



# Data-Driven Soft Sensor Model Based on Deep Learning for Quality Prediction of Industrial Processes

Xianglin Zhu<sup>1</sup> · Khalil Ur Rehman<sup>1</sup> · Wang Bo<sup>1</sup> · Muhammad Shahzad<sup>1</sup> · Ahmad Hassan<sup>2</sup>

Received: 27 August 2020 / Accepted: 24 December 2020 / Published online: 15 January 2021  
© The Author(s), under exclusive licence to Springer Nature Singapore Pte Ltd. part of Springer Nature 2021

## Abstract

Fermentation process is a time-varying, nonlinear and multivariable dynamic coupling system. Therefore, it is difficult to directly measure the key biological variables using traditional physical sensors during the process of fermentation, which makes the monitoring and real-time control impossible. To resolve this problem, a data-driven soft sensor modeling method based on deep neural network (DNN) is proposed in this paper. This method is suitable for large amount of data and it enjoys high efficiency and robustness. At the same time, an adaptive moment estimation (Adam) algorithm is used to optimize the hyper-parameters of the DNN model, which is a technique for efficient stochastic optimization that only requires first-order gradients with little memory requirement. The consistent correlation method is used to determine the auxiliary variables of the soft sensor model. The penicillin and L-lysine fermentation processes are taken as the research object, substrate concentration, cell concentration, and product concentration are selected as a target variable. The performance of established soft sensor model is evaluated through the indexes of mean square error (MSE), root-mean-square error (RMSE), and mean absolute error (MAE). The simulation results show that the prediction performance of the soft sensor model based on DNN-Adam is good and compared with model based on stochastic gradient descent (SGD) with momentum optimization algorithm. It is verified that the proposed method can make a more accurate real-time prediction of quality variables in the fermentation process, and it has higher prediction accuracy than DNN-SGD method.

**Keywords** Fermentation process · Soft sensor model · Deep learning · Quality prediction · Optimization algorithm

## Introduction

Fermentation is a common production method in modern process industry, and it is widely used in the fields of medicine, food, and chemical industry. However, the fermentation process is a highly nonlinear, time-varying, multivariable, and strongly coupled biochemical reaction process. Due to the growth and reproduction of living organisms, the internal mechanism is very complex. Some key variables that directly reflect the quality of the fermentation process are difficult to measure in real time through traditional measurement methods. The offline test has a large time lag and cannot meet the needs of real-time optimization control on

site. The soft sensor model uses easily measurable process variables to estimate the target variables that are difficult to measure in real time, and provides an effective way to solve the above problems. The concept is commonly referred to as software sensors, intelligent sensor, and model-based sensors. Today the concept is well established in engineering, science, and other parts of the process industry [1]. It was first introduced in bioengineering in the early 1990s and is well established and successfully been applied in academic research [2–4]. At present, the commonly used soft sensor modeling methods mainly include: mechanism modeling, data-driven modeling, and hybrid modeling. Mechanism modeling requires a deep understanding of the mechanism of the fermentation process. Due to the high complexity of the fermentation process, the mechanism research process requires considerable resources and a long time. Therefore, the application of mechanism modeling to the fermentation process has greater limitations. The data-driven modeling algorithm is also called the “black box model”. This method does not need to understand the mechanism of the process.

✉ Khalil Ur Rehman  
khaliluaf@hotmail.com

<sup>1</sup> School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, Jiangsu, China

<sup>2</sup> School of Computer and Information Engineering, Jiangxi Normal University, Nanchang 330022, China

It filters out decisive information from the input and output historical data of the process, and obtains the mathematical relationship between auxiliary variables and key variables. Hybrid modeling is a modeling method that combines mechanism modeling and black box modeling; that is, the mechanism modeling is used for the part of the system with known mechanism, and the black box modeling is used for the part of the system with unknown mechanism.

In recent years, many studies have been carried out by utilizing data-driven soft sensor models to predict the key biological variables of the fermentation process. The most commonly used data-driven modeling methods mainly include neural networks (NN), support vector machine (SVM), least square support vector machine (LS-SVM), partial least square (PLS), principal component regression (PCR), and other methods [5]. Because of its powerful nonlinear approximation ability, NN has been successfully applied by many scholars in the soft sensor modeling of biochemical processes in recent years [6, 7]. However, because NN training is based on the principle of experience-based risk minimization, over-learning is prone to lead to a decline in generalization ability. In addition, there are defects such as the difficulty of determining the network structure and the excessive dependence on large sample set learning. SVM is a supervised machine learning method proposed by [8] based on statistical learning theory. It uses the principle of structural minimization to improve its generalization ability, and solves practical problems such as small samples, nonlinearity, high dimensionality, and local minima, while for large data sets, the computation cost is unaffordable. It has been widely used in the field of biochemical process modeling and control [9]. LS-SVM is an extension of the standard SVM. The LS-SVM model selects the form of the error (relaxed variable) two norm in the optimization objective function, and the quadratic programming that needs to be solved in the SVM. The problem is converted into a linear equation system to solve the problem, thereby reducing the computational complexity and effectively improving the learning speed, so it is more suitable for the research of industrial process soft sensor modeling [10, 11]. PCR is a regression technique that projects the variables into lower dimensions and finds principal components, while cannot reflect the relation between X and Y spaces. PLS maximizes the covariance between X and Y spaces, and is commonly used in soft sensor modeling [12]. However, they both require large number of data samples and are weak in nonlinear process modeling.

