Self-Attention-Based Convolutional Parallel Network: An Efficient Multi-Input Deep Learning Model for Endpoint Prediction of High-Carbon BOF Steelmaking

TIAN-YI XIE [ID], FEI ZHANG, YI-REN LI, QUAN ZHANG, YAN-WEI WANG, and HAO SHANG

In this work, a data-driven model for endpoint prediction of basic oxygen furnace (BOF) steelmaking based on both tabular features (information about hot metal, scrap, additives, blowing practices) and time series (curves of off-gas profiles, sonar slagging, and blowing practices) was developed and implemented. The model was designed with the following distinctive artificial intelligence (AI) characteristics: convolutional neural networks, patching embedding, wavelet decomposition, a parallel structure, a self-attention mechanism, a collaborative attention mechanism, and so on. The model presented in this work is named the self-attention-based convolutional parallel network (SabCP) and was applied to high-carbon steelmaking scenarios. SabCP predicts the endpoint of molten steel temperature (Temp) and chemistry (contents of carbon (C), phosphorus (P), and sulfur (S)). For training, validation, and testing, historical data from 13,656 heats were collected. The testing results show that the mean absolute errors (MAEs) of SabCP for temperature and the contents of carbon, phosphorus, and sulfur are 6.374 °C, 7.192×10^{-3}, 2.390×10^{-3}, and 2.224×10^{-3} pct, respectively, while the mean square errors (MSEs) are 67.345, 1.132×10^{-4}, 1.306×10^{-5}, and 1.298×10^{-5}, respectively, which are lower than those of other published models with same dataset. Relevant importance analyses for tabular features, time series time steps, and channels are also performed. SabCP has been implemented in a prediction module, and the practical results show its strong robustness and generalizability. This model provides significant feasibility for fully eliminating the conventional physical temperature, sampling, and oxygen test (TSO test), which may greatly decrease the cost of BOF steelmaking.

I. INTRODUCTION

ENDPOINT control for making high-carbon steel through the BOF steelmaking process is challenging; this is because within the carbon content range of 0.08~0.35 pct, both the decarbonization rate and slag condition rapidly change and become unstable during oxygen blowing. There are two conventional methods for ensuring that the endpoint temperature and chemistry are within the specified range. Decarbonization occurs at a much lower level than that of the spec, and then, re-carbonization practices are conducted after tapping and lab chemistry analysis. The second method is to perform a temperature, sampling, and oxygen test (TSO test) before tapping. Both methods are very costly and time-consuming. If an accurate endpoint prediction model can be developed, significant benefits will be brought to BOF steelmaking, especially in high-carbon scenarios. However, it is a complex, nonlinear, and vigorous process involving both physical and chemical transformations. Therefore, AI models designed for analyzing complex nonlinear systems are highly suitable for modeling and predicting the steelmaking process.

TIAN-YI XIE is with the Material Technology Research Institute, Hesteel Group, Shijiazhuang City 050023, P.R. China and also with Institute of Engineering Technology, University of Science and Technology Beijing, Beijing 102206, P.R. China. Contact e-mail: txie1516@gmail.com FEI ZHANG is with the Material Technology Research Institute, Hesteel Group and also with the Metallurgical Technology Research Institute, Central Iron and Steel Research Institute Co., Ltd., Beijing 100081, P.R. China. YI-REN LI is with the Hesteel Group, Shijiazhuang City 050023, P.R. China. QUAN ZHANG is with the Tangsteel Company, Hesteel Group, Tangshan City 063611, P.R. China. YAN-WEI WANG and HAO SHANG are with the Digital Technology Company, Hesteel Group, Tangshan City 063611, P.R. China.

Multiple studies have already applied AI models to various aspects of steelmaking. For example, Wang et al.[1] and Zhang et al.[2] analyzed additive amounts for BOF steelmaking with various machine learning models. Xin et al.[3] and Feng et al.[4] predicted the endpoint temperature of a ladle furnace (LF) with a modified deep neural network and Bayesian belief network. In regard to direct reduced iron (DRI) and electric arc furnace (EAF) steelmaking, Son et al.[5] studied the slag foaming estimation of an EAF with a long short memory network (LSTM), and Devlin et al.[6] utilized machine learning models to analyze the optimal locations for renewable energy-based steel production. At the microscopic level, particularly concerning impurities, there are also many examples. Abdulsalam et al.[7] applied unsupervised machine learning models for the detection of nonmetallic inclusion clusters. A survey conducted by Webler et al.[8] highlighted a method proposed by Alatarvas et al.[9] for automatic classification of inclusions via logistic regression.

There are multiple existing studies on the endpoint prediction of BOF steelmaking. They mainly adopt the following technical approaches: (1) using an original or modified backpropagation (BP) neural network with tabular data that consist of information derived from hot metal, scrap, additive, and blowing practices, such as in References 10 through 20; (2) using an original or modified support vector machine (SVM) with tabular data such as in References 18,21 through 25; (3) using SVM or convolution neural networks with flame features (radiation, temperature, and image features) to make predictions, such as in References 26 through 28] and (4) using a case-based reasoning (CBR) model with tabular data for predictions, such as in References 29 through 31.

The above studies have indicated the significant potential of AI data-driven models for solving problems related to steelmaking and have achieved good results. However, they might be improved by overcoming the following limitations:

(1) Inefficient models adopted. Some research has used classical machine learning models (SVM, random forest, etc.) and neural networks (LSTM, BP, etc.). According to research and surveys,[32–37] such models have less robustness and predictive capacity than the state-of-the-art (SOTA) machine learning and deep learning models. The SOTA models should be tested and compared.
(2) Insufficient modifications in the backbones. Some of the deep-learning-based studies introduced new data preprocessing and loss function optimization algorithms. However, they did not modify the existing feature extraction networks (backbones). Some widely proven effective backbone improving techniques, such as attention mechanisms and residual connections, have not been tried and implemented.
(3) There is a lack of ablation experiments. Some studies have proposed new algorithms for data preprocessing and loss function optimization. However, ablation experiments were not conducted.

Whether the added algorithms are useful and the specific parts contributing to the improvement of the model have not been determined.
(4) Simple features. Their input features are relatively simple, such as tabular features or flame images, which can only represent a part of the period of steelmaking. Moreover, the cases selected in such studies are very limited.

In the present work, a method has been proposed to achieve endpoint prediction of the BOF steelmaking and compensate for the shortcomings of the aforementioned research. The originality of this method stems from (1) the design of a multi-input deep learning algorithm, the reliability of which was validated by achieving the challenging high-carbon BOF steelmaking endpoint prediction; (2) the introduction of the hybrid input and the time series data that can fully represent the entire steelmaking process for BOF endpoint prediction; (3) the analysis of the importance of each channel and time step of the time series for the endpoint with self-attention-based networks; and (4) extensive experimentation, ample data acquisition, and comprehensive analysis are conducted.

The importance of the proposed prediction method is (1) demonstrating the applicability of time series and hybrid input for endpoint prediction, thereby designing a data-driven approach to predicting; (2) achieving endpoint prediction through the utilization of efficient deep learning algorithms and hybrid inputs; and (3) laying the technical groundwork for eliminating the TSO test, carbon catching, endpoint automation control, and future unsupervised or self-supervised pre-training model development. The advantages of this method lie in its high backbone efficiency, utilization of data with high information density, small storage footprint, and ease of preprocessing, making it highly suitable for widespread adoption. For steel plants lacking robust statistical digital infrastructure, endpoint prediction can be achieved with the numerical data automatically recorded by their coal gas recovery and level-two systems. A more detailed description of the specific contents of this study is as follows:

(i) For the first time, the challenging high-carbon endpoint prediction of BOF steelmaking was implemented using composite inputs, including tabular data and time series data. Good results are achieved.
(ii) A matching deep learning model with inputs of both time series and tabular features was proposed. It is based on a self-attention mechanism and a convolutional neural network. At the same time, the patching method and wavelet decomposition are used to process the input time series. Ablation experiments were carried out, and the performance of the proposed model was compared with that of modified SOTA models.
(iii) A strong correlation was found between the BOF endpoint and the time series, including the off-gas profile and blowing practice curves. This demonstrated that the endpoint can be relatively accurately predicted with only time series.

(iv) A BOF endpoint prediction module based on SabCP was programmed and applied to external validation. The results show that SabCP has good prediction accuracy and robustness. Its performance and human–machine interface (HMI) and results are listed in detail.

(v) The data have been carefully analyzed. For tabular data, data cleaning was conducted, followed by statistical analysis. The filtered data were utilized with machine learning models to predict the endpoint and analyze the importance of the features. For time series data, data cleaning has also been applied, and a transformer model has been utilized to analyze the importance of each time step and channel for the target. Ultimately, various deep learning models have been employed to predict endpoints using time series data.

II. DATA DESCRIPTION

The raw history data were collected from 13,656 heats. After data cleaning and feature engineering, 8328 heats remained. A total of 5328 heats of the remaining data for each target were randomly divided into the training set, 1500 heats into the validation set, and 1500 heats into the test set. The main steel grades considered in this study are those requiring high-carbon endpoints (≥0.08 pct), including HRB400, HRB500, Q355, and Q420. To ensure the generalization capability and robustness of the model, some steel grades with lower required endpoint carbon contents (0.04–0.06 pct) produced in this work were also considered, such as ER70, TH10Mn2, and H08.

A. Tabular Data

Columns in tabular data determine features, and rows determine values of each heat in such features. Tabular features include information of preset value, hot metal, scarp, additives and blowing practices. Tabular data are divided into three parts, and their statistical descriptions are detailed in Table I (std: standard deviation). The description of each part is as follows:

- Preset data: preset values of oxygen blowing practices and endpoints. The preset oxygen blowing before the Temperature, Sample, Carbon (TSC), and TSO tests come from the second calculation of the level-two system. That is, after determining the relevant data of the loaded hot metal and scrap, the level-two system calculates the oxygen blowing amount more accurately.
- Static data: physical or chemical qualities of hot metal, scrap, and additives (additive data before and after the TSC test were separately input into the model).
- Dynamic data: dynamic oxygen blowing practices and TSC-related values.

Feature engineering and importance analysis of tabular data. The importance of each feature is the average of the importance analysis results of the light gradient boosting machine (LightGBM)[38] and categorical boosting (Catboost)[39] algorithms. For LightGBM, the importance of features was defined by the total number of splits and the total/average information gain during the training process. The importance of a feature is greater when it is chosen more often as a split node. For Catboost, the loss function change[23] method is adopted for importance analysis. It refers to counting the change in the model's loss that contains and does not contain a certain feature. The calculation process is described in Eqs. [1] and [2],[23] where $E_i v$ is the mathematical expectation of the formula value without the i th feature, the metric is the loss function specified in the training parameters, and v is the vector with formula values for the dataset.

$$importance_i = \pm(metric(E_i v) - metric(v)) \qquad [1]$$

$$importance_i = abs(metric(E_i v) - best\_value) \\ - abs(metric(v) - best\_value). \qquad [2]$$

The results of the feature importance analysis for different tasks are visualized in the Results and Discussion section.

