Chinese Journal of Chemical Engineering 89 (2026) 132–144

Contents lists available at ScienceDirect

Chinese Journal of Chemical Engineering

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

Full Length Article

Attention-enhanced multi-time scale LSTM for soft sensor modeling of corn starch liquefaction

Yu Zhuang [1], Zhongyi Zhang [1], Jin Tao [2], Yi Li [2], Fan Li [2], Yu Wang [2], Lei Zhang [1], Jian Du [1,*]

[1] Institute of Chemical Process Systems Engineering, School of Chemical Engineering, Dalian University of Technology, Dalian 116024, China
[2] COFCO Biotechnology Co., Ltd., Beijing 100005, China

ARTICLE INFO

Article history:
Received 13 June 2025
Received in revised form
27 August 2025
Accepted 1 September 2025
Available online 27 October 2025

Keywords:
Multi-scale dilated causal convolution
Neural networks
Soft sensor
Systems engineering attention mechanism
Biochemical engineering

ABSTRACT

Data-driven deep learning modeling has been increasingly applied to quality prediction in complex chemical processes. However, the data show complex temporal features due to different residence times and strong coupling relationships among chemical entities. This study proposes a multi-scale temporal feature extraction module to extract local dynamic temporal features across different time scales and combines it with long short-term memory (LSTM) networks to capture global temporal patterns, thereby taking full advantage of available data. In addition, variable-wise channel attention is integrated into the model to enhance attention on the essential parts of the feature maps and improve predictive performance. Furthermore, by analyzing the attention weights, the model quickly identifies the key variables that significantly affect the predictions. Finally, the model is applied to a real corn starch liquefaction process and achieves an accurate product quality prediction with an R^2 value of 0.9392, which represents a 4% to 9% improvement over traditional models and demonstrates the superiority of the proposed approach.

1. Introduction

Corn is one of the world's top three food crops and a major source of starch. The corn processing sector is widespread around the world [1]. However, it faces notable challenges that include a long production process, complex production procedures, and low levels of automation and intelligence. These issues highlight the urgent need for intelligent transformation to improve efficiency. To address these challenges, modern chemical industries, including petroleum refining [2] and food processing [3], are increasingly focusing on intelligent manufacturing to achieve cost reduction, efficiency improvement, safe and green production [4,5]. These improvements require the implementation of operational parameter optimization, rapid decision-making, and advanced process control strategies [6] that rely on feedback from critical indicators in process production. In the enzymatic hydrolysis of corn starch to produce glucose, the dextrose equivalent (DE) serves as a key indicator of intermediate product quality and a vital reference for operational adjustments. However, due to the limited deployment of online instrumentation, many facilities still rely on offline laboratory sampling for DE analysis. This method introduces several hours of measurement delay, leading to lagging process monitoring and making it difficult to evaluate product quality in real time or to make timely operational adjustments to maintain high product standards.

The soft sensor provides a solution for online estimation of product quality. This technology can estimate hard-to-measure key variables by using easy-to-measure variables and building mathematical models. Soft sensor modeling can be generally categorized into two types: (1) first-principle models(FPMs) [7] and (2) data-driven models [8]. FPMs use physical and chemical knowledge to establish relationships between variables, which often require extensive domain knowledge and become increasingly difficult to model as the process scale increases. In contrast, data-driven methods, which have become increasingly popular in industry and academia [9], use the vast amounts of historical data available due to the widespread use of distributed control systems (DCS), require less expert knowledge, and have lower modeling cost and faster running modeling speeds. There are numerous data-driven soft measurement modeling methods, including those based on multivariate statistics and traditional machine learning

* Corresponding author.
E-mail address: dujian@dlut.edu.cn (J. Du).

https://doi.org/10.1016/j.cjche.2025.09.016

approaches, such as multiple linear regression (MLR), partial least squares (PLS) [10], principal component regression (PCR) [11], random forest [12], support vector regression (SVR) [13,14], gradient boosting regression [15], and some shallow artificial neural networks [16,17]. While these methods may be effective in relatively simple systems, they often struggle to achieve satisfactory predictive performance in large-scale chemical processes characterized by significant non-linearity and strong coupling. Deep learning techniques, through layer stacking and advanced network architecture design, can thoroughly extract various deep features from the data, thereby providing a better fit for the relationships between different variables in industrial production.

With the development of computing technology, deep learning has made rapid progress in computer vision, natural language processing, and other fields. Recently, deep learning has been increasingly applied to industrial processes to build data-driven models. Shang *et al.* [18] used deep belief networks (DBN) to develop a soft sensor model and achieved an accurate estimation of the 95% cut point of heavy diesel in a crude oil distillation unit. Similarly, Xie *et al.* [19] used a variational autoencoder (VAE) for soft sensor modeling to achieve good predictions even in cases of missing data. Yuan *et al.* [20] proposed a semi-supervised stacked autoencoder (SS-SAE) based on stacked autoencoders (SAE) to deal with the limited labeled data and is validated on two refining industries of a debutanizer column and a hydrocracking process. Typically, DBN and AE are used for static process models. However, real-world chemical processes often exhibit dynamic features, requiring deep learning models that can extract such dynamic features. These models include convolutional neural networks (CNN) [21, 22], long short-term memory networks (LSTM) [23–25], time convolutional networks (TCN) [26], and Transformer [27]. Hong and Tian [28] used LSTM to extract time-series relationships in catalytic cracking processes and optimized model hyperparameters with swarm intelligence algorithms (SIA), achieving an accurate prediction of reaction temperature. This simple application commonly focuses on global data information and does not fully exploit local data features. CNN is widely used for feature extraction of local information. Wang [22] developed a soft sensor model using CNN for real-time estimation of quality variables and extraction of local correlations between variables. Zha *et al.* [29] combined CNN with LSTM, first employing CNN for local feature extraction and then LSTM for time dependency learning. The results showed that this model outperformed LSTM alone in predicting gas field production. Rather than simply combining CNN with LSTM, Yuan *et al.* [30] proposed a multi-scale attention-based convolutional neural network (MSACNN), which extracted data features using convolutional kernels of different sizes and designed a channel-wise attention mechanism to further improve prediction performance. The superiority of the model was finally validated on the hydrocracking process dataset and the debutanizer column dataset. In contrast to traditional CNN that uses a full-range convolutional kernels to extract features on the feature maps, causal convolution [31] only uses past and present information for prediction, effectively preventing information leakage and better suiting process prediction. In chemical processes, the use of convolutional kernels of different sizes to extract causal features from data can facilitate improving the predictive effect of the final model by fully capturing dynamic time-series characteristics at different scales due to different residence times in different devices.

The attention mechanism is a powerful technique that assigns different weights to different parts of the data, enabling the model to automatically learn the relationships between data inputs and outputs during the modeling process. In recent years, attention mechanisms have been widely applied to data-driven modeling, effectively improving model prediction performance and enhancing model interpretability. Bi and Zhao [32] applied orthogonal self-attention and variational autoencoders to fault diagnosis and extracted the correlations between different variables and temporal dependency among different timesteps, so that fault detection and identification tasks can be effectively performed and interpretable results simultaneously obtained. Han *et al.* [23] combined attention mechanisms with LSTM for production analysis to explore the correlations between inputs and outputs. They calculated the weights of hidden layer units for prediction through attention computation. Attention mechanisms are also widely used in soft sensor modeling to enhance the recognition of the relationships between input and output variables, thereby improving the prediction accuracy of target variables [33–36].

Therefore, this paper proposes a novel model that combines multi-scale dilated causal convolutional layers with LSTM for time series data prediction. In addition, a spatial attention structure is introduced into the network, which improves the prediction performance of the model and provides insights into the impact of different input variables on the prediction results. Finally, the proposed model is successfully applied to a real starch liquefaction process. The main contributions of this paper are summarized as follows:

(1) A neural network structure is designed that combines multi-scaleconvolution and LSTM, enabling the extraction of temporal features at multiple scales, and better adapting to the residence time differences between different devices during the liquefaction process.
(2) The combination of the attention mechanism with the neural network renders the model local interpretability, while the analysis of the attention weights of the samples reveals the key input features that contribute to the outcome.
(3) The model is applied to the corn starch liquefaction process, achieving a preferable prediction accuracy of $R^2 = 0.9392$.

2. Methodology

In this section, the architecture of the attention-enhanced multi-time scale long short-term memory network (AMT-LSTM) is introduced. Key components of the network are briefly described, including the variable-wise channel attention mechanism, multi-scale temporal feature extraction module, and the long short-term memory (LSTM) network.

2.1. Basic network structure

(1) Variable-wise channel attention mechanism

Attention mechanisms dynamically adjust the weight distribution of input data in neural networks, enabling the model to focus on critical information. In computer vision, channel attention and spatial attention are widely used to enhance local features by computing region-specific attention scores. As shown in Fig. 1, drawing inspiration from channel attention, the proposed variable-wise channel attention mechanism dynamically weights features at the sample level to strengthen input feature representations. This approach allows for enhanced interpretability through weight visualization, revealing the relative importance of different features within localized samples.

