is applied to finely separate high-frequency and low-frequency components.

Step 2: The weights of high- and low-frequency features are adaptively adjusted through VI, and dynamic importance is assigned according to their relevance and uncertainty in prediction.

Step 3: Dynamic weighting mechanism between the likelihood term and the KL divergence term is introduced in the ELBO optimization to dynamically adjust the balance between reconstruction fidelity and regularization during the optimization process.

Table 1. Predictor variables considered for soft sensor modeling

No.	Tags	Descriptions	No.	Tags	Descriptions
1	PIC-10326	N_2 pressure of TPA tank	17	DIC-10317	The density of slurry
2	SIC-10343	TPA rotary valve speed	18	FI-10654	1223-H03 water flow rate
3	FIC-10302	EST to EG slurry mix tank flow	19	LI-10804	Catalyst spray to tank level
4	FIC-10301	Circulation of EG	20	TJE-10657	1223 outlet temperature
5	LIC-14208	Cp circulating tank level	21	LIC-11406	Level of UFPP 16th
6	FIC-11716	Fresh EG added	22	LIC-11602A	Inlet liquid level of FIN
7	LI-11602B	Outlet liquid level of FIN	23	PDI-11407	UFPP differential pressure
8	LIC-10313	Level control of slurry tank	24	PIC-11603	FIN pressure
9	FIC-10401	Slurry to feed tank flow	25	TI-10646A	Esterified water temperature
10	FIC-10406A	Slurry A to esterification flow	26	FIC-11008	The flow of DEG
11	LI-10901	TiO2 spray to tank level	27	PI-10408B	Pressure of PTA sizing agents
12	FIC-10406B	Slurry B to esterification flow	28	TIC-10601	Temperature of first plate
13	II-10312	Slurry mix tank current agitator	29	FIC-10909	Oligomer flow
14	LI-11002	DEG tank level	30	PI-10525	Pressure of siphon
15	FIC-10630	Flow of the reflux liquid	31	VIC-11808	Melt viscosity intrinsic
16	PI-10408A	Pressure of EG sizing agents	-	-	-

Step 4: The high- and low-frequency features are fused by weighted averaging to generate a new dataset without noise.

Step 5: The new dataset is sent into the GCN to finish prediction.

4. CASE STUDY: APPLICATION TO POLYMERISATION PROCESS

4.1 Process description

Taking the polymerization process of a polyester factory as an example, the predictive performance of the AVEMDG-GCN soft sensor model was evaluated. The necessary assumptions for this process include the following: the data is complete, and the data used for modeling needs to be able to represent the behavior of the entire system (the data obtained includes the three stages of the polyester fiber polymerization process, and a total of 10,000 data pieces including a complete production cycle of the polymerization process are collected). The flow chart of this process is shown in Figure 4. Initially, terephthalic acid (PTA) is combined with ethylene glycol (EG) in a precise ratio in the slurry tank. The resulting slurry is then pumped into a buffer tank and continuously fed into an esterification rotary reactor, where esterification occurs. The purified mixture is then moved from the bottom to the precondensation reactor, where it flows upward to form the prepolymer while the EG evaporates. This prepolymer is then transferred to the final polycondensation reactor to produce the final product. The viscometer (VIC 11808) is installed in the pipe carrying the final product and measures the melt viscosity index (MVI). However, due to the frequent failures and high cost of procurement and maintenance of hardware sensors (approximately $46,466), real-time MVI measurements are impractical in terms of both accuracy and cost. Therefore, it is necessary to estimate MVI online using soft sensors. In this study, we developed a soft sensor model based on avemdg-gcn that combines statistical characteristics of the data and integrates probabilistic and deep learning methods to achieve accurate MVI predictions. The model is suitable for data-driven modeling of complex industrial processes that depend on time series data. Specifically, the method assumes that the input data is in the form of a time series with high integrity and properly processed noise. In addition, the model is especially suitable for dealing with industrial processes with complex dynamic behaviors such as non-linear and multivariable coupling, which is usually assumed in most complex industrial processes, so the method has good universality.
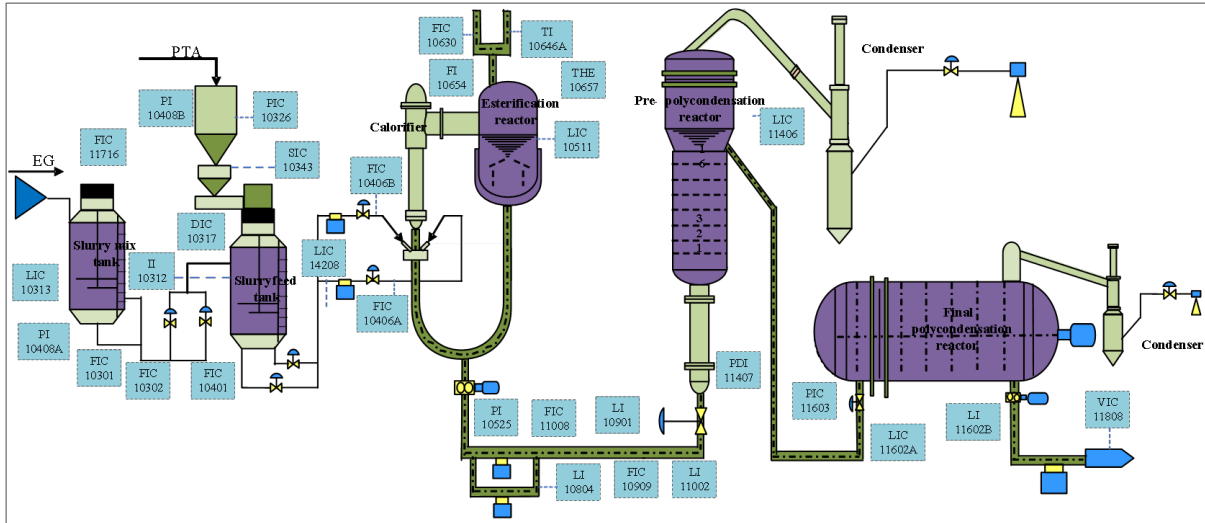
Figure 4. Flow diagram of the polymerization process.

