Novel multimodal data fusion soft sensor modeling framework based on meta-learning networks for complex chemical process

Gao Xiaoyong*. Liu Yanchao. Xie Yi***. Huang Dexian******

**Department of Automation, China University of Petroleum, Beijing 102249, China*
(Tel: 18513957176; e-mail: x.gao@cup.edu.cn).

***Department of Automation, China University of Petroleum, Beijing 102249, China*
(e-mail: lyc6162021@163.com).

**** State Pipeline Group Beijing Pipeline Co. LTD, Beijing, 100101, China (e-mail:*
394464988@qq.com)

***** Department of Automation, Tsinghua University, Beijing,100084, China*
(e-mail: huangdx@tsinghua.edu.cn)

Abstract: Data-driven soft sensor has been widely used in industrial processes. However, complex industrial processes all exhibit nonlinear and multimodal characteristics due to varying operating conditions. Multimodal data characteristics will cause the deterioration of soft sensor performance. Therefore, in this article, a modified Model-Agnostic Meta-Learning (MAML) based on K-Means (KM) is proposed. Firstly, the KM method is introduced to cluster the multimodal processed data, and then extract the clustered data to form multiple tasks. After that, MAML method is adopted to train a group of initialization parameters. The sum of each task's loss function is introduced to adjust the initial parameters by one or more steps of gradient. The proposed model is finally applied and verified in the Purified Terephthalic Acid (PTA) solvent system. Compared to some conventional methods, the prediction accuracy is improved by more than 70%. The result demonstrates the superiority in the proposed method.

Keywords: data-driven, multimodal, soft sensor, K-Means, MAML

1. INTRODUCTION

With the continuous development of the process industry, the requirements for monitoring and controlling key quality indicators in the production process are becoming higher and higher (Kano M et al. 2008; Yuan X F et al. 2018; Kaneko H et al. 2014; Wang J L et al. 2018). Since hardware sensors are usually expensive, difficult-to-maintain and failure-prone (Kadlec P et al. 2011). Soft sensor has been widely used in industrial fields (Gao X Q et al. 2016; Yuan X F et al. 2019). Currently, there are mechanism and/or data-driven based modeling methods in soft sensor research (Sun Q et al. 2019). Due to the complexity of the production processes, it is laborious and hard to implement using a mechanism based method. In contrast, the historical data typically used for monitoring allows nonlinear black or gray-box process models to identify. However, process variables are highly correlated, and the process data have multimodal characteristics. Such large number of process variables and data are much larger than their dimensionality, manifesting as data-rich and information-poor (Park S Y et al. 2000). In this case, the more mainstream approach is to solve the problem by using the principal components analysis (PCA), and the least squares method (Dong D et al. 1996). However, in the case of non-Gaussian distribution, the principal components derived from the PCA method may not be optimal (Lin K Z et al. 2021). The least squares method not only requires a large amount of data as support, but also is difficult to deal with complex multimodal data well. The support vector machine (SVM) approach can solve the problem of small samples very well (Popli K et al. 2018), thus avoids the dimensional disaster. Yet, its computation shows an exponential growth with the samples number. Although the artificial neural network is a good fit for the nonlinear model (Gonzaga J C B et al. 2009; Ko Y D et al. 2011; Mohammadpoor M et al. 2018), there are still problems of local optimum and failure to converge. Model mismatch of soft sensor under varying state and dynamic conditions is the major challenge. Thus, when the conditions change, a large amount of multimodal data will be generated, which leads to model mismatches. How to process complex multimodal data and make the soft sensor precisely track the varying conditions is the main concern of this paper.