To optimize computational efficiency, a window-level feature weighting strategy is adopted for each individual input sample. Instead of assigning independent weights to each timestep, this

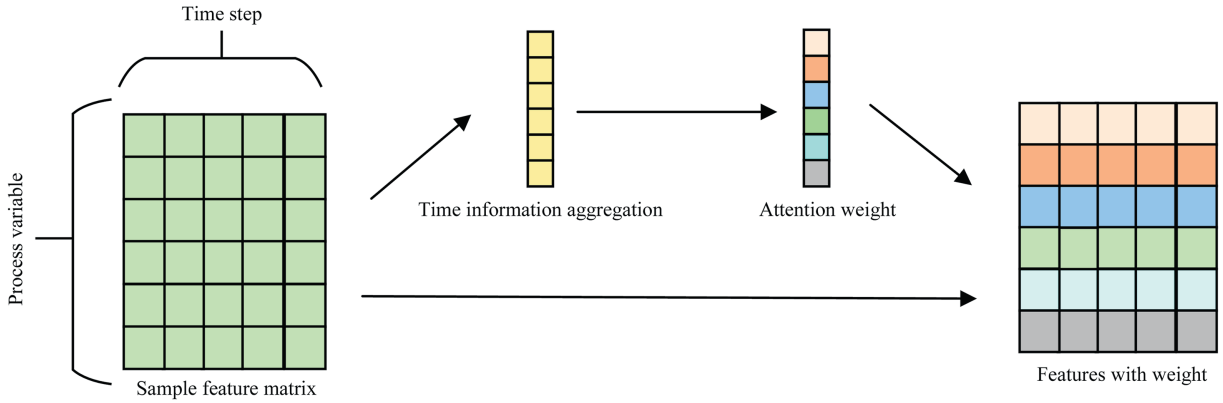

Fig. 1. Variable channel attention mechanism.

method aggregates features across the entire time series, reducing the number of parameters by a factor of T (where T is the sequence length). Afterward, a dimensionality expansion and reduction transformation is applied to obtain attention weights for each feature. These weights are then multiplied with the corresponding input features to produce attention-enhanced representations. This design significantly lowers computational resource consumption, mitigates convergence challenges, and improves modeling performance. The mechanism is formally described by Eqs. (1)–(3).

$$Z_i^{\text{avg}} = \frac{1}{T}\sum_{t=1}^{T} x_{i,t} \tag{1}$$

$$A_i = \sigma\left(\alpha \cdot \tanh\left(W_2^{\mathrm{T}}\text{Relu}\left(W_1^{\mathrm{T}} Z_i^{\text{avg}} + b_1\right) + b_2\right)\right) \tag{2}$$

$$X'_{i,t} = X_{i,t} \odot A_i + X_{i,t} \tag{3}$$

where X represents the input data sample, i represents the i-th feature, $Z \in \mathbb{R}^d$ is the average-pooled value, $W_1 \in \mathbb{R}^{d\times r}$ and $W_2 \in \mathbb{R}^{r\times d}$ are weight matrices, $b_1 \in \mathbb{R}^r$ and $b_2 \in \mathbb{R}^d$ denote bias terms, σ represents the sigmoid activation function, A corresponds to the computed attention weight matrix, and $\odot$ indicates element-wise multiplication.

(2) Multi-scale temporal feature extraction module

The multi-scale convolution technique employs convolutional kernels of varying sizes to extract hierarchical features across different temporal scales. This approach enhances the network's capability to capture multi-resolution information while improving robustness against frequency variations. Typical implementations utilize parallel convolutional branches with differently sized kernels to obtain multi-scale temporal representations.

In conventional temporal convolution operations, the convolution result at timestep t depends on both preceding and subsequent data points due to symmetric padding. As illustrated in Fig. 2, our causal convolution architecture employs front-zero-padding and output truncation mechanisms to preserve temporal causality. For each layer in the causal convolution network, the output at timestep t strictly depends on historical inputs up to t. For example, take the collected data of a certain variable i as an example, the data is $X_i = (x_{i,1}, x_{i,2}, \cdots, x_{i,\mathrm{T}})$ and the convolution kernel is $K = (w_1, w_2, \cdots, w_K)$, the standard convolution operation is shown in Eq. (4), whereas the computational formula using causal convolution is shown in Eq. (5). The standard convolution uses the first and last complementary zero padding length of $(K-1)/2$, while the use of causal convolution requires more padding on both sides of the length of K−1. After causal convolution, the right end of the extra $(K-1)/2$ data, you need to trim off the excess data to ensure that the length of the data is unchanged.

$$x'_{\text{conv}}[t] = \sum_{k=0}^{K-1} w[k] \cdot x[t \cdot s + k - p] \tag{4}$$

$$x'_{\text{causal}}[t] = \sum_{k=0}^{K-1} w[k] \cdot x[t \cdot s + k - (K-1)] \tag{5}$$

where $x'[t]$ denotes the output at timestep t, subscripts conv and causal distinguish standard and causal convolution operations, s represents the stride length, and p specifies the zero-padding length.

Traditional multi-scale convolution networks typically increase receptive fields through larger kernel sizes, which inevitably introduces additional parameters and computational complexity. To address this, we implement dilated convolution layers that strategically introduce spacing (dilation rates) between kernel elements. As shown in Fig. 3, this design achieves expanded receptive fields without parameter inflation while naturally suppressing high-frequency components through its inherent low-pass filtering characteristics. Our architecture employs two-layer dilated convolution blocks with constant dilation rates across parallel branches rather than progressively increasing rates in cascaded layers. This configuration ensures focused extraction of local temporal patterns while maintaining computational efficiency.

(3) long short-term memory network (LSTM)

LSTM is a special type of recurrent neural network that overcomes the problems of the vanishing gradient and the exploding gradients through a gated structure. It can process and predict time series data of certain lengths. LSTM primarily consists of input gates, output gates, and forget gates, the structure of which is illustrated in Fig. 4. Where X_t represents the input at time t. C_{t-1}, and C_t represent the cell state at time $t-1$ and t. h_{t-1}, and h_t represent the output at time $t-1$ and t. Tanh and σ are activation functions, and o_t, i_t, and f_t represent the output gate, input gate, and forget gate. The information transmission process of the LSTM is given by Eqs. (6)–(11).

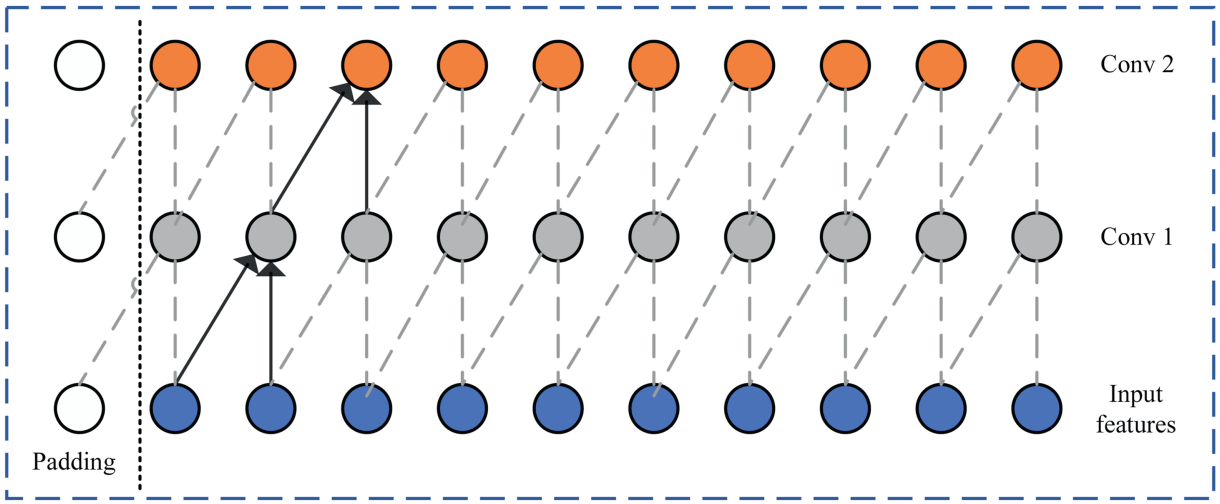

Fig. 2. Architecture of causal convolutional layer network.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{6}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{7}$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{8}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{9}$$

$$h_t = o_t \odot \tanh(C_t) \tag{10}$$

$$C_t = f_t \odot C_{t-1} + i_t * \tilde{C}_t \tag{11}$$

where $\tilde{C}_t$ represents the current cell state. W_f, W_i, W_o, and W_c are the weight matrices for the forget gate, input gate, output gate, and state update. b_f, b_i, b_o, and b_c are the biases for the forget gate, input gate, output gate, and state update, and $\odot$ represents the multiplication of vector elements. The forget gate and input gate selectivity update information about the cell state, while the output gate uses the activation function and cell state to produce the final output.

2.2. Attention-enhanced multi-time scale long short-term memory network model

In real-world chemical processes, process variables (such as pressure and concentration) exhibit distinct frequency differences. Furthermore, different devices have varying residence times, leading to disparate temporal features among variables. Traditional deep learning models, such as long short-term memory (LSTM), often struggle to effectively extract these multi-scale temporal features, resulting in performance bottlenecks when handling complex time-series data. To address this issue, our research employs multiple convolutional kernels with different receptive fields to capture local information at various time scales. By doing so, features of different scales can be synchronized in the model, allowing for a more comprehensive capture of local dynamic features among variables. Additionally, we utilize LSTM to extract global features, ensuring the integrity of overall information. This approach not only enhances the model's flexibility in handling complex time-series data but also improves its ability to extract diverse local features, ultimately providing a new method for analyzing dynamic features in chemical processes. The network structure of the 2.2 attention-enhanced multi-time scale long short-term memory network proposed in this study is shown in Fig. 5, and the data flow process in the model is as follows.

