ACS OMEGA

http://pubs.acs.org/journal/acsodf

Article

Horizontal Data Augmentation Strategy for Industrial Quality Prediction

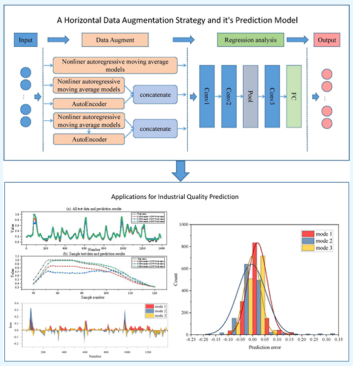
Shiwei Gao, Qingsong Zhang,* Ran Tian, Zhongyu Ma, and Xiaochao Dang

Cite This: *ACS Omega* 2022, 7, 30782−30793

Read Online

ACCESS | Metrics & More | Article Recommendations

ABSTRACT: In recent years, neural network-based soft sensor technology has been widely used in industrial production processes and has excellent optimization, monitoring, and quality prediction performance. This paper proposes a horizontal data augmentation strategy to provide highly available data for subsequent prediction models, called the combined autoencoder data augmentation (CADA) strategy. This paper has developed a CADA-based convolutional neural network (CADA-CNN) soft sensor model and applied it to the process of industrial debutanizer and industrial steam volume. In terms of method validation, this paper compares the output data of the proposed CADA by the Spearman correlation coefficient to verify the strategy's feasibility. Then, the output data of the CADA strategy is fed into the artificial neural network (NN), support vector regression (SVR), and convolutional neural network (CNN) for comparison experiments. The final experimental results show that our proposed CADA-CNN model has lower prediction error and better prediction error distribution.



1. INTRODUCTION

In the industrial production process, to better control the product quality, various advanced control, optimization, and monitoring technologies are widely used.[1,2] Applying these advanced technologies often requires many instruments and relies on real-time feedback on product quality.[3] However, some analytical tools have the characteristics of long sampling periods and high latency. Therefore, in complex industrial production processes, product quality data often require a high cost to be measured, including time cost, labor cost, and capital cost.[4] Soft sensor technology is considered a substitute for traditional analytical instruments due to its rapid response, low maintenance cost, and simple operation.[5] It can provide predictive estimates of key variables by building mathematical models from easily measurable auxiliary variables such as pressure, temperature, and flow.

Soft sensors can generally be classified into two main categories: mechanism-based modeling and data-driven modeling. Mechanism-based modeling is based on a deep understanding of the process mechanism through macroscopic or microscopic equilibrium equations to determine the mathematical relationships between key variables and easily measurable auxiliary variables.[6,7] This modeling method has high requirements for modelers, and the modeling process is time-consuming and difficult to maintain. With the rapid development of computer technology, data-driven modeling methods are being used more and more extensively in industrial production processes.[8,9] These modeling approaches use to process data exclusively without considering its physical meaning, and the modeling is simple and easy to maintain. Typical data-driven modeling approaches include multivariate analysis, statistical theory, and neural network modeling. The development of artificial neural network algorithms has been hot in recent years, and this approach is also widely used in soft sensor modeling. For example, artificial neural network (NN) and support vector regression (SVR), which are used extensively as baseline methods;[10,11] deep belief networks (DBN), which build a joint probability distribution between data and labels;[11,12] autoencoder networks (AE), which use input data for supervision to guide the network in learning mapping relationships;[2−4,6,13] long- and short-term memory networks (LSTM), which can "remember" and can be applied to time series;[1,14−17] and convolutional neural networks (CNN), which is based on visual principles and pays more attention to local features.[18−23] For soft sensor modeling, neural networks extract useful features from many easily accessible auxiliary variables and then build a model between the key variables and the extracted features for prediction.

With the development of artificial neural networks for many years, the improvements and applications in various research directions can demonstrate their excellent feature representation capabilities.[10] Typically, the abundant data collected in the process plant are high-dimensional with strong correlations

Received: March 23, 2022
Accepted: August 16, 2022
Published: August 24, 2022

and high redundancy, which is also known as data-rich but information-poor.[2] The ability to represent the features of a neural network comes from the data. Therefore, a large amount of representative data is essential to capture the hidden characteristics of the data and the characteristics of the data distribution. Although the process auxiliary variables are easily accessible, acquiring key variables is still costly.[1] Data augmentation is an effective strategy that can not only create data samples for model training but also help to improve the generalization ability of the model.[3]

This paper proposes and applies a combined autoencoder data augmentation (CADA) strategy to soft sensor modeling. On the one side, this paper uses the proven nonlinear autoregressive moving average model to expand the dataset with historical data. On the other hand, this paper uses the autoencoder network to perform initial feature extraction on the data. It regards the extracted features as the dataset's coarse screening features and uses them to enhance the data features. Then, the data obtained by the two methods are combined and used as sample input data for the subsequent regression prediction model. Instead of generating a new virtual sample, this paper expanded the data features through the adaptive combination of the two methods based on the original data, which helps express more valuable data features in subsequent regression predictions. In the regression analysis phase, the CNN model extracts high-value features from the input data and adds key variables to the top layer of the network to fine-tune the entire CNN network.

This paper adopts a structural adaptive approach to discuss the feasibility of the CADA strategy. This paper built the complete CADA-CNN soft sensor model and compared the experiments with artificial neural network (NN) and support vector regression (SVR) regression models. The results demonstrate that our proposed CADA-CNN model has a lower prediction error and better prediction error distribution than the comparable models.

In this paper, our main contributions are summarized as follows.

(1) This paper proposes a combined autoencoder data augmentation (CADA) strategy, a generic framework, and a preliminary exploration is carried out in this paper.
(2) In this paper, three models of the CADA strategy are built and the feasibility of the strategy is explored by conducting a correlation analysis.
(3) In this paper, a CADA-CNN soft sensor model is designed based on the proposed CADA strategy and the hyperparameters in the model are experimentally analyzed.

The rest of this article is structured as follows. Section 2 shows the related working studies of the proposed method and how the combined method works. Section 3 provides a detailed description of the combined autoencoder data augmentation strategy and the overall process of the soft sensor model under this strategy. Then, Section 4 presents results and a discussion on the process debutanizer unit and the process steam volume to show the effectiveness of the proposed strategy. In Section 5, the main work of the paper is summarized, and an outlook for future research is provided.

2. RELATED WORK

Neural network models require large amounts of data to support them, which is expensive and time-consuming for many applications to obtain. Therefore, this paper focuses on finding data augmentation strategies that combine with the current research hotspots. Our goal is to find more efficient data augmentation strategies and provide high-quality data for subsequent regression prediction models, and this is a data preprocessing process. Guided by extensive expert experience, this paper proposes a combined autoencoder data augmentation strategy for soft sensor modeling. Our proposed strategy is related to two aspects of the research literature: First, this paper investigated widely used data augmentation methods and their application to soft sensor modeling. Second, this paper investigates methods for updating and improving the autoencoder neural network and how to use it for modeling.

2.1. Data Augmentation. Data augmentation is a simple and effective strategy that provides a large representative sample of data for effective model learning but also helps to improve the generalization ability of the model.[3] In general, data augmentation methods can be divided into horizontal and vertical augmentation in terms of the distribution of the augmented data. If the data containing the auxiliary variables and the corresponding labeled variables is considered a complete piece of data, the vertical augmentation of the data can be seen as increasing the number of entries in the dataset. For example, graphic image processing[24−27] may generate new data by flipping, cropping, and adding noise. These methods are considered to help improve the generalization of the model. In regression analysis, such as predicting the weather, industrial product quality forecasting, and soft sensor modeling prediction, data augmentation is typically performed using generative adversarial networks (GAN)[28−31] and linear interpolation methods.[3,32] The horizontal augmentation of data can be seen as expanding the number of attributes for each piece of data while maintaining the current data size. For example, the horizontal dimension of the data is raised to a larger extent using an autoregressive moving average model.[33]

The methods mentioned above are often dataset-dependent and are realized by trial and error under the guidance of much expert knowledge.[3] The horizontal and vertical data augmentation methods described above can be seen as mutations or redistributions of local data, thus reducing the model's sensitivity to small changes to improve the model's generalization ability. However, mutations can introduce foreign features not inherent in the dataset, and redistribution may change the original distribution of features in the data. Furthermore, this corruption is persistent and can misguide feature extraction when passed between layers of the model, which may eventually lead to a weakened feature representation of the model. Thus, we propose a combined autoencoder data augmentation (CADA) strategy, hoping to use global features extracted by neural networks to alleviate the above problem.

