Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

A Model Combining Seq2Seq Network and LightGBM Algorithm for Industrial Soft Sensor

Yanrui Li * **Chunjie Yang** * **Hanwen Zhang** * **Chao Jia** **

* *College of Control Science and Engineering, State Key Laboratory of Industrial Control Technology, Zhejiang University, Zheda Road 38, Hangzhou, China(e-mail: liyanrui@zju.edu.cn, cjyang999@zju.edu.cn, zhanghanwen@zju.edu.cn).*
** *Software Engineering and Appraisal Center Electronics Standardization Institute, Beijing, China (e-mail: jiachao@cesi.cn).*

Abstract: As a key technology for industry 4.0, data-driven soft sensing plays an important role in the control and optimization of industrial processes. However, due to the large-scale, nonlinear and dynamic characteristics of industrial data, it is difficult to process industrial data. To solve these difficulties, a soft sensor modeling method based on a sequence to sequence model and a gradient boosting tree algorithm is developed. In this method, an unsupervised trained Seq2Seq model is used to extract dynamic features at first. Then a high-precision model based on LightGBM is constructed with the dynamic features and the original features as inputs. The developed method is validated on pulping data and compared with other machine learning methods such as RNN and SVR. The result shows the developed method has a better performance.

Keywords: Soft sensing, dynamic feature, sequence to sequence, data integration, process industry

1. INTRODUCTION

In practical industrial processes, measuring quality variables and other key variables is important to ensure the quality of products. However, some key product qualities are difficult to measure because of technical or economic limitations. Therefore, soft sensors have been widely studied and implemented in the process industry over the past thirty years. There are three typical soft sensor modeling methods: mechanism modeling, knowledge-based modeling and data-driven modeling. With the rapid development of computer hardware and machine learning methods, data-driven soft sensors have become an important development direction of industry 4.0 for its excellent ability in large-scale data processing and high precision modeling .

Data plays a key role in data-driven soft sensor. However, industrial data processing faces many difficulties:

1) Large scale:

Industrial processes often involve complex reactions and numerous processes. The equipment characteristics, external working conditions, process formulation and even material parameters of each process are closely related to the final product quality, which means industrial data have high dimension. Industrial data have diverse sampling intervals with large differences in duration. For example, the sampling rate of vibration and current may be in milliseconds whereas some other quality variables may be in hours. For these reasons, industrial data are much larger than their effective scale, which is termed as data rich but information poor.

2) Poor quality

Industrial data have poor quality, which reflect in three aspects. Data invalidation (inaccuracy): happens when sensors stop working properly. Data formats chaos: manual labeled data and automatically collected data are mixed and missing values appear from time to time. Uneven distribution of data: the proportion of anomaly data is very small.(Zhang et al., 2017)

3) Dynamicity

Industrial data are generally time series data which contain dynamic characteristics of industrial processes.

Many methods have been applied to solve these problems. As the most popular multivariate statistical approaches, principal component analysis (PCA) and partial least squares (PLS) are used to reduce data dimensionality (Geladi and Kowalski, 1986; Zheng and Qian, 2006), but they can not deal with time series data. So, dynamic PCA and dynamic PLS were proposed to handle dynamic characteristic (Perera et al., 2006). In order to deal with nonlinearity, many kernel-based algorithms, such as support vector regression (SVR), kernel-driven fisher discriminant analysis (KFDA) and kernel principal component analysis (KPCA) were proposed (Jain et al., 2007; Lee et al., 2004; Ge et al., 2016). However, in recent years, two methods have been proven to have better performance on many

⋆ This work is supported by the National Science Foundation of China (61933015 and 61903326)