4.2 Dataset description

The data utilized in this study were obtained from the historical records of the DCS of the polyester plant in question. Drawing from the operators' expertise and process knowledge, 30 secondary variables that exhibit a strong correlation with MVI were chosen as predictors (as shown in Table 1) for the development of the AVEMDG-GCN-based soft sensor. The historical dataset consists of 2,000 observations. The data were organized chronologically: 60% were allocated for training, the subsequent 20% served as the validation set, and the final 20% were used to assess the generalization performance of the soft sensor.

4.3 Experimental results analysis

4.3.1 Model evaluation metrics

To evaluate the model's performance, we used six evaluation metrics: mean absolute error (MAE), mean absolute percentage error (MAPE), root mean square error (RMSE), symmetric MAPE (SMAPE), mean square error (MSE), and correlation coefficient (R^2), which can be expressed as follows:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{27}$$

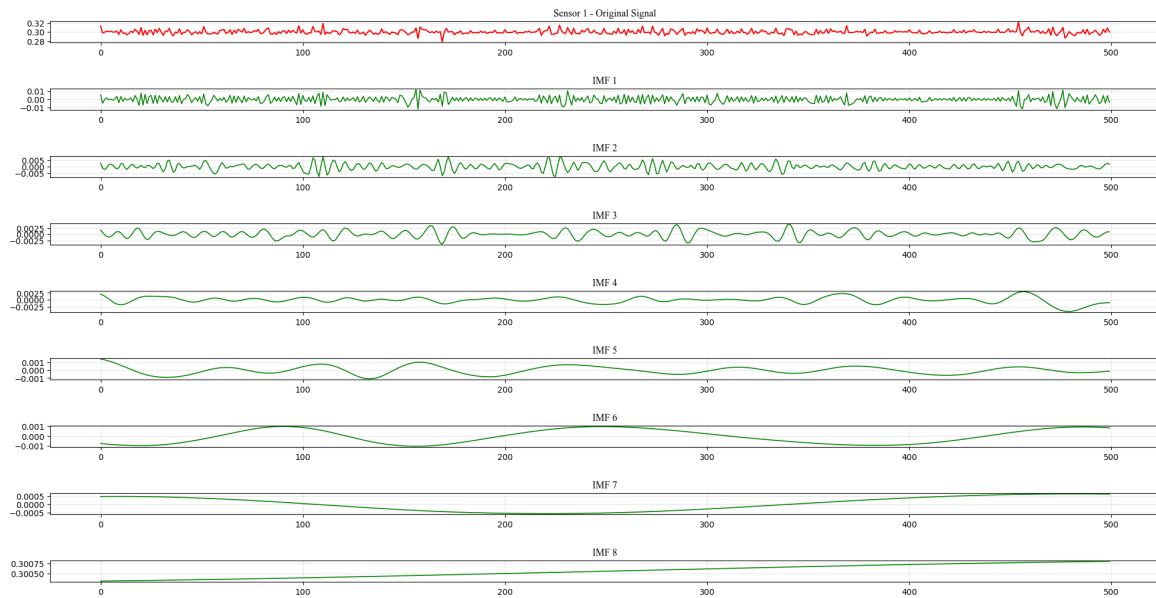$$\text{MAPE} = \frac{100}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{28}$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{29}$$

$$\text{SMAPE} = \frac{100}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{|y_i| + |\hat{y}_i|} \tag{30}$$

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{31}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{32}$$

Where n is the total number of data points. y_i represents the actual value of the i-th data point. $\hat{y}_i$ stands for the value of the i-th data point in the predicted value. $\bar{y}$ indicates the average of the actual values.