In this paper, a soft sensor model of fermentation process based on deep neural network (DNN) is established, which has good prediction performance. With large amount of data, DNN enjoys high efficiency and robustness, and it has been commonly used in image classification [13] and natural language processing (NLP) [14],

and shows great performance in chemical industrial processes. The accuracy and validity of the process measurement data largely determine the performance of the soft sensor model. However, the measurement data collected from the fermentation control system platform are inevitably accompanied by errors due to the influence of instrument accuracy and measurement environment, and sometimes, there are even serious significant errors (negligence errors). At present, most of the methods for removing significant errors and reducing random errors from the measured data are simple pre-processing of the data, which is prone to errors of discarding truth or falsifying, which directly affects the modeling effect. The proposed method is based on the deep learning model, combined with improved normal distribution weighting rules, and assigns different weights to each modeled sample to eliminate the effect of random errors on the performance of the soft sensor model.

In the process of soft sensor modeling, some parameters (for example, hyper-parameter) of the model need to be optimized. There is an optimization algorithm called momentum, or gradient descent with momentum [15] which works faster than the standard gradient descent algorithm [16]. The basic concept is to compute an exponentially weighted average of your gradients and then utilize that gradient to update your weights instead. Root-mean-squared propagation or RMSprop optimization algorithm is using a similar idea of the exponentially weighted average of the gradients like gradient descent with momentum, but there is one difference between which is the update of parameters or how the gradients are calculated. It was first described in a Coursera class on neural network taught by Geoffrey Hinton [17]. In this paper, Adam optimization algorithm [18] is used to optimize the selection of DNN model parameters, which can be looked at as a combination of stochastic gradient descent with momentum and RMSprop. Adam is a very famous optimization algorithm in deep learning, which used in many applications. Finally, based on the data of the penicillin and L-lysine fermentation process, the DNN-Adam method was used to establish a soft sensor model with good prediction accuracy and robust performance, which laid the foundation for the optimal control of the biochemical process. The simulation results show that the DNN-Adam modeling method works well in practice and compares favorably to other stochastic optimization techniques.

The rest of the article is prepared as follows: the “[Methodology](#)” describes the structure of soft sensor model based on DNN-Adam. The soft sensor modeling procedure of deep learning is also presented in the “[Methodology](#)”. Real case studies about the prediction of key variables of penicillin and L-lysine fermentation processes and discussions are provided in the “[Results and Discussion](#)”. Finally, some concluding remarks are given in the section “[Conclusion](#)”.

## Methodology

### Deep Neural Networks (DNN)

DNN has some special capabilities that is why it can be used in data-based process modeling [19]. DNN is a combination of more than one hidden layer of nodes or artificial neurons. They are different extensively in design. While single-layer neural networks were useful early in AI development, it should be noted that the vast majority of today’s networks have a multi-layer model. In the process of learning, we start by defining a task and a model. The prediction model consists of architecture and different parameters. For a particular architecture, the values of the parameters ascertain how accurately the model performs the task. However, how do you find good values? By describing a loss function that evaluates how well the model performs. The objective is to minimize the loss and thereby to find parameter values that match predictions with reality. This is the core of training. Numerous optimization algorithms had been utilized to train the parameters of the network for regression and classification problems. The conventional gradient descent algorithm or stochastic gradient descent (SGD) is a very famous and popular optimization algorithm for training the neural networks, which is a derivative-based optimization algorithm that is applied to search for the local minimum of a function, but it has many disadvantages [16], that is why in this paper, Adam optimization algorithm is utilized.

The basic structure diagram of a fully connected soft sensor model based on DNN is shown in Fig. 1. In general, there are two components of a DNN. The first one is the architecture of network, which defines how many layers, how many neurons, and how the neurons are connected. Second is the parameters or values of a network: also known as weights. Consider a four-layer DNN model, on the left-hand side,

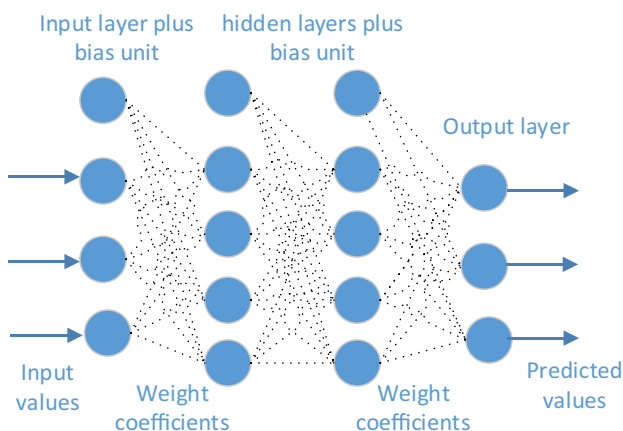


Fig. 1 The basic structure of deep learning model

the input layer is fed from external data (fermentation data)  $x^{(1)}, \dots, x_n^{(i)}$ . The first hidden layer, notice how all inputs are connected to all nodes or neurons in the next layer, so this is called fully connected layers.