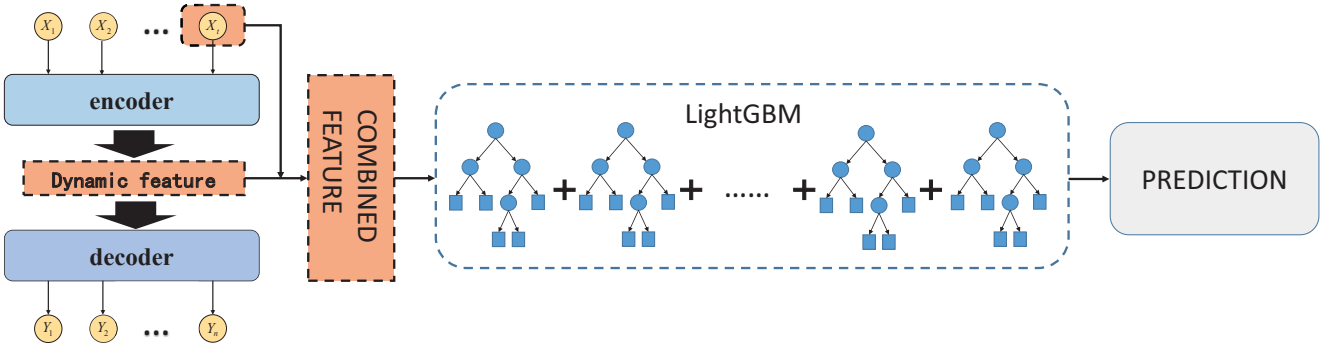

Fig. 1. The framework of the proposed method.

tasks: recurrent neural network (RNN) for dynamic issues and gradient boosting decision tree (GBDT) for large-scale problems. it is a good idea to apply these methods to an industrial data process.

GBDT is an ensemble machine learning method for regression and classification problems, which ensembles decision trees as weak prediction models to generate a strong prediction model. Models are built in a stage-wise fashion and generalized by the optimization of an arbitrary differentiable loss function. The concept of gradient boosting was first proposed by (Friedman, 2001), and developed by (Friedman, 2002). Then, LightGBM, which used a variety of engineering techniques to speed up the construction of decision trees without reducing the prediction accuracy, was proposed by (Ke et al., 2017).

RNN have made a breakthrough in the field of natural language processing (NLP), which shows its ability in dealing with time series (Lu and Tsai, 2008). However, conventional RNN requires that the dimension of inputs and outputs is known and fixed, which is inconvenient in many NLP tasks. So, RNN encoder-decoder, also called sequence to sequence (Seq2Seq), modeling was proposed (Sutskever et al., 2014; Cho et al., 2014). The main idea of Seq2Seq is using an RNN to map an input sequence into a vector and using another RNN to map the vector into a variable-length output. Because the decoder only uses the vector as input, the vector contains dynamic information of whole input sequence and even the future tendency.

GBDT has great performance in dealing with large-scale data whereas RNN model especially Seq2Seq model is good at processing time series data. In order to complement the advantages of each, many scholars have combined GBDT, RNN and other methods together to deal with dynamic and nonlinear data. For example, Sun Qinqiang proposes probabilistic sequential network (PSN) based on Gaussian-Bernoulli Restricted Boltzmann Machine (GRBM) and the recurrent neural network (RNN) structure (Qingqiang and Zhiqiang, 2018). Zhu J proposes Deep Embedding Forest by integrating the deep neural network with GBDT (Zhu et al., 2017). Yun Ju combines convolutional neural network and LightGBM to forecast ultra-short-term wind power (Ju et al., 2019). In this paper, we construct a model based on Seq2Seq and LightGBM for the purpose of soft sensor application. The modeling algorithm consists of two parts: unsupervised dynamic feature extraction and supervised modeling for prediction. It is suitable for industrial data and has a good expansibility, which can process both time series and categorical features. And it is only suited for soft sensors but also for fault diagnosis in industrial processes.

The layout of this paper is given as follows. The modeling method is explained in detail in section 2. In section 3, the effectiveness and feasibility of the proposed method is validated on a real industrial soft sensor application. Finally, conclusions are made.

2. PROPOSED MODELING METHOD

This paragraph include sections. First, we introduce the structure of our proposed model, secondly, we present its training method.

2.1 Structure of the proposed model

In this paper, we construct a model combined with sequence to sequence network and LightGBM algorithm. Fig 1 shows the structure of the modeling method. First, at each time step, the input sequence data are serialized and put into the trained sequence to sequence model. Then, the sequence to sequence model extracts dynamic features and combines it to the original features at every timestep. Finally, a prediction model, which is constructed using LightGBM makes a prediction according to the integrated features. In summary, the model consists of two parts: an unsupervised trained Seq2Seq model to extract dynamic features and LightGBM to make predictions.