**Figure 5.** EMD explosion diagram. EMD: Empirical mode decomposition.

4.3.2 EMD results

The decomposition results for the data from the first sensor (PIC-10326) using EMD method are shown in Figure 5. The figure shows different signals of data collected from the sensor PIC-10326. The original signal is depicted in red at the top, showing some irregular fluctuations around a mean value. Below, the EMD results are presented, breaking down the original signal into eight IMFs. Each IMF represents a component with distinct frequency characteristics, from high-frequency fluctuations in IMF 1 to progressively lower frequencies in subsequent IMFs. The last few IMFs, particularly IMF 7 and IMF 8, display slow, smooth oscillations, representing low-frequency trends or the general trend of the signal over time. We then use the dichotomy method to classify the decomposed IMF into high-frequency features and low-frequency features, and then use VI to adaptively weight to highlight the importance of high and low frequencies. To more effectively illustrate the weight variations captured by various sensors, the data from two representative sensors has been selected for detailed examination and is presented in Figure 6.

The plot illustrates the optimization process of alpha and beta weights for two sensors over 15 iterations. For Sensor 1, the alpha weight (blue line) gradually increases and stabilizes around 0.6, while the beta weight (orange dashed line) decreases steadily, approaching 0.4. This trend suggests a growing preference for the high-frequency components represented by alpha in Sensor 1's reconstruction. Similarly, for Sensor 2, the alpha weight (green line) initially rises and then levels off at a slightly lower value than Sensor 1's alpha, while the beta weight (red dashed line) shows a consistent decline, mirroring Sensor 1's beta. Overall, the optimization process favors the influence of high-frequency components (alpha) over low-frequency components (beta) for both sensors, which likely reflects their relative importance in accurately reconstructing the sensor data.

4.3.3 Ablation experiment

In order to evaluate the contribution of different components in the proposed AVEMDG-GCN model, ablation experiments were conducted. The main purpose of these experiments is to separate and evaluate the effectiveness of adaptive variational EMD and GCN in predicting the KPI of the polyester polymerization process. We compared three models:

(1) AVEMDG-GCN: This model incorporates adaptive variational EMD for feature extraction and employs a
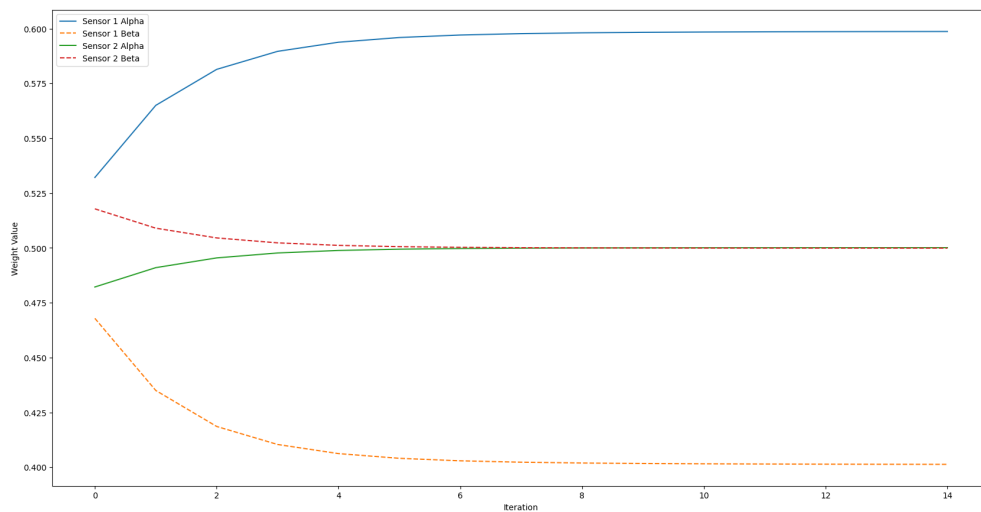
Figure 6. Adaptive weight change graph.

GCN based on KNN to model the spatial interdependencies among various sensors. By leveraging adaptive variational EMD, the model can decompose sensor signals into IMFs, adaptively capturing high- and low-frequency information that is crucial for accurate prediction. Then, the GCN based on KNN processes these adaptive weighted IMFs to effectively learn the relationships between different sensor data. Through these two strategies, the proposed AVEMDG-GCN enhances the model's interpretability, which is helpful for operators to better understand the polyester fiber polymerization production process.

(2) EMD-GCN: In this model, we use EMD to preprocess the data to extract features and then pass them to a standard GCN. While this model retains the advantages of EMD in feature extraction, it does not incorporate the adaptation mechanisms and enhancements present in AVEMDG-GCN, allowing us to evaluate the impact of these additional layers.

(3) GCN: This baseline model utilizes raw sensor data without any preprocessing. By applying GCN directly to the raw data, we can establish a performance baseline. This model is intended to highlight the importance of preprocessing steps such as adaptive variational EMD in enhancing the predictive power of the model.

The data used in this study was split chronologically, with 60% for training, 20% for validation, and the remaining 20% for testing. All models were trained using the same data split to ensure a fair comparison. The six evaluation metrics described previously were used to quantify the predictive performance of each model. The experimental results of the ablation experiment are shown in Table 2.