We need to find a base data augmentation method according to the following conditions to validate our proposed CADA strategy. First, we need to find a data augmentation method within the soft sensor field as a baseline method; Second, the baseline method must be rigorously proven; Third, the baseline method must be validated over a long period by a multiliterature study. In the course of the thesis research, we found that the nonlinear autoregressive moving average model met the above requirements. The specific rationales are as follows: First, the method was rigorously proven. L. Fortuna et al.[33] used a nonlinear autoregressive moving average model for data augmentation on the debutanizer column dataset in 2005

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

and provided rigorous proof of their proposed nonlinear fourth-order model. Second, the method has a long history of extended research. In 2018, Yuan et al.[2] proposed a novel variable-wise weighted stacked autoencoder (VWSAE) model based on this method and experimentally verified the superior performance of the model. In 2019, Zhou et al.[4] proposed a stacked quality-driven autoencoder approach based on this method to construct a high-performance soft sensor model and experimentally verified that the model has better prediction results. In 2020, Ren et al.[17] proposed a supervised long short-term memory network based on this method to capture hidden features in dynamic data and experimentally verify the effectiveness of the network. Generating adversarial networks is a promising approach to data augmentation that uses games between generators and discriminators to generate highly credible data. It can be seen as a vertical augmentation method to raise the number of data entries. The CADA strategy proposed in this paper is a horizontal augmentation method, which increases the attribute columns of the data while maintaining the original amount of data. Hence, the vertical expansion methods in refs 3 and 31 are not discussed in this paper.

The nonlinear autoregressive moving average model is used to fit the real input/output data,[33] and the model output can be expressed as

$$y(k) = F(y(k-n), \cdots, y(k-1), u_1(k-n_1), \cdots, u_1(k), \cdots, u_m(k-n_m), \cdots, u_m(k)), \quad (1)$$

where $y(K)$ is the current system output estimation, $y(k-i)$ is a generic lagged sample of the system output, and $u_i(k-j)$ is a lagged sample of the i-th system input. The maximum output delay of the model is assumed to be n, and n_i represents the i-th maximum delayed input. The unknown function $F(\cdot)$ is the regression analysis function, and only the proven fourth-order model is extracted as the baseline method in this paper, so the regression function is not discussed. The specific use of this fourth-order model is described in the case studies in Section 4.

2.2. Autoencoder Neural Network (AE). The autoencoder is an unsupervised learning model based on a backpropagation algorithm with optimization methods.[2,3] The single autoencoder is a three-layer network structure as in Figure 1, with an input layer on the left, a hidden layer in the middle, and an output layer on the right. The whole network model can be divided into two parts: the encoding part and the decoding part. This network model's encoding and decoding parts are symmetrical, i.e., the number of nodes in the input layer is equal to that in the output layer. The middle hidden layer can be a single layer or multiple layers. When there are numerous hidden layers, they can be considered various AEs stacking to form a stacked autoencoder. The autoencoder uses the input data $\boldsymbol{X}$ as supervision to guide the neural network to learn a mapping relationship that reconstructs the output $\boldsymbol{X}^{\mathrm{R}}$.

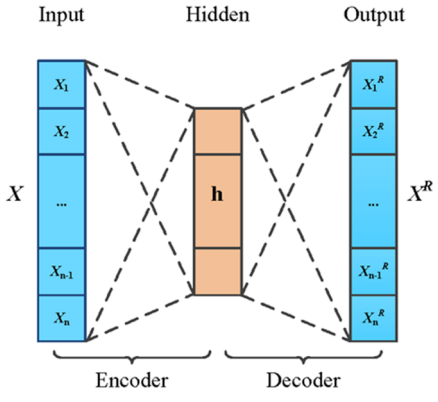


Figure 1. Autoencoder (AE) neural network model diagram.

The AE model has some sparsity and can complete the automatic selection of data features and the automatic completion of the dimensionality reduction process, thus forcing the neural network to learn high-value features. As shown in Figure 1, the encoding process of AE is from the input layer to the hidden layer, where the high-dimensional input data x is encoded into the low-dimensional hidden variable h through the nonlinear mapping function $f(\cdot)$

$$h = f(Wx + b) \quad (2)$$

where W is the weight matrix and b is the bias vector. The decoding process of AE is a process from the hidden layer to the output layer, reflecting the hidden layer data through the inverse mapping function g and reconstructing the input data $\tilde{x}$ in the output layer

$$\tilde{x} = g(\tilde{W}h + \tilde{b}) \quad (3)$$

where $\tilde{W}$ and $\tilde{b}$ are the corresponding weight matrices and bias vectors in the decoding process. The objective of the model is to minimize the reconstruction error, i.e., the error between the input data x and the output data $\tilde{x}$ so that more high-value features are retained in the parameter set $\theta = \{W, \tilde{W}, b, \tilde{b}\}$. Denote the raw observed input dataset as $x_i \in \{x_1, x_2, \ldots, x_n\}$, To obtain the parameter set θ, the reconstruction error can be minimized by calculating the loss function as

$$l(W, \tilde{W}, b, \tilde{b}) = \frac{1}{2n}\sum_{i=1}^{n} \| \tilde{x}_i - x_i \|^2 \quad (4)$$

The AE network forces the hidden layer to extract high-value features through extraction and reconstruction operations. Subsequent regression prediction models can directly use these extracted features.[2,4,6] Hence, the extracted high-value features can be considered globally relevant and do not destroy the feature distribution of the original data.

3. SOFT SENSOR MODELING

This section will detail the proposed combined autoencoder data augmentation strategy and the complete soft sensor modeling steps. Our introduction will be divided into the following two aspects: first, we introduce the combined autoencoder data augmentation strategy and its internal modes of structural adaptation and present a validation method for the strategy. Second, we introduce the modeling process of the soft sensor modeling and the evaluation metrics of the model.

3.1. Combined Autoencoder Data Augmentation (CADA) Strategy. The main idea of this paper is derived from ref 3: the data enhancement approach aims to provide highly representative training data for subsequent regression models. And in refs 2 and 4, we learn that autoencoder networks have the characteristics of automatic compression and forced extraction of high-value features. Therefore, this

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

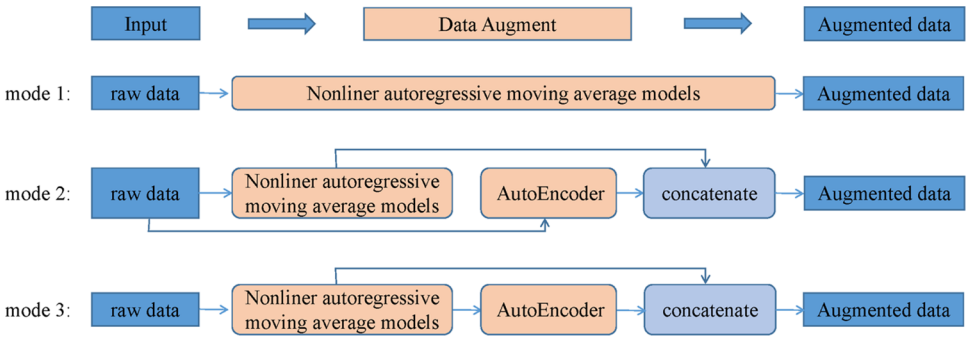


Figure 2. Flowchart of the three different modes in the CADA strategy (mode 1 is the baseline mode, and modes 2 and 3 are the research modes with different structural assignments for the two methods).