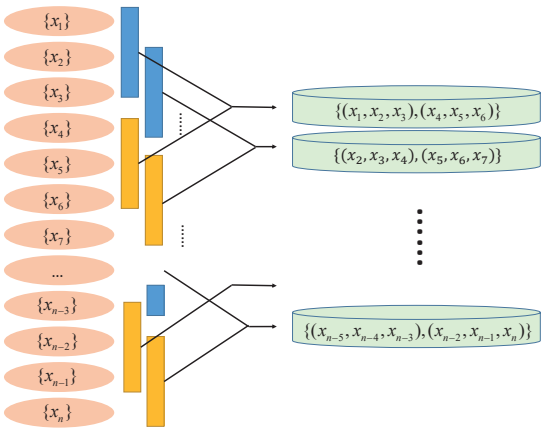

Fig. 2. Procedure of serialization with length t=3.

2.2 Model training method

Serializing the original data are first step for Seq2Seq model training. Define the original data as $D = \{X, Y\} = \{(x_i, y_i)\}$, $i \in 1, 2, ..., n$ where x_i and y_i denote the process variables and quality variables. Use a time window to scan the data and obtain the serialized data $D^s = \{X_j^s, Y_j^s\} = \{(x_j^s, y_j^s)\}$. In this paper we trained Seq2Seq in an unsupervised way, $x_j^s = (x_j, x_{j+1}, ..., x_{j+t-1})$ and $y_j^s = (x_{j+t}, x_{j+t+1}, ..., x_{j+2t-1})$ where t is the time window width. Fig. 2 shows the procedure of serialization with length t=3. Time window width decides the sequence length and influences the performance of the model which will be discussed further in the later part.

After the data are serialized, the Seq2Seq model is trained in an unsupervised way as Fig 3 shown. By encoder and decoder processing, the final output sequence $\hat{Y}_j$ is generated. Then Y_j^s and $\hat{Y}_j$ are used to compute the sequence loss and train the model through back propagation. The training of Seq2Seq model only requires data without labels.

After that, dynamic features are extracted for every time step based on the pre-trained Seq2Seq model. Finally, an ensemble decision tree model is trained to predict the quality variables. The dataset used to train the Seq2Seq model and LightGBM is different to avoid overfitting. The flowchart of the proposed procedures for the soft sensor modeling methods is displayed in Fig 4.

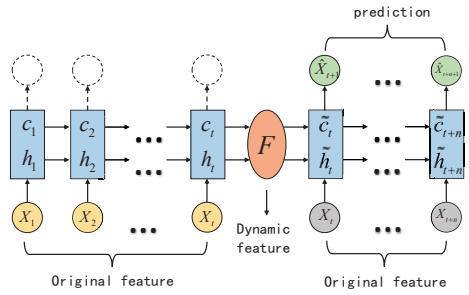

Fig. 3. Sequence to sequence framework.

3. CASE STUDY

The proposed method is validated on a real industrial dataset, a froth flotation process.

3.1 Froth flotation process

Froth flotation is a process of selectively separating hydrophobic materials from hydrophilic materials, and used in the mining industry to obtain high grade concentrate. In this process, the plant uses froth flotation to get high grade iron ore for the iron smelting industry. Silicon content is an important reference index in the ironmaking process and closely related to the iron ore. Under stable conditions, in order to keep the ironmaking process stable, control of the silicon content between 0.4% and 0.6% is necessary (see (Zhou et al., 2017)).When silicon content is less than 0.4%, the furnace temperature is generally lower than 1500 °C, and the ironmaking environment can not be achieved; when silicon content is higher than 0.6%, the temperature is too high and much energy is wasted((Zhou et al., 2019)). So, for an appropriate silicon content in ironmaking process, the ore silicon content needs to be appropriate and fluctuations of it needs to be as smooth as possible. However, silicon concentration is obtained through lab measurement, which means that it takes at least two hours for the process engineers to have this value. Two hours is too long for real-time closed-loop control. Therefore, we need to predict silicon concentration, and building a soft sensor is meaningful.