The model first employs a channel attention mechanism to extract spatial features from the input data. As shown in Eq. (12), the variable-wise channel attention structure obtains weight coefficients for each input feature variable. Subsequently, the attention matrix is multiplied with the input variables to enhance their features. Finally, through residual connections, the original input data and the enhanced data are added together to form an attention-enhanced feature map, thereby increasing the feature information contained in the data samples.

$$X'_{d,t} = X_{d,t} + \text{Attention}\left(X_{d,t}\right), 1 < d < D, 1 < t < T \tag{12}$$

Next, the enhanced input sample feature map is fed into the multi-scale temporal feature extraction module to effectively extract local dynamic features of different sizes from the time-series data. In this module, the model uses multiple one-dimensional convolutional kernels with different sizes in parallel. When the kernel size is greater than 5, dilated convolutions are used to form convolutional kernels with different receptive fields. As shown in Eq. (13), the convolutional kernels with different receptive fields are applied to the sample feature map through convolutional operations, generating multiple feature maps of the same size that are consistent in time steps. Finally, as shown in Eq. (14), these feature maps are concatenated along the feature dimension to form a composite feature map with multi-scale temporal features.

$$H_t^{(\text{causal},i)} = \sum_{k=0}^{K_i-1}\left(W_k^{(\text{causal},i)} \cdot X'_{t-d_i \cdot k,d} + b_k^{(\text{causal},i)}\right), \quad t \geq d_i(K_i - 1) + 1 \tag{13}$$

$$H_t^{\text{causal}} = \text{Concat}\left(H_t^{(\text{causal},1)}, H_t^{(\text{causal},2)}, \cdots, H_t^{(\text{causal},i)}\right) \tag{14}$$

The concatenated feature map with multi-scale temporal features is then fed into the LSTM network, where long-term sequential features inherent in the time-series data are gradually extracted. This effectively captures the global temporal features

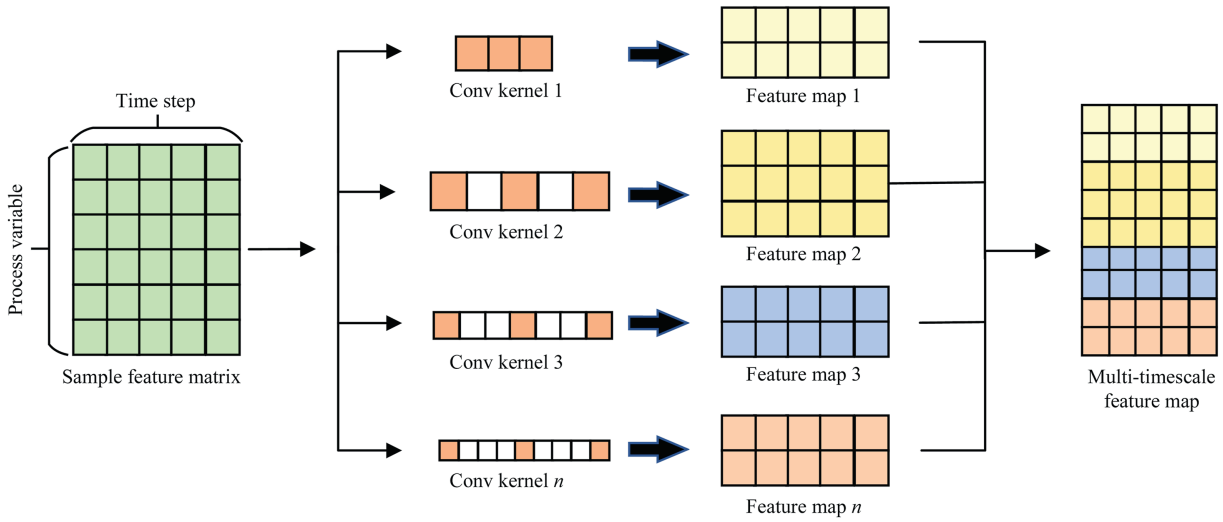

Fig. 3. Architecture of multi-scale dilated convolutional network.

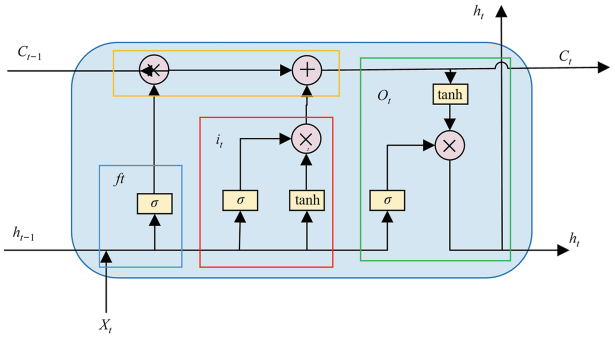

Fig. 4. Structure of the LSTM model.

while simultaneously reducing the dimensionality of the feature space, as shown in Eq. (15).

$$H_t^{\text{lstm}} = \text{LSTM}\left(H_t^{\text{causal}}\right) \tag{15}$$

The model ultimately employs a multi-layer perceptron (MLP) layer to perform regression tasks. The output data from the LSTM layer is flattened to meet the input requirements of the fully connected layer. The fully connected layer is responsible for predicting the regression target variable, which is achieved through linear transformations and non-linear activations of the flattened data, ultimately producing the model's output result.

$$\widehat{Y} = \phi\left(W^{\text{mlp}} \cdot \text{Flatten}\left(H_t^{\text{lstm}}\right) + b^{\text{mlp}}\right) \tag{16}$$

In Eqs. (12)–(16), Attention(·) and LSTM(·) denote channel-wise attention and LSTM-based feature extraction, respectively. Concat (·) represents the concatenation of different feature maps. Flatten(·) indicates the flattening of 2D data. *X*, *H*, and *Y* represent the input data, intermediate features, and model output, respectively, with different superscripts indicating the results of different modules.

2.3. Data-driven soft sensor development process

The modeling process framework for real-time estimation of product quality based on the proposed model is shown in Fig. 6, which consists of the following three steps:

Step 1. Data preprocessing and sample splitting

After determining the prediction target variable, we first combine process knowledge and analyze historical data characteristics [37] to perform feature engineering and select an appropriate subset of input variables. Next, we collect relevant historical data and perform missing value imputation, outlier treatment [38], noise reduction, and other preprocessing steps to improve data quality and ensure the accuracy of subsequent modeling [39]. Then, sliding window technology is used to expand 1-D data into 2-D data to capture time series features. Finally, the processed data will be divided into training sets, validation sets, and test sets for subsequent model training and evaluation.

Step 2. Offline training

The first step in offline training is to determine the model structure. Next, a set of initial hyperparameters and model parameters is established. Using the training set, multiple iterations are conducted to continuously optimize the model gradients and weights based on the loss function. Throughout the training process, the accuracy of the validation set is continuously monitored to adjust the number of iterations, evaluate the efficacy of hyperparameters, and ascertain the optimal conditions for the model performance on the validation set. Additionally, various optimization algorithms, such as genetic algorithm, particle swarm algorithm, or Bayesian optimization, can be integrated during model training and validation to identify a more optimal set of hyperparameter settings. Finally, the trained neural network model is evaluated on the test set to verify its generalization ability and to ensure its overall efficiency and practicality.

Step 3. Online use

After completing offline training, real-time data is input into the trained model to enable online product quality estimation. Concurrently, real-time data is continuously updated in the historical database. It is important to note that devices may experience model drift due to equipment aging and other operational problems, affecting prediction performance. Therefore, it is essential to regularly repeat the above training process to update the model, ensuring that its prediction performance remains consistently high.

In this study, the Adam optimizer is selected, one of the most widely used optimizers in deep learning, to enhance model performance. The mean absolute error (MAE) was employed as the

loss function to guide the direction of the model gradient and parameter optimization. To evaluate the model fitting effectiveness, the coefficient of determination (R^2) is regarded as the assessment metric. Additionally, an early stopping strategy is implemented to prematurely halt the training iterations, thereby preventing the unnecessary consumption of training resources. The MAE, RMSE and R^2 formulas are presented in Eqs. (17)–(19).

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\widehat{y}_i - y_i| \tag{17}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(\widehat{y}_i - y_i)^2}{n}} \tag{18}$$

$$R^2 = \frac{1 - \sum_{i=1}^{N}\left(\widehat{y}_i - y_i\right)}{\sum_{i=1}^{N}\left(\widehat{y}_i - \overline{y}\right)} \tag{19}$$

where N is the number of samples. $\widehat{y}$, y and $\overline{y}$ are the predicted output value, true value, and mean value of the sample, respectively.