B. Time Series Data

A time series is a sequence composed of times and their corresponding observed values over a continuous period. The sampling frequency of the time series is once per second. The time series data in this work consisted of the following curves:

(a) Curves of the off-gas profile, which directly represent the carbon–oxygen reaction. The off-gas profile is a curve consisting of the real-time content of the main gases in the exhaust gas of steelmaking. These gases include carbon monoxide, carbon dioxide, nitrogen, oxygen, and hydrogen, and curves of their gas content percentage (GP) in off gas are abbreviated as GP-CO, GP-CO_2, GP-N_2, GP-O_2, and GP-H_2 in this work. The gas flow curve (denoted as Gas-Flow) is also included in the off-gas profile.

(b) The curve of sonar slagging directly represents the slag condition. Sonar slag is a method of measuring slag condition by sound intensity. The curve shows the real-time slag condition. In this work, it is denoted as Sonar.

(c) The curves directly represent the blowing practices. These curves include the total oxygen blowing curve (denoted as O_2-Blown), the instantaneous oxygen blowing intensity curve (denoted as O_2-Blowing), and the oxygen lance height curve (denoted as Lc-height). The curves of blowing practice can comprehensively summarize the oxygen blowing conditions of each heat.

Table I. Descriptive Statistics of the Tabular Features

Class	Features	Units	Max	Min	Mean	Std
Preset Values	set oxygen consumption before TSC test	Nm^3	5652.000	3715.000	4503.892	206.627
	set oxygen consumption before TSO test	Nm^3	6563.000	4508.000	5011.015	272.456
	target steel temperature	°C	1680.000	1600.000	1644.488	12.024
	target [C]	pct	0.150	0.040	0.074	0.013
Static Values	hotmetal [C]	pct	4.999	3.589	4.432	0.194
	hotmetal [Si]	pct	0.742	0.024	0.300	0.085
	hotmetal [Mn]	pct	0.367	0.084	0.149	0.029
	hotmetal [S]	pct	0.346	0.001	0.033	0.010
	hotmetal [P]	pct	0.159	0.067	0.117	0.010
	hotmetal [V]	pct	0.069	0.018	0.036	0.008
	hotmetal [Ti]	pct	0.160	0.005	0.071	0.027
	hotmetal weight	kg	142800.000	102000.000	123906.168	2108.320
	hotmetal temperature	°C	1448.000	1250.000	1354.521	38.508
	self-produced script weight	kg	20000.000	0.000	4239.091	2670.897
	iron lump weight	kg	12369.000	0.000	141.823	816.252
	hot pressed iron weight	kg	9300.000	0.000	62.207	405.943
	slag–steel weight	kg	8800.000	0.000	973.260	1242.862
	heavy scrap weight	kg	10200.000	0.000	134.452	727.892
	other scrap weight	kg	6300.000	0.000	1.002	79.455
	lime weight (before TSC)	kg	5808.000	500.000	3286.633	472.200
	dolomite weight (before TSC)	kg	3624.000	148.000	2105.257	387.073
	limestone weight (before TSC)	kg	3498.000	0.000	362.646	558.044
	raw dolomite weight (before TSC)	kg	3519.000	0.000	696.000	504.905
	inhibitor weight (before TSC)	kg	567.000	0.000	8.468	40.519
	ferrous cold material weight (before TSC)	kg	10858.000	0.000	3243.380	1692.903
	other material weight (before TSC)	kg	9808.000	0.000	394.985	1199.756
	lime weight (after TSC)	kg	505.000	0.000	1.242	22.116
	dolomite weight (after TSC)	kg	579.000	0.000	5.805	46.635
	limestone weight (after TSC)	kg	1091.000	0.000	7.183	52.831
	raw dolomite weight (after TSC)	kg	1026.000	0.000	4.575	42.635
	inhibitor weight (after TSC)	kg	293.000	0.000	0.609	10.261
	ferrous cold material weight (after TSC)	kg	5583.000	0.000	411.945	439.155
	other material weight (after TSC)	kg	4904.000	0.000	48.178	215.990
Dynamic Values	oxygen consumption before TSC test	Nm^3	5510.000	3972.000	4585.501	188.145
	oxygen consumption before TSO test	Nm^3	6134.000	4388.000	5093.427	215.333
	TSC thermal arrest [C]	pct	0.760	0.100	0.398	0.153
	TSC temperature	°C	1650.000	1510.100	1605.652	22.461

Data preprocessing: For data cleaning (the cleaning standards are detailed in the Appendix), all time series are padded to 1024 time steps with zero. An example of an time series in one typical heat is shown in Figure 1

The importance of each time step and channel is calculated by a transformer encoder.[40]

The weight of the multihead attention layer is the importance. The structure of the transformer encoder is shown in Figure 2. The results are detailed in the Results and Discussion section.

C. Targets and Metrics

The temperature of the hot steel (TSO-Temp (°C)), the carbon content of the hot steel (TSO-C (pct)), the phosphorus content of the hot steel (TSO-P (pct)), and the sulfur content of the hot steel (TSO-S (pct)), were measured via the TSO test. Figure 3 shows the distribution and relationship between each target. Combinatorial metrics were used to evaluate the model. The basic metrics adopted were as follows:

$$R^2 = 1 - \frac{\sum \left(Y_i - \widehat{Y}\right)^2}{\sum \left(Y_i - \overline{Y}\right)^2} \quad \text{and} \quad \text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|Y_i - \widehat{Y}\right|,$$

where R^2 is the coefficient of determination; MAE is the mean absolute error; Y_i is the true value with index i in dataset Y; $\widehat{Y}$ is the predicted value of Y_i; $\overline{Y}$ is the average of Y; and n is the number of samples contained in Y.

To better evaluate the manufacturability of the model, the BOF steelmaking quality control requirements and key process indices (KPIs) of the BOF steelmaking plant were used. The following hit rate criteria are added.

- For TSO-Temp, the proportion of the predicted value within ±10°C, ±15°C, and 20°C of the true value.
- For TSO-C, the proportion of the predicted value within ±0.01 pct, ±0.015 pct, and ±0.02 pct of the true value.
- For TSO-P and TSO-S, the proportion of the predicted value within ±0.001pct, ±0.003pct, and ±0.005pct of the true value.
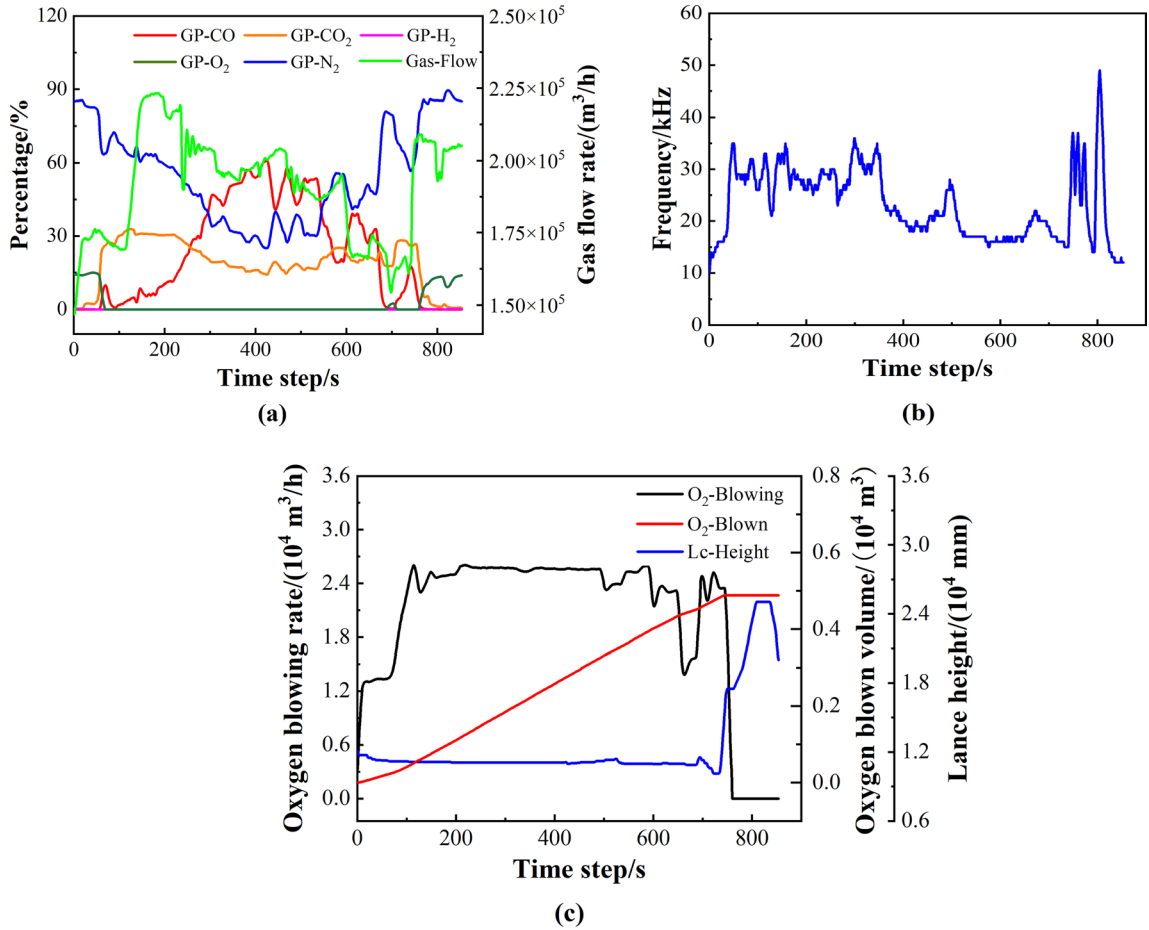
Fig. 1—An example of time series data in one typical heat, including (*a*) off-gas-related curves, (*b*) a sonar slagging curve, and (*c*) blowing practice-related curves, s—second.
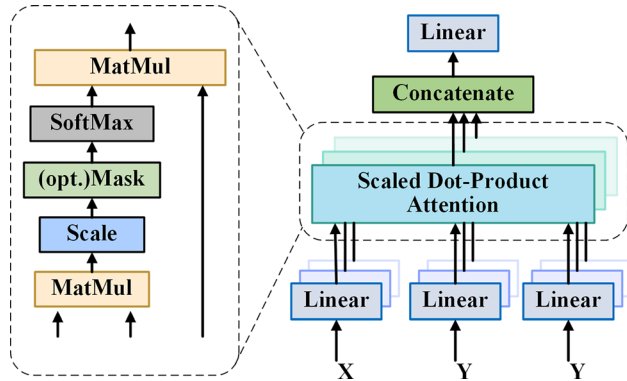

Fig. 2—Structure of the transformer encoder.[40] X—input; Y—target.