With the improvement of computing power, deep learning has become a hot research topic and many related algorithms have been applied to soft sensor modeling (Sun Q Q et al. 2019). Many scholars have done relevant studies. Shang C et al. introduced deep learning to soft sensor modeling firstly in 2014 and demonstrated the effectiveness (Shang C et al. 2014). Lian P L et al. proposed a soft sensor method for air preheater based on (deep belief network) DSN- (improved particle swarm optimization) IPSO- (support vector regression) SVR, using a particle swarm optimization algorithm. However, the model construction process is complicated, which reduces the generalization ability of the model (Lian P L et al. 2020). Wang J L et al. proposed a dynamic soft sensor based on convolutional neural networks, which obtained better prediction results by fully mining the information of historical data (Wang J L et al. 2020). Zhu Q Y proposed a hybrid learning algorithm based on Evolutionary Extreme Learning Machine, which used the differential evolutionary algorithm to select the input weights, resulting in a fast solution speed. However, the above methods require a complex model design process and do not consider the multimodality of the data.

In recent years, Model-Agnostic Meta-Learning (MAML) has been successfully applied to speech recognition (Yildirim S et al. 2021; Xu Z X et al. 2021) and image feature extraction (Zhou F et al. 2021), nonlinear extension has also been widely studied (Ye H L et al. 2021). This method can also be used to establish the required nonlinear soft sensor modeling. To make full use of multimodal data of industrial processes, this paper proposed a multimodal data recognition soft sensor modeling framework based on K-Means and Model-Agnostic Meta-Learning method. Since MAML is a gradient-based updating method, through a gradient descent, a set of initialization parameters is obtained. Applying this initialization parameters to neural network will get better prediction effect. The Back Propagation (BP) neural network model is chosen as an example in this paper. The modified MAML method can solve the problem of small sample data and multi-modal data at the same time, so it has better performance.

1.1 K-means model

The training unit of the MAML method is defined as task, and multiple tasks compose a batch. The objects of MAML are tasks composed of samples from sample points. In the iterative updating process of MAML, the selection of tasks is random. Therefore, to extract the characteristics of the data at a deeper level, the K-Means method is firstly used to preprocess the data. The data is first clustered, and the samples are divided into several clusters based on the characteristics of the samples. During tasks sampling, samples are sampled based on the clustered data to ensure the sample features of each category can be selected.

The core goal of K-Means is to divide a given data set into K clusters (K is a super parameter). Firstly, the data is preprocessed, then K centers are randomly selected and denoted as : $\{\mu_1^{(0)}, \mu_2^{(0)}, \ldots, \mu_k^{(0)}\}$.The loss function can be defined as:

$$J(c,\mu) = \sum_{i=1}^{M} ||x_i - \mu_{c_i}||^2 \qquad (1)$$

where x_i represents the sample number i, c_i is the cluster to x_i, μ_{c_i} represents the center point corresponding to the cluster, and M is the total number of samples. For each sample x_i, it is assigned to the nearest center:

$$c_i^t < -argmin_k ||x_i - \mu_k^t||^2 \qquad (2)$$

Then, for each class center K, the center of the cluster is recalculated:

$$\mu_k^{(t+1)} < -argmin_\mu \sum_{i:c_i^t=k}^{b} ||x_i - \mu||^2 \qquad (3)$$

The iterative calculation is repeated until the loss function converges. The flow chart of the proposed training task acquisition process is shown in Fig. 1.

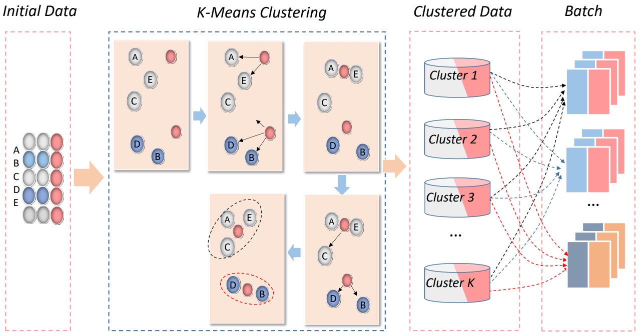

Figure 1. Flow chart of the proposed training task acquisition process.