3. Case study: Starch liquefaction process

3.1. Introduction of starch liquefaction process

A case study of the corn starch liquefaction process is presented, which is sourced from the corn deep processing factory of Jilin province. The flowchart illustrating this process is shown in Fig. 7. This process uses the enzymatic hydrolysis method to liquefy corn starch, which includes the steps of corn starch emulsion preparation, starch gelatinization and retrogradation, and starch enzyme hydrolysis reaction. The process consists of heat exchangers (E-1, E-2, and E-3), storage tanks (V-1, V-2, and V-3), flash tank (F-1, F-2, and F-3), high-temperature injectors (I-1 and I-2), high-temperature maintenance pipes (M-1 and M-2), liquefaction reactors (R-1, R-2, R-3, and R-4), and pumps (P-1, P-2, P-3, P-4, P-5, and P-6). Firstly, the process begins with corn starch emulsion from the wet grinding workshop, which is heated and diluted to achieve the desired concentration and temperature. The alkaline solution is added to storage tank V-1 to adjust the pH value. The enzyme is then introduced into storage tank V-2, and the mixture is thoroughly stirred. Next, the emulsion is injected into the high-temperature injector, where it is mixed with high-temperature steam to undergo high-temperature gelatinization. After the emulsion undergoes flash evaporation and cooling, it is subjected to a second injection, and then the emulsion is sent to the liquefaction reactor R-1. The emulsion is hydrolyzed by the enzyme to break the 1,4-glycosidic bond, resulting in dextrin, maltose, maltotriose and maltopentose [40]. Finally, the emulsion is subjected to further processing and quality inspection and then sent to the sugar production section. The hydrolysis of corn starch is a typical bioprocess that exhibits pronounced multi-scale temporal dynamics and complex multivariate coupling. Firstly, different variables in the process fluctuate at different frequencies, and the residence times of materials in different units vary significantly—from a few minutes to several tens of minutes—resulting in multi-timescale temporal characteristics. Secondly, due to the complex physicochemical changes involved in starch transformation and the mutual influence of process variables across equipment units, the process exhibits a high degree of inter-variable coupling.

In the production process, the factory must monitor and control the dextrose equivalent (DE) value of the liquefied product. The DE value [41] indicates the percentage of reducing sugar in the dry matter of sugar syrup. If the DE value is too low, the emulsion is prone to retrogradation, resulting in high viscosity, which is not conducive to operation. Additionally, high molecular weight dextrin that is not completely liquefied will be brought into the sugarization process, directly affecting the quality of saccharification. Conversely, if the DE value is too high, it is not conducive to the formation of the complex structure between the enzyme and the substrate in the saccharification process, reducing catalytic efficiency. Currently, due to technical and economic constraints, the factory relies on offline sampling at regular intervals to measure the DE value of the emulsion, which introduces significant time delays. Therefore, through data-driven modeling, rapid estimation of the product DE value is of great significance for improving product quality, and immediate changes to operation strategies can be made accordingly.

The data collection period spanned from February 1, 2023, to April 30, 2023. After feature engineering based on worker operation experience and data analysis, a total of 25 process variables were selected to establish a data-driven model for the target variable. The variables are shown in Table 1. The dataset consists of 3310 data samples and was divided into training sets, validation sets, and test sets in a ratio of 8:1:1.

3.2. Selection of sliding window size

The sliding window [42] is a widely used technique for time series prediction, as illustrated in Fig. 8. By defining a window within the dataset, it is moved forward along the time axis, transforming the data from a 1D input variable to a 2D input variable with time steps, thereby obtaining the dynamic characteristics of the data to deal with data delays. However, the sliding window size selection is an optimized target variable. If the window size is too small, it may not adequately cover the entire process residence time, resulting in the omission of important information and poor model prediction performance. Conversely, if the window size is too large, it may introduce redundant information, increasing the number of parameters required by the model and complicating the training process. This can lead to premature convergence and a decline in model performance.

The selection of the sliding window size is an optimization target. If the window size is too small, it cannot cover the entire process residence time, resulting in incomplete information and poor model prediction performance. On the other hand, if the window size is too large, redundant information is included in the data, leading to increased model parameters and training difficulty, causing the model to converge prematurely and resulting in poor performance. To reduce the optimization cost of subsequent models, this study selects a time length of 100 min, covering a complete cycle of starch milk flow after liquefaction treatment, resulting in a fixed time window size of 20 time steps.

3.3. Hyperparameter optimization

Hyperparameters are parameters in machine learning that must be set before training, such as learning rate, network structure and optimizer. Unlike model parameters, hyperparameters cannot be learned directly from the training data. Inappropriate hyperparameters can lead the network to local optima, significantly affecting model performance. Therefore, adjusting

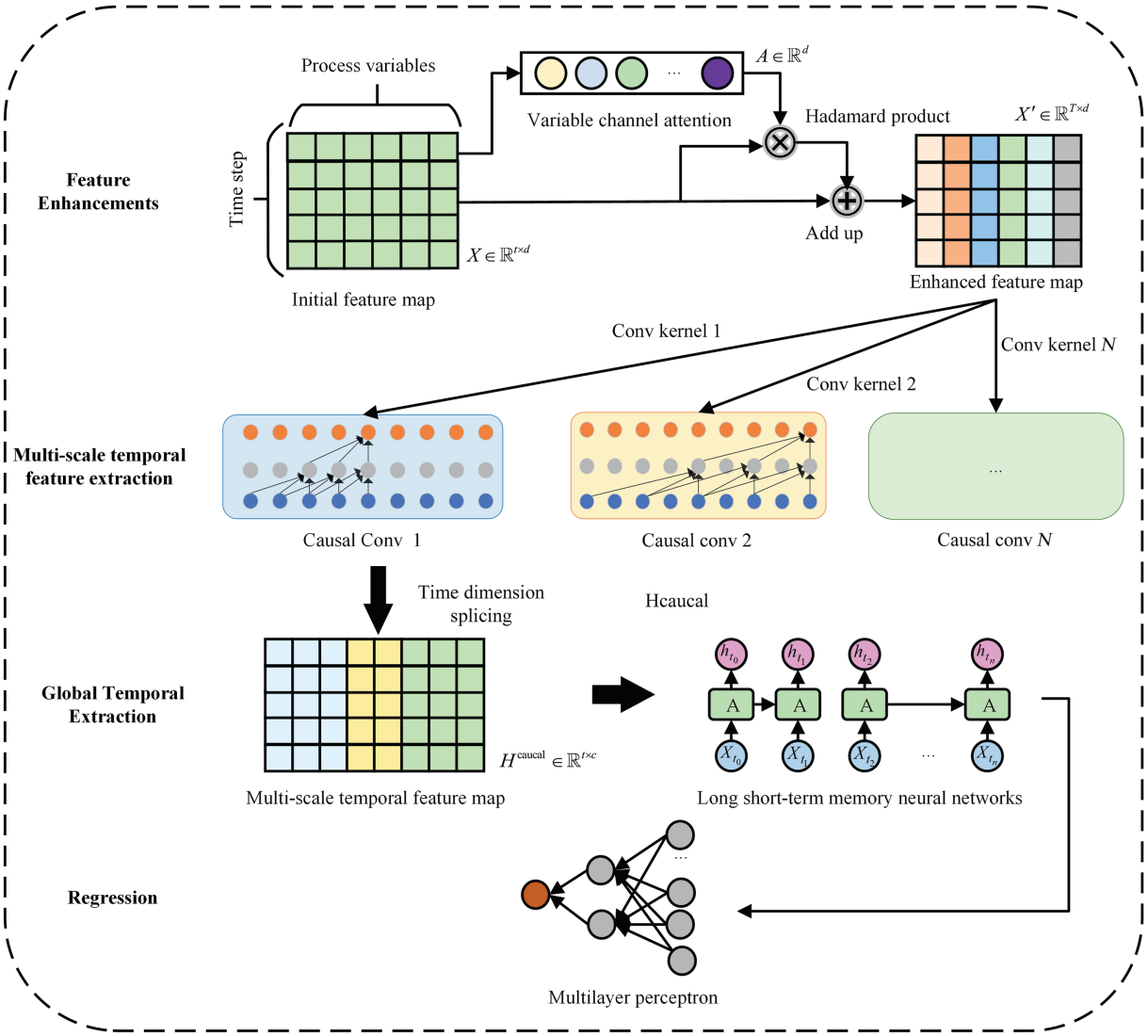

Fig. 5. The structure of the AMT-LSTM.

hyperparameters has a significant impact on model performance. Compared to the grid search method and genetic algorithm, the Bayesian optimization [43] can find a better hyperparameter combination with fewer iterations. Bayesian optimization builds a probabilistic surrogate model based on historical evaluations of the objective function and gradually finds the set of hyperparameters that minimizes or maximizes the objective function through optimization of the acquisition function. When using Bayesian optimization for hyperparameters, it is generally necessary to determine several components: the objective function, which is the metric to be minimized or maximized. In this study, the objective is to maximize the R^2 value on the validation set. There is also the domain space, which is the range of values for the hyperparameters to be searched, primarily including network architecture and some training parameters. The probabilistic surrogate model is used to model the objective function, with common functions including Gaussian process regression (GP), sequential model-based algorithm Configuration (SMAC), and Tree-structured Parzen Estimator (TPE). The acquisition function is used to explore the next sampling point, commonly including expected improvement (EI), probability of improvement (PI), and upper confidence bound (UCB). Additionally, the results history stores the results of the objective function evaluations. In this study, since it takes about 10 min to train a set of hyperparameters, using Bayesian optimization can help reduce the training costs of the model.