The performance comparison in the table highlights the superiority of the AVEMDG-GCN model over the EMD-GCN and GCN models across all evaluation metrics. Specifically, AVEMDG-GCN achieves the highest R^2 score of 0.7439, indicating a stronger correlation between predicted and actual values, and demonstrates the lowest MSE of 0.0020 and RMSE of 0.0450, reflecting its higher accuracy in predicting the MVI. Furthermore, it achieves the lowest MAE of 0.0343 and significantly reduces MAPE and SMAPE to 5.0347 and 5.0121, respectively, outperforming other models in minimizing prediction error. These results underscore the effectiveness of integrating adaptive EMD with GCNs, which enhances predictive accuracy and robustness. To provide a more comprehensive evaluation, it is valuable to analyze the contributions of different components

Table 2. Performance comparison of models based on different metrics

Model	R^2	MSE	RMSE	MAE	MAPE	SMAPE
AVEMDG-GCN	0.7439	0.0020	0.0450	0.0343	5.0347	5.0121
EMD-GCN	0.7175	0.0022	0.0473	0.0361	5.2336	5.2110
GCN	0.6804	0.0025	0.0503	0.0388	5.6586	5.6600

R^2: Correlation coefficient; MSE: mean square error; RMSE: root mean square error; MAE: mean absolute error; MAPE: mean absolute percentage error; SMAPE: symmetric MAPE; AVEMDG-GCN: adaptive variational empirical mode decomposition-guided graph convolutional network; EMD: empirical mode decomposition; GCN: graph convolutional network.

within the AVEMDG-GCN model. For example, exploring the role of adaptive EMD in preprocessing and its impact on feature extraction could clarify its influence on the model's performance. Additionally, assessing the model's behavior under varying working conditions, such as changes in input data characteristics or operational scenarios, would help establish its robustness and generalizability. Such analyses would not only validate the practical reliability of AVEMDG-GCN but also provide actionable insights for optimizing its application in industrial contexts.

Figure 7 shows the comparison between the actual and predicted values of the three models: AVEMDG-GCN, EMD-GCN, and GCN. In each image, the blue line represents the true value, while the red line indicates the predicted value. It is evident from the plots that the AVEMDG-GCN model is closest to the true values, reflecting its superior prediction accuracy. The model more accurately captures underlying trends and fluctuations, minimizing deviations from the true values. In contrast, the EMD-GCN model shows a moderately accurate fit, following the overall trend, but with some observable deviations, especially in areas where values change rapidly. Finally, the GCN model, while able to approximate the overall trend, has the largest deviation from the true values, indicating lower prediction performance compared to the other models. These plots intuitively confirm that AVEMDG-GCN provides the most reliable predictions, followed by EMD-GCN, while GCN has the lowest accurate predictions. This observation is consistent with the quantitative metrics presented previously, further supporting the effectiveness of AVEMDG-GCN in the prediction task.

4.3.4 Related baseline model comparison

In order to verify the effectiveness of the proposed prediction method, we compared four different models to evaluate their performance using the actual dataset collected from the polyester polymerization process. Our model is AVEMDG-GCN, which combines the advantages of adaptive weighting mechanisms and GCNs. For comparison, we selected three baseline models: a CNN + long short-term memory (LSTM) model, which is a traditional sequence modeling approach capable of capturing temporal dependencies; GraphSAGE, which generates node representations through a sampling method suitable for large-scale graph data, and graph attention network (GAT), which enhances the modeling of relationships between nodes by introducing an attention mechanism. In addition, we evaluated separate LSTM and gated recursive unit (GRU) models, which are widely used in sequence data analysis due to their excellent performance in capturing temporal dependencies. By comparing the performance of these models, we aim to gain a comprehensive understanding of the strengths and limitations of AVEMDG-GCN under different data characteristics and task settings.