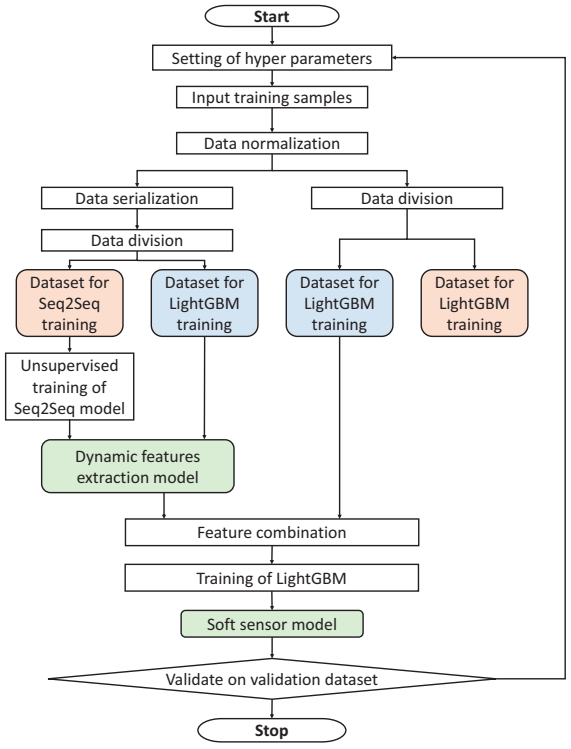

Fig. 4. Complete training process of the proposed model.

The flotation column diagram and its explanation are shown in Fig 5. In this plant, there are several flotation columns placed in series or parallel to extract concentrates from pulp as much as possible. In this process there are total 21 process variables and 1 quality variables, the detailed descriptions of these process variables are listed in Table 1.

3.2 Experiment

There are total of 730000 samples collected from a distributed control sysytem (DCS) with a sampling interval of 20 seconds in this dataset [1]. The iron feed and silicon concentrate are analyzed every 2 hours in a laboratory; other variables are measured through sensors. For the convenience of comparing to other methods, we downsample the data with interval of 5 minutes and get 32000 training samples, 8000 validation samples and 8647 samples for testing; this partitioning ensure that the test dataset is completely unseen for model.

[1] The dataset is downloaded from https://www.kaggle.com/edumagalhaes/quality-prediction-in-a-mining-process

Table 1. Process variables for soft sensor.

No	Tag	Explanation
1	Iron Feed	% of Iron that comes from the iron ore that is being fed into the flotation cells
2	Silica Feed	% of silica (impurity) that comes from the iron ore that is being fed into the flotation cells
3	Starch Flow	Starch (reagent) flow measured in m^3/h
4	Amina Flow	Amina (reagent) flow measured in m^3/h
5	Ore Pulp Flow	t/h
6	Ore Pulp pH	pH scale from 0 to 14
7	Ore Pulp Density	Density scale from 1 to 3 kg/m^3
8-14	Flotation Column 01-07 Air Flow	Air flow that goes into the flotation cell 01-07 measured in Nm^3/h
15-21	Flotation Column 01-07 Level	Froth level in the flotation cell 01-07 measured in mm (millimeters)
22	Silicon Concentrate[1]	% of silicon in the end of the flotation process (generally 0-6%, lab measurement)

[1] Note that the silica content demand of the froth floatation process is different from the content demand in iron making process. In this dataset, the sillicon conten fluctuates between 0% and 6%.

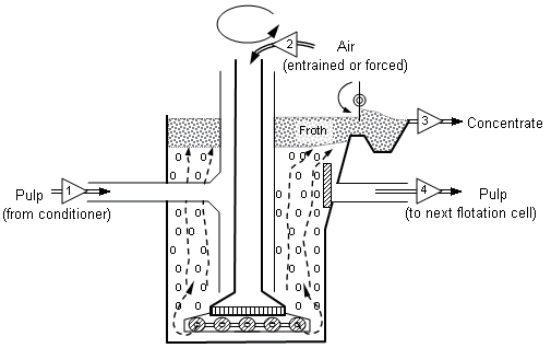

Fig. 5. Flotation column diagram, where numbered triangles show direction of stream flow. Pulp which is a mixture of water and ore enter to the bottom of the column. A vertical impeller passes down air and breaks the air stream into small bubbles by shearing forces. Then the mineral concentrate froth is collected from the top of cell and the pulp flows to the next flotation column.

To better show the proposed model's performance, comparison will be done among four different algorithms. In the deep learning field an RNN model is used, in time series process field an SVR model is used, besides, two LightGBM models with and without dynamic features are constructed respectively.