Optuna [44] is an open-source Python library for hyperparameter optimization that can be used to implement TPE Bayesian optimization easily. In this study, the main optimized hyperparameters include the parallel dilated causal convolution structure, the LSTM network structure, the MLP layer structure, and the learning rate of the model training. Specifically, the initial kernel size for the convolutional layers is set to 3, and for each additional parallel convolutional layer, a dilated convolution is added with the dilation rate increasing exponentially by a factor of 2. In addition to the above hyperparameters, this study also set an early stopping strategy with a patience value of 100, a batch size of 128 for model training, and used the Adam optimizer for optimization. The range of each hyperparameter and the optimization results are summarized in Table 2. Where N_{branch} is the number of

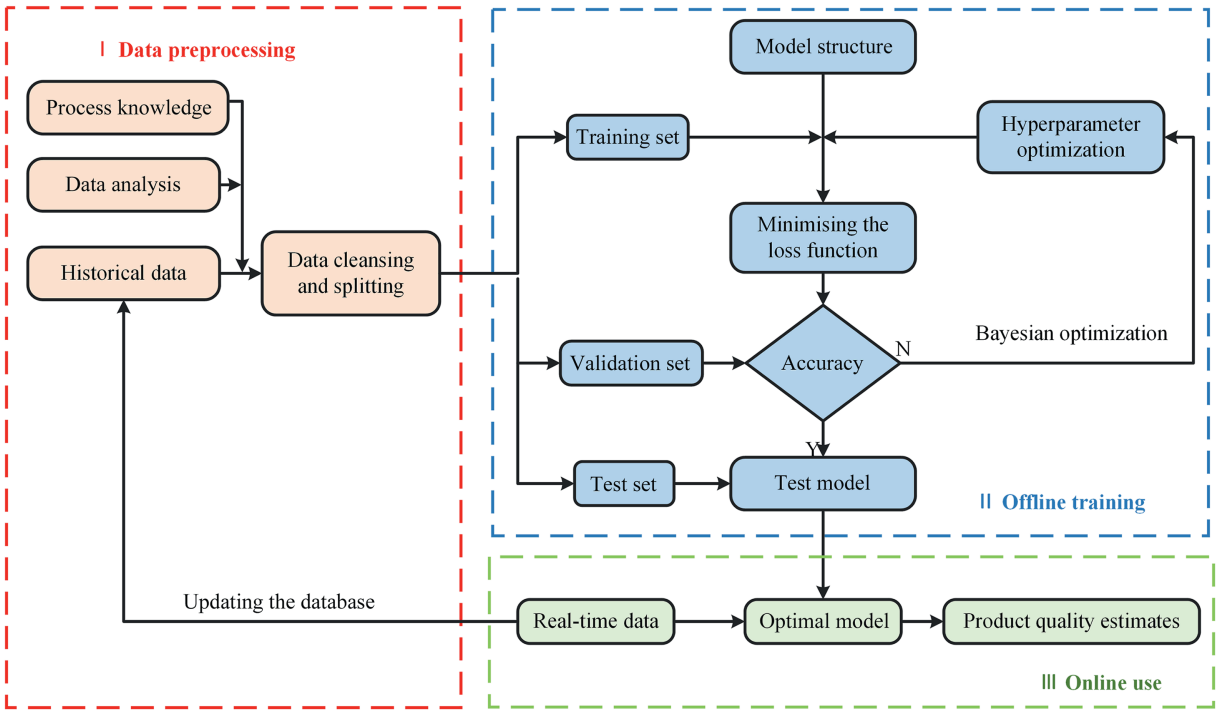

Fig. 6. Framework of soft sensor modeling based on AMT-LSTM.

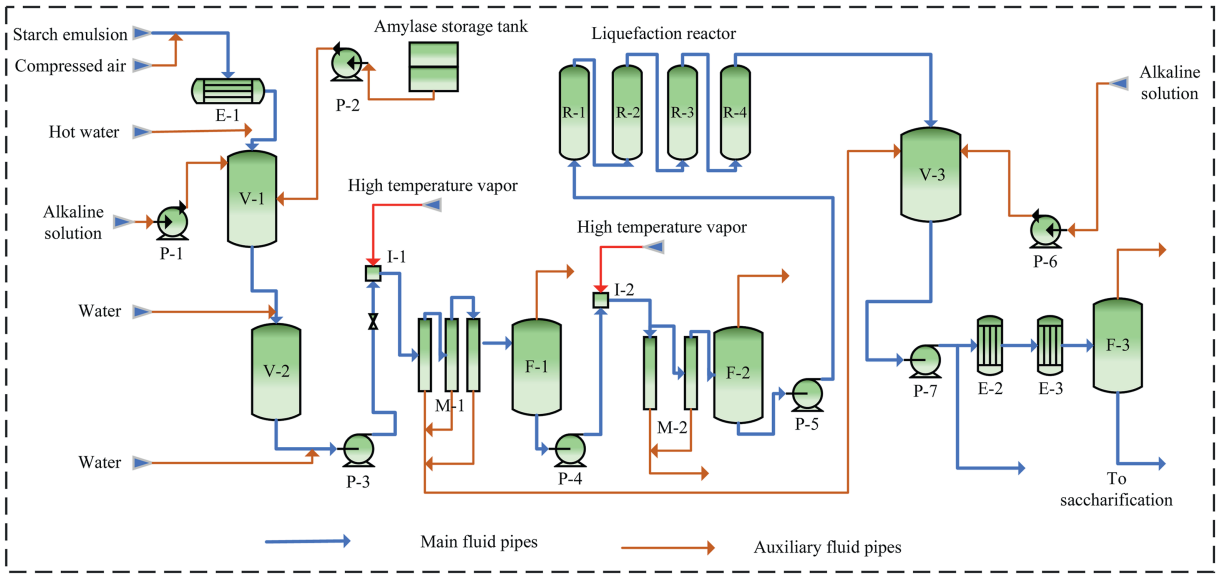

Fig. 7. Flowchart of starch liquefaction.

parallel convolutional layers. $C_{channel}$ is the number of hidden units in convolutional layers. L_{lstm} is the number of layers in a stacked LSTM layer. H_{lstm} is the number of hidden units in each layer of LSTM. L_{mlp} is the number of linear layers in a fully connected layer. H_{mlp} is the number of hidden units in a fully connected layer. Lr is the learning rate used in the model training.

4. Results and Discussion

A data-driven model was developed based on the corn liquefaction process to perform a soft sensor of the DE value of the liquefied product. The model performance was compared with various predictive models, and some of the prediction results were analyzed.

4.1. Accuracy comparison of different models

To validate the superior performance of the proposed AMT-LSTM model in temporal modeling, five representative models were selected for comparative analysis: multi-layer perceptron (MLP), CNN, LSTM, temporal convolutional network (TCN), and Transformer. It should be noted that while MLP is not specifically designed for temporal modeling and CNN can process sequential information, LSTM, TCN, and Transformer are all established deep

Table 1
Input and output variables of the starch liquefaction process.

Output: The DE value of liquefied liquid	
Inputs:	
X_1	Pressure I of injector (I-1)
X_2	Pressure II of injector (I-1)
X_3	Level of configuration tank (V-2)
X_4	Pressure of injector (I-2)
X_5	Level of flash tank (F-3)
X_6	Flowrate of alkaline solution
X_7	Control of pump (P-1) for alkaline solution
X_8	Steam conditioning valve for injectors (I-2)
X_9	Outflow from starch emulsion tank (V-2)
X_{10}	Temperature of the inlet to maintenance tube (M − 1)
X_{11}	Temperature of inlet to flash tank (F-1)
X_{12}	Control of injector (I-2)
X_{13}	Temperature of PH conditioning tanks (V-3)
X_{14}	Temperature of liquefaction reactor
X_{15}	Pressure of flash tanks (F-1)
X_{16}	Inflow of starch emulsion from the previous process
X_{17}	Inflow to the liquefaction reactor
X_{18}	Control of pump (P-5) for liquefaction reactors
X_{19}	Quantity of liquefied enzyme
X_{20}	Signal feedback II from injector (I-1)
X_{21}	Pressure of inlet to flash tank (F-1)
X_{22}	Dry matter content of starch emulsion
X_{23}	Temperature of the inlet to the maintenance tube (M-2)
X_{24}	Display of pump for liquefaction into the reactor
X_{25}	Dry matter content of configuration tank (V-2)

learning models for time series analysis. Table 3 presents the MAE, RMSE, and R^2 metrics of the six models across training and testing sets. As evidenced in Table 3, MLP and CNN models exhibit significantly lower R^2 values compared to dedicated temporal models. Although LSTM and Transformer demonstrate superior performance on training data, their test set performance degrades substantially, indicating poor generalization capability. The TCN network achieves the second-best performance on the validation set after the proposed AMT-LSTM model, showing the lowest degree of overfitting. The proposed AMT-LSTM model demonstrates significant advantages across all metrics, achieving 3% to 8% improvement in R^2 compared to conventional deep learning models for temporal modeling.

Fig. 9 presents a comparison of the predicted and actual values of all models. By contrast, we find that MLP merely flattens the input 2D matrix and performs linear and nonlinear

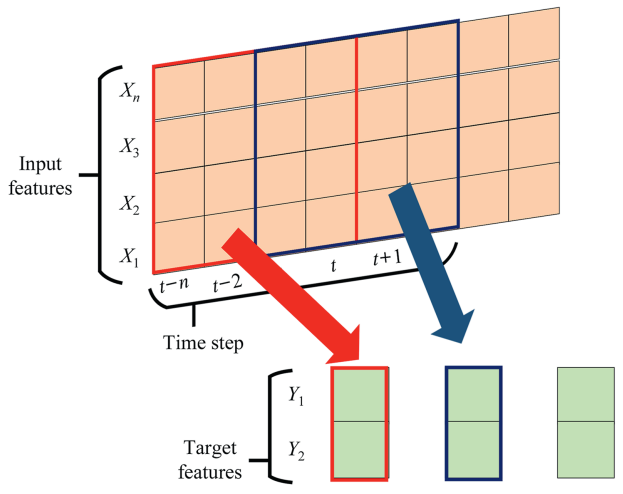

Fig. 8. Sliding window transformation from 1D to 2D data.

Table 2
The optimized hyperparameter for the AMT-LSTM model.