Figure 3. Diagram of CADA-CNN soft sensor model.

paper attempts to use a traditional nonlinear autoregressive moving average model combined with an autoencoder to find a data augmentation method with higher performance gains.

Our proposed CADA strategy is a preliminary exploration of an adaptive combination of the two methods. So in this paper, we explore three modes, one original mode (the baseline mode) and two other research modes (the structural adaptive comparison modes), the specific mode flow diagram shown in Figure 2. Mode 1 uses a fourth-order nonlinear autoregressive moving average model, demonstrated in ref 33. We have embedded this method in the CADA strategy and used it as our baseline model for comparison purposes. In mode 2, we used an AE network to perform coarse feature extraction from the raw data. We combined the output of the hidden layer with the expanded data from the fourth-order nonlinear autoregressive moving average model of mode 1. In mode 3, we first use mode 1 to expand the raw data and then use the AE network to perform coarse extraction of features on the expanded data. After the calculation is completed, the hidden layer output of the AE network is extracted and combined with the expanded data of mode 1. As the CADA strategy is horizontal in this paper, the "connection" in modes 2 and 3 is to expand the data to a higher number of columns. The two methods are reusable in the CADA strategy, and all exist in a single model. Only the input and output interfaces of the data need to be adjusted between the different modes. The details of the data flow are shown in Figure 2.

In this paper, Spearman's rank correlation coefficient is used to verify the feasibility of the CADA strategy. In this paper, the correlation coefficient is an indication of the direction of correlation between the auxiliary variable X and the key variable Y. When X increases and Y tends to increase, the Spearman correlation coefficient is positive; when X increases and Y tends to decrease, the Spearman correlation coefficient is negative. In particular, when the Spearman correlation coefficient is zero, indicating no convergence of Y as X increases, the Spearman correlation coefficient increases in absolute value as X and Y get closer to a complete monotonic correlation. The Spearman correlation coefficient is defined as the Pearson correlation coefficient between rank variables. For a sample with a capacity of n rows and m columns in this paper, the correlation coefficient for the m data columns is

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}} \quad (5)$$

3.2. CADA-CNN Soft Sensor Model. The soft sensor model in this paper is divided into two stages. The first stage is the data augmentation stage, where we augment the data using the CADA strategy. The second stage is the regression prediction stage. We use a convolutional neural network (CNN) that focuses more on local features to perform the regression prediction of features, as the features are augmented for local data in the first stage. Therefore, the complete soft sensor model is called the CADA-CNN model. Intuitively, Figure 3 shows the CADA-CNN soft sensor model diagram.

This paper shows the specific CNN network structure in the regression analysis stage in Figure 3. In this stage, we set up three convolutional layers, interspersed with a pooling layer in the second and third convolutional layers, and finally used a fully connected neural network for the predictive representation of the features and to obtain the predicted output in the output layer. The specific algorithmic flow of the CNN network is shown in Table 1.

In this paper, the three modes of CADA strategy are modeled, respectively, and the modeling process is shown in Figure 4, with the following modeling steps.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

Table 1. Convolutional Neural Network Algorithm Flow

	algorithm: convolution regression
	input: Output of CADA stage $X(DA)$, key variables $\boldsymbol{Y}$
	output: key variables for prediction $\boldsymbol{Y}_{\text{pred}}$
1:	parameter setting: batch size, epochs, learning rate.
2:	loss function: mean absolute error (MAE).
3:	optimizers: Adam.
4:	conv parameter setting: kernel size, padding, activation function.
5:	initial weight.
6:	repeat:
7:	loss (MAE) $\leftarrow \frac{1}{n}\sum_{j=1}^{n}\lvert y_j - y_{\text{pred}}\rvert$
8:	weight $\leftarrow$ updated parameters by gradient descent
9:	until: convergence of weight

(1) Step 1: The auxiliary variable selection, collection, and preprocessing.
(2) Step 2: Determine train and test datasets.
(3) Step 3: The autoencoder network in the CADA strategy is pre-trained, the number of iterations and learning rate of this network is determined, and the feasibility of the CADA strategy is verified by correlation analysis.
(4) Step 4: Pre-training the CADA-CNN model and determining the learning rate of the CNN network in this model.
(5) Step 5: The CADA-CNN soft sensor model is trained according to the hyperparameters determined in steps 3 and 4.
(6) Step 6: Fine-tune the overall network and modify the network parameters slightly.
(7) Step 7: Testing the test set and evaluating the performance of the soft sensor model.

This paper uses the three model indicators used in refs 2−4 to evaluate the model.

Mean absolute error (MAE) is defined as

$$\text{MAE} = \frac{1}{n}\sum_{j=1}^{n}\lvert y_j - y_{\text{pred}}\rvert \tag{6}$$

Root mean square error (RMSE) defined as

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - y_{\text{pred}})^2} \tag{7}$$

R-square (R^2) is defined as

$$R^2 = 1 - \sum_{j=1}^{n}(y_j - y_{\text{pred}})^2 / \sum_{j=1}^{n}(y_j - \overline{y})^2 \tag{8}$$

4. RESULTS AND DISCUSSION

This section performs a comparative ablation study of CADA strategies using a debutanizer column and an industrial steam volume dataset. We will describe and analyze the following four aspects. First, we introduce the dataset used for this case study and its associated variables. Second, we present the usage of the baseline method identified in this paper and the model structure parameters of the neural network used. Third, we experimentally set the hyperparameters of the AE network in the CADA strategy and performed a correlation analysis on the output data. Fourth, we experimentally determine the

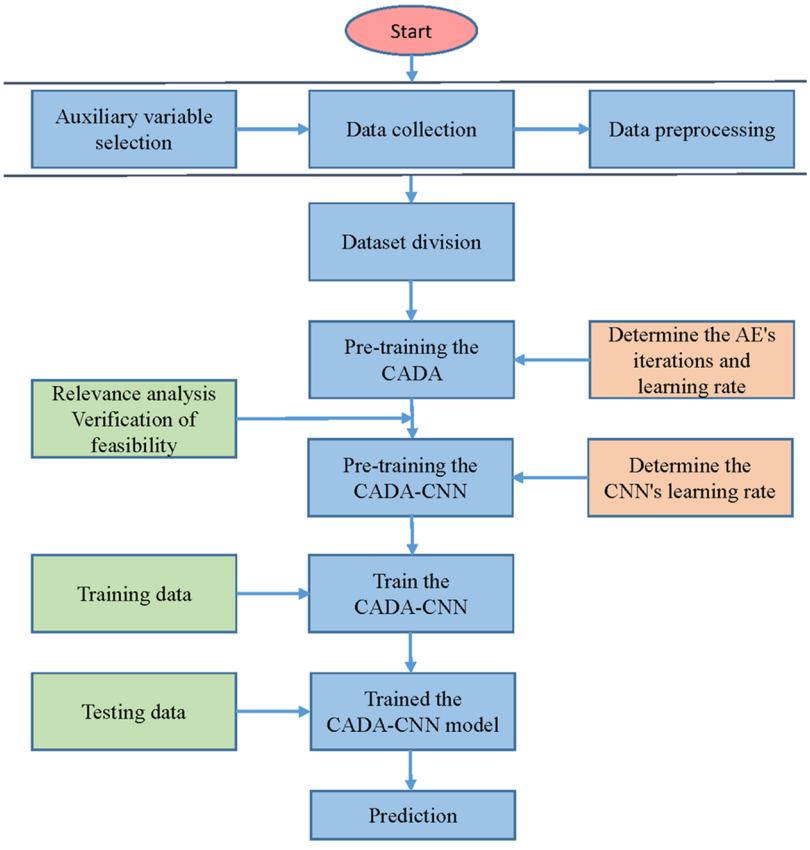


Figure 4. Flowchart of CADA-CNN soft sensor modeling.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