The experimental results of the baseline experiment are shown in Table 3. This table provides a comprehensive comparison of the predictive performance of six models: GAT, GraphSAGE, CNN + LSTM, LSTM, GRU, and AVEMDG-GCN. Among these, the AVEMDG-GCN model stands out as the most accurate, achieving the highest R^2 value of 0.7439, which indicates it explains the largest variance in the data. It also has the lowest error metrics, including an MSE of 0.0020, RMSE of 0.0450, and MAE of 0.0343, reflecting its minimal prediction error. Additionally, in percentage-based errors, AVEMDG-GCN maintains its superiority with the
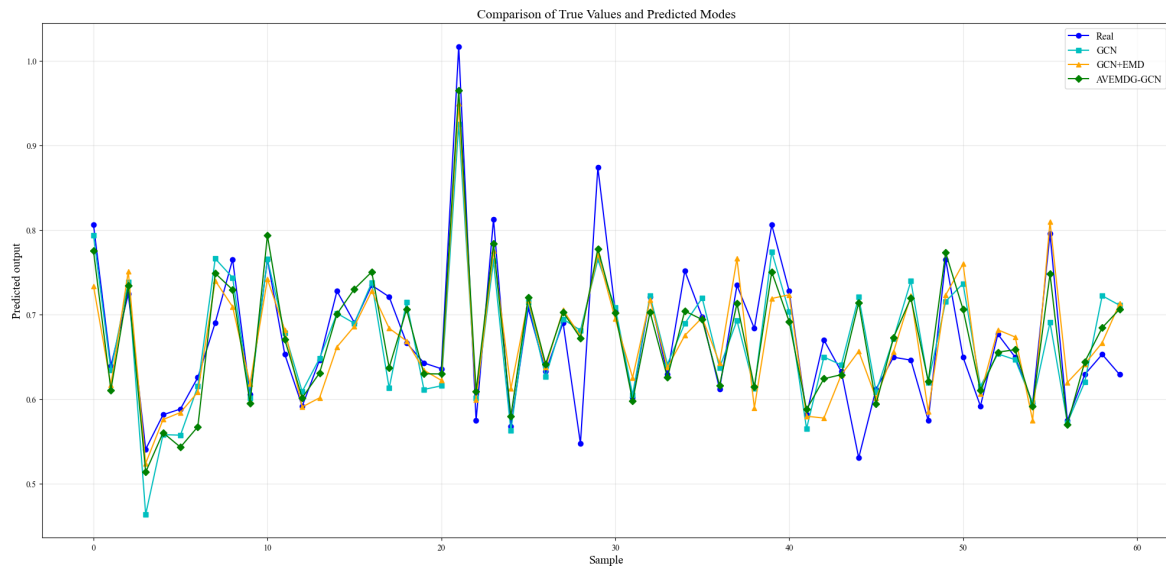
Figure 7. Comparison chart of actual value and predicted value.

Table 3. Baseline comparison results

Model	R²	MSE	RMSE	MAE	MAPE	SMAPE
GAT	0.6472	0.0037	0.0615	0.0469	6.9795	6.9097
GraphSAGE	0.6267	0.0035	0.0595	0.0464	6.8229	6.8165
CNN + LSTM	0.6029	0.0033	0.0572	0.0441	6.4550	6.3851
LSTM	0.5848	0.0040	0.0634	0.0504	7.4310	7.3715
GRU	0.5911	0.0040	0.0629	0.0492	7.2610	7.2069
AVEMDG-GCN	0.7439	0.0020	0.0450	0.0343	5.0347	5.0121

R²: Correlation coefficient; MSE: mean square error; RMSE: root mean square error; MAE: mean absolute error; MAPE: mean absolute percentage error; SMAPE: symmetric MAPE; GAT: graph attention network; CNN: convolutional neural net-work; LSTM: long short-term memory; GRU: gated recursive unit; AVEMDG-GCN: adaptive variational empirical mode decomposition-guided graph convolutional network.

lowest MAPE (5.0347) and SMAPE (5.0121), showcasing its high accuracy in both absolute and relative terms. Among the baseline models, GRU outperforms LSTM slightly, with a higher R² value (0.5911 *vs*. 0.5848) and marginally lower RMSE (0.0629 *vs*. 0.0634) and MAE (0.0492 *vs*. 0.0504). However, both GRU and LSTM still lag behind the performance of GAT, GraphSAGE, and CNN + LSTM. GAT, while the second-best performer overall, exhibits a higher RMSE (0.0615) and SMAPE (6.9097), indicating a greater margin of error compared to AVEMDG-GCN. GraphSAGE and CNN + LSTM perform moderately but are less competitive in R² and error metrics, particularly in RMSE and MAPE. Overall, AVEMDG-GCN is consistently the most reliable model across all metrics, making it the optimal choice for tasks requiring high predictive accuracy and minimal error.