Name of the variable	Range of values	Final selection result
N_{branch}	2–6	3
$C_{channel}$	15–40	25,30,30
L_{lstm}	1–4	2
H_{lstm}	30–80	42,80
L_{mlp}	1–4	2
H_{mlp}	40–200	1,44
Lr	0.0005–0.005	0.0015

Table 3
The MAE, RMSE and R^2 of the studied deep learning models on training, and testing datasets.

Model	Training			Testing		
	MAE	RMSE	R^2	MAE	RMSE	R^2
MLP	0.6552	0.9234	0.1778	0.5566	0.7515	0.1558
CNN	0.3297	0.5747	0.7649	0.3786	0.6195	0.5479
LSTM	0.1019	0.1620	0.9700	0.2027	0.308	0.8582
TCN	0.1921	0.3338	0.9147	0.1685	0.3002	0.9037
Transformer	0.0896	0.1511	0.9780	0.1809	0.3097	0.8566
AMT-LSTM	0.0802	0.1340	0.9827	0.1111	0.2016	0.9392

transformations, failing to capture the temporal features across different time steps within the data, resulting in poor performance in the time-series dynamic data regression task. The traditional CNN model extracts correlations between variables along the time dimension using convolutional kernels, its results are oversimplified. It tends to focus more on local correlations between variables, making it difficult to capture long-term global features. LSTM and Transformer can capture global dynamic features, enabling them to fit the overall curve. However, during the information transmission process, they gradually lose some local details between features, resulting in large errors between predicted and actual values. The TCN model, on the other hand, captures long-term dynamic features through causal convolution and also extracts local dynamic features, leading to relatively better prediction performance. However, the model's use of dilated convolutions means it is not adept at performing multi-scale feature extraction on the same time-series data, resulting in a weaker ability to capture local features of different frequencies and scales. In contrast, AMT-LSTM, through its parallel dilated causal convolution, captures dynamic features at different time scales and combines the strengths of recurrent networks in extracting long-term features, making it more sensitive to local dynamic changes and better able to capture local features of varying sizes, thus producing predicted values that are closer to the actual values.

To visually illustrate the prediction differences among the models, a box plot was created, as shown in Fig. 10. The box plot indicates that the prediction errors of the AMT-LSTM model are more tightly clustered around zero, featuring a smaller interquartile range and fewer outliers. Additionally, the maximum error sample value for the AMT-LSTM model is lower than that of the other models. This demonstrates the superior performance of the AMT-LSTM model.

4.2. Ablation experiment

Ablation experiments are conducted to evaluate the impact of individual components on the overall performance of the deep learning model by modifying its partial structure. This section aims to verify the effectiveness of each module in the proposed AMT-LSTM model by comparing four variants: 1) AM-LSTM, which

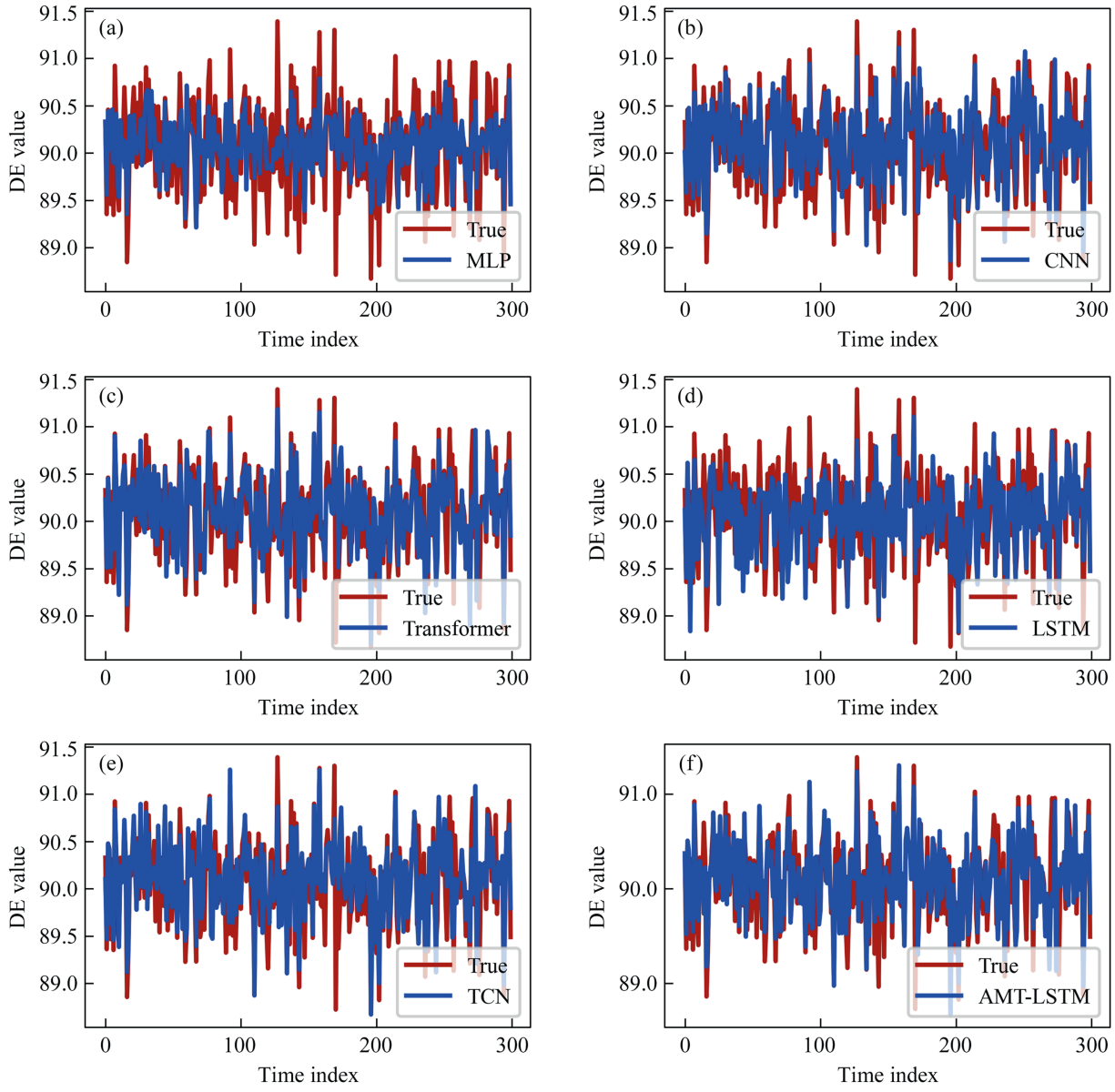

Fig. 9. Comparison of predicted and true values of the six models: (a) MLP, (b) CNN, (c) Transformer, (d) LSTM, (e) TCN, (f) AMT-LSTM.

only contains the variable channel attention mechanism; 2) MT-LSTM, which only includes the multi-scale temporal feature extraction layer; 3) AMT-LSTM (without dilation), which uses an attention-enhanced module without dilation; 4) the complete AMT-LSTM network. With consistent hyperparameters for the same modules, the models are trained, and the prediction accuracy and fitting curves of the real and predicted values are shown in Table 4.

Through the ablation experiment, it is found that the main contribution to the performance improvement of the model comes from the multi-scale temporal feature extraction module, which extracts temporal features of different scales using convolutional kernels with multiple receptive fields. Moreover, the use of dilated convolutional kernels instead of regular convolutional kernels is more suitable for mathematical modeling of the corn starch liquefaction process, resulting in a 2.1% increase in the R^2 index. Additionally, the channel attention mechanism assigns dynamic weights to different process variables, slightly improving the model's prediction accuracy by approximately 1% of the R^2 index. Although MT-LSTM has a slightly higher accuracy on the validation set than AMT-LSTM, its accuracy on the test set is lower, indicating that the channel attention mechanism can significantly reduce the model's overfitting. Through the combined effect of these modules, the AMT-LSTM model outperforms the ablated models in terms of MAE, RMSE, and R^2 on the independent test set, fully verifying the applicability of the proposed model for corn starch liquefaction process modeling.

4.3. Attention weighting analysis

The trained AMT-LSTM model effectively fits the mapping relationship between the input variables and the quality variables. By analyzing the weights of the attentional layers in the AMT-LSTM network, we determine which weights are occupied by the different variables in the current sample for the predicted target. For the analysis, we selected a relatively low-quality sample of the target. We visualized the channel attention in the attention structure as a bar chart, as shown in Fig. 11.