### Adam Optimization Algorithm

Optimization algorithms are important part of deep learning, understanding how they work would help you to select which one to utilize for your practical application. Adam optimization algorithms is one of those algorithms that work well across a wide range of deep learning architectures. It is recommended by many well-known deep learning algorithm experts. There are two main advantages of Adam: low memory requirements and works well with a little tuning of hyper-parameters. The implementation steps of Adam are as follows:

- Variables initialization  $M_{dw}, N_{dw}, M_{db}$ , and  $N_{db}$  to zero.
- Compute the derivatives  $dw$  and  $db$  using current batch on iteration T.
- Update the variables  $M_{dw}$  and  $M_{db}$  like momentum:

$$M_{dw} = B_1 x M_{dw} + (1 - B_1) x dW, \tag{1}$$

$$M_{db} = B_1 x M_{db} + (1 - B_1) x db. \tag{2}$$

- Let us update  $N_{dw}$  and  $N_{db}$ :

$$\begin{aligned} N_{dw} &= B_2 x N_{dw} + (1 - B_2) x dW^2 \\ N_{db} &= B_2 x N_{db} + (1 - B_2) x db^2. \end{aligned} \tag{3}$$

- During the implementation of Adam, we implement bias correction:

$$\begin{aligned} M_{dw}^{corrected} &= M_{dw} / (1 - B_1^t) \\ M_{db}^{corrected} &= M_{db} / (1 - B_1^t) \\ N_{dw}^{corrected} &= N_{dw} / (1 - B_2^t) \\ N_{db}^{corrected} &= N_{db} / (1 - B_2^t). \end{aligned} \tag{4}$$

- Let us update  $W$  and  $b$  parameters:

$$\begin{aligned} W &= W - \alpha x (M_{dw}^{corrected} / \text{sqrt}(N_{dw}^{corrected} + \epsilon)) \\ b &= b - \alpha x (M_{db}^{corrected} / \text{sqrt}(N_{db}^{corrected} + \epsilon)). \end{aligned} \tag{5}$$

Sometimes, the value of  $N_{dw}$  could be relatively small. Then, the value of weights could blow up, so that is why in the above equation, a parameter is included with the name of epsilon ' $\epsilon$ ' in the denominator which is set to a small value. The main purpose of using this parameter is to prevent the gradients from blowing up. Where  $B_1$  and  $\beta_2$  are hyper-parameters, and their suggested values are 0.9 and 0.999, respectively. Alpha is the learning rate and the default value is 0.001.

## Soft Sensor Model Based on DNN-Adam

In this research, the Adam optimization algorithm has been used to improve the DNN soft sensor model. The training process of a DNN soft sensor model occurs by selecting the best parameters of the weights of neurons in-between the hidden layer and output layer and the parameter spreads to function mainly in the hidden layer, center hidden layer, and bias of the neurons in output layer. In spite of the fact that the DNN can successfully be utilized as a soft sensor model, nevertheless, total number of neurons in the hidden layer of the DNN influences the complexity of the network and the generalization ability of the network, so deciding the number of hidden neuron layers is only a small part of the problem. If we use too many neurons in the hidden layer, the learning of DNN fails to correct convergence, or overfitting situations may occur. And, if we are using too few neurons in the hidden layer, then underfitting situations will occur [20, 21]. Adding dropout and Early stopping regularization techniques are used to overcome the problems of overfitting and underfitting as explained in [22, 23]. These are the most effective and most commonly used techniques for controlling overfitting and underfitting problem in machine learning, especially neural networks. The learning procedure of deep learning-based soft sensor model is as follows:

- Starting with values for the model parameters ( $w$  weights and  $b$  biases).
- Take a set of samples of fermentation data and propagate data forward to the output layer through the input layer to obtain their predictions.
- Based on the predicted value, calculate the error (the difference between the predicted outcomes and expected outcomes). The error needs to be minimized.
- To discover the optimal weights for the neurons of the DNN soft sensor model, back-propagate the error.
- Update the parameters of the DNN soft sensor model using the propagated information with the Adam optimization

algorithm in a way that the total loss is decreased and a more effective model is attained.

Repeat the steps given above over multiple epochs to learn ideal weights or until we consider that we have a good model (Fig. 2).

### Soft Sensor Modeling Procedure

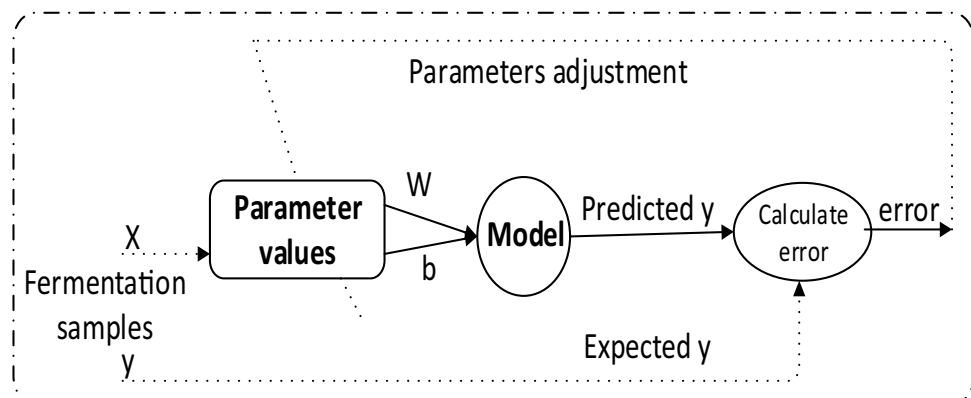
In this research, the training procedure of the DNN was conducted on laptop, with Intel Core i3 @2.40 GHz, 8 GB RAM. The software tools are Windows 10 pro, MathType, Visio 2016, Anaconda, Python version 3.7.1, Tensorflow 1.14.0, and Keras 2.2.5. The training time is within the range of 20 s–1 m. DNN takes more time than simple NN network, and more layers take more computation time. Two fully connected hidden layers are used to build the DNN model, 64 neurons in the first hidden layer and 32 neurons in the second hidden layer, respectively, and the Rectified Linear Unit (ReLU) activation function is used for this experiment. Figures 3 and 6 shows the actual and predicted curves based on both soft sensor models. The graphs and tables with error information of different approaches are listed in Tables 2 and 4, relating to Figs. 4, 5, 7, and 8, which proves that DNN-Adam offers the narrowest error range.