1.2 K-means &Model-agnostic meta-learning

MAML is a two-phase learning process. In the first phase process, it generates a set of initialization parameters of BP neural network. In the second phase process, the neural network parameters are optimized and adjusted. Since KM-MAML ultimately aims to solve the problem of better learning on the new tasks, the learning experience in the previous tasks should be transferred. So, the training process consists of two phases. Firstly, the data are divided into two parts, $D_{meta\text{-}training}$ data and $D_{meta\text{-}testing}$ data. $D_{meta\text{-}training}$ data is for the first gradient update and $D_{meta\text{-}testing}$ data is for the second. In the first training phase, $D_{meta\text{-}training}$ data is clustered based on K-Means model to form multiple data clusters. Then, each data cluster is randomly sampled to form tasks, and multiple tasks form a batch. These batches are used to train BP neural network respectively, and the loss under each batch is calculated. Each task still has a training set and testing set. The training set is called support set, and the testing set is called query set. The sampling points corresponding to different tasks are also different. After training on several such meta-training tasks, meta-testing or reasoning should be carried out on a new task. In the second fine-tuning phase, the $D_{meta\text{-}training}$ data also needs to be clustered and sampled to form batch. However, the K-Means parameters will be different in this case. The process of data processing is the same as the first gradient update. But at this time, it uses the sum of each task's loss function to adjust the initial parameters. In the first training phase, the obtained parameters are intermediate parameters, and are not used in the final network. While the parameters of the network are really updated in the second fine-tuning phase. Since the testing tasks are samples of several clusters that have never been seen before, the model can be fine-tuned on these samples. However, the model has already learned the "experience" during the first training, so it can quickly adapt to the testing tasks in the second fine-tuning phase. The flow chart of the modeling process is shown in Fig. 2.