From the analysis of the attention weights, it can be seen that we can identify the most critical input variables in each sample,

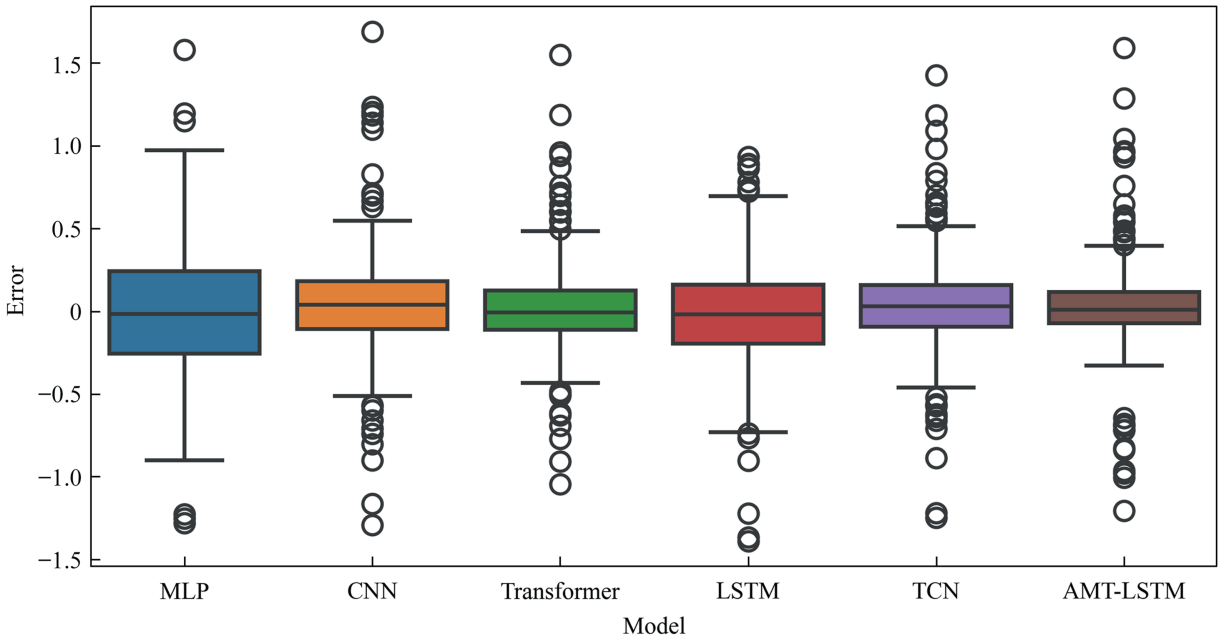

Fig. 10. Box plots of prediction errors of six models.

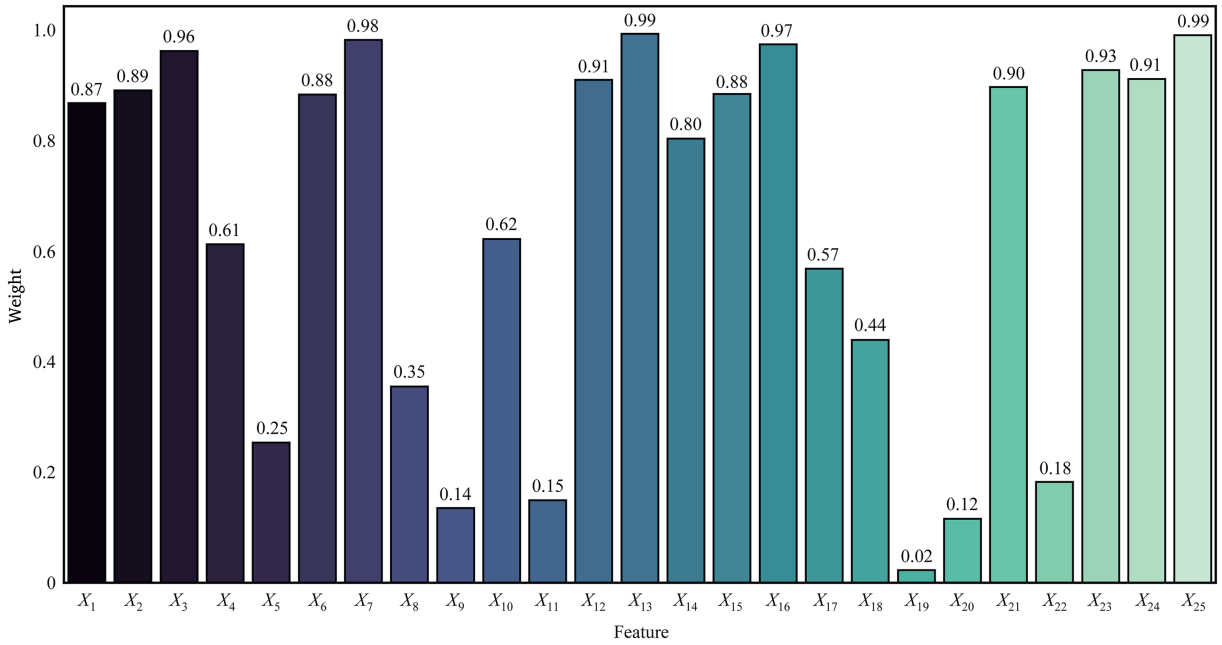

Fig. 11. Bar of attention weights for the sample.

thus revealing the local interpretability within the model to a certain extent and enhancing the credibility of the model. In addition, when there is a low-quality product, the model can analyze the most important input variables at the moment by using the attention weights. This can help engineers perform root cause analysis based on the importance ranking of input variables, facilitating a more effective operational response to achieve improvements in product specifications within the optimal range.

Table 4
Fitting effects of the ablation experimental model on the training and testing sets.

Model	Training			Testing		
	MAE	RMSE	R^2	MAE	RMSE	R^2
AM-LSTM	0.1033	0.1656	0.9736	0.1579	0.2632	0.8614
MT-LSTM	0.086	0.1400	0.9811	0.1040	0.1896	0.9282
AMT-LSTM(no dilation)	0.0841	0.1493	0.9824	0.1161	0.2082	0.9134
AMT-LSTM	0.0802	0.1340	0.9827	0.0943	0.1743	0.9392

5. Conclusions

To address the complex temporal characteristics in the starch liquefaction process data, this paper proposes a novel AMT-LSTM model. The model effectively extracts multi-time scale information from the data by integrating a parallel multi-scale dilated causal convolution with the LSTM network. Compared to traditional models, this method achieves an R^2 value of 0.9392 in the corn starch liquefaction case, demonstrating its high accuracy. Moreover, the model introduces a spatial attention mechanism, which automatically assigns weights to different process variables. By analyzing the attention heat map of individual samples, the model identifies key variables that significantly influence the outcome. This process provides valuable references for workers in formulating operational strategies or optimizing operational parameters.

Currently, the input format of the data is still a sequence. However, in the chemical industry, different operating variables exist in a complex non-Euclidean space. Therefore, further research is needed to find more suitable data representation methods, such as using graph-based structures [45–47] to represent the input data. In this approach, variables are treated as nodes in the graph, with connections between nodes determined by the underlying relationships among variables. This structure can integrate the prior knowledge of the process into the internal structure of the model, thereby enhancing the generalization and credibility of the model. However, constructing such an explicit graph structure is inherently challenging and constitutes a key focus of our future research.

CRediT Authorship Contribution Statement

Yu Zhuang: Writing – review & editing, Methodology, Funding acquisition, Conceptualization. Zhongyi Zhang: Writing – original draft, Validation, Software, Methodology, Conceptualization. Jin Tao: Software, Resources. Yi Li: Resources, Project administration. Fan Li: Visualization, Project administration, Investigation. Yu Wang: Project administration, Investigation, Data curation. Lei Zhang: Writing – review & editing, Validation, Software. Jian Du: Writing – review & editing, Supervision, Funding acquisition.

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.

Acknowledgements

The authors would like to gratefully acknowledge the financial support provided by the Special Foundation for State Major Basic Research Program of China (2021YFD2101005) and National Natural Science Foundation of China (22478057, 22178045).