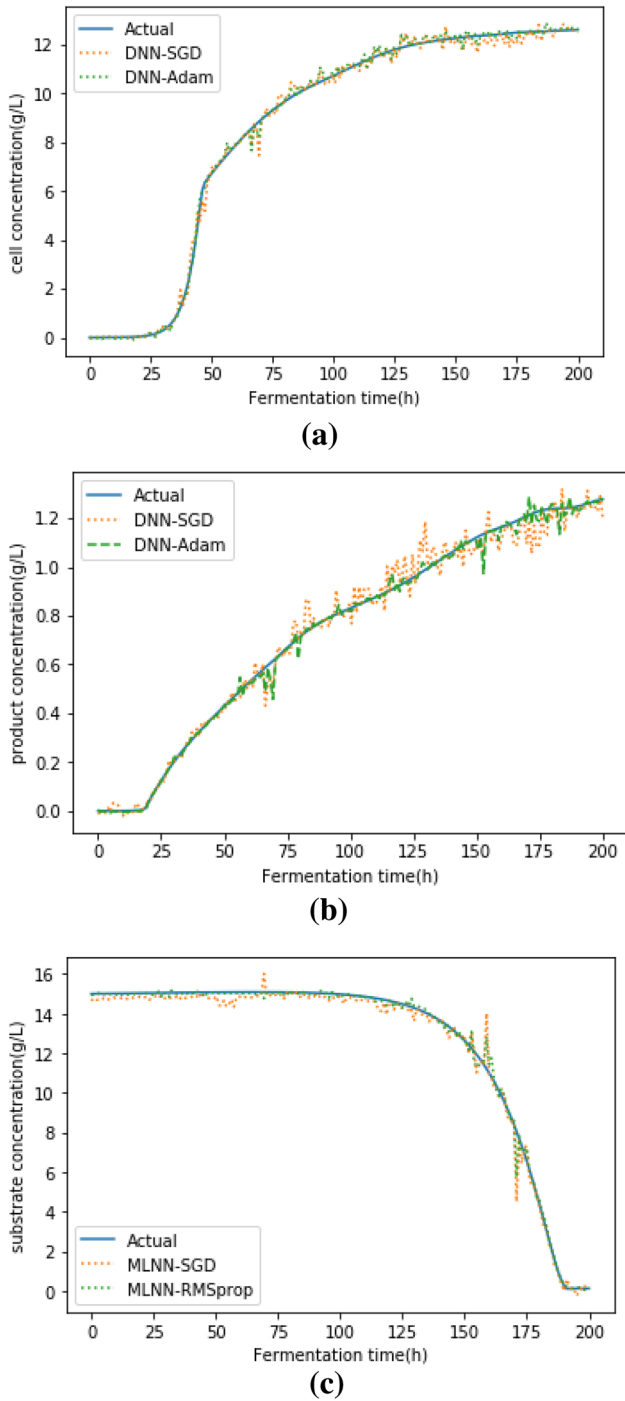
## Results and Discussion

### Case Study I: Penicillin Fermentation Process

To verify the accuracy of a soft sensor model based on DNN-Adam for real-time on-line estimation of key biological variables of penicillin fermentation, this section uses the simulation software for testing and analysis. The penicillin fermentation process is one of the important industrial processes of biochemical reactions, and the establishment of a prediction model of its key variables is of great significance for the optimal control of the fermentation process.

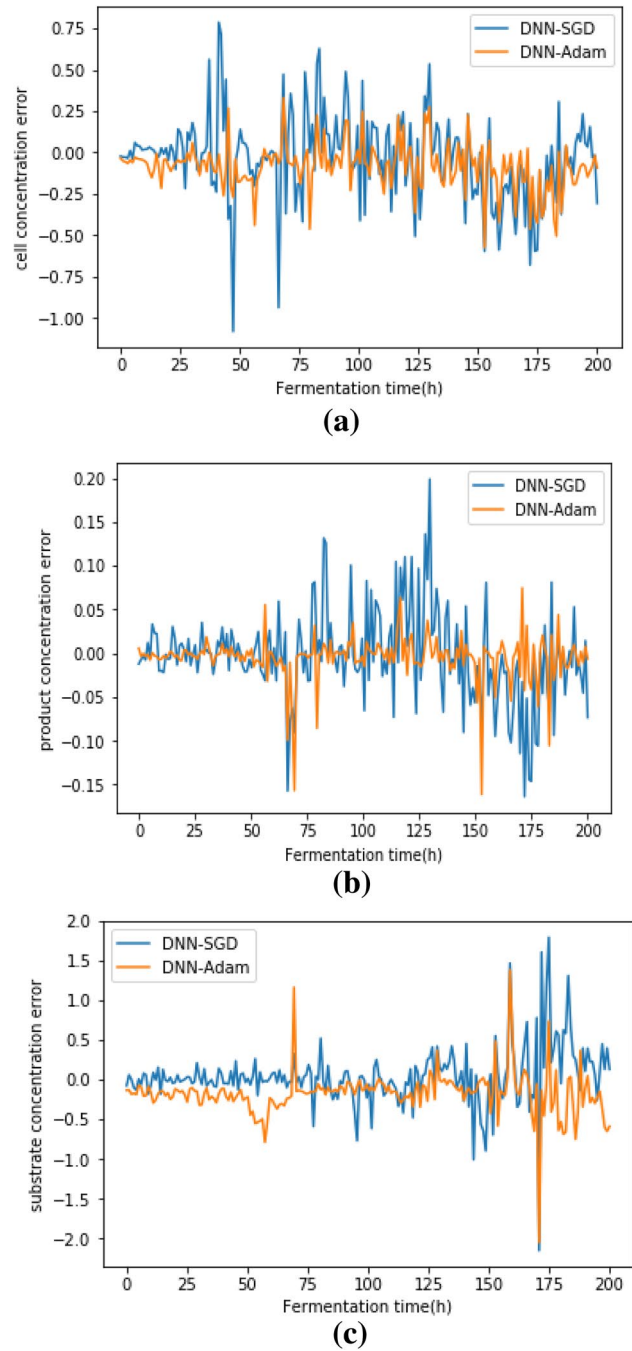
**Fig. 2** Learning process of a deep learning-based soft sensor model in a general way