The hyper parameters to train different models are shown in Table 2; these parameters are the best results we get from multiple experiments. SVR takes the Radial Basis Function (RBF) as its kernel function. For LightGBM+Seq2Seq algorithm, the normalized data is serialized by a sliding window of width 20 to obtain a sequence input. Because the Seq2Seq model is only used to extract features, the encoder and decoder RNN are set two layers with 5 hidden units each. The main parameters in LightGBM are "learning rate", "feature fraction" and "bagging fraction", we set these parameters manually by choosing the best results from multiple experiments too.

The predicted and real values of four algorithms for the testing data are displayed in Fig 6 and the prediction error is shown in Fig 7. Four numerical evaluation indices are calculated in Table 3 include coefficient of determination (R^2) which is closer to 1 the better, mean squared error (MSE), mean absolute deviation (MAD) and median absolute deviation (MdAD) which has small impact of outliers.

Table 2. Hyper parameters of four methods.

SVR	C	Epsilon	Gamma
	100	0.1	0.1
LSTM	Layers	Units	Learning rate
	3	64	0.01
LightGBM	Bagging fraction	Feature fraction	Learning rate
	0.78	0.64	0.05
seq2Seq + LightGbm	Bagging fraction	Feature fraction	Learning rate
	0.75	0.45	0.05

Table 3. Evalution indices of four methods.

indices	SEQLGB	LGB	LSTM	SVR
R^2	**0.6981**	0.6504	0.2982	0.5379
MSE	**0.4090**	0.4737	0.9510	0.6262
MAD	**0.5099**	0.5535	0.7782	0.6099
MdAD	**0.4267**	0.4736	0.6572	0.4909

3.3 Discussion

Several conclusions are draw from the above results: 1) In this case, the small size RNN has poor performance. 2) The low MSE of LightGBM indicates that LightGBM has the ability of high precision modeling. 3) After integrating dynamic features extracted by a Seq2Seq model, the LightGBM model has better performance on silicon content prediction, which proves that this algorithm does combine the advantages of Seq2Seq and LightGBM, and lets the LightGBM lets the LightGBM learn dynamic characteristic and construct a better model.

The time window width influences the performance of model. In this case, we set the time width to be 20, based on the model results with different time window widths shown in Table 4. However, this approach is not the best way to utilize the RNN model, because RNN only extracts information in a short time period instead of aggregate information ovwe the whole time frame. If we do not

Fig. 6. Predicted results of LSTM,SVR,LightGBM and proposed algorithm.

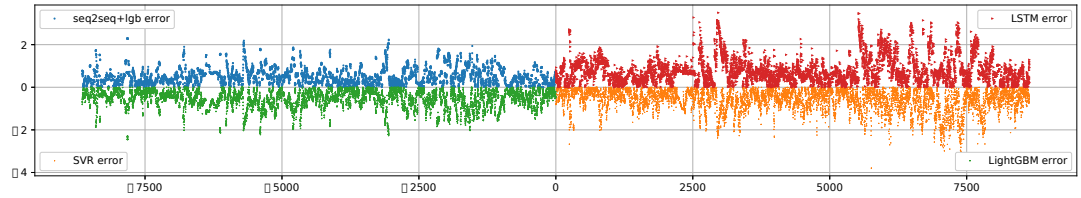

Fig. 7. Prediction error by LSTM,SVR,LightGBM and proposed algorithm.

use the time window, the training and predicting process will be much easier because we do not need to serialize the input data but only update the RNN unit when new samples come in. As Table 4 shows, we try to build the model without the time window(time window width is ∞), although the MSE is not the best, it has an advantage in computing time.

Table 4. Influence of different time window width.

time window width	MSE	feature construction time
1[1]	0.4563	21s
3	0.4221	34s
10	0.4112	50s
20	**0.4090**	87s
50	0.4196	210s
100	0.4214	527s
∞[2]	0.4219	**17s**

[1] The time window width is zero means the Seq2Seq only use the present data to train.
[2] The time window width is infinity means the Seq2Seq model uses all the history data to extract the features.

Another parameter that affects the results is the number of dynamic features. Through simple experiments (results shown in Fig. 8), we draw the conclusion that the appropriate number of dynamic features is 20 in this case.