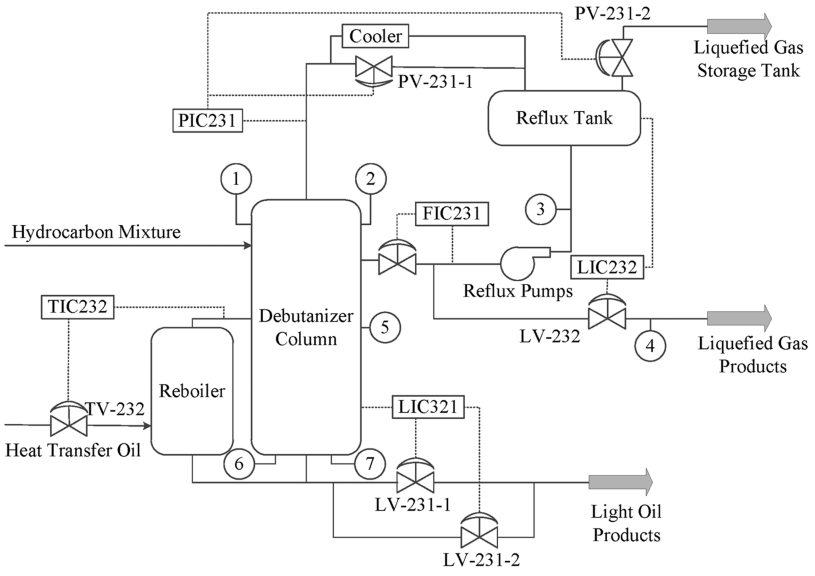


Figure 5. Debutanizer column flowchart.

hyperparameters of the CADA-CNN soft sensor model and analyze the model's index scores and prediction results on the test set.

4.1. Debutanizer Column. Separating crude oil is a very complex and important refining process in the petroleum industry. The debutanizer column is an important industrial refinery furnace for separating liquefied petroleum gas and stabilized light hydrocarbons, mainly for desulphurization and naphtha splitting. The flowchart of the debutanizer column is shown in Figure 5. To ensure product quality, the butane content at the bottom of the debutanizer column must be minimized. As a result, the real-time measurement of the butane content in the column is the key point for the accurate control of the refinery process. As a result, the real-time measurement of the butane content in the column is the key point for the accurate control of the refinery process. However, the concentration of C4, which can reflect the butane content, cannot be measured directly but requires continuous measurement and analysis of the subsequent overheads of the deisopentane tower with the aid of a gas chromatograph.

In summary, the gas chromatograph has a serious delay in measuring butane content, and the equipment is expensive to maintain, which cannot guarantee the real-time control of the refinery process. To alleviate these problems, soft sensor technology, which is easy to operate and low maintenance, predicts the C4 content. The seven points in Figure 5 are the data collection points for the auxiliary variables, and Table 2 describes the auxiliary and key variables.

Table 2. Variable Description for the Debutanizer Column

input variables	variable description
u_1	top temperature
u_2	top pressure
u_3	reflux flow
u_4	flow to next process
u_5	6th tray temperature
u_6	bottom temperature A
u_7	bottom temperature B
y	butane content

4.2. Baseline Method and Model Structural Parameters. In this subsection, we present the following two aspects. First, we present the specific operation of the determined baseline method, i.e., the fourth-order nonlinear autoregressive moving average model, on the debutanizer column dataset. Second, we present the model structure parameters of the two neural networks in the proposed CADA-CNN model, namely, the autoencoder and the convolutional neural network.

There are seven auxiliary variables and one key variable in the debutanizer column dataset. The dataset is expanded according to the proven fourth-order nonlinear autoregressive moving average model using historical data for the u_5 attribute and the key variable y. The specific data expansion is shown in the augmentation matrix (9).[33] A total of 2390 data samples are collected in this process, of which 1000 samples are used as the training dataset and the remaining samples as the test dataset.

$$[u_1(k), u_2(k), u_3(k), u_4(k), u_5(k), u_5(k-1), u_5(k-2), u_5(k-3), (u_6(k)+u_7(k))/2, y(k-1), y(k-2), y(k-3), y(k-4)]^T \tag{9}$$

In this paper, three modes are set up in the CADA strategy, where mode 1 uses data augmentation such as the augmentation matrix (9), and in modes 2 and 3, the autoencoder network (AE) is used. Therefore, we need to configure the AE network structure, which is referenced in ref 2 and set to [13 8 3]. Since there are 13 variables in the augmented variable vector of the data after the fourth-order nonlinear autoregressive moving average model, the number of neurons in the input layer of AE is 13. The high-value features extracted from the hidden layer of the AE network were expanded into the data vector and the data were passed into the CNN network as $k \times k$, thus setting the middle hidden layer neurons of the AE network to three. In the regression analysis stage, the structure of the CNN network is shown in Figure 3. The Adam optimizer is used for optimization, the loss function is set to MAE, the convolutional kernel size is 2×2, the padding method is the same, and the relu function is used as the activation function.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

4.3. CADA Parameter Determination and Correlation Analysis. In this subsection, we present the following two aspects. First, we experimentally determine the hyperparameters for the CADA stage. Second, we perform a correlation analysis of the output data from the CADA stage.

In the CADA strategy, both mode 2 and mode 3 use the autoencoder network, so we need to experimentally limit the number of iterations and learning rate of the autoencoder network. In exploring the number of iterations, we refer to the setting in ref 2 and set the learning rate tentatively at 0.01 (this learning rate will be experimentally validated subsequently), with 2000 iterations on mode 2 and mode 3, respectively, whose network loss varies with the number of iterations as shown in Figure 6. In Figure 6, as can be seen, the pattern of

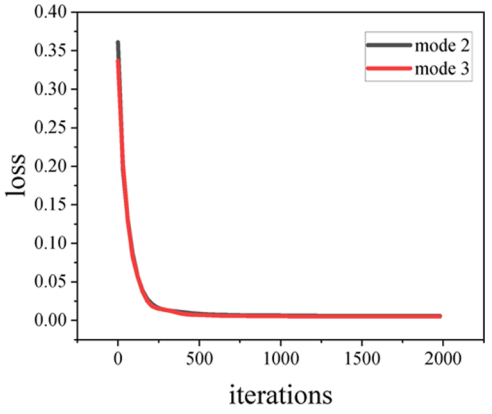


Figure 6. CADA stage, mode 2, and mode 3 loss variation diagram.

loss change is the same for both modes. The loss of the autoencoder network stopped decreasing after nearly 1000 iterations, so we set the number of iterations for each mode in the CADA stage at 1000.

We tentatively set the learning rate at 0.01 and experimentally determined the number of iterations to be 1000 when exploring the variation pattern of the number of iterations versus loss. Therefore, seven sets of experiments are conducted to set the learning rate. Respectively, set the learning rate (lr) to {0.001,0.005,0.01,0.05,0.1,0.5,1}, the relationship between the learning rate, loss, and iterations is shown in Figure 7. As seen in Figure 7, the loss of both mode 2 and mode 3 decreases smoothly as the number of iterations increases when the learning rate is 0.001 and 0.005. As the learning rate continues to rise, the loss of mode 2 and mode 3 fluctuate as the number of iterations rises. Hence, we can determine that the change in loss is close to a critical state at a learning rate of around 0.005. Meanwhile, to reduce the fluctuations during multiple independent experiments, we selected the learning rate of the CADA stage as 0.001.