**Fig. 3** Comparison of soft sensor results (a cell concentration; b product concentration; c substrate concentration)

However, due to the complex biochemical reaction process, the fermentation process has multivariable, nonlinear, time-varying, and uncertain process characteristics, especially some key biological variables cannot be measured in real time due to technical or economic reasons. The precise model of the fermentation process faces great difficulties,



**Fig. 4** Error curve comparison diagrams (a cell concentration error; b product concentration error; c substrate concentration error)

which directly affects the control and optimization of the entire fermentation process. To solve the above problems, a soft sensor modeling method is introduced.

In the process of soft sensor modeling, the selection of auxiliary variables is also very important. The general gray correlation method can be used to determine the auxiliary variables of the soft sensor model, but this method has defects in data processing. In this paper, the consistent

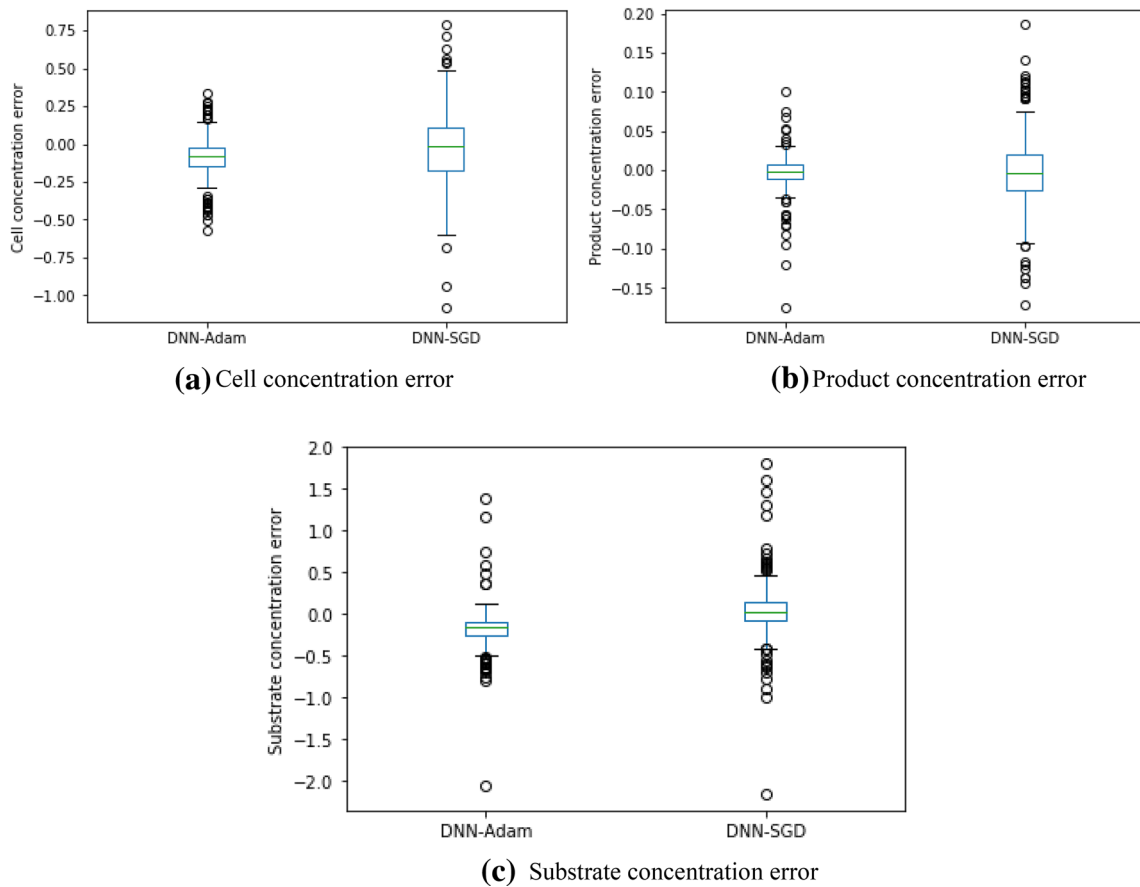


Fig. 5 Error distribution of the penicillin fermentation process

Table 1 Process variables used for soft sensor in penicillin process

Variable	Description
$u_1$	Glucose flow
$u_2$	Corn pulp flow rate
$u_3$	Flow rate of potassium
$u_4$	Calcium carbonate flow rate
$x_5$	Flow rate of gluten powder
$x_6$	Dissolved oxygen concentration
$x_7$	Carbon dioxide concentration
$x_8$	[H <sup>+</sup> ] concentrations
$x_9$	Fermentation broth volume
$X$	Cell concentration
$S$	Substrate concentration
$P$	Product concentration

correlation method is used to calculate the correlation between each environmental variable and the dominant variable, and to determine the auxiliary variables of the soft sensor model. The cell concentration, substrate concentration, and product concentration were selected as target variables. Table 1 lists the quality variables used in soft sensor model.