References

[1] D.J. Rose, G.E. Inglett, S.X. Liu, Utilisation of corn (Zea mays) bran and corn fiber in the production of food components, *J. Sci. Food Agric.* 90 (6) (2010) 915–924.
[2] Q.F. Min, Y.G. Lu, Z.Y. Liu, C. Su, B. Wang, Machine learning based digital twin framework for production optimization in petrochemical industry, *Int. J. Inf. Manag.* 49 (2019) 502–519.
[3] P.P. Mondal, A. Galodha, V.K. Verma, V. Singh, P.L. Show, M.K. Awasthi, B. Lall, S. Anees, K. Pollmann, R. Jain, Review on machine learning-based bioprocess optimization, monitoring, and control systems, *Bioresour. Technol.* 370 (2023) 128523.
[4] H. Cartwright, S. Curteanu, Neural networks applied in chemistry. II. Neuro-evolutionary techniques in process modeling and optimization, *Ind. Eng. Chem. Res.* 52 (36) (2013) 12673–12688.
[5] M. Kano, Y. Nakagawa, Data-based process monitoring, process control, and quality improvement: recent developments and applications in steel industry, *Comput. Chem. Eng.* 32 (1–2) (2008) 12–24.
[6] Z.Y. Pang, P. Huang, C. Lian, C. Peng, X.C. Fang, H.L. Liu, Data-driven prediction of product yields and control framework of hydrocracking unit, *Chem. Eng. Sci.* 283 (2024) 119386.
[7] Y.C. Jiang, S. Yin, J.W. Dong, O. Kaynak, A review on soft sensors for monitoring, control, and optimization of industrial processes, *IEEE Sens. J.* 21 (11) (2020) 12868–12881.
[8] Z.Q. Ge, Z.H. Song, S.X. Ding, B. Huang, Data mining and analytics in the process industry: the role of machine learning, *IEEE Access* 5 (2017) 20590–20616.
[9] M.R. Dobbelaere, P.P. Plehiers, R. Van de Vijver, C.V. Stevens, K.M. Van Geem, Machine learning in chemical engineering: strengths, weaknesses, opportunities, and threats, *Engineering* 7 (9) (2021) 1201–1211.
[10] R. Sharmin, U. Sundararaj, S. Shah, L. Vande Griend, Y.J. Sun, Inferential sensors for estimation of polymer quality parameters: industrial application of a PLS-based soft sensor for a LDPE plant, *Chem. Eng. Sci.* 61 (19) (2006) 6372–6384.
[11] E. Marengo, M. Bobba, E. Robotti, M.C. Liparota, Modeling of the polluting emissions from a cement production plant by partial least-squares, principal component regression, and artificial neural networks, *Environ. Sci. Technol.* 40 (1) (2006) 272–280.
[12] Z. Zhou, C. Qiu, Y.F. Zhang, A comparative analysis of linear regression, neural networks and random forest regression for predicting air ozone employing soft sensor models, *Sci. Rep.* 13 (1) (2023) 22420.
[13] N. Parveen, S. Zaidi, M. Danish, Development of SVR-based model and comparative analysis with MLR and ANN models for predicting the sorption capacity of Cr(VI), *Process. Saf. Environ. Prot.* 107 (2017) 428–437.
[14] M.M. Zhang, X.G. Liu, A soft sensor based on adaptive fuzzy neural network and support vector regression for industrial melt index prediction, *Chemometr. Intell. Lab. Syst.* 126 (2013) 83–90.
[15] Y.R. Xu, X.H. Zeng, S. Bernard, Z. He, Data-driven prediction of neutralizer pH and valve position towards precise control of chemical dosage in a wastewater treatment plant, *J. Clean. Prod.* 348 (2022) 131360.
[16] J.C.B. Gonzaga, L.A.C. Meleiro, C. Kiang, R. Maciel Filho, ANN-based soft-sensor for real-time process monitoring and control of an industrial polymerization process, *Comput. Chem. Eng.* 33 (1) (2009) 43–49.
[17] Y.L. He, Z.Q. Geng, Q.X. Zhu, Data driven soft sensor development for complex chemical processes using extreme learning machine, *Chem. Eng. Res. Des.* 102 (2015) 1–11.
[18] C. Shang, F. Yang, D.X. Huang, W.X. Lyu, Data-driven soft sensor development based on deep learning technique, *J. Process Control* 24 (3) (2014) 223–233.
[19] R.M. Xie, N.M. Jan, K.R. Hao, L. Chen, B. Huang, Supervised variational autoencoders for soft sensor modeling with missing data, *IEEE Trans. Ind. Inf.* 16 (4) (2020) 2820–2828.
[20] X.F. Yuan, C. Ou, Y.L. Wang, C.H. Yang, W.H. Gui, A novel semi-supervised pre-training strategy for deep networks and its application for quality variable prediction in industrial processes, *Chem. Eng. Sci.* 217 (2020) 115509.
[21] Y.M. Han, Y. Wang, Z.W. Chen, Y. Lu, X. Hu, L.C. Chen, Z.Q. Geng, Multiscale variational autoencoder regressor for production prediction and energy saving of industrial processes, *Chem. Eng. Sci.* 284 (2024) 119529.
[22] K.C. Wang, C. Shang, L. Liu, Y.H. Jiang, D.X. Huang, F. Yang, Dynamic soft sensor development based on convolutional neural networks, *Ind. Eng. Chem. Res.* 58 (26) (2019) 11521–11531.
[23] Y.M. Han, C.Y. Fan, M. Xu, Z.Q. Geng, Y.H. Zhong, Production capacity analysis and energy saving of complex chemical processes using LSTM based on attention mechanism, *Appl. Therm. Eng.* 160 (2019) 114072.
[24] W.S. Ke, D.X. Huang, F. Yang, Y.H. Jiang, Soft sensor development and applications based on LSTM in deep neural networks, in: 2017 IEEE Symposium Series on Computational Intelligence (SSCI). Honolulu, HI, USA, IEEE, 2017.
[25] X.F. Yuan, L. Li, Y.A.W. Shardt, Y.L. Wang, C.H. Yang, Deep learning with spatiotemporal attention-based LSTM for industrial soft sensor model development, *IEEE Trans. Ind. Electron.* 68 (5) (2021) 4404–4414.
[26] X.F. Yuan, S.B. Qi, Y.L. Wang, K. Wang, C.H. Yang, L.J. Ye, Quality variable prediction for nonlinear dynamic industrial processes based on temporal convolutional networks, *IEEE Sens. J.* 21 (18) (2021) 20493–20503.
[27] Y.M. Bai, J.S. Zhao, A novel transformer-based multi-variable multi-step prediction method for chemical process fault prognosis, *Process. Saf. Environ. Prot.* 169 (2023) 937–947.
[28] J. Hong, W.D. Tian, Prediction in catalytic cracking process based on swarm intelligence algorithm optimization of LSTM, *Processes* 11 (5) (2023) 1454.
[29] W.S. Zha, Y.P. Liu, Y.J. Wan, R.L. Luo, D.L. Li, S. Yang, Y.M. Xu, Forecasting monthly gas field production based on the CNN-LSTM model, *Energy* 260 (2022) 124889.
[30] X.F. Yuan, L.F. Huang, L.J. Ye, Y.L. Wang, K. Wang, C.H. Yang, W.H. Gui, F.F. Shen, Quality prediction modeling for industrial processes using multiscale attention-based convolutional neural network, *IEEE Trans. Cybern.* 54 (5) (2024) 2696–2707.
[31] J.H. Wang, G.F. Lin, M.J. Chang, I.H. Huang, Y.R. Chen, Real-time water-level forecasting using dilated causal convolutional neural networks, *Water Resour. Manag.* 33 (11) (2019) 3759–3780.
[32] X.T. Bi, J.S. Zhao, A novel orthogonal self-attentive variational autoencoder method for interpretable chemical process fault detection and identification, *Process. Saf. Environ. Prot.* 156 (2021) 581–597.

[33] Y.J. Wang, C. Qian, S.J. Qin, Attention-mechanism based DiPLS-LSTM and its application in industrial process time series big data prediction, *Comput. Chem. Eng.* 176 (2023) 108296.
[34] Z.Y. Yang, K. Wang, L.J. Ye, X.F. Yuan, Y.L. Wang, C.H. Yang, W.H. Gui, A difference metric attention with position distance-based weighting for transformer in data sequence modeling of industrial processes, *IEEE Trans. Ind. Inf.* 21 (2) (2025) 1803–1812.
[35] X.F. Yuan, N. Xu, L.J. Ye, K. Wang, F.F. Shen, Y.L. Wang, C.H. Yang, W.H. Gui, Attention-based interval aided networks for data modeling of heterogeneous sampling sequences with missing values in process industry, *IEEE Trans. Ind. Inf.* 20 (4) (2024) 5253–5262.
[36] X.F. Yuan, L. Li, Y.L. Wang, C.H. Yang, W.H. Gui, Deep learning for quality prediction of nonlinear dynamic processes with variable attention-based long short-term memory network, *Can. J. Chem. Eng.* 98 (6) (2020) 1377–1389.
[37] Y. Tong, M. Shu, M.X. Li, Y.W. Liu, R. Tao, C.C. Zhou, Y. Zhao, G.X. Zhao, Y. Li, Y. C. Dong, L. Zhang, L.L. Liu, J. Du, A neural network-based production process modeling and variable importance analysis approach in corn to sugar factory, *Front. Chem. Sci. Eng.* 17 (3) (2023) 358–371.
[38] M.P. Maples, D.E. Reichart, N.C. Konz, T.A. Berger, A.S. Trotter, J.R. Martin, D. A. Dutton, M.L. Paggen, R.E. Joyner, C.P. Salemi, Robust chauvenet outlier rejection, *Astrophys. J. Suppl.* 238 (1) (2018) 2.
[39] S. Xu, B. Lu, M. Baldea, T.F. Edgar, W. Wojsznis, T. Blevins, M. Nixon, Data cleaning in the process industries, *Rev. Chem. Eng.* 31 (5) (2015) 453–490.
[40] S. Morales, H. Álvarez, C. Sánchez, Dynamic models for the production of glucose syrups from cassava starch, *Food Bioprod. Process.* 86 (1) (2008) 25–30.
[41] C.M. Li, D. Fang, Z.F. Li, Z.B. Gu, Q.W. Yang, L. Cheng, Y. Hong, An improved two-step saccharification of high-concentration corn starch slurries by granular starch hydrolyzing enzyme, *Ind. Crops Prod.* 94 (2016) 259–265.
[42] J.G. Monroy, E.J. Palomo, E. López-Rubio, J. Gonzalez-Jimenez, Continuous chemical classification in uncontrolled environments with sliding windows, *Chemometr. Intell. Lab. Syst.* 158 (2016) 117–129.
[43] J. Wu, X.Y. Chen, H. Zhang, L.D. Xiong, H. Lei, S.H. Deng, Hyperparameter optimization for machine learning models based on Bayesian optimization, *J. Electron. Sci. Technol.* 17 (2019) 26–40.
[44] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: a next-generation hyperparameter optimization framework, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage AK USA, ACM, 2019.
[45] W.W. Guo, J.L. Zhu, X.Y. Yu, M.W. Jia, Y. Liu, Temporal graph convolutional network soft sensor for molecular weight distribution prediction, *Chemometr. Intell. Lab. Syst.* 252 (2024) 105196.
[46] X.Y. Lin, Z.H. Li, Y.M. Han, Z.W. Chen, Z.Q. Geng, Novel spatiotemporal graph attention model for production prediction and energy structure optimization of propylene production processes, *Comput. Chem. Eng.* 181 (2024) 108507.
[47] Y. Wang, F.F. Shen, L.J. Ye, A knowledge-refined hybrid graph model for quality prediction of industrial processes, *Eng. Appl. Artif. Intell.* 139 (2025) 109711.