In particular, models may have conflicting MAEs and R^2 values due to predicted value distribution (*e.g.*, lower MAEs but lower R^2 values). In this case, since the factory takes the MAE as the key model evaluation index, and the MAE's statistical evaluation of the model is more intuitive and interpretable, the model with a lower MAE will be regarded as the better model.

III. SELF-ATTENTION-BASED CONVOLUTIONAL PARALLEL NETWORK (SABCP)

SabCP is a deep learning model proposed for accurately predicting the endpoint of BOF steelmaking; it incorporates tabular and time series data, with a network structure comprising data preprocessing and parallel backbone components. SabCP follows a patchwise approach, dividing inputs into patches for independent feature extraction using the same backbone model. The feature vectors from each patch are then merged and projected to produce the final output. Further details on SabCP are provided, with its general structure depicted in Figure 4. The training strategy, list of hyperparameters, and tuning results are shown in Appendix.

A. Data Preprocessing Module

The data preprocessing module is mainly composed of two parallel parts, namely, a time series patching and wavelet transform network and a tabular feature embedding network. The outputs of these two parallel
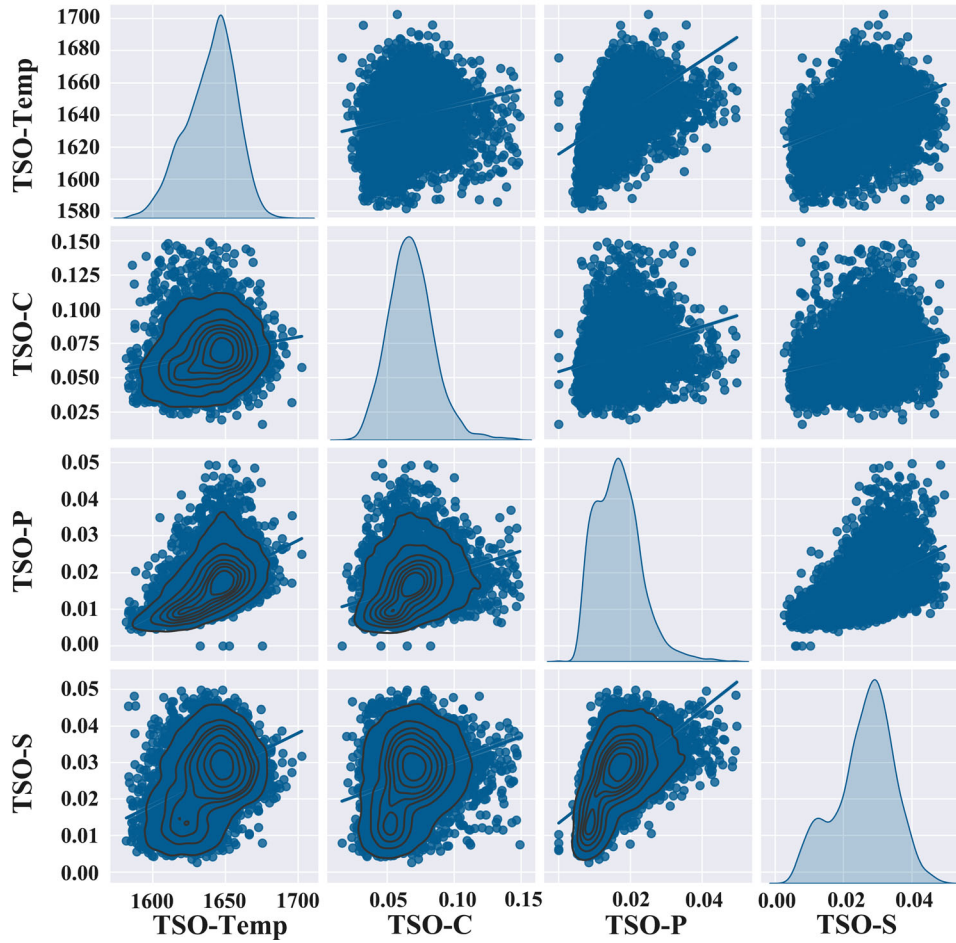
Fig. 3—Distribution and relationship between each target.

parts are concatenated. The final output of the data preprocessing module are multiple sequence tensors. Detailed descriptions of each part are provided in Fig. 6.

1. Time series patching and the wavelet transform network

The time series wavelet decomposition network extracts features from different sequence segments and frequencies in time series data. Research by Nie et al.[41] demonstrated that the patchwise method is more suitable for processing long time series data; it treats a patch of time series as an individual token, which decreases the model complexity to avoid overfitting. Because each patch hardly affect the other in the model training process, the locality of the time series is preserved. In addition, SabCP uses the self-attention mechanism, and its time complexity of attention computation is proportional to the square of the time series length. The patchwise model greatly reduces the computational time complexity of the self-attention mechanism. Furthermore, this patchwise model can be used to develop unsupervised or self-supervised pre-trained models for time series in the future. Initially, the processed time series data are divided into a_1 (where $a_1 = 2^i$) equal

patches. Each patch then undergoes two-dimensional discrete wavelet transformation (2D-DWT). This process enables separate extraction of features from different frequency components of the time series, thus enhancing the model's overall feature extraction capability. Eq. [3] represents the father scaling function ϕ for any input x, while Eq. [4] represents the mother wavelet basis ψ.

$$\phi(x) = \begin{cases} 1 & if 0 \leq x <; 1 \\ 0 & otherwise \end{cases} \quad [3]$$

$$\psi(x) = \begin{cases} 1 & 0 \leq x < 0.5 \\ -1 & 0.5 \leq x < 1 \\ 0 & elsewhere \end{cases} \quad [4]$$

For a two-dimensional (2D) input (x, y), Eq. [5] is a 2D scaling function $\phi(x, y)$. The 2D wavelet functions for the horizontal edge (H), vertical edge (V), and diagonal direction (D) are ψ^H (Eq. [6]), ψ^V (Eq. [7]), and ψ^D (Eq. 8]), respectively.

$$\phi(x, y) = \phi(x)\phi(y) \quad [5]$$
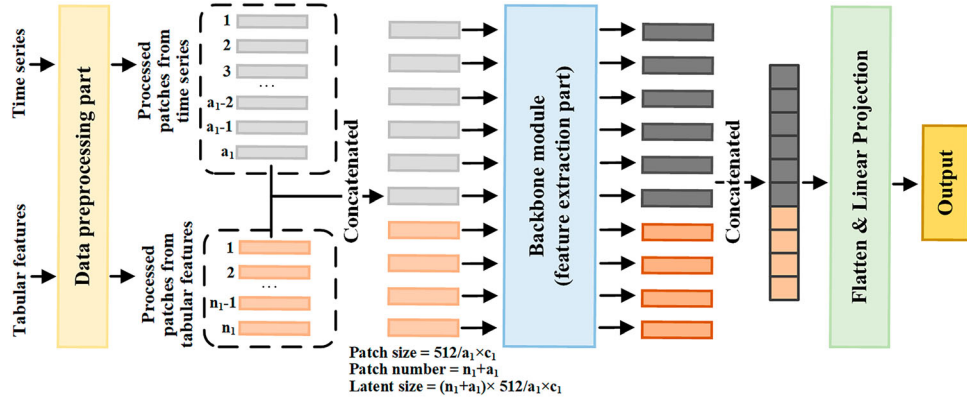
Fig. 4—General structure of SabCP. a_1—number of patches from the time series, n_1—number of patches from tabular features, c_1—embedding dimension of each embedded patch.
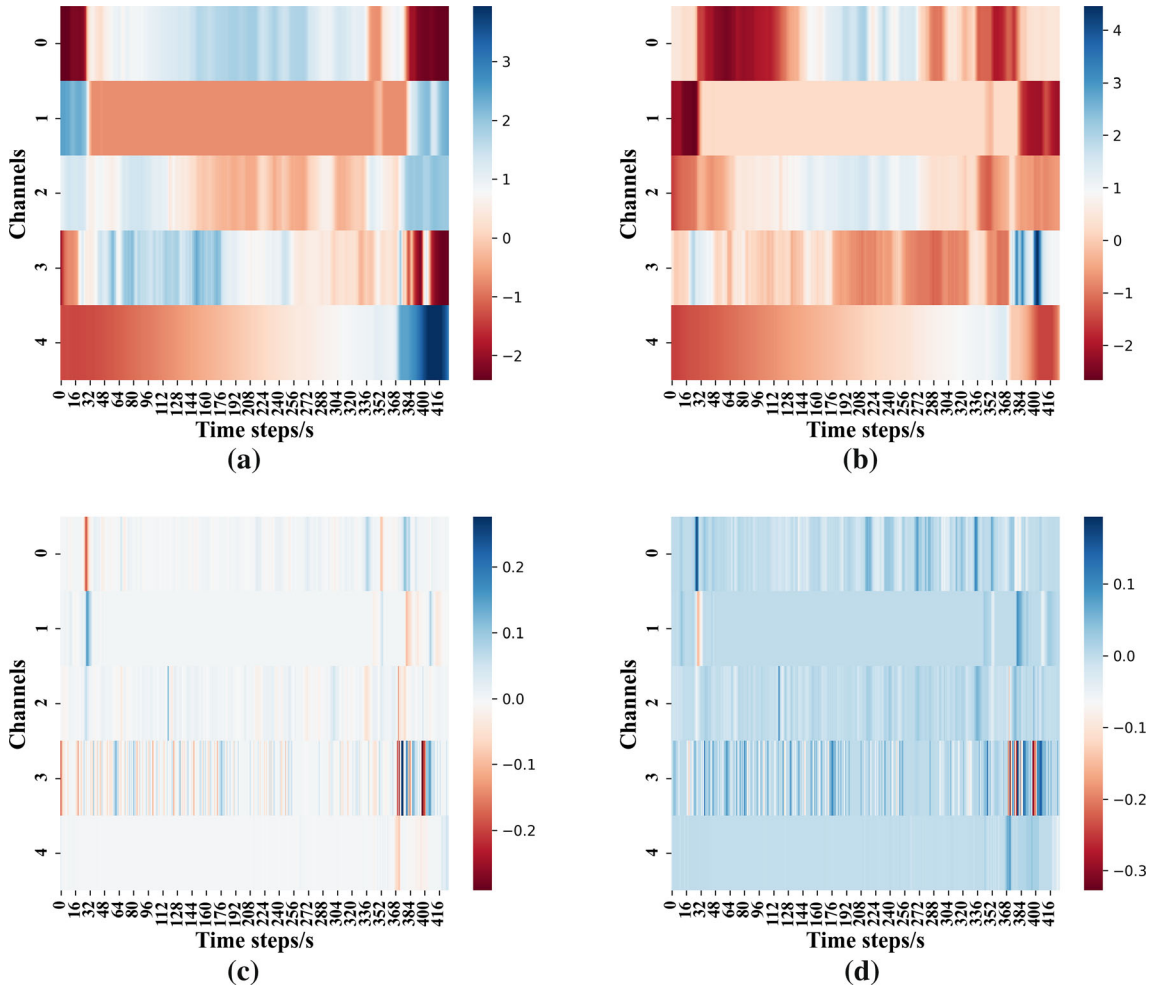
Fig. 5—An example of transformed time series by single-stage 2D-DWT. The components are cA (a), cH (b), cV (c), cD (d).