In this example, ten batches of experimental data were collected with 72 h span between every batch. Among them, the first six batches of the fermentation data were used to train the DNN model for minimum model error. The 7th and 8th batch of fermentation data are used for cross-validation and the last two batches were used to test the final soft sensor model. To improve the accuracy, the sample data should be normalized with the normalization formula as follows:

$$i' = \frac{i_{\max} - i}{i_{\max} - i_{\min}}, \tag{6}$$

where  $i'$  is determine to the variable name where normalized fermentation dataset will be stored;  $i$  is the actual fermentation samples,  $i_{\max}$  is the maximum value of the fermentation sample dataset, and  $i_{\min}$  is the minimum value of the fermentation sample dataset. To verify the performance and effectiveness of the Adam optimization algorithm, compared to another common optimization technique like stochastic gradient descent (SGD) with momentum. The ReLU activation function is adopted in both algorithms, the parameter  $\alpha=0.001$ , and  $\beta=0.9$ . The soft sensor curves and relative error curves of the key biological variables of penicillin

fermentation based on DNN-Adam and DNN-SGD model are shown in Figs. 3 and 4. The predicted results show that the DNN-Adam-based soft sensor model provides comparable or better results when compared to the other modeling methods. The boxplot with the error report of DNN-Adam and DNN-SGD is listed in Fig. 5. As demonstrated in Fig. 5, the boxes in blue specify the ranges between the upper and lower quartiles. It can be clearly seen that the developed DNN-Adam approach offers the narrowest error range, which verifies that it delivers a better quality prediction result than other techniques.

To determine the performance evaluation of soft sensor model based on DNN-Adam, Table 2 displays the predicted mean square error (MSE), root-mean-square error (RMSE), and mean absolute error (MAE) results of both soft sensor models. It can be seen that the values of MSE, RMSE, and MAE are less than the SGD algorithm. It shows that the predictive value of this technique has less uncertainty, which means that the results of the soft sensor model based on Adam are more reliable. Through experiments and tests, it was concluded that the soft sensor model based on DNN-Adam is better than DNN-SGD for the estimation of variables of the penicillin fermentation process.

### Case Study II: L-lysine Fermentation Process

In this section, a real case study of DNN-Adam soft sensor model based on L-lysine fermentation process is presented, which is an important biological research object. The L-lysine fermentation process is a time-varying, nonlinear, random multivariable coupling system. Involving complex processes of microbial cell growth and metabolism, the influencing factors are complex, and the correlation of variables is serious. The cell concentration and lysine product concentration during fermentation are important biochemical process variables that directly reflect the fermentation quality. Effective control of these process parameters plays an important role in implementing optimal control of the fermentation process, constructing an optimal growth environment for microorganisms, and improving L-lysine yield and quality [24, 25]. These biochemical process variables that directly reflect the state information of the fermentation

process are determined by various nutrients added (commonly known as feed) during the fermentation process, and are also related to the environmental variables of the fermentation process (fermentation broth pH, dissolved oxygen concentration DO, temperature, pressure, airflow rate, and motor stirring). Table 3 lists the quality variables used in soft sensor model. The relationship between them is complicated; for example, only through the observation of environmental variables, based on experience to adjust the production process to achieve manual operation. If it is improperly controlled or regulated, it will affect the fermentation production at the lightest level, and the whole fermentation process will fail at the worst, causing serious economic losses. Therefore, the optimization control of lysine fermentation process needs to be solved urgently.

In this case study, ten batches of experimental data were collected with 72 h span between every batch. Among them, the first nine batches of the fermentation data are applied to train the DNN model for minimum model error. And the 10th batch is selected to examine the identification precision of the model.

The actual results and the predicted results of L-lysine fermentation process variables based on DNN-Adam soft sensor model and DNN-SGD with momentum are shown in Figs. 6 and 7. Though some outliers of DNN-Adam can be observed with higher prediction errors, the boxplot in Fig. 8 shows that the total outlier number of DNN-Adam is less than other techniques. In comparison with the use of the DNN-SGD soft sensor model, DNN-Adam soft sensor

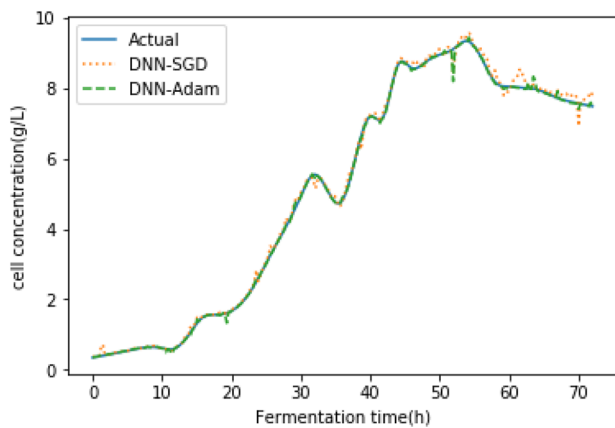
**Table 2** Quality prediction results of penicillin fermentation process

Dominant variables	DNN-SGD Prediction model			DNN-Adam Prediction model		
	MSE	RMSE	MAE	MSE	RMSE	MAE
<i>X</i>	0.238	0.488	0.378	0.137	0.370	0.301
<i>P</i>	0.006	0.082	0.066	0.005	0.072	0.057
<i>S</i>	0.475	0.689	0.459	0.354	0.595	0.445