The learning curves corresponding to the best runs of the six models - CNN + LSTM, GraphSAGE, GAT, AVEMDG-GCN, LSTM, and GRU - are shown in Figure 8. To better display the differences between models, the plot starts from the 50th epoch. The curves highlight AVEMDG-GCN as the most effective, achieving the lowest final loss with a stable and consistent decline throughout training. GAT follows closely with competitive performance, showing a low and steady loss. LSTM and GRU demonstrate moderate improvements, but their losses remain higher, indicating relatively less effective learning. CNN + LSTM stabilizes at a higher
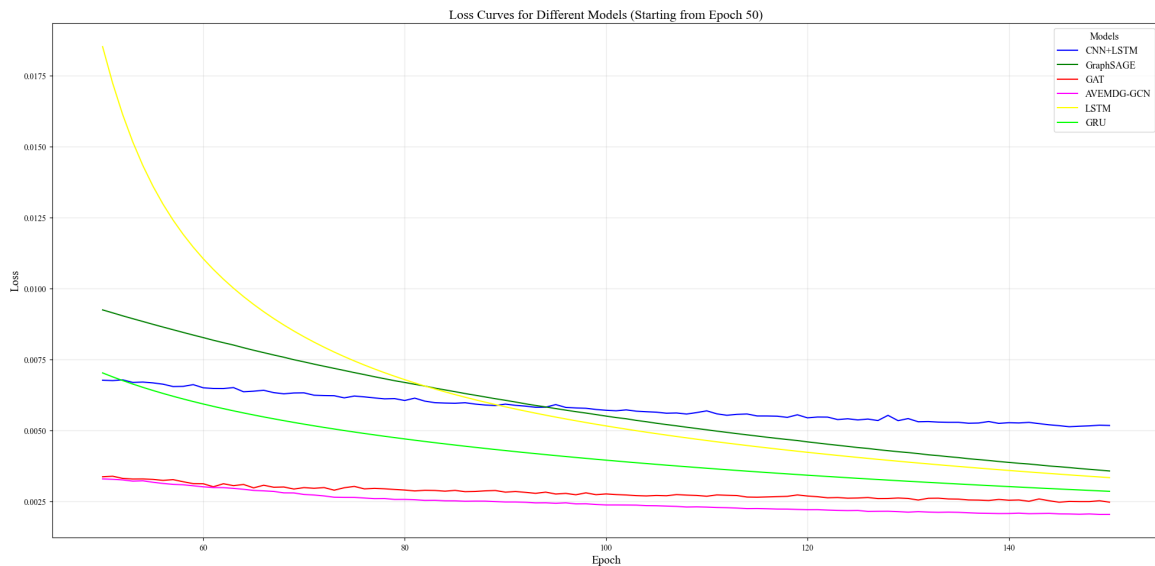
Figure 8. Convergence curve of the soft sensor models.

loss, reflecting limited performance, while GraphSAGE, despite a sharp initial decline, ends with the highest final loss, making it the least suitable for this task. Overall, AVEMDG-GCN outperforms the other models, showcasing superior learning and generalization.

The comparison between the actual and predicted values of the six models is shown in Figure 9. This image illustrates the prediction performance of CNN + LSTM, GAT, GraphSAGE, AVEMDG-GCN, LSTM, and GRU. Among them, AVEMDG-GCN demonstrates the strongest ability to capture both overall trends and local fluctuations, maintaining a curve closely aligned with the actual values. CNN + LSTM also shows competitive performance but exhibits slightly higher deviations in some regions. GAT performs reasonably well but struggles in areas with pronounced peaks and valleys. The GraphSAGE model effectively captures smoother trends but lacks precision in modeling local fluctuations. Similarly, LSTM and GRU display moderate capabilities, with GRU slightly outperforming LSTM in aligning with actual values. Overall, AVEMDG-GCN and CNN + LSTM emerge as the most effective models, with AVEMDG-GCN demonstrating the best generalization across the prediction task.

5. CONCLUSIONS

We present state-of-the-art results for predicting the key quality indicator MVI of polyester production using a novel AVEMDG-GCN model that combines adaptive variational EMD with GCN based on KNN. The model leverages a learnable weighting mechanism to decompose high- and low-frequency features, along with graph-based learning to capture the complex spatial dependencies between sensor data, which enhances the generalization ability of the model under different conditions. In addition, we gain new theoretical insights into the combination of probabilistic and deep learning methods in soft sensor modeling, showing how adaptive VI can improve the robustness and interpretability of the model. To further enrich this research, future work will focus on extending AVEMDG-GCN through adaptive graph neural networks to better capture the dynamic changes over time. In addition, we plan to improve the adaptive weighting mechanism with more advanced VI techniques to enhance interpretability and improve uncertainty quantification. Extending the application scope of the framework to other industrial fields with different sensor configurations and evaluating its performance under different operating conditions will also be key priorities. Finally, we aim to optimize the
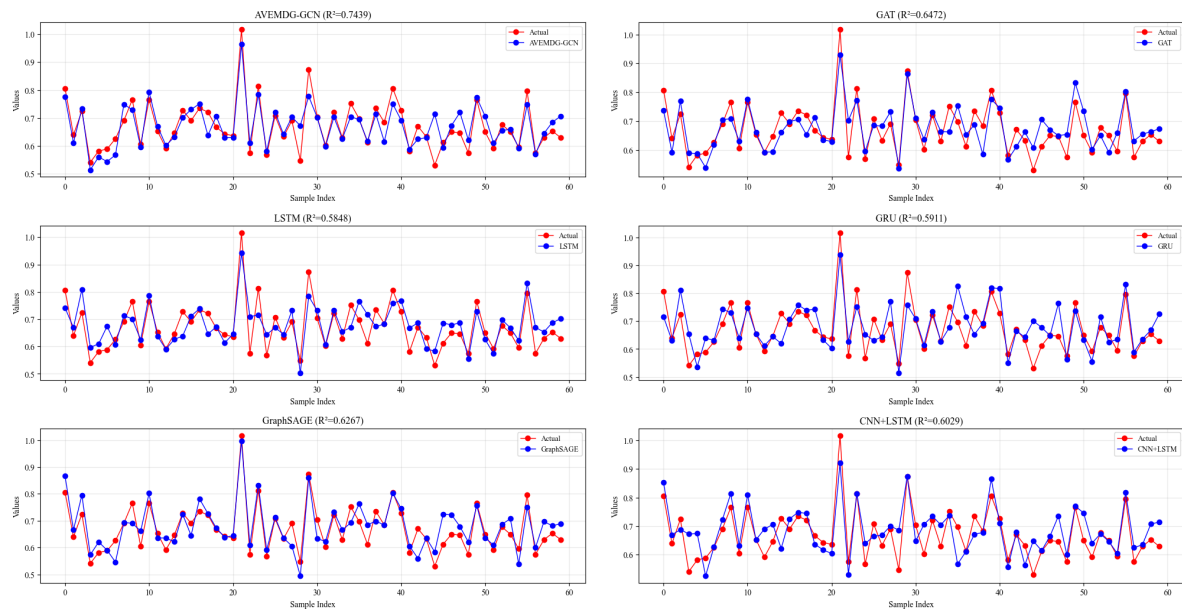
Figure 9. Comparison of actual and predicted values of different models.