$$\psi^H(x, y) = \psi(x)\phi(y) \qquad [6]$$

$$\psi^D(x, y) = \psi(x)\psi(y). \qquad [8]$$

$$\psi^V(x, y) = \phi(x)\psi(y) \qquad [7]$$

If the length and width of the input (x, y) are M and N, respectively, in the j series of transformation, the scale basis function (Eq. [9]), and shift basis function (Eq. [10]) are described as follows ($M = 2^m$, $N = 2^n$):

$$\phi_{0,m,n}(x,y) = 2^{j/2}\phi\left(2^j x - m, 2^j - n\right) \quad [9]$$

$$\psi^i_{j,m,n}(x,y) = 2^{j/2}\psi\left(2^j x - m, 2^j - n\right), i = \{H, V, D\}. \quad [10]$$

Finally, the 2D-DWT transforms a multichannel time series (2D input, channels × time steps) into four components: a low-frequency component W_ϕ (Eq. [11]) and three high-frequency components $W^i_\psi, i = \{H, V, D\}$ (Eq. [12]), where $f(x, y)$ is a discrete form of (x, y).

$$W_\phi(0, m, n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\phi_{0,m,n}(x,y) \quad [11]$$

$$W^i_\psi(j, m, n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\psi^i_{j,m,n}(x,y), i$$
$$= \{H, V, D\}. \quad [12]$$

In SabCP, 2D-DWT is performed in a single stage ($j = 1$); that is, a multichannel time series is directly decomposed into low-frequency components (cAs), horizontal high-frequency components (cHs), vertical high-frequency components (cVs), and diagonal high-frequency components (cDs). Each component is a tensor of five channels.

After 2D-DWT, to maintain tensor integrity, each tensor is padded and processed by the same two-dimensional convolutional network (Conv2d) with c_1 kernels of size 5. Figure 5 illustrates the wavelet-transformed matrix from the unpadded time series shown in Figures 1 and 6(a) shows the structure of the time series patching and wavelet transform network.

2. *Tabular feature embedding network*

The tabular feature embedding network embeds tabular data, increases data dimension, and extracts initial features; it includes an embedding layer, a one-dimensional dilation convolutional layer,[42] allowing interval sampling and enlarging receptive fields with a dilation rate parameter, and Conv2ds. For an input sequence x of length M, kernel weight θ, dilation parameter d, and time series information s, the dilated convolution function is expressed as Eq. [13].

$$F(s) = \sum_{i}^{M-1}\theta(i) * x_{s-d*i}. \quad [13]$$

The dilation convolution layer expands the time series dimension and enhances feature extraction by enlarging the receptive field. Initially, the input length is changed from n_0 to n_1 by the embedding layer. Then, a one-dimensional (1D) dilation convolutional layer processes the extended input to add dimensions and further extract features.
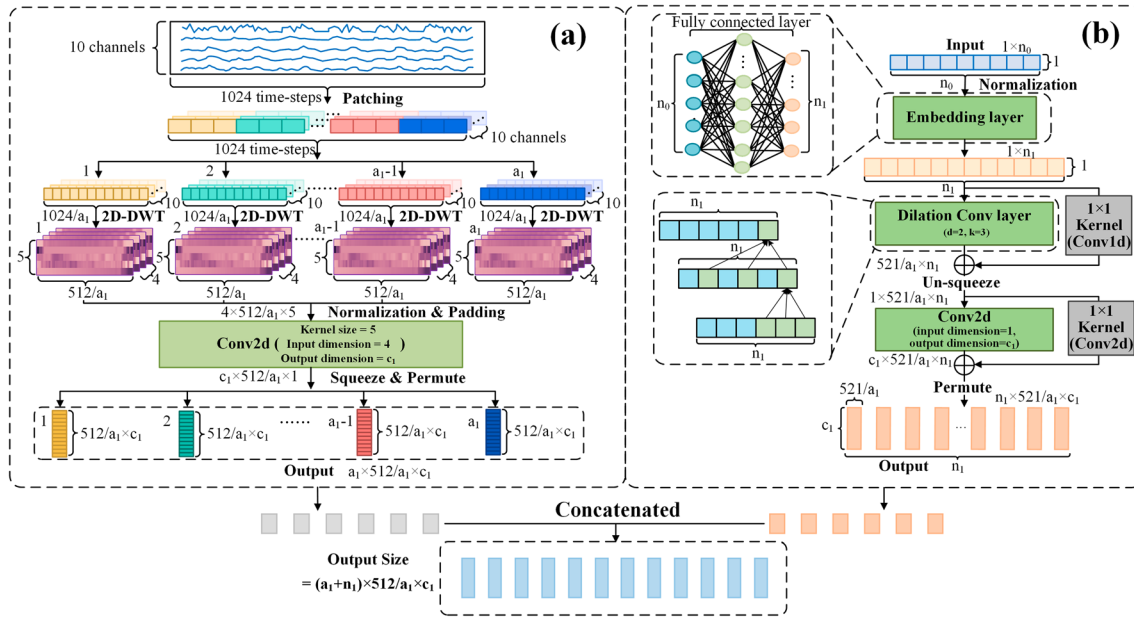
Fig. 6—Structure of the proposed data preprocessing module: (*a*) time series patching and wavelet transform network, (*b*) tabular feature embedding network.

The input dimension of its first layer is 1, resulting in an output dimension of $512/a_1$. The extended input is then residually connected to the output after dimension augmentation via a 1×1 convolution kernel. The resulting output forms a tensor with a length of n_1 and a width of $512/a_1$. These tensors are further processed by Conv2ds with c_1 kernels. Figure 6(b) illustrates the structure of the tabular feature embedding network.

B. *Parallel Backbone*

The parallel backbone efficiently extracts features; it includes an attention-based convolutional temporal network for local feature extraction and a gated self-attention network for global feature extraction. The output of the data preprocessing module is independently fed into these two components. Their outputs are added, processed by a Conv2d layer, flattened, and projected to a single output. The structure of the parallel backbone is shown in Figure 7.

1. *Attention-based convolutional temporal network*

According to previous research,[42–44] convolutional neural networks (CNNs) have demonstrated a robust ability to extract local features from time series. An attention-based convolutional temporal network comprises multiple attention-based convolutional temporal blocks, each integrating a causal dilation convolution layer, collaborative attention module, layer normalization layer, rectified linear unit (ReLU) function, and dropout layer. Causal dilation convolution layer ensures a one-way model structure. The causal convolution part in a casual dilation convolution layer is expressed as Eq. [14], where x is the input sequence, H is the length of x, $p(x)$ is the final output, and $p(x_t|x_1, x_2\dot{s}\dot{s}, x_{t-1})$ is the output of the previous layers.