To compare the correlation of the data constructed by the three modes in the CADA strategy more intuitively, we numbered the data in the three modes. The numbering description table is shown in Table 3. The data columns

Table 3. Description of Data Column Numbers

data description	number
raw data	1−7
mode 1 output	1−5,8−15
mode 2 output	1−5,8−18
mode 3 output	1−5,8−15,19−21

numbered 1−7 are the raw data columns, those numbered 1−5 and 8−15 are the data columns outputted by mode 1, those numbered 1−5 and 8−18 are the data columns outputted by mode 2, and those numbered 1−5 and 8−15 and 19−21 are the data columns outputted by mode 3. The key variable *y* data column was used to calculate the Spearman correlation coefficient with the original seven attribute columns in the dataset and the output data columns of the three modes. The Spearman correlation coefficient calculation results are shown in Table 4 and Figure 8. The correlation coefficients calculated

Table 4. ρ for the Output Data of the Three Modes in the CADA Strategy

number	ρ	number	ρ
1	0.068678652	12	0.996632657
2	0.21090934	13	0.987065435
3	0.248085121	14	0.971603143
4	0.149349171	15	0.950779782
5	0.21023562	16	0.096137048
6	0.064929623	17	0.128862912
7	0.043576496	18	0.074797045
8	0.24673177	19	0.471020202
9	0.286921231	20	0.820837668
10	0.330141462	21	0.790777659
11	0.053040193		

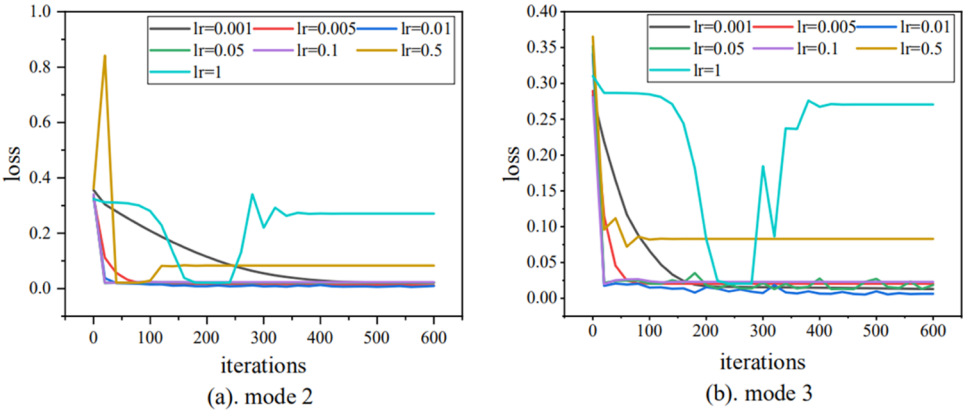


Figure 7. CADA stage, loss, iterations, and learning rate variation diagram.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

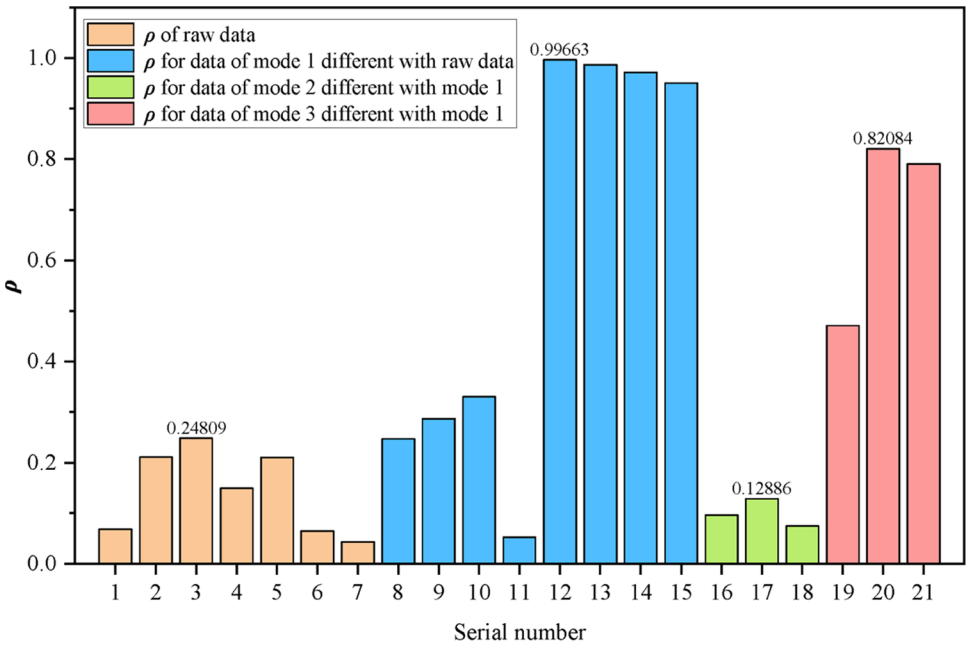


Figure 8. Histogram of ρ for each data column (the maximum ρ values for each part of the legend are marked in the figure).

for the data columns numbered 1−15 are constant. In contrast, the data columns numbered 16−21 are calculated from the high-value features extracted by the AE network and will change each time. Therefore, we use the network settings determined from the above experiments to ensure the stability of the AE network, set the number of iterations to 1000 and the learning rate to 0.001, and repeat 20 times to calculate its mean value.

As shown in Table 4 and Figure 8, the correlation coefficients of the raw data are low. However, after the fourth-order nonlinear moving average method of mode 1, the expanded data columns have high correlation coefficients, as shown in the data columns numbered 9,10,12,13,14,15, respectively. The high-value features extracted by the AE network also have similarly high correlation coefficients, as shown in the data columns numbered 20,21 respectively. The data numbered 16−18 are the high-value features extracted from the AE network in mode 2. The results show that mode 2 has a lower correlation coefficient than mode 3, and mode 3 has a higher correlation coefficient than some of the data in mode 1.

Our proposed CADA strategy can significantly expand the data columns with a higher correlation on the base method, thus demonstrating the strategy's feasibility.

4.4. CADA-CNN Soft Sensor Model. In this subsection, we present the following three aspects. First, the hyperparameters of the CADA-CNN soft sensor model are determined. Second, the experimental results of the model and the scores of the model evaluation indicators are analyzed. Third, the prediction error of the model is analyzed.

In this paper, we use 1000 data as the training set and the remaining data as the test set, so we set the batch size to 50 and the epochs to 20 by referring to the setting in ref 4. We conducted five groups of experiments for the CNN regression network to determine the size of a learning rate of {0.001,0.003,0.005,0.008,0.01}. The variation of its learning rate and MAE loss with increasing epochs is shown in Figure 9.

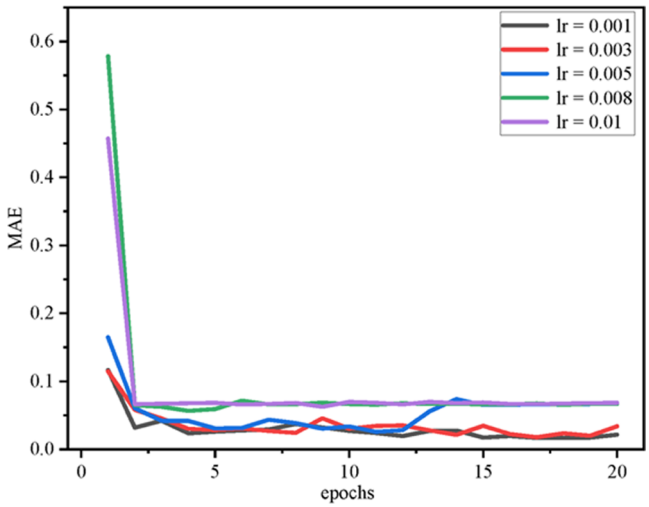


Figure 9. Plot of CADA-CNN model learning rate and MAE loss with epoch.

As shown in Figure 9, there is a substantial decrease in loss during the first three epochs of the experiment and a slight decrease in loss during subsequent training. The loss in the first epoch of the model decreases when the learning rate decreases. In addition, the smaller the learning rate, the smaller the MAE loss when training is completed with 20 epochs. Therefore, to minimize the training error, we set the learning rate of the CNN regression network in the CADA-CNN model to 0.001.