computational efficiency of the model to enable real-time deployment of large-scale continuous monitoring systems, ensuring its scalability and practicality for industrial applications.

DECLARATIONS

Acknowledgments
We thank the Editor-in-Chief and all reviewers for their comments.

Authors' contributions
Conceptualization of this study, methodology, software, writing - original draft: Chen, Y.
Conceptualization of this study, review and editing, methodology: Zhu, X.
Review and editing: Wang, P.
Writing - review and editing: Hao, K.
Editing: Sheng, K.

Availability of data and materials
The data are available upon request. If needed, please contact the corresponding author by email.

Conflicts of interest
All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate
Not applicable.

Consent for publication

Not applicable.

Copyright

© The Author(s) 2025.

REFERENCES

1. Yuan X, Wang Y, Wang C, et al. Variable correlation analysis-based convolutional neural network for far topological feature extraction and industrial predictive modeling. *IEEE Trans Instrum Meas* 2024;73:1–10. DOI

2. Zhang J, Tian J, Alcaide AM, et al. Lifetime extension approach based on the levenberg–marquardt neural network and power routing of DC–DC converters. *IEEE Trans Power Electron* 2023;38:10280–91. DOI

3. Ding Y, Ding P, Zhao X, Cao Y, Jia M. Transfer learning for remaining useful life prediction across operating conditions based on multisource domain adaptation. *IEEE/ASME Transn Mechatronics* 2022;27:4143–52. DOI

4. Yuan X, Xu N, Ye L, et al. Attention-based interval aided networks for data modeling of heterogeneous sampling sequences with missing values in process industry. *IEEE Trans Ind Informat* 2023;20:5253–62. DOI

5. Ge Z. Distributed predictive modeling framework for prediction and diagnosis of key performance index in plant-wide processes. *J Process Control* 2018;65:107–17. DOI

6. Wang S, Ju Y, Fu C, Xie P, Cheng C. A reversible residual network-aided canonical correlation analysis to fault detection and diagnosis in electrical drive systems. *IEEE Trans Instrum Meas* 2024;73:1–10. DOI

7. Zhao F, Jiang Y, Cheng C, Wang S. An improved fault diagnosis method for rolling bearings based on wavelet packet decomposition and network parameter optimization. *Meas Sci Technol* 2023;35:025004. DOI

8. Shang C, You F. Data analytics and machine learning for smart process manufacturing: recent advances and perspectives in the big data era. *Engineering* 2019;5:1010–6. DOI

9. Jiang Y, Yin S, Dong J, Kaynak O. A review on soft sensors for monitoring, control, and optimization of industrial processes. *IEEE Sensors J* 2021;21:12868–81. DOI

10. Yin S, Ding SX, Xie X, Luo H. A review on basic data-driven approaches for industrial process monitoring. *IEEE Trans Ind Electron* 2014;61:6418–28. DOI

11. Chen J, Fan S, Yang C, Zhou C, Zhu H, Li Y. Stacked maximal quality-driven autoencoder: deep feature representation for soft analyzer and its application on industrial processes. *Inform Sci* 2022;596:280–303. DOI

12. Åström KJ, Bell RD. Drum-boiler dynamics. *Automatica* 2000;36:363–78. DOI

13. Lampinen M, Laari A, Turunen I. Kinetic model for direct leaching of zinc sulfide concentrates at high slurry and solute concentration. *Hydrometallurgy* 2015;153:160–9. DOI

14. Zamani MG, Nikoo MR, Rastad D, Nematollahi B. A comparative study of data-driven models for runoff, sediment, and nitrate forecasting. *J Environ Manage* 2023;341:118006. DOI

15. Yan F, Zhang X, Yang C, Hu B, Qian W, Song Z. Data-driven modelling methods in sintering process: current research status and perspectives. *Can J Chem Eng* 2023;101:4506–22. DOI

16. Edeh E, Lo WJ, Khojasteh J. Review of partial least squares structural equation modeling (PLS-SEM) using R: a workbook. *Struct Equ Model* 2022;30:165-7. DOI