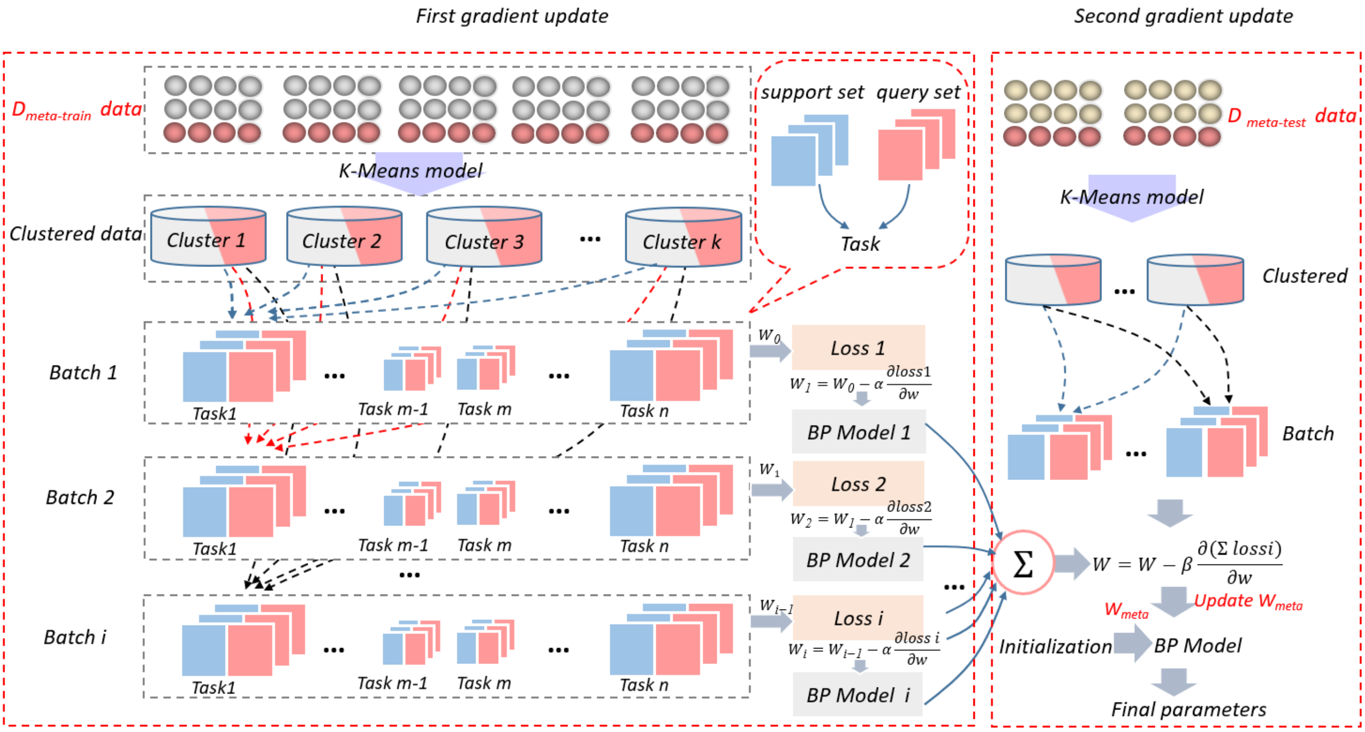

Figure 2. Flow chart of the modeling process.

2. MODELING AND ANALYSIS

The goal of MAML is to train models for a variety of learning tasks, so new learning tasks can be solved by using only a small number of training samples. Suppose the training data, $Data = \{X, Y\}$, includes two parts, input data $X \in R^{n\times m}$and output data $Y \in R^{n\times 1}$, which obey the distribution $p(\tau)$. Formally, each task $\tau = \{L(x_1, a_2, \dots, x_H, a_H), q(x_1), q(x_{t+1}|x_t, a_t)\}$ consists of a loss function L, a distribution over initial observations $q(x_1)$, a transition distribution $q(x_{t+1}|x_t, a_t)$. Consider that a model F maps observations x to outputs y, then samples K new tasks τ_i from $p(\tau)$. The model F is trained with K samples and the feedback is from the corresponding loss L_{τ_i} from τ_i. Then test on new samples from τ_j to complete the entire training process of the model.

The initial parameters θ of the neural network are defined as:

$$\theta^0 = [\theta_1, \theta_2, \dots, \theta_n]^T \tag{4}$$

Suppose a batch of samples are randomly selected, including K samples. Each sample has n input features and m outputs corresponding label data. The input matrix of the sample is (the rows represent the sample, and the columns represent the characteristics (dimensions) of the input sample):

$$M_{in} = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_n^1 \\ x_1^2 & x_2^2 & & x_n^2 \\ & \vdots & \ddots & \vdots \\ x_1^l & x_2^l & \cdots & x_n^l \end{bmatrix} \tag{5}$$

The corresponding output matrix is:

$$M_{pred}(\theta) = \begin{bmatrix} \hat{y}_1^1(\theta) & \hat{y}_2^1(\theta) & \cdots & \hat{y}_m^1(\theta) \\ \hat{y}_1^2(\theta) & \hat{y}_2^2(\theta) & & \hat{y}_m^2(\theta) \\ & \vdots & \ddots & \vdots \\ \hat{y}_1^l(\theta) & \hat{y}_2^l(\theta) & \cdots & \hat{y}_m^l(\theta) \end{bmatrix} \tag{6}$$

A mean square error loss function $L_\tau(\theta)$ is defined to measure the performance of the neural network model:

$$L_\tau(\theta) = MSE_\tau = \frac{1}{2n \times m} \times [\sum_i^n \sum_j^m (\hat{y}_j^i(\theta) - y_j^i(\theta))^2] \tag{7}$$

The derivative of the loss function $L_\tau(\theta)$ to θ is:

$$g = \nabla_\theta L_\tau(\theta) = \frac{\partial L_\tau(\theta)}{\partial \theta} = \begin{bmatrix} \partial L_\tau(\theta)/\partial\theta_1 \\ \partial L_\tau(\theta)/\partial\theta_2 \\ \vdots \\ \partial L_\tau(\theta)/\partial\theta_n \end{bmatrix} \tag{8}$$