$$p(x) = \prod_{t=1}^{H} p(x_t|x_1, x_2\dot{s}\dot{s}, x_{t-1}). \qquad [14]$$

The collaborative attention module effectively weights the eigenvectors, considering both the channel and time step dimensions. Yu *et al.*[45] demonstrated the efficacy of standard deviation pooling (std-pooling)
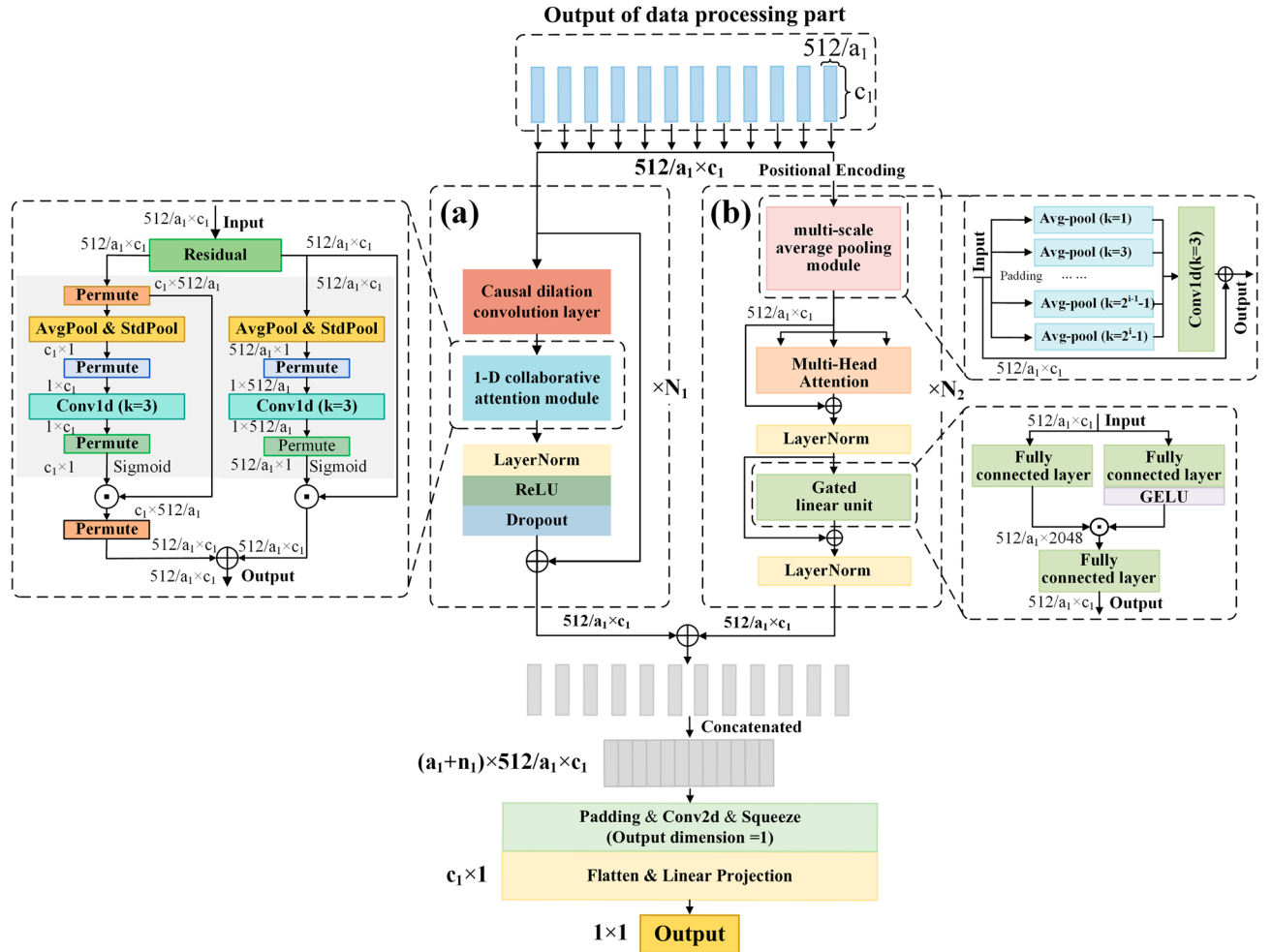
Fig. 7—The structure of the proposed parallel backbone: (*a*) structure of the attention-based convolutional temporal network; (*b*) structure of the gated self-attention network. N_1—the number of attention-based convolutional temporal blocks, N_2—the number of gated self-attention layers.
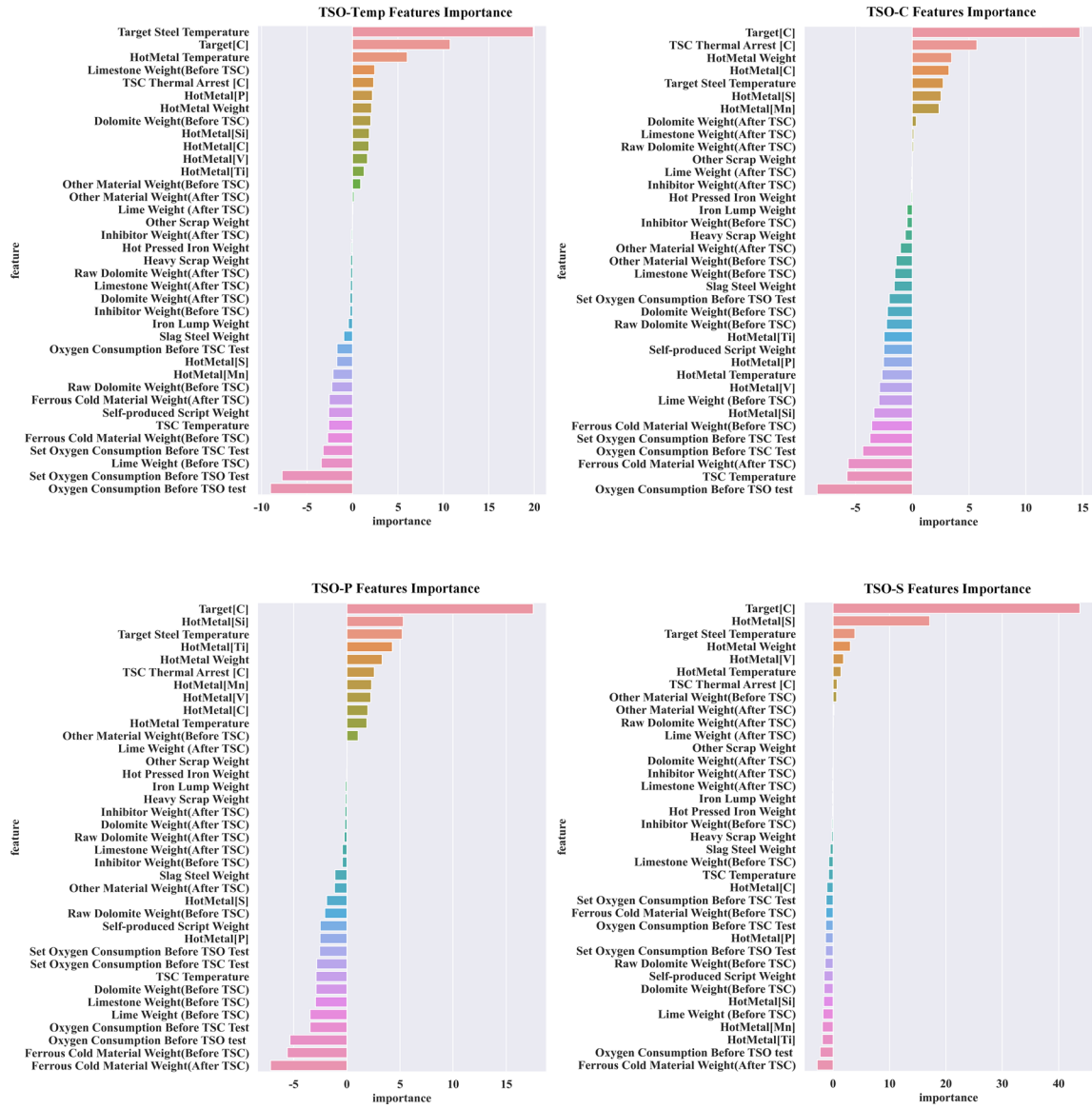
Fig. 8—Importance of visualized tabular features.

and average pooling (avg-pooling) for feature extraction and weight calculation. Initially, std-pooling and avg-pooling operations are conducted on each channel and time step of the original sequence. Subsequently, Conv1d is employed for the excitation transformation of the output feature vector post squeeze operation, thereby forming the final attention weight. The structure of the attention-based convolutional temporal network is shown in Figure 7(a).

2. Gated self-attention network

According to previous research,[40,46,47] models employing the self-attention mechanism exhibit superior capability in extracting global features from sequence inputs. Additionally, Wang et al.[48] demonstrated that incorporating avg-pooling at various scales can enhance the model's sensitivity to global correlations within time series data. Inspired by these studies, the gated self-attention network is designed with multiple gated

self-attention layers, and each layer contains a multi-scale avg-pooling module and a gated self-attention module. The input is processed by positional encoding. The structure of the gated self-attention network is shown in Figure 7(b).

In a multiscale avg-pooling module, the input undergoes processing through multiple avg-pooling layers, each employing different kernel size. Specifically, for the i-th avg-pooling layer, the kernel size is determined to be $2^{(i-1)} - 1$, with padding that maintain consist output dimensions. Subsequently, the outputs of these avg-pooling layers are padded, aggregated and processed by Conv1d with k = 3. The input is also connected residually to the output.

In a gated self-attention layer, the input is processed by a transformer encoder layer with multiple heads (default = 8). The feedforward neural network (FFN) in the transformer encoder is replaced by gated linear units (GLUs).[49] GLUs utilize two fully connected
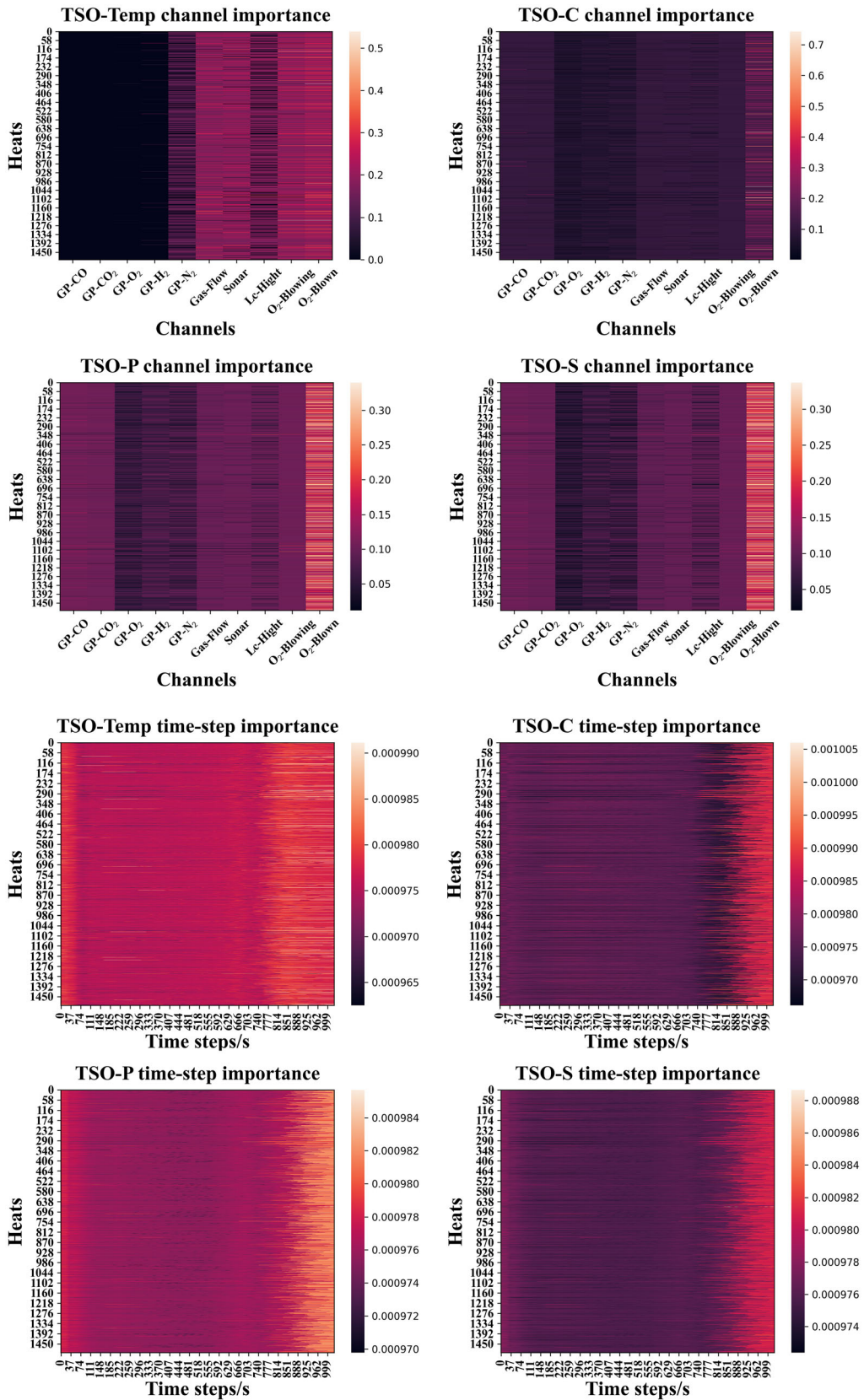
Fig. 9—The importance of channels and time steps in different tasks.

layers, with one employing a Gaussian error linear unit (GELU) as the activation function. The Hadamard products of their outputs are fed into a third fully connected layer to produce the final GLU output.
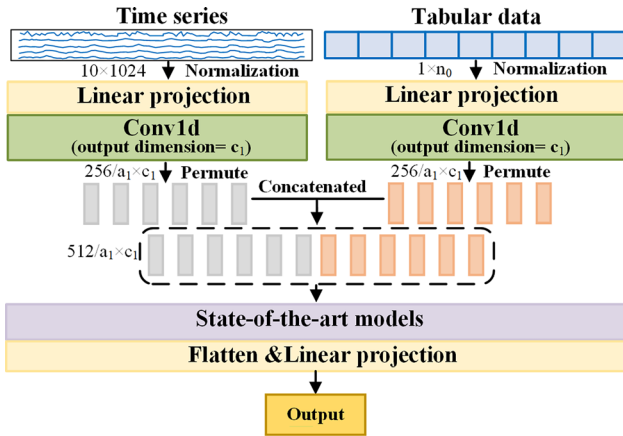
Fig. 10—Structure of the modified prediction method with SOTA models.