This paper uses the parameters described above to build the CADA-CNN soft sensor model and conduct experiments. In which we use the base regressors as used in ref 2 for comparison tests in the regression analysis stage, which are multilayer artificial neural networks (NN) with the structure of [13 10 7 4 1][2], support vector regression (SVR). And two citation comparison models are used, VWSAE-NN[2] and SQAE-NN[4]. The complete experimental indicator scores are shown in Table 5.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

Table 5. Results of CADA-CNN Model Metrics

CADA	model	MAE	RMSE	R^2
AE-only	NN	0.0705	0.0910	0.7781
	SVR	0.0741	0.1053	0.7022
	CNN	0.0562	0.0791	0.8323
mode 1 (baseline mode)	NN	0.0259	0.0491	0.9321
	SVR	0.0519	0.0656	0.8846
	CNN	0.0284	0.0421	0.9478
	VWSAE-NN[2]	0.0277	0.0379	0.9444
	SQAE-NN[4]	0.0220	0.0303	0.9646
mode 2	NN	0.0350	0.0646	0.8764
	SVR	0.0468	0.0649	0.8869
	CNN	0.0318	0.0471	0.9404
mode 3	NN	0.0267	0.0449	0.9434
	SVR	0.0433	0.0599	0.9035
	CNN	**0.0273**	**0.0361**	**0.9651**

From the evaluation metrics in Table 5, as can be seen, in mode 3, the MAE, RMSE, and R^2 metrics of the CADA-CNN model outperformed the comparison model VWSAE-NN. The MAE and RMSE metrics of the CADA-CNN model are slightly higher due to the different data selection and less improvement, but the R^2 metric is better than that of the SQAE-NN model. Overall, the CADA-CNN model outperformed mode 2 and mode 1 (baseline mode) under mode 3. As shown in Table 4 and Figure 8, this result indirectly illustrates the lower correlation coefficients calculated for the high-value features extracted by the AE neural network in mode 2 and the higher correlation coefficients in mode 3. As can be seen from the results of the ablation experiments only involving autoencoders in Table 5, the AE-only experimental metrics are inferior and cannot be compared to the better models available. To provide a more intuitive understanding of the prediction results of the soft sensor model, we extracted the prediction results of the regression model as a CNN for each of the three modes, which is represented in Figure 10.

From Figure 10a, we can see that the prediction results under the three modes of the CADA strategy are significantly different. In mode 1, the prediction curve for the data augmentation mode using the fourth-order nonlinear autoregressive moving average model is in the middle of the three modes. However, the prediction curves for mode 2 and mode 1 are essentially the same, but in some regions, such as around the data point with the test set number in [70,120] in Figure 10b, the prediction curve for mode 2 is lower than that for mode 1. Possible reasons for this occurrence are fluctuations in the model when predicting particular data points, inadequate support of feature data, etc. As can be seen in Figure 10a,b, mode 3, i.e., after the expansion of mode 1 and then using the AE network for coarse feature extraction, and combining the outputs of the two methods, has a high degree of fit with the real value, which also reflects that mode 3 has a high score among the various evaluation indicators obtained in Table 5. From Figure 10, it is only possible to see whether the predictions fit the real value, so we calculated the error between the predicted and real value for each mode in the CADA-CNN model, which is derived from the difference between the predicted and real value. The detailed prediction errors for each mode are shown in Figure 11.

Figure 11 presents the difference between the predicted and true values using an area plot. The area chart is bounded by the prediction error curve, using the area between the curve and the zero axis to represent the magnitude of the error value and the fluctuations in the prediction. From Figure 11, we can visualize that in mode 1, the prediction error is between

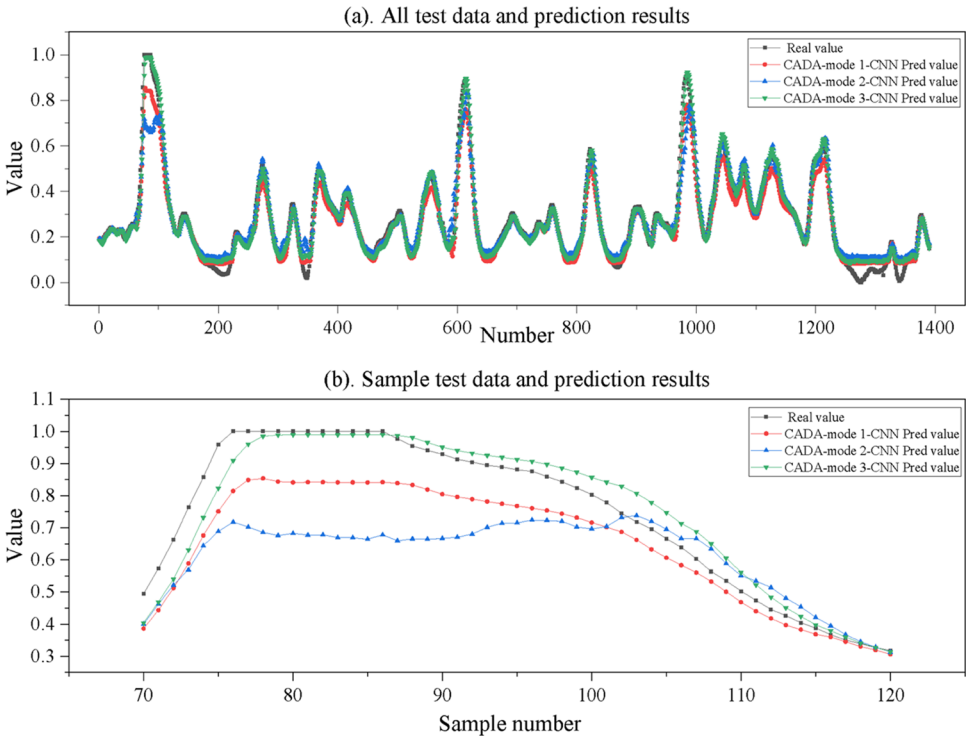


Figure 10. Graph of prediction results versus true values for the CADA-CNN model in three modes: (a) comparison of prediction results for all test data and (b) comparison of prediction results for test data number 70 to 120.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

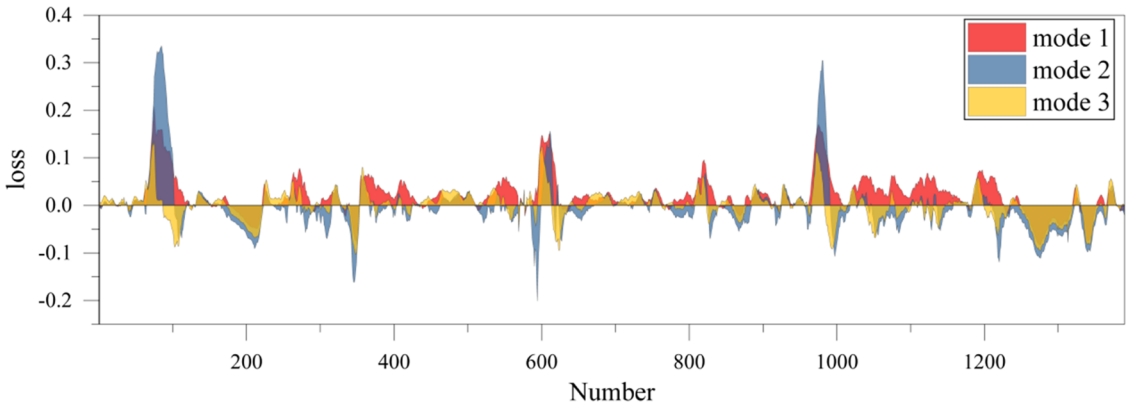


Figure 11. Error area chart of the predicted and true value of CADA-CNN model in three modes.

[−0.1,0.2], and the experiment in this mode serves as our baseline. In mode 2, there is a significant fluctuation in the prediction error, which expands to a range between [−0.2,0.35]. In this mode, the prediction error decreases for most of the data points in the test set. Still, it increases significantly around some particular points, such as the data point with the test set number [70,120]. In mode 3, the range of the prediction error is further reduced to [−0.1,0.15], and the prediction error in the entire test set is significantly reduced compared to baseline.