The first gradient descent can be expressed as:

$$g_1 = \nabla_\theta L_\tau(\theta_0)$$

$$\theta^1 = \theta^0{}_0 - \alpha g_1 \tag{9}$$

Then consider the case of performing k inner gradient steps, k≥1. Starting with the initial model parameter θ_{meta}:

$$\theta_0 = \theta_{meta}$$

$$\theta_1 = \theta_0 - \alpha\nabla_\theta L^{(0)}(\theta_0)$$

$$\theta_2 = \theta_1 - \alpha\nabla_\theta L^{(0)}(\theta_1)$$

$$\cdots\cdots \tag{10}$$

$$\theta_k = \theta_{k-1} - \alpha\nabla_\theta L^{(0)}(\theta_{k-1})$$

$$\begin{aligned} g_{MAML} &= \nabla_\theta L^{(1)}(\theta_k) \\ &= \theta_k L^{(1)}(\theta_k) \cdot (\nabla_{\theta-1}\theta_k) \cdots (\nabla_{\theta_0}\theta_1)(\nabla_\theta\theta_0) \\ &= \theta_k L^{(1)}(\theta_k) \cdot \left(\prod_{i=1}^{k} \nabla_{\theta_{i-1}}\theta_i\right) \cdot I \\ &= \theta_k L^{(1)}(\theta_k) \cdot \prod_{i=1}^{k} \nabla_{\theta_{i-1}}(\theta_{i-1} - \alpha\nabla_\theta L^{(0)}(\theta_{i-1})) \\ &= \theta_k L^{(1)}(\theta_k) \cdot \prod_{i=1}^{k} (I - \alpha\nabla_{\theta_{i-1}}\left(\nabla_\theta L^{(0)}(\theta_{i-1})\right)) \quad (11) \end{aligned}$$

The objective of MAML is to find a set of initial parameters θ of the model, so that the loss function can be minimized after k times of gradient update. The model is faced with randomly selected new tasks τ_i. In mathematical language, namely:

$$\begin{aligned} \theta^* &= \arg\min_\theta \sum_{\tau_i \sim p(\tau)} L_{\tau_i}^{(1)}(F_{\theta_i}) \\ &= \arg\min_\theta \sum_{\tau_i \sim p(\tau)} L_{\tau_i}^{(1)}(F_{\theta-\alpha\nabla_\theta L_{\tau_i}^{(0)}(f_\theta)}) \quad (12) \end{aligned}$$

3.INDUSTRIAL CASE STUDY

3.1 Purified terephthalic acid (PTA) reaction generation process

Experiments were conducted by using PTA production process data and comparative experiments with BP, SVM and Partial Least Squares (PLS) methods. The measures used are the following three commonly used methods:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \quad (13)$$

$$R2Score = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(y_i - y^2)^2} \quad (14)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \quad (15)$$

PTA is mainly from Paraxylene (PX) oxidation reaction, and the main devices include the main reactor, the dewatering condenser, and the crystallizer, as the flow chart of PX oxidation reaction process shown in Fig. 3. The main reactor is a continuous stirred tank reactor, under the conditions of PX and oxygen as raw materials, cobalt, manganese, bromine as catalysts for a series of oxidation to produce PTA. The 4-CBA is the main quality index of PTA products, and its unit is *ppm*. It will have a great impact on product quality.

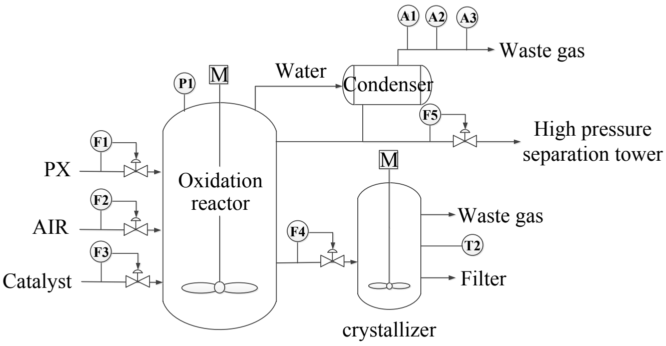

Figure 3. Flow chart of PX oxidation reaction process.

3.2 Variable selection

The process variables used as soft sensor inputs is based on the physical context, as shown in Table 1.