IV. RESULTS AND DISCUSSION

A. *Results of the Importance Analysis of Tabular Features*

Figure 8 illustrates the visualization of the importance of each tabular feature. The columns on the right indicate positive correlations, while those on the left indicate negative correlations. The length of each column corresponds to the strength of correlation with the target feature.

According to Figure 8, it can be concluded that the target [C] and steel temperature are significant for each target. Additionally, the oxygen content at each stage and the initial state of the hot metal also exhibit considerable importance. Furthermore, the significance of pre-TSC additives outweighs that of post-TSC additives.

Table II.   Performance of SabCP and Other Models in Different Tasks

| Targets	Models	R^2	MAE (°C)	±10°C	±15°C	±20°C
TSO-Temp (°C)	SabCP	0.785	6.374	0.801	0.921	0.977
	informer	0.780	6.417	0.797	0.924	0.971
	autoformer	0.782	6.380	0.796	0.921	0.971
	SCINet	0.780	6.405	0.798	0.921	0.977
	FEDformer	0.781	6.398	0.800	0.925	0.977
	D-linear	0.778	6.438	0.797	0.922	0.972
	PatchMixer	0.784	6.376	0.794	0.917	0.976
TSO-C (Pct)	models	R^2	MAE (pct)	±0.01 pct	±0.015 pct	±0.02 pct
	SabCP	0.650	7.192×10^{-3}	0.755	0.869	0.941
	informer	0.640	7.321×10^{-3}	0.747	0.868	0.936
	autoformer	0.646	7.198×10^{-3}	0.755	0.877	0.939
	SCINet	0.645	7.260×10^{-3}	0.758	0.871	0.939
	FEDformer	0.646	7.259×10^{-3}	0.752	0.867	0.944
	D-linear	0.644	7.276×10^{-3}	0.757	0.869	0.935
	patchMixer	0.647	7.195×10^{-3}	0.760	0.873	0.936
TSO-P (Pct)	models	R^2	MAE (pct)	±0.01 pct	±0.003 pct	±0.005 pct
	SabCP	0.705	2.390×10^{-3}	0.326	0.734	0.896
	informer	0.690	2.439×10^{-3}	0.313	0.724	0.881
	Autoformer	0.701	2.399×10^{-3}	0.327	0.729	0.898
	SCINet	0.696	2.409×10^{-3}	0.335	0.735	0.887
	FEDformer	0.699	2.401×10^{-3}	0.325	0.731	0.899
	D-linear	0.691	2.437×10^{-3}	0.322	0.729	0.901
	patchMixer	0.703	2.393×10^{-3}	0.322	0.735	0.899
TSO-S (Pct)	models	R^2	MAE (pct)	±0.01 pct	±0.003 pct	±0.005 pct
	SabCP	0.843	2.224×10^{-3}	0.378	0.781	0.911
	informer	0.829	2.398×10^{-3}	0.339	0.748	0.900
	autoformer	0.841	2.228×10^{-3}	0.384	0.775	0.907
	SCINet	0.831	2.352×10^{-3}	0.361	0.752	0.901
	FEDformer	0.840	2.230×10^{-3}	0.399	0.777	0.908
	D-linear	0.837	2.234×10^{-3}	0.433	0.761	0.894
	patchMixer	0.842	2.227×10^{-3}	0.393	0.770	0.907
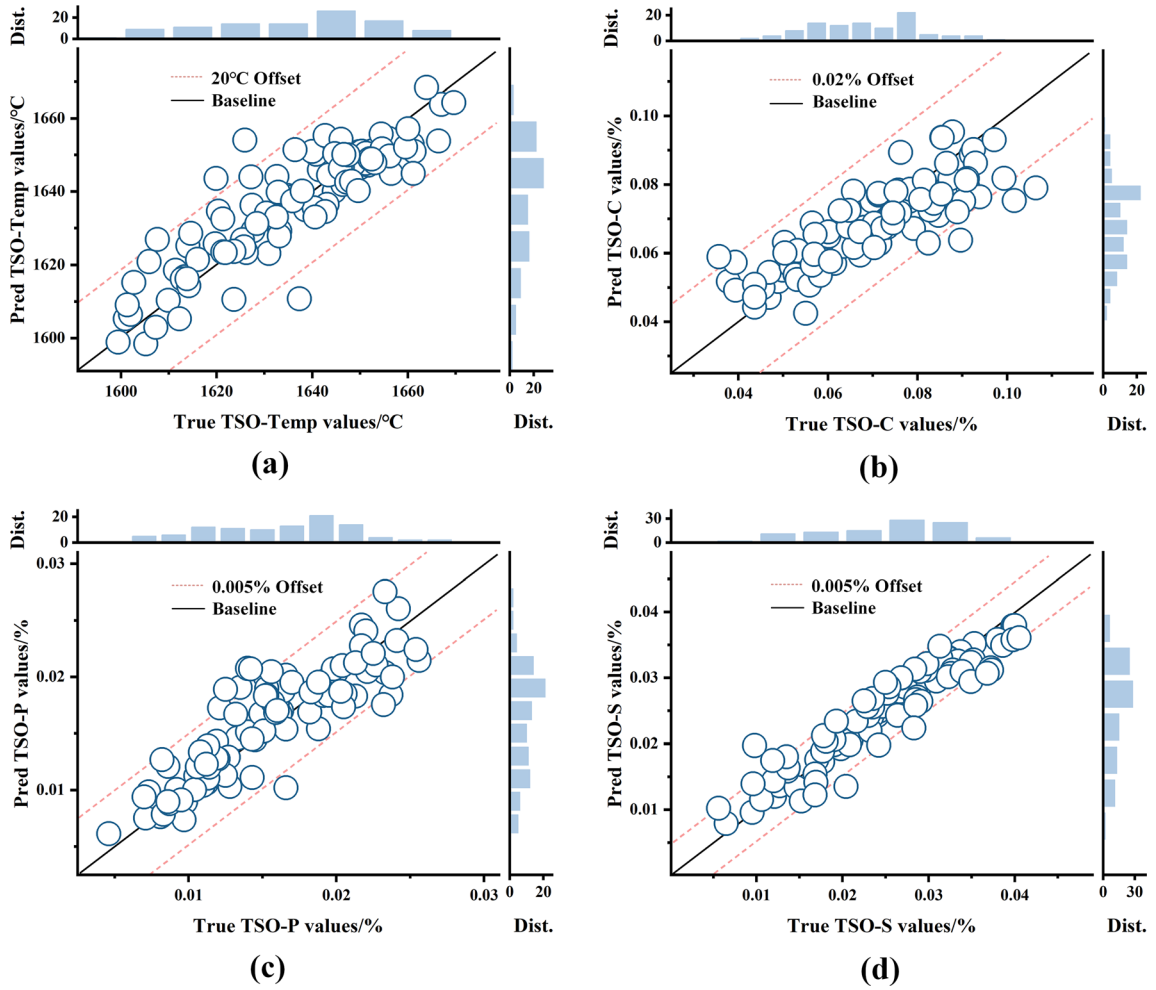
Fig. 11—The prediction results of the SabCP for TSO-Temp (*a*), TSO-C (*b*), TSO-P (*c*), and TSO-S (*d*), Pred—predicted, Baseline—Pred values = True values, Offset—max range of the hit rates criterion, Dist.— statistical distribution of the samples.

B. *Results of the Importance Analysis of Time Series Data*

The importance of the channels and time steps of the time series data is shown in Figure 9. Heatmaps illustrate the importance of channels and time steps in time series data. The total oxygen blowing curve and instantaneous oxygen blowing intensity curve are notably crucial across all tasks. There are slight differences in heatmaps for predicting TSO-P and TSO-S. GP-O2 and GP-H2 are less important across all tasks. Overall, later time steps are more important, while time steps exhibit consistent importance for the TSO-Temp, TSO-C, and TSO-S predictions. However, in predicting the TSO-P, the middle time step is less important.

C. *Results of the Comparison of SabCP with SOTA Time Series Analysis Models*

Table II provides a comparison of the performance of SabCP with that of SOTA time series prediction and classification models. The control SOTA models include the Informer model proposed by Zhou *et al.*,[50] the Autoformer model proposed by Wu *et al.*,[51] the

SCINet model proposed by Liu *et al.*,[52] the FEDformer model proposed by Zhou *et al.*,[47] the D-linear model proposed by Zeng *et al.*,[53] and the PatchMixer model proposed by Gong *et al.*[54]

Since these models were not originally designed as multi-input models, they have been modified to maintain the same input patterns as SabCP. The time series and tabular features are embedded using linear and conv1d layers. This modification preserves data information while resizing the data to match the input format of the SabCP backbone. The modified method is illustrated in Figure 10. The results are listed in Table II. The data for the best values in each task are bolded.

According to Table II, it can be concluded that SabCP performs better in all tasks than the modified multi-input SOTA models. For predicting TSO-Temp/C/P/S, R^2 reaches 0.785, 0.650, 0.705, and 0.843, and the MAEs reach 6.374 °C, 7.192×10^{-3} pct, 2.390×10^{-3} pct, and 2.224×10^{-3} pct, respectively. In addition to the prediction of TSO-P, for other targets, the hit rate is more than 90 pct. To show this more clearly, one

Targets	Models	R^2	MAE (°C)	±10°C	±15°C	±20°C
TSC-Temp (°C)	SabCP	0.785	6.374	0.801	0.921	0.977
	SabCP-rTpw	0.701	7.550	0.733	0.875	0.951
	Xgboost	0.711	7.431	0.729	0.893	0.957
	LightGBM	0.713	7.375	0.736	0.891	0.957
	catboost	0.714	7.328	0.738	0.890	0.959
	tabnet	0.703	7.551	0.715	0.885	0.959
	node	0.702	7.547	0.728	0.890	0.953
TSC-C (Pct)	models	R^2	MAE (pct)	±0.01 pct	±0.015 pct	±0.02 pct
	SabCP	0.650	7.192×10^{-3}	0.755	0.869	0.941
	SabCP-rTpw	0.594	8.366×10^{-3}	0.689	0.846	0.935
	Xgboost	0.607	8.040×10^{-3}	0.710	0.869	0.934
	lightGBM	0.611	7.935×10^{-3}	0.713	0.869	0.933
	catboost	0.609	7.999×10^{-3}	0.715	0.864	0.932
	tabnet	0.603	8.062×10^{-3}	0.706	0.862	0.927
	node	0.603	8.050×10^{-3}	0.715	0.861	0.933
TSC-P (Pct)	models	R^2	MAE (pct)	±0.001 pct	±0.003 pct	±0.005 pct
	SabCP	0.705	2.390×10^{-3}	0.326	0.734	0.896
	SabCP-rTpw	0.644	2.729×10^{-3}	0.275	0.661	0.865
	Xgboost	0.652	2.659×10^{-3}	0.283	0.681	0.873
	lightGBM	0.657	2.650×10^{-3}	0.276	0.685	0.872
	catboost	0.653	2.655×10^{-3}	0.286	0.687	0.865
	tabnet	0.646	2.700×10^{-3}	0.274	0.673	0.867
	node	0.648	2.686×10^{-3}	0.275	0.671	0.869
TSC-S (Pct)	models	R^2	MAE (pct)	±0.001 pct	±0.003 pct	±0.005 pct
	SabCP	0.843	2.224×10^{-3}	0.378	0.781	0.911
	SabCP-rTpw	0.753	3.064×10^{-3}	0.253	0.641	0.836
	Xgboost	0.776	2.952×10^{-3}	0.256	0.659	0.843
	lightGBM	0.777	2.906×10^{-3}	0.247	0.651	0.859
	catboost	0.776	2.933×10^{-3}	0.243	0.655	0.854
	tabnet	0.762	3.005×10^{-3}	0.250	0.644	0.847
	node	0.759	3.008×10^{-3}	0.243	0.639	0.853

hundred heats were randomly sampled from the test results. The SabCP prediction results for the test set are detailed in Figure 11.