The prediction results are statistically presented to reflect the prediction error distribution of the CADA-CNN model under the three modes. The complete histogram of the prediction error distribution is shown in Figure 12. As can be

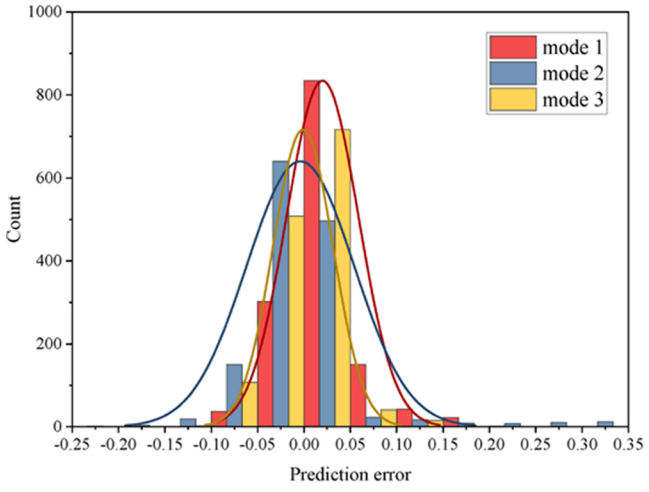


Figure 12. Histogram of error distribution statistics for the CADA-CNN model in the three modes (the bars in the figure are the number of error statistics in that range, and the curves in the figure indicate the distribution of that error).

seen from the error distribution curve in Figure 12, the error distribution of the baseline method, i.e., the CADA-CNN model under mode 1, is biased to the right of the zero labels, indicating an uneven error distribution. The prediction error distributions for modes 2 and 3 are not skewed and are evenly distributed around the zero labels. Meanwhile, the sharper the error distribution curve, the more concentrated the distribution. In Figure 12, the error distribution curve for mode 2 is flatter than that for mode 3, which means that mode 2 has a larger prediction error than mode 3. It also shows that the CADA-CNN model has a better prediction error distribution under mode 3.

4.5. Industrial Steam Volume. Thermal power generation uses the released heat energy when fuel is burned to heat the water in the boiler to produce steam. The steam is accumulated in a special pressure tank and is used to drive the turbine. As a result, the turbine rotates the generator for electricity production. The flowchart of thermal power generation is shown in Figure 13. In this process, the energy conversion efficiency of the boiler is the key to the efficiency of electricity generation. In other words, the transformation efficiency of the fuel is realized when the fuel is burned to heat the water in the boiler and to produce high temperature and pressure steam. The factors affecting the energy transfer of this process are complex, including the boiler's adjustable parameters, such as fuel charge, ventilation air volume, boiler water volume, and boiler operating conditions, such as boiler bed temperature, bed pressure, furnace chamber temperature, pressure, etc.

There are 38 auxiliary variables and 1 key variable in the data. A total of 2884 data samples are collected, of which 2500 samples are used as training data and the rest as test data. In this experiment, we focus on testing the effectiveness of the CADA strategy and the performance of each model. Therefore, we use the same parameter configuration as in the previous experiments. The specific data expansion for the baseline model is shown in the augmentation matrix (10).[33] In addition, we should adjust the number of output neurons of the AE network to 5 to facilitate the integration of data from modes 2 and 3. The complete experimental indicator scores are shown in Table 6.

$$[u_1(k), u_2(k), u_3(k), u_4(k), \cdots, u_{36}(k), u_{36}(k-1), u_{36}(k-2), u_{36}(k-3), (u_{37}(k)+u_{38}(k))/, y(k-1), y(k-2), y(k-3), y(k-4)]^T \quad (10)$$

From the evaluation metrics in Table 6, the trends in the overall experimental results are the same as the previous debutanizer column experiments. The results for mode 3 were all better than the other ablation experiments, the results for AE-only and mode 1 were essentially the same, and the results for mode 2 were slightly better than the former two baselines. Since the experimental results on both datasets trended the same, we did not extract and analyze the industrial steam volume experiment results.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793

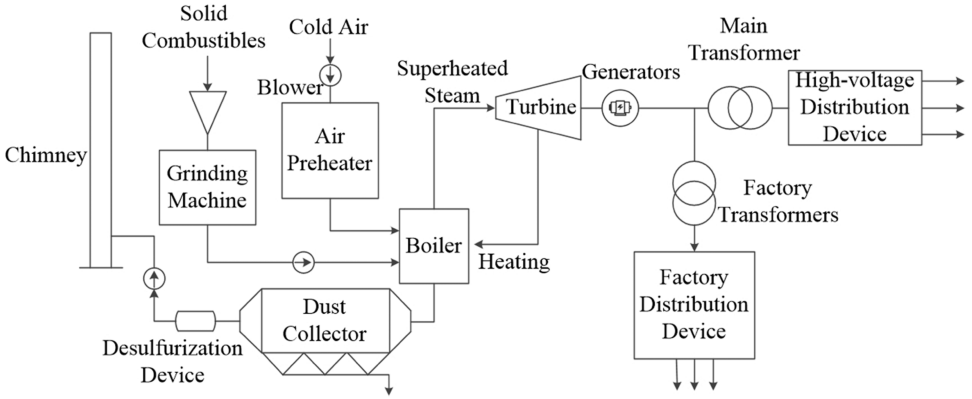


Figure 13. Thermal power flowchart.

Table 6. Results of CADA-CNN Model Metrics

CADA	model	MAE	RMSE	R^2
AE-only	NN	0.0560	0.0835	0.8110
	SVR	0.0670	0.0906	0.7775
	CNN	0.0614	0.0854	0.8021
mode 1 (baseline mode)	NN	0.0618	0.0872	0.7953
	SVR	0.0597	0.0845	0.8078
	CNN	0.0634	0.0878	0.7915
mode 2	NN	0.0552	0.0802	0.8271
	SVR	0.0560	0.0795	0.8299
	CNN	0.0562	0.0792	0.8312
mode 3	NN	0.0528	0.0766	0.8421
	SVR	0.0569	0.0794	0.8302
	CNN	**0.0528**	**0.0739**	**0.8530**

5. CONCLUSIONS

This paper discusses the feasibility of the data augmentation strategy, which combines the autoencoder network with the nonlinear autoregressive moving average model. Meanwhile, a CADA-CNN soft sensor model is designed, and the effectiveness of the strategy and model is validated by experiments in an industrial process debutanizer column and ablation testing of CADA strategies on an industrial steam volume dataset. The experimental results show that our proposed CADA strategy has a large improvement in the prediction performance of the subsequent regression model. The proposed CADA-CNN model has a smaller prediction error and a better error distribution at mode 3.

In this paper, subject to several requirements mentioned in the paper, our proposed CADA strategy is only combined with the proven fourth-order nonlinear autoregressive moving average model, which may be combined more effectively with other methods. Moreover, in this paper, we only use the autoencoder network, and there may be more efficient networks to replace its position. The strategy validated in this paper also offers the possibility of further exploration in different areas. For example, the CADA strategy could be useful in classification problems, where autoencoder networks have many research applications.

AUTHOR INFORMATION

Corresponding Author

Qingsong Zhang − *College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China*; orcid.org/0000-0002-6223-772X; Email: 2020211983@nwnu.edu.cn

Authors

Shiwei Gao − *College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China*

Ran Tian − *College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China*

Zhongyu Ma − *College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China*

Xiaochao Dang − *College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.2c01747

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors acknowledge Northwest Normal University for providing resources and supporting this research. They express their sincere gratitude to the financial support of the National Natural Science Foundation of China (71961028), Scientific Research Projects of Colleges and Universities in Gansu Province (2019B-038), the Scientific Research Project of the Lanzhou Science and Technology Program (2018-01-58), and the Industrial Support Program Project of Gansu Province (2021CYZC-06).

REFERENCES

(1) Kong, X.; Ge, Z. Adversarial Attacks on Neural-Network-Based Soft Sensors: Directly Attack Output. *IEEE Trans. Ind. Inf.* **2022**, *18*, 2443−2451.