Table 1. Process variables selected

Input variables	Tag	Description
U1	F1	PX feed flow
U2	F2	AIR feed flow
U3	F3	Catalyst feed flow
U4	P1	Reactor pressure
U5	T1	Reactor temperature
U6	T2	Crystallizer temperature
U7	A1	CO concentration
U8	A2	CO_2 concentration
U9	A3	O_2 concentration

In this case study, all modeling data and testing data were obtained from actual process data measured, and recorded at a refinery in northern China. The quality variable 4-CBA content was analyzed by laboratory sampling assays every four hours. A dataset containing 400 process quality samples corresponding to the last two months of process operations was selected. For the KM-MAML, the $D_{meta\text{-}training}$ data was split into a training set of 120 samples and a testing set of 40 samples. In addition, the $D_{meta\text{-}testing}$ data includes a training set of 100 samples and a testing set of 40 samples for fine-tuning. Other 100 samples are validation set. For other methods (BP、PLS、SVM), the ratio of training set, testing set to validation set is 2:1:1.

In the KM-MAML model framework, the input layer、hidden layer、output layer of BP neural network model are 9, 20, 1.For the KM-MAML, the $D_{meta\text{-}training}$ data is processed for clustering, and the input data is 9-dimensional data. The clustered data is used to select tasks and applyed to the training model. Do the same operation for $D_{meta\text{-}testing}$ in the second fine-tuning phase.

3.3 Results and discussions

The comparison results in Table 2 show that the method of K-Means (KM)-MAML has the best performance with the MAE of 0.252, comparing the results of SVM, PLS, and BP with 0.687, 0.621, and 0.841. Obviously, its prediction error is smaller; For MSE, the prediction accuracy of KM-MAML of 0.172 is significantly better than that of SVM (0.740), PLS (0.586), and BP (0.787). And for R2Score, the fitting performance of the KM-MAML method is closer to 1, which indicates a good match between the proposed model and the real process.

Table 2. Performance comparison metrics results

METRICS	SVM	PLS	BP	KM-MAML
MSE	0.740	0.586	0.787	0.172
R2SCORE	0.307	0.451	-0.031	0.805
MAE	0.687	0.621	0.841	0.252

Meanwhile, the prediction curves of the SVM, the PLS, the BP, and the KM-MAML in Fig. 6 show that SVM and PLS are still very sensitive to data outliers. While BP can overcome this situation, but its random initial parameters will lead to unsatisfactory prediction accuracy. Compared to others, the KM-MAML can better handle outliers and enhance the robustness of the model to improve the prediction accuracy because of the use of previously learned "experience".