According to Figure 11, accurate predictions are more likely for true values distributed in the middle and low ranges for each prediction task. Conversely, large true values, particularly extremely large values, pose greater difficulty in prediction.

In line with the principle of BOF steelmaking, a higher C/P/S content in hot steel corresponds to a more intense oxidation process. Consequently, the rate of change in such element content is greater, leading to lower predictability. Additionally, the scarcity of endpoint samples with high temperature or element (C/P/S) content results in insufficient weighting and difficulty in prediction. Furthermore, the limited number of endpoint samples with high temperatures or element contents exacerbate the challenge of prediction.

D. Results of the Comparison of SabCP with SOTA Machine Learning Models

Currently, the prevailing approach for BOF endpoint prediction involves developing machine learning (ML) data-driven models utilizing tabular features.[1–25] This section juxtaposes SabCP with a variant model, denoted as SabCP-rTpw, which excludes its time series input and processing component. This enables a comparison with SOTA machine learning models employing identical tabular inputs. The models used are extreme gradient boosting (Xgboost),[55] LightGBM, Catboost, the tabular attention network (TabNet),[56] and neural oblivious decision ensembles (NODE).[57] The comparative results between the SabCP and SOTA machine learning models are presented in Table III.

In each task, SOTA machine learning models utilizing tabular data exhibited inferior performance compared to SabCP with multiple inputs, with R^2 scores ranging from 5 to 10 pct lower. However, they all outperformed the SabCP-rTpw model, which also utilizes only tabular data, with R^2 scores ranging from 1 to 3 pct higher.

Furthermore, in each task, decision tree-based models (XGBoost, LightGBM, and Catboost) demonstrated superior performance compared to neural network-based models (TabNet and NODE), which was particularly evident when predicting TSO-Temp and TSO-C.

Table IV. Results of the Ablation Experiments

Targets	Models	R^2	MAE (°C)	±10°C	±15°C	±20°C
TSO-Temp (°C)	SabCP	0.785	6.374	0.801	0.921	0.977
	SabCP-rTf	0.744	6.979	0.764	0.909	0.962
	SabCP-rTpw	0.701	7.550	0.733	0.875	0.951
	SabCP-rTp	0.770	6.559	0.789	0.924	0.971
	SabCP-rDwt	0.770	6.546	0.779	0.913	0.977
	SabCP-rCtn	0.747	6.904	0.764	0.903	0.969
	SabCP-rGsn	0.760	6.654	0.770	0.918	0.973
	SabCP-rCam	0.775	6.495	0.798	0.912	0.973
	SacCP-rMt	0.777	6.479	0.791	0.924	0.973
TSO-C (Pct)	models	R^2	MAE (pct)	±0.01 pct	±0.015 pct	±0.02 pct
	SabCP	0.650	7.192×10^{-3}	0.755	0.869	0.941
	SabCP-rTf	0.612	8.006×10^{-3}	0.713	0.871	0.935
	SabCP-rTpw	0.594	8.366×10^{-3}	0.689	0.846	0.935
	SabCP-rTp	0.638	7.374×10^{-3}	0.747	0.872	0.933
	SabCP-rDwt	0.641	7.278×10^{-3}	0.751	0.869	0.932
	SabCP-rCtn	0.623	7.517×10^{-3}	0.739	0.865	0.926
	SabCP-rGsn	0.634	7.478×10^{-3}	0.744	0.864	0.933
	SabCP-rCam	0.637	7.369×10^{-3}	0.747	0.874	0.935
	SacCP-rMt	0.640	7.278×10^{-3}	0.748	0.863	0.931
TSO-P (Pct)	models	R^2	MAE (pct)	±0.001 pct	±0.003 pct	±0.005 pct
	SabCP	0.705	2.390×10^{-3}	0.326	0.734	0.896
	SabCP-rTf	0.672	2.550×10^{-3}	0.294	0.709	0.885
	SabCP-rTpw	0.644	2.729×10^{-3}	0.275	0.661	0.865
	SabCP-rTp	0.690	2.448×10^{-3}	0.320	0.725	0.891
	SabCP-rDwt	0.694	2.425×10^{-3}	0.329	0.733	0.889
	SabCP-rCtn	0.686	2.460×10^{-3}	0.317	0.721	0.889
	SabCP-rGsn	0.689	2.449×10^{-3}	0.314	0.724	0.888
	SabCP-rCam	0.688	2.457×10^{-3}	0.311	0.723	0.893
	SacCP-rMt	0.692	2.428×10^{-3}	0.331	0.729	0.894
TSO-S (Pct)	models	R^2	MAE (pct)	±0.001 pct	±0.003 pct	±0.005 pct
	SabCP	0.843	2.224×10^{-3}	0.378	0.781	0.911
	SabCP-rTf	0.805	2.654×10^{-3}	0.284	0.711	0.878
	SabCP-rTpw	0.753	3.064×10^{-3}	0.253	0.641	0.836
	SabCP-rTp	0.831	2.366×10^{-3}	0.357	0.760	0.893
	SabCP-rDwt	0.833	2.271×10^{-3}	0.390	0.767	0.903
	SabCP-rCtn	0.826	2.378×10^{-3}	0.361	0.752	0.895
	SabCP-rGsn	0.836	2.241×10^{-3}	0.383	0.781	0.905
	SabCP-rCam	0.835	2.262×10^{-3}	0.405	0.759	0.897
	SacCP-rMt	0.839	2.233×10^{-3}	0.421	0.758	0.903

E. Results of the Ablation Experiment

To assess the rationality of the model structure, an ablation experiment was conducted on SabCP. Various components were either removed or replaced, including (1) removal of the tabular feature embedding network (SabCP-rTf), (2) removal of the time series patching and wavelet transform network (SabCP-rTpw), (3) removal of the patching process in the time series patching and wavelet transform network (SabCP-rTp), (4) removal of the 2D-DWT in the time series patching and wavelet transform network (SabCP-rDwt), (5) removal of the parallel convolutional temporal network in the data feature extraction part (SabCP-rCtn), (6) removal of the gated self-attention network in the data feature extraction part (SabCP-rGsn), (7) removal of the collaborative attention module in the parallel convolutional temporal network (SabCP-rCam), and (8) replacement of the gated self-attention network with a traditional transformer layer (SabCP-rMt). The results of the ablation experiments are presented in Table IV.

Based on the table results, the absence of either time series or tabular data significantly impacts model performance. Time series data are more crucial than tabular data, with the model exhibiting its poorest performance when lacking time series and its processing network. Regarding time series processing, removing patching or 2D-DWT diminishes model performance, which is particularly evident in predicting TSO-Temp, TSO-P, and TSO-S. Concerning model structure, eliminating any major component or module decreases performance, with alterations to the convolutional temporal network exerting the greatest impact on model performance.