(2) Yuan, X.; Huang, B.; Wang, Y.; Yang, C.; Gui, W. Deep Learning-Based Feature Representation and Its Application for Soft Sensor Modeling With Variable-Wise Weighted SAE. *IEEE Trans. Ind. Inf.* **2018**, *14*, 3235−3243.

(3) Yuan, X.; Ou, C.; Wang, Y.; Yang, C.; Gui, W. A Layer-Wise Data Augmentation Strategy for Deep Learning Networks and Its Soft Sensor Application in an Industrial Hydrocracking Process. *IEEE Trans. Neural Networks Learn. Syst.* **2021**, *32*, 3296−3305.

(4) Yuan, X.; Zhou, J.; Huang, B.; Wang, Y.; Yang, C.; Gui, W. Hierarchical Quality-Relevant Feature Representation for Soft Sensor Modeling: A Novel Deep Learning Strategy. *IEEE Trans. Ind. Inf.* **2020**, *16*, 3721−3730.

(5) Shao, W.; Yao, L.; Ge, Z.; Song, Z. Parallel Computing and SGD-Based DPMM For Soft Sensor Development With Large-Scale Semisupervised Data. *IEEE Trans. Ind. Electron.* **2019**, *66*, 6362−6373.

(6) Yao, L.; Ge, Z. Deep Learning of Semisupervised Process Data With Hierarchical Extreme Learning Machine and Soft Sensor Application. *IEEE Trans. Ind. Electron.* **2018**, *65*, 1490−1498.
(7) Yan, W.; Tang, D.; Lin, Y. A Data-Driven Soft Sensor Modeling Method Based on Deep Learning and its Application. *IEEE Trans. Ind. Electron.* **2017**, *64*, 4237−4245.
(8) Wang, X.; Han, L. Soft Sensor Based on Stacked Auto-encoder Deep Neural Network for Air Preheater Rotor Deformation Prediction. *Adv. Eng. Inf.* **2018**, *36*, 112−119.
(9) Cheng, T.; Harrou, F.; Sun, Y.; Leiknes, T. Monitoring Influent Measurements at Water Resource Recovery Facility Using Data-Driven Soft Sensor Approach. *IEEE Sens. J.* **2019**, *19*, 342−352.
(10) Sun, Q.; Ge, Z. A Survey on Deep Learning for Data-Driven Soft Sensors. *IEEE Trans. Ind. Inf.* **2021**, *17*, 5853−5866.
(11) Lian, P.; Liu, H.; Wang, X.; Guo, R. Soft sensor based on DBN-IPSO-SVR approach for rotor thermal deformation prediction of rotary air-preheater. *Measurement* **2020**, *165*, No. 108109.
(12) Yuan, X.; Gu, Y.; Wang, Y. Supervised Deep Belief Network for Quality Prediction in Industrial Processes. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1−11.
(13) Osman, Y. B. M.; Li, W. Soft Sensor Modeling of Key Effluent Parameters in Wastewater Treatment Process Based on SAE-NN. *J. Control Sci. Eng.* **2020**, *2020*, No. 6347625.
(14) Zhou, J.; Wang, X.; Yang, C.; Xiong, W. A Novel Soft Sensor Modeling Approach Based on Difference-LSTM for Complex Industrial Process. *IEEE Trans. Ind. Inf.* **2022**, *18*, 2955−2964.
(15) Yin, X.; Niu, Z.; He, Z.; Li, Z.; Lee, D. Ensemble deep learning based semi-supervised soft sensor modeling method and its application on quality prediction for coal preparation process. *Adv. Eng. Inf.* **2020**, *46*, No. 101136.
(16) Pan, H.; Su, T.; Huang, X.; Wang, Z. LSTM-based soft sensor design for oxygen content of flue gas in coal-fired power plant. *Trans. Inst. Meas. Control.* **2021**, *43*, 78−87.
(17) Ren, L.; Wang, T.; Laili, Y.; Zhang, L. A Data-Driven Self-Supervised LSTM-DeepFM Model for Industrial Soft Sensor. *IEEE Trans. Ind. Inf.* **2022**, *18*, 5859−5869.
(18) Wang, K.; Shang, C.; Liu, L.; Jiang, Y.; Huang, D.; Yang, F. Dynamic Soft Sensor Development Based on Convolutional Neural Networks. *Ind. Eng. Chem. Res.* **2019**, *58*, 11521−11531.
(19) Geng, Z.; Chen, Z.; Meng, Q.; Han, Y. Novel Transformer Based on Gated Convolutional Neural Network for Dynamic Soft Sensor Modeling of Industrial Processes. *IEEE Trans. Ind. Inf.* **2022**, *18*, 1521−1529.
(20) Lu, Y.; Yang, D.; Li, Z.; Peng, X.; Zhong, W. Neural Network Based on Windowed Convolutional Transformation to Extract Features in Time Domain and Its Application on Soft Sensing. *IEEE Trans Instrum Meas.* **2022**, *71*, 1−13.
(21) Yuan, X.; Qi, S.; Shardt, Y. A. W.; Wang, Y.; Yang, C.; Gui, W. Soft sensor model for dynamic processes based on multichannel convolutional neural network. *Chemom. Intell. Lab. Syst.* **2020**, *203*, No. 104050.
(22) Zhao, Y.; Ding, B.; Zhang, Y.; Yang, L.; Hao, X. Online cement clinker quality monitoring: A soft sensor model based on multivariate time series analysis and CNN. *ISA Trans.* **2021**, *117*, 180−195.
(23) Jin, H.; Li, Z.; Chen, X.; Qian, B.; Yang, B.; Yang, J. Evolutionary optimization based pseudo labeling for semi-supervised soft sensor development of industrial processes. *Chem. Eng. Sci.* **2021**, *237*, No. 116560.
(24) Loey, M.; Manogaran, G.; Khalifa, N. M. A deep transfer learning model with classical data augmentation and CGAN to detect COVID-19 from chest CT radiography digital images. *Neural Comput. Appl.* **2020**, 1−13.
(25) Lashgari, E.; Liang, D.; Maoz, U. Data augmentation for deep-learning-based electroencephalography. *J. Neurosci. Methods* **2020**, *346*, No. 108885.
(26) Shorten, C.; Khoshgoftaar, T. M. A survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, No. 60.
(27) Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random Erasing Data Augmentation. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 13001−13008.
(28) Yang, H.; Li, J.; Lim, K.; Pan, C.; Van Truong, T.; Wang, Q.; Li, K.; Li, S.; Xiao, X.; Ding, M.; Chen, T.; Liu, X.; Xie, Q.; Alvarado, P. V.; Wang, X.; Chen, P. Automatic strain sensor design via active learning and data augmentation for soft machines. *Nat. Mach. Intell.* **2022**, *4*, 84−94.
(29) Ye, J.; Nakwijit, P.; Schiemer, M.; Jha, S.; Zambonelli, F. Continual Activity Recognition with Generative Adversarial Networks. *ACM Trans. Internet Things.* **2021**, *2*, 1−25.
(30) Guo, R.; Liu, H. A Hybrid Mechanism- and Data-Driven Soft Sensor Based on the Generative Adversarial Network and Gated Recurrent Unit. *IEEE Sens. J.* **2021**, *21*, 25901−25911.
(31) Gao, S.; Qiu, S.; Ma, Z.; Tian, R.; Liu, Y. SVAE-WGAN based Soft Sensor Data Supplement Method for Process Industry. *IEEE Sens. J.* **2022**, *22*, 601−610.
(32) Guo, F.; Xie, R.; Huang, B. A deep learning just-in-time modeling approach for soft sensor based on variational autoencoder. *Chemom. Intell. Lab. Syst.* **2020**, *197*, No. 103922.
(33) Fortuna, L.; Graziani, S.; Xibilia, M. G. Flexible sensors for monitoring the quality of products in the distilling column of the debugger. *Control Eng. Pract.* **2005**, *13*, 499−508.

https://doi.org/10.1021/acsomega.2c01747
ACS Omega 2022, 7, 30782−30793