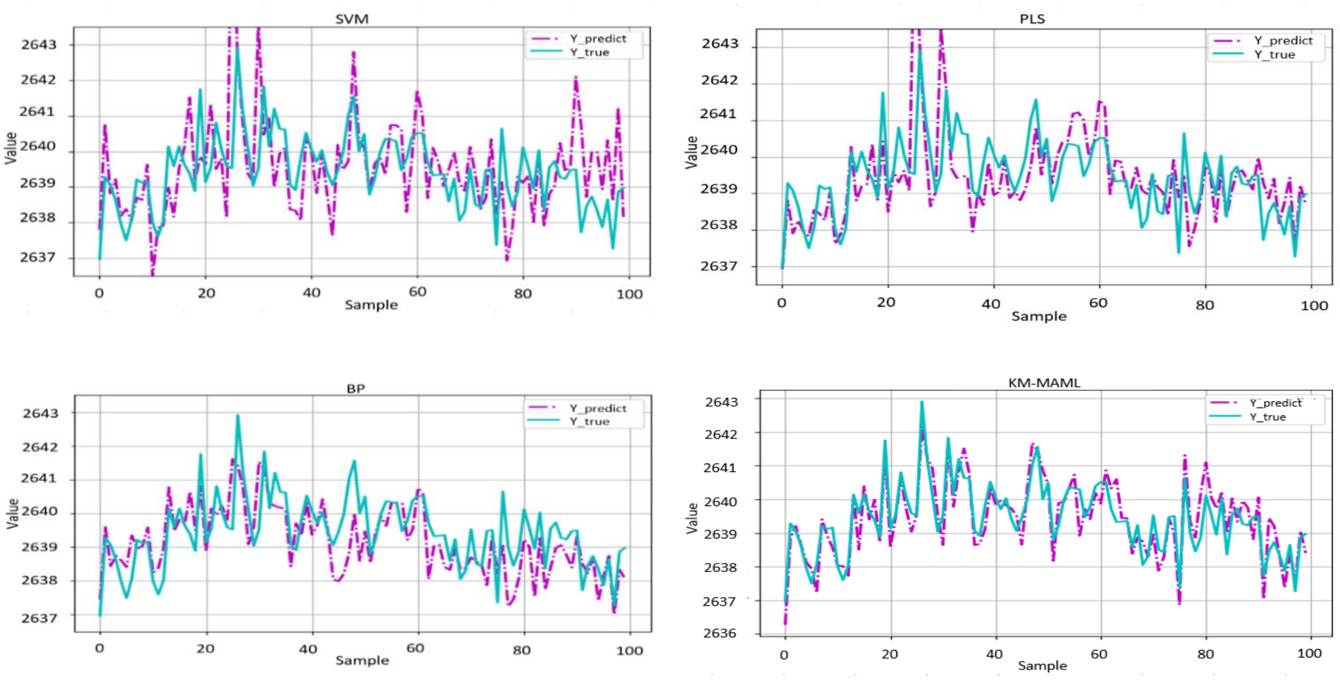

Figure 6. The predict curves of the SVM, the PLS, the BP, and the KM-MAML.

Therefore, the KM-MAML model parameters have better generalization ability. The modeling work can be performed faster and more conveniently by using the existing model parameters for training when changing system working conditions. The model has better generalization ability, and the accuracy of prediction is also significantly improved.

3. CONCLUSIONS

This paper introduced the K-Means and Meta-Learning technique as a novel data-driven soft sensor modeling method. Advantages and characteristics of the multimodal data fusion soft sensor modeling method were analyzed and applied. The experimental results of SVM, PLS, BP, and KM-MAML on actual production data of PTA show that the proposed model can achieve better performance. The KM-MAML can reduce the computational burden of processing data and improve prediction accuracy. So, the modeling process of this method has better performance, which is important for soft sensor modeling in real industrial processes. In the future, we will consider more complex scenarios, and explore the adaptive capability of the model more fully.

ACKNOWLEDGEMENT

This research was supported by the National Natural Science Foundation of China (No. 22178383, 21706282), and Research Foundation of China University of Petroleum (Beijing) (No. 2462020BJRC004, 2462020YXZZ023)

REFERENCES

Dong, D., McAvoy, T.J. (1996). Nonlinear principal component analysis—Based on principal curves and neural networks[J]. *Computers & Chemical Engineering*, 20(1), 65-78.

Gao, X.Q., Yang, F., Huang, D. X., Ding Y.J. (2014). An iterative two-level optimization method for the modeling of wiener structure nonlinear dynamic soft sensors[J]. *Industrial & Engineering Chemistry Research*, 53(3), 1172–1178.

Gonzaga, J.C.B., Meleiro, L.A.C. Kiang, C. (2009). ANN-based soft-sensor for real-time process monitoring and control of an industrial polymerization process[J]. *Computers and Chemical Engineering*, 2009, 33(1), 43-49.

Kaneko, H., Funatsu, K. (2014) Adaptive soft sensor based on online support vector regression and Bayesian ensemble learning for various states in chemical plants[J]. *Chemometrics and Intelligent Laboratory Systems*, 137, 57-66.

Kano, M. and Nakagawa Y. (2008). Data-based process monitoring, process control, and quality improvement: Recent developments and applications in steel industry[J]. *Computers & Chemical Engineering*, 32(1-2), 12-24.

Kadlec, P., Grbić, R., Gabrys, B. (2011). Review of adaptation mechanisms for data-driven soft sensors[J]. *Computers & Chemical Engineering*, 35(1), 1-24.

Ko, Y.D., Shang, H.(2011). A neural network-based soft sensor for particle size distribution using image analysis[J]. *Powder Technology*, 212(2), 359-366.