17. Hyvärinen A, Khemakhem I, Morioka H. Nonlinear independent component analysis for principled disentanglement in unsupervised deep learning. *Patterns* 2023;4:100844. DOI

18. Song P, Zhao C. Slow down to go better: a survey on slow feature analysis. *IEEE Trans Neural Netw Learn Syst* 2022;35:3416–36. DOI

19. Hardoon DR, Szedmak S, Shawe-Taylor J. Canonical correlation analysis: an overview with application to learning methods. *Neural Comput* 2004;16:2639–64. DOI

20. Li S, Zheng Y, Li S, Huang M. Data-driven modeling and operation optimization with inherent feature extraction for complex industrial processes. *IEEE Trans Autom Sci Eng* 2023;21:1092–106. DOI

21. Zhang Z, Tao D. Slow feature analysis for human action recognition. *IEEE Trans Pattern Anal Mach Intell* 2012;34:436–50. DOI

22. Zhou P, Lu SW, Chai T. Data-driven soft-sensor modeling for product quality estimation using case-based reasoning and fuzzy-similarity rough sets. *IEEE Trans Autom Sci Eng* 2014;11:992–1003. DOI

23. Zhang X, Zou Y, Li S. Enhancing incremental deep learning for FCCU end-point quality prediction. *Inform Sci* 2020;530:95–107. DOI

24. Shao W, Ge Z, Yao L, Song Z. Bayesian nonlinear Gaussian mixture regression and its application to virtual sensing for multimode industrial processes. *IEEE Trans Autom Sci Eng* 2019;17:871–85. DOI

25. Yuan X, Zhou J, Huang B, et al. Hierarchical quality-relevant feature representation for soft sensor modeling: a novel deep learning strategy. *IEEE Trans Ind Informat* 2020;16:3721–30. DOI

26. Reigosa D, Briz F, Charro CB, Garcia P, Guerrero JM. Active islanding detection using high-frequency signal injection. *IEEE Trans Ind Appl* 2012;48:1588–97. DOI

27. Zhang X, Deng X, Cao Y, Xiao L. Nonlinear predictable feature learning with explanatory reasoning for complicated industrial system fault diagnosis. *Knowl Based Syst* 2024;286:111404. DOI

28. Wang L, Mao M, Xie J, Liao Z, Zhang H, Li H. Accurate solar PV power prediction interval method based on frequency-domain decom-

position and LSTM model. *Energy* 2023;262:125592. DOI

29. Fazi MB. Beyond human: deep learning, explainability and representation. *Theory Cult Soc* 2021;38:55–77. DOI

30. Hosain MT, Jim JR, Mridha MF, Kabir MM. Explainable AI approaches in deep learning: advancements, applications and challenges. *Comput Electr Eng* 2024;117:109246. DOI

31. Jang K, Pilario KES, Lee N, Moon I, Na J. Explainable artificial intelligence for fault diagnosis of industrial processes. *IEEE Trans Ind Informat* 2023:1–8. DOI

32. Kidambi Raju S, Ramaswamy S, Eid MM, et al. Enhanced dual convolutional neural network model using explainable artificial intelligence of fault prioritization for industrial 4.0. *Sensors* 2023;23:7011. DOI

33. Ferraro A, Galli A, Moscato V, Sperlì G. Evaluating eXplainable artificial intelligence tools for hard disk drive predictive maintenance. *Artif Intell Rev* 2023;56:7279–314. DOI

34. Wang T, Zheng X, Zhang L, Cui Z, Xu C.   A graph-based interpretability method for deep neural networks.   *Neurocomputing* 2023;555:126651. DOI

35. Xie T, Grossman JC. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys Rev Lett* 2018;120:145301. DOI

36. Boudraa AO, Cexus JC. EMD-based signal filtering. *IEEE Trans Instrum Meas* 2007;56:2196–202. DOI

37. Kopsinis Y, McLaughlin S. Development of EMD-based denoising methods inspired by wavelet thresholding. *IEEE Trans Signal Process* 2009;57:1351–62. DOI

38. Xiong Z, Yao J, Huang Y, Yu Z, Liu Y. A wind speed forecasting method based on EMD-MGM with switching QR loss function and novel subsequence superposition. *Appl Energy* 2024;353:122248. DOI

39. Zhang S, Li X, Zong M, Zhu X, Wang R. Efficient kNN classification with different numbers of nearest neighbors. *IEEE Trans Neural Netw Learn Syst* 2017;29:1774–85. DOI

40. Kipf TN, Welling M. Semi-supervised slassification with graph convolutional networks. *arXiv* 2016. arXiv:1609.02907. Available online: https://arxiv.org/abs/1609.02907. (accessed 11 Jan 2025).

41. Lv M, Li Y, Liang H, et al. A spatial–temporal variational graph attention autoencoder using interactive information for fault Detection in complex industrial processes. *IEEE Trans Neural Netw Learn Syst* 2024;35:3062–76. DOI