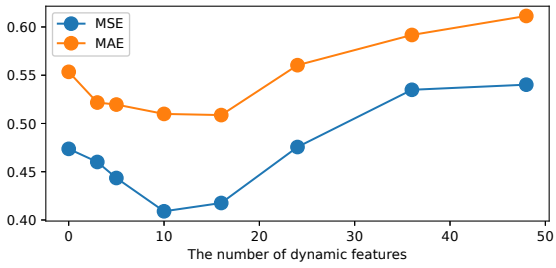

Fig. 8. The influence of changing dynamic feature numbers.

4. CONCLUSION

In this paper, taking the difficulties of industrial data into consideration, a framework using deep learning to extract features and LightGBM for modeling is constructed for the purpose of a soft sensor application. Taking advantage of the unsupervised dynamic feature extraction ability of Seq2Seq and large-scale data processing ability of LightGBM, the proposed algorithm have a good performance which have the MSE less than 0.5. So, it is reasonable to believe this method can provide accurate and reliable prediction of sillic content for field engineers to help them control the process.

ACKNOWLEDGEMENTS

This work is supported by the National Science Foundation of China (61933015 and 61903326).

REFERENCES

Cho, K., Van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *Computer Science*.

Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189–1232.

Friedman, J.H. (2002). Stochastic gradient boosting. *Computational statistics and data analysis*, 38(4), 367–378.

Ge, Z., Zhong, S., and Zhang, Y. (2016). Semisupervised kernel learning for fda model and its application for fault classification in industrial processes. *IEEE Transactions on Industrial Informatics*, 12(4), 1403–1411.

Geladi, P. and Kowalski, B.R. (1986). Partial least-squares regression: a tutorial. *Analytica chimica acta*, 185, 1–17.

Jain, P., Rahman, I., and Kulkarni, B. (2007). Development of a soft sensor for a batch distillation column using support vector regression techniques. *Chemical Engineering Research and Design*, 85(2), 283–287.

Ju, Y., Sun, G., Chen, Q., Zhang, M., Zhu, H., and Rehman, M.U. (2019). A model combining convolutional neural network and lightgbm algorithm for ultra-short-term wind power forecasting. *IEEE Access*, 7, 28309–28318. doi:10.1109/ACCESS.2019.2901920.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in neural information processing systems*, 3146–3154.

Lee, J.M., Yoo, C., Choi, S.W., Vanrolleghem, P.A., and Lee, I.B. (2004). Nonlinear process monitoring using kernel principal component analysis. *Chemical engineering science*, 59(1), 223–234.

Lu, C.H. and Tsai, C.C. (2008). Adaptive predictive control with recurrent neural network for industrial processes: An application to temperature control of a variable-frequency oil-cooling machine. *IEEE Transactions on Industrial Electronics*, 55(3), 1366–1375.

Perera, A., Papamichail, N., Bârsan, N., Weimar, U., and Marco, S. (2006). On-line novelty detection by recursive dynamic principal component analysis and gas sensor arrays under drift conditions. *IEEE Sensors Journal*, 6(3), 770–783.

Qingqiang, S. and Zhiqiang, G. (2018). Probabilistic sequential network for deep learning of complex process data and soft sensor application. *IEEE Transactions on Industrial Informatics*.

Sutskever, I., Vinyals, O., and Le, Q.V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, 3104–3112.

Zhang, H., Chen, M., Xi, X., and Zhou, D. (2017). Remaining useful life prediction for degradation processes with long-range dependence. *IEEE Transactions on Reliability*, 66(4), 1368–1379.

Zheng, X.X. and Qian, F. (2006). Soft sensor modeling based on pca and support vector machines. *Journal of System Simulation*, 18(3), 739–741.

Zhou, H., Yang, C., Liu, W., and Zhuang, T. (2017). A sliding-window ts fuzzy neural network model for prediction of silicon content in hot metal. *IFAC-PapersOnLine*, 50(1), 14988–14991.

Zhou, H., Zhang, H., and Yang, C. (2019). Hybrid-model-based intelligent optimization of ironmaking process. *IEEE Transactions on Industrial Electronics*, 67(3), 2469–2479.

Zhu, J., Shan, Y., Mao, J., Yu, D., Rahmanian, H., and Zhang, Y. (2017). Deep embedding forest: Forest-based serving with deep embedding features. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1703–1711.