Lian, P.L., Liu, H., Wang, X., Guo, R.Y. (2020). Soft sensor based on DBN-IPSO-SVR approach for rotor thermal deformation prediction of rotary air-preheater[J]. *Measurement*, 165, 108109.

Lin, K.Z., Lei, J., Roeder, K.(2021). Exponential-Family Embedding With Application to Cell Developmental Trajectories for Single-Cell RNA-Seq Data[J]. *Journal of the American Statistical Association*, 116(534), 457-470.

Mohammadpoor, M., Torabi, T. (2018). A new soft computing-based approach to predict oil production rate for vapour extraction (VAPEX) process in heavy oil reservoirs[J]. *The Canadian Journal of Chemical Engineering*, 96(6), 1273-1283.

Popli, K., Maries, V., Afacan, A., Liu, Q., Prasad,V.（2018）. Development of a vision-based online soft sensor for oil sands flotation using support vector regression and its application in the dynamic monitoring of bitumen extraction[J]. *The Canadian Journal of Chemical Engineering*, 96(7), 1532-1540.

Park, S.Y., Han, C.H. (2000). A nonlinear soft sensor based on multivariate smoothing procedure for quality estimation in distillation columns[J]. *Computers & Chemical Engineering*, 24(2-7), 871-877.

Sun, Q. and Ge, Z. (2019). A Survey on Deep Learning for Data-Driven Soft Sensors[J]. *IEEE Transactions on Industrial Informatics*, 17(9), 5853-5866.

Sun, Q.Q. and Ge, Z.Q. (2019). Probabilistic sequential network for deep learning of complex process data and soft sensor application[J]. *IEEE Transactions on Industrial Informatics*, 15(5), 2700-2709.

Shang, C., Yang, F., Huang, D.X. (2014). Data-driven soft sensor development based on deep learning technique[J]. Journal of Process Control, 24(3), 223–233.

Wang, J.L., Qiu, K.P., Guo, Y.Q., Wang, R.T., Zhou, X.J. (2020). Soft sensor development based on improved just-in-time learning and relevant vector machine for batch processes[J]. *The Canadian Journal of Chemical Engineering*, 99(1),334-344.

Wang, K.C., Shang, C., Liu, L., Jiang, Y.H., Huang, D.H., Yang, F. (2019). Dynamic Soft Sensor Development Based on Convolutional Neural Networks[J]. *Industrial & Engineering Chemistry Research*, 58(26), 11521-11531.

Wen, H.X., Zhu, X.H., Zhang L.F., Li, F. (2020). A gated piecewise CNN with entity-aware enhancement for distantly supervised relation extraction[J]. *Information Processing & Management*, 57(6), 102373.

Xu, Z.X., Chen.X.L., Tang, W., Lai, J., Cao, L. (2021). Meta weight learning via model-agnostic meta-learning[J]. *Neurocomputing*, 432, 124-132.

Yuan, X.F., Huang, B., Wang, Y.L., Yang C.H., Gui, W.H. (2018). Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted sae[J]. *IEEE Transactions on Industrial Informatics*, 14(7), 3235-3243.

Yuan, X.F., Li, L., Wang, Y.L., Yang, C.H., Gui, W.H. (2019). Deep learning for quality prediction of nonlinear dynamic processes with variable attention-based long short-term memory network[J]. *The Canadian Journal of Chemical Engineering*, 98(6), 1377-1389

Yildirim, S., Kaya, Y., Kılıç, F. (2021). A modified feature selection method based on metaheuristic algorithms for speech emotion recognition[J]. *Applied Acoustics*, 173, 107721.

Ye, H.L., Wang, Y., Cao.F.L.(2021). A novel meta-learning framework: multi-features adaptive aggregation method with information enhancer[J]. *Neural Networks*, 144, 755-765.

Zhou, F., Cao, C.T., Zhong, T., Geng.J. (2021). Learning meta-knowledge for few-shot image emotion recognition[J]. *Expert Systems with Applications*, 168, 114274.

Zhu, Q.Y., Qin, A.K., Suganthan P.N., Huang, G.B. (2005). Evolutionary extreme learning machine[J], *Pattern Recognition*,38(10), 1759-1763.