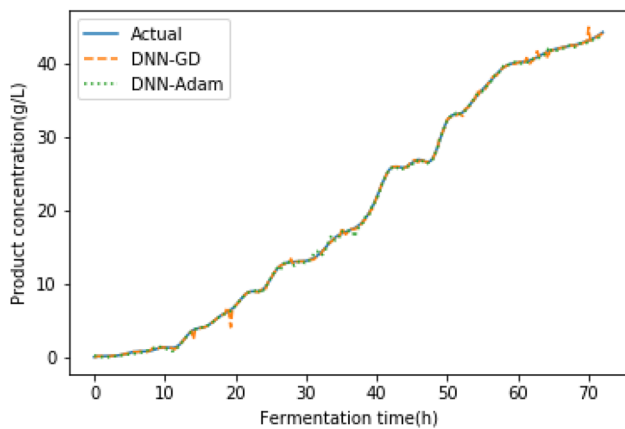
*X* cell concentration, *P* product concentration, *S* substrate concentration, *MSE* mean square error, *RMSE* root-mean-square error, *MAE* mean absolute error

**Table 3** Process variables used for soft sensor in L-lysine process

Variables	Description
$x_1$	Fermentation broth pH
$x_2$	Dissolved oxygen concentration
$x_3$	Fermenter tank temperature
$x_4$	Fermenter tank pressure
$x_5$	Airflow rate
$x_6$	Motor stirring
<i>X</i>	Cell concentration
<i>P</i>	Product concentration



(a)



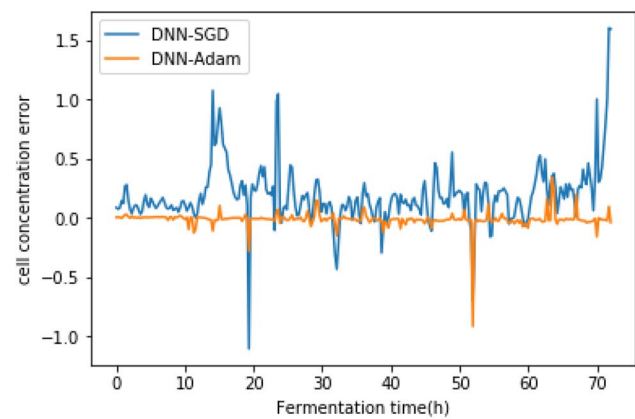
(b)

**Fig. 6** Comparison of soft sensor results (a cell concentration; b product concentration)

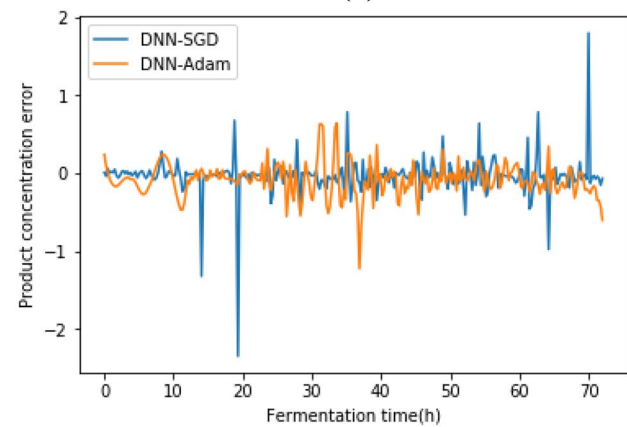
model produces prediction results that are closer to real values. To determine the performance evaluation of corresponding soft sensor model, Table 4 displays the predicted mean square error (MSE), root-mean-square error (RMSE), and mean absolute error (MAE) results of both soft sensor models on test dataset. It can be seen that the values of MSE, RMSE, and MAE are less than the SGD algorithm. It is verified that the proposed soft sensor model based on DNN-Adam can perform more accurate real-time prediction of key variables of lysine fermentation process and it has higher prediction accuracy and adaptability.

## Conclusion

To solve the problems of real-time measurement of key biological variables (such as cell concentration, substrate concentration, product concentration, and so on) in the



(a)

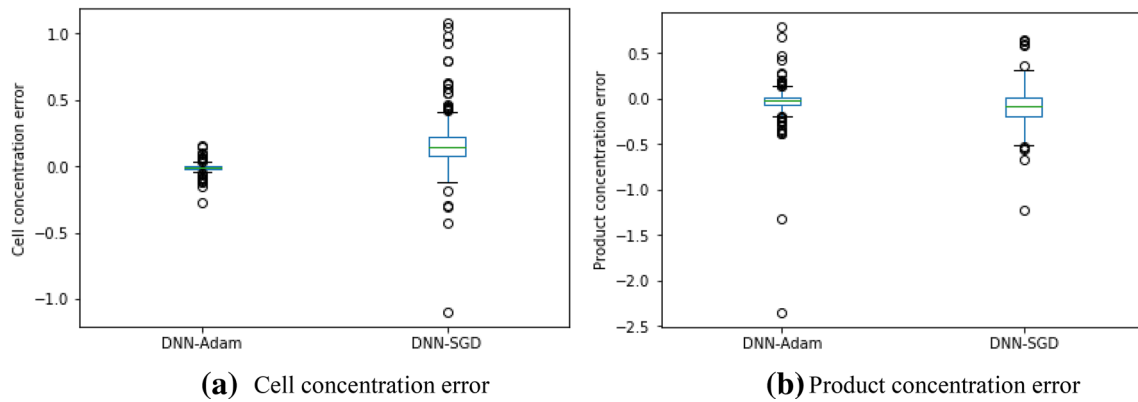


(b)

**Fig. 7** Error curve comparison diagrams (a cell concentration error; b product concentration error)

microbial fermentation process, a soft sensor modeling method based on DNN was proposed. At the same time, the Adam algorithm is used to optimize the hyper-parameters of the DNN model, which is a technique for efficient stochastic optimization that only requires first-order gradients with little memory requirement. The auxiliary variables and the dominant variables of soft sensor model were determined based on the analysis of the mechanism of fermentation process using the consistent correlation method. According to the sample fitting error, combined with improved normal distribution weighting rules, assigns different weights to each modeled sample to reduce the impact of random errors on the performance of the model. In this paper, the penicillin and L-lysine fermentation processes are taken as the research object, and then, soft sensor model was established by DNN based on Adam optimization algorithm. The simulation results show that the soft sensor model based on DNN-Adam has higher prediction accuracy and better generalization ability than DNN-SGD soft sensor model.





**Fig. 8** Error distribution of the L-lysine fermentation process

**Table 4** Quality prediction results of L-lysine fermentation process

Dominant variables	DNN-SGD Prediction model			DNN-Adam Prediction model		
	MSE	RMSE	MAE	MSE	RMSE	MAE
$X$	0.059	0.243	0.143	0.023	0.152	0.085
$P$	1.303	1.141	0.683	0.904	0.951	0.591

$X$  cell concentration,  $P$  product concentration,  $MSE$  mean square error,  $RMSE$  root-mean-square error,  $MAE$  mean absolute error

**Acknowledgements** The National Science Research Foundation of China (41376175), The Natural Science Foundation of Jiangsu Province (BK20140568, BK20151345), and a project funded by the priority academic program development of Jiangsu higher education institutions (PAPD).

## Compliance with Ethical Standards

**Conflict of Interest** The authors declare that they have no conflict of interest.

## References

- Kadlec P, Gabrys B, Strandt S. Data-driven soft sensors in the process industry. *Comput Chem Eng*. 2009;33(4):795–814.
- Chérury A. Software sensors in bioprocess engineering. *J Biotechnol*. 1997;52(3):193–9.
- Luttmann R, Bracewell DG, Cornelissen G, Gernaey KV, Glassey J, Hass VC, Kaiser C, Preusse C, Striedner G, Mandenius CF. Soft sensors in bioprocessing: a status report and recommendations. *Biotechnol J*. 2012;7(8):1040–8.
- Randek J, Mandenius C-F. On-line soft sensing in upstream bioprocessing. *Crit Rev Biotechnol*. 2018;38(1):106–21.
- Zhu X, Rehman KU, Wang B, Shahzad M. Modern soft-sensing modeling methods for fermentation processes. *Sensors*. 2020;20(6):1771.
- Nasr N, Hafez H, El Naggat MH, Nakhla G. Application of artificial neural networks for modeling of biohydrogen production. *Int J Hydrogen Energy*. 2013;38(8):3189–95.
- Sivapathasekaran C, Sen R. Performance evaluation of an ANN-GA aided experimental modeling and optimization procedure for enhanced synthesis of marine biosurfactant in a stirred tank reactor. *J Chem Technol Biotechnol*. 2013;88(5):794–9.
- Cortes C, Vapnik V. Support-vector networks. *Mach Learn*. 1995;20(3):273–97.
- Wang B, Sun Y, Ji X, Huang Y, Ji L, Huang L. Soft-sensor modeling for lysine fermentation processes based on PSO-SVM inversion. *CIESC J*. 2012;63(9):3000–7.
- Gu Y, Zhao W, Wu Z. Least squares support vector machine algorithm [J]. *J Tsinghua Univ (Sci Technol)*. 2010;7:1063–6.
- Ou Yang H-B, Li S, Zhang P, Kong X. Model penicillin fermentation by least squares support vector machine with tuning based on amended harmony search. *Int J Biomath*. 2015;8(03):1550037.
- Kresta J, Marlin T, MacGregor J. Development of inferential process models using PLS. *Comput Chem Eng*. 1994;18(7):597–611.
- Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, 2012. pp 1097–1105
- Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. *arXiv preprint*; 2013. [arXiv:1301.3781](https://arxiv.org/abs/1301.3781)
- Qian N. On the momentum term in gradient descent learning algorithms. *Neural Networks*. 1999;12(1):145–51.
- Sutton R. Two problems with back propagation and other steepest descent learning procedures for networks. In: *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, 1986, 1986. pp 823–832
- Tieleman T, Hinton G. Lecture 6.5-rmsprop, coursera: *Neural networks for machine learning*. University of Toronto, Technical Report; 2012

18. Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint; 2014. [arXiv:1412.6980](https://arxiv.org/abs/1412.6980)
19. Lin Y, Yan W Study of soft sensor modeling based on deep learning. In: 2015 American Control Conference (ACC), 2015. IEEE, pp 5830–5835
20. Hinton GE, Osindero S, Teh Y-W. A fast learning algorithm for deep belief nets. *Neural Comput.* 2006;18(7):1527–54.
21. Hornik K. Approximation capabilities of multilayer feedforward networks. *Neural Networks.* 1991;4(2):251–7.
22. Plaut D, Nowlan S, Hinton G. Experiments on learning by back propagation. Computer Science Department Carnegie-Mellon University Pittsburgh; 1986.
23. Chollet F. Deep learning mit Python und Keras: Das Praxis-Handbuch vom Entwickler der Keras-Bibliothek. Wachtendonk: MITP-Verlags GmbH & Co; 2018.
24. Wang B, Shahzad M, Zhu X, Rehman KU, Uddin S. A non-linear model predictive control based on grey-wolf optimization using least-square support vector machine for product concentration-control in l-lysine fermentation. *Sensors.* 2020;20:3335.
25. Wang B, Shahzad M, Zhu X, et al. Soft-sensor modeling for L-lysine fermentation process based on hybrid ICS-MLSSVM. *Sci Rep.* 2020;10:11630.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.