Table V. Comparison of the Best Models in This and Previous Research

| Targets	Models	R^2	MAE	MSE
TSO-Temp (°C)	SabCP	0.785	6.374	67.345
	FWA-TSVR[25]	—	—	342.028
	CS-TSVR[25]	—	—	382.750
	WTWTSVR[58]	0.797	4.738	—
	MI_IWSVM[59]	—	—	76.631
	DRSupAE-NN[60]	—	—	67.544
	DRSupAE-JITRN[60]			44.084
	DSupAE-JITRN[60]			77.338
TSO-C (Pct)	SabCP	0.650	7.192×10^{-3}	1.132×10^{-4}
	FWA-TSVR[25]	—	—	8.237×10^{-4}
	CS-TSVR[25]	—	—	8.283×10^{-4}
	WTWTSVR[58]	0.693	2.600×10^{-3}	—
	MI_IWSVM[59]	—	—	2.783×10^{-4}
	DRSupAE-NN[60]	—	—	3.984×10^{-4}
	DRSupAE-JITRN[60]	—	—	1.911×10^{-4}
	DSupAE-JITRN[60]	—	—	4.805×10^{-4}
TSO-P (Pct)	SabCP	0.705	2.390×10^{-3}	1.306×10^{-5}
	PCA-BP model[10]	0.791	—	—
	BP model[16]	0.759		—
	PCA–GA–BP model[20]	—	—	2.250×10^{-5}
	GBR model[61]	0.599	—	1.018×10^{-5}
	RFR model[61]	0.608		1.056×10^{-5}
	M-c BP[62]	0.846	—	9.000×10^{-6}
	LWOA-TSVR[63]	0.911	6.690×10^{-3}	7.850×10^{-5}
TSO-S (Pct)	SabCP	0.843	2.224×10^{-3}	1.298×10^{-5}
	LWOA-TSVR[63]	0.903	8.230×10^{-3}	7.430×10^{-5}
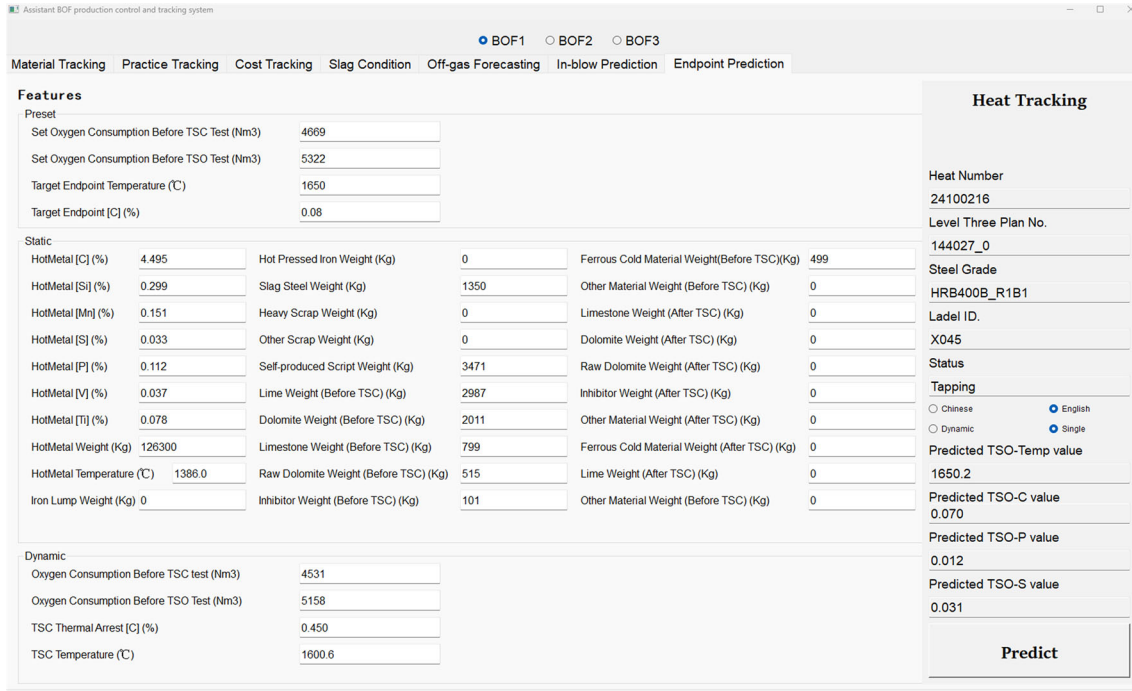
Fig. 12—The HMI of the endpoint prediction system with SabCP.
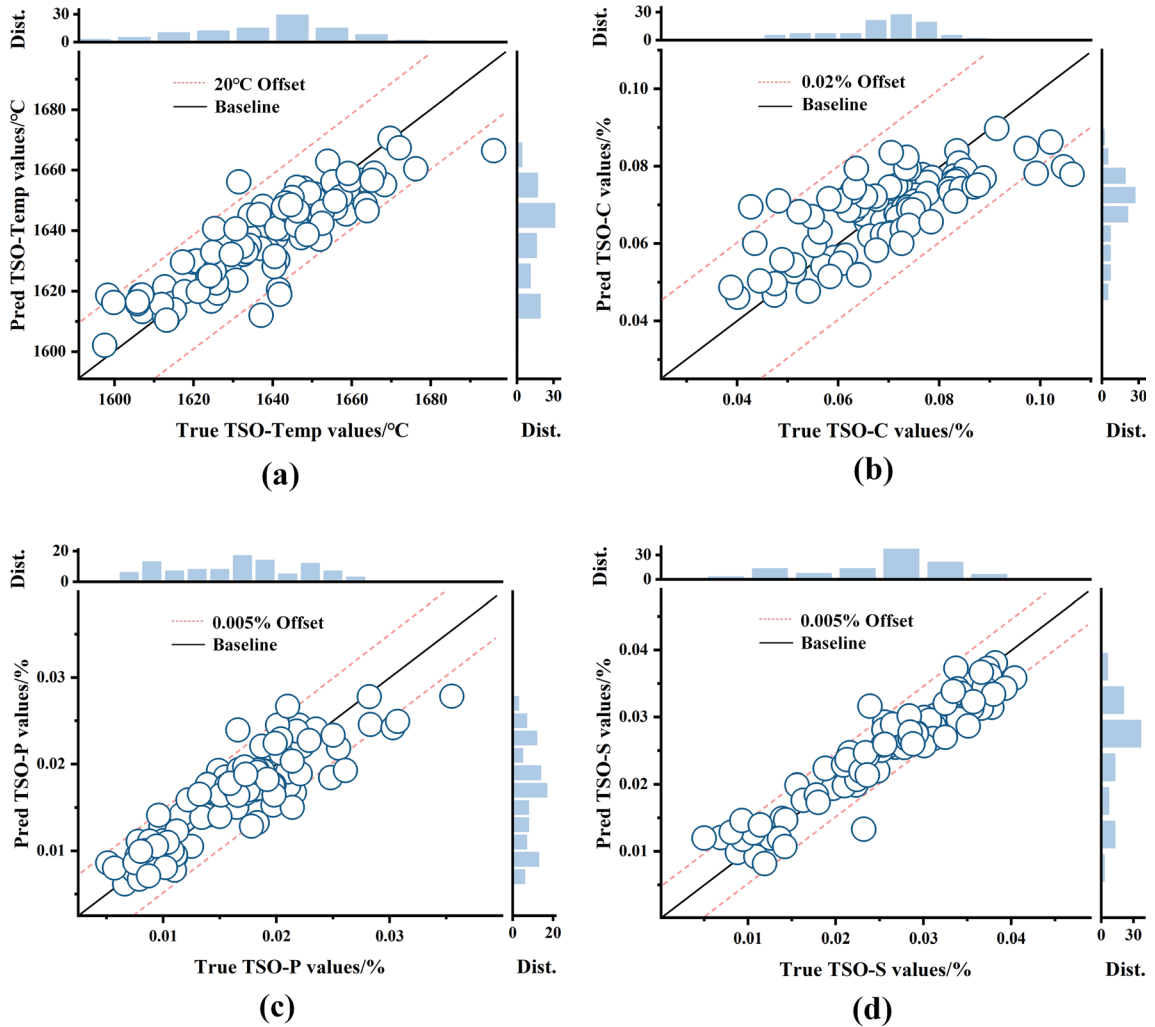
Fig. 13—The external validation results of SabCP for TSO-Temp (*a*), TSO-C (*b*), TSO-P (*c*), and TSO-S (*d*).

F. *Results of Comparing SabCP with Other Public Works*

The results of BOF endpoint prediction from other studies are juxtaposed with those of the SabCP model. Despite variations in data quality, objectives, application scenarios (high- or low-endpoint carbon), BOF vessel structure, and steelmaking processes across these studies compared to the scenarios addressed in this study, they remain valuable for reference and comparison. Given that certain studies have reported the MSEs of their models, this section includes a specific calculation of the prediction MSE for the SabCP model. The results are shown in Table V.

V. EXTERNAL VALIDATION

A. *The HMI of the Endpoint Prediction Module*

Figure 12 illustrates the (HMI) of the endpoint prediction module, implemented in Python. During prediction, data are retrieved from the programmable logic controller (PLC), level one system, or database. The data undergo automated preprocessing before being sent to the SabCP model for prediction. Basic data and predicted values are displayed in separate grids. The module offers two modes:

"Dynamic," it provides continuous predictions. Due to the lower sample weights for cases with extremely high or low oxygen blowing amount, this mode is used when the oxygen blowing amount is between 4700 and 5500 m^3 (mean $\pm$ 2 sigma) to ensure accuracy. Once the oxygen blowing amount exceeds 4700 m^3, the model outputs a set of predictions (TSO-Temp/C/P/S) every second to guide the process until the point when oxygen blowing stops.

"Single," where predictions occur upon pressing the "Predict" button. This mode simulates the tests with sub-lance, which involve point measurements. A single prediction can be made at the moment of stopping oxygen blowing to replace the TSO test. In the external validation, a single prediction is conducted simultaneously with the TSO test.

B. *Results of the External Validation*

Actual prediction results from 300 consecutive heats with no abnormal inputs were recorded. A comparison of these predicted values with the true values is shown in Figure 13. The prediction accuracy of SabCP slightly lags that of the test set. The R^2 for TSO-Temp, TSO-C, TSO-P, and TSO-S were 0.740, 0.570, 0.697, and 0.831. And the MAEs were 7.232 °C, 8.065×10^{-3} pct, 2.817×10^{-3} pct, and 2.712×10^{-3} pct, respectively. Specifically, for TSO-Temp/C/P/S, the MAE increased by 0.858 °C, 8.722×10^{-4} pct, 4.265×10^{-4} pct, and 4.875×10^{-4} pct, respectively. This finding underscores the robustness and generalizability of SabCP. One hundred samples were randomly selected from the external validation results and shown in Figure 13.

VI. CONCLUSION

(i) The R^2 values of the designed multi-input deep learning model (SabCP) for TSO-Temp, TSO-C, TSO-P, and TSO-S were 0.785, 0.650, 0.705 and 0.843, respectively, and the MAEs were 6.374 °C, 7.192×10^{-3} pct, 2.390×10^{-3} pct, and 2.224×10^{-3} pct, respectively. These results show that SabCP has better performance than other models.

(ii) The ablation experiments show that changes in the input categories or structure of SabCP will decrease its prediction accuracy, especially once the time series inputs are eliminated or the parallel convolutional temporal network is removed.

(iii) In practical external validation of 300 heats, the prediction MAEs of SabCP for TSO-Temp, TSO-C, TSO-P, and TSO-S were 7.232 °C, 8.065×10^{-3} pct, 2.817×10^{-3} pct, and 2.712×10^{-3} pct, respectively, which are close to the testing results. This proves that SabCP has good robustness and generalizability.

(iv) According to the importance analysis, the preset values and blowing practice-related features in tabular data have greater importance, but the additive features have less importance. On the other hand, the curves related to blowing practices have greater importance; however, the curves related to gas percentages have less importance. Moreover, the later time steps of the curves are more important than the middle or early time steps.

CONFLICT OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

APPENDICES: DESCRIPTION OF ERRORS IN THE DATA

Since the division is random, the data used for the model learning (training dataset) and the data used for evaluating the model (validation and test dataset) contain the same types of errors. The following are these errors and the approaches to address them.

Measurement errors: For tabular data, measurement errors are primarily attributed to systematic errors. For instance, in measuring molten steel temperature, the typical error associated with a thermocouple ranges from 1 °C ~ 3 °C. However, there are also random errors present, such as when humid air causes powdered material to adhere to the walls of a hopper, resulting in underfeeding.

In the case of time series data, errors tend to be much smaller and predominantly systematic. For example, there may be slight errors associated with oxygen flow meters, while data obtained from absolute value encoders for parameters like lance height are virtually error-free.

Sampling Errors: Due to the large volume of collected data, certain infrequent steel grades, such as EH14, have lower sample weights, resulting in suboptimal prediction results. Consequently, the universal model derived from this study will be employed as a pre-trained backbone and subsequently fine-tuned using data from various steel grades to better fit the specific steel grade.

Labeling Errors: In rare cases, the values of the TSC test may be erroneously recorded as the values of the TSO test (target labels). To filter out this incorrect information, it is necessary to ensure that the carbon content for each set of TSC tests is higher than that of the TSO test, and the temperature is lower than that of the TSO test.

To minimize data errors, tabular data must conform to the criteria outlined in Table VI, as per the practical steelmaking process. Additionally, a large volume of data is being collected to reduce the prediction error associated with erroneous data. Introducing the time series can also solve part of the problem.

The time series data are automatically collected and stored by the low-maintenance gas analyzing system (LOMAS) and PLC system, ensuring high data quality. The primary source of data errors arises from abnormal smelting processes. Therefore, after filtering out recorded instances of abnormal steelmaking batches (such as reblowing, stoppage, and slopping), sequences with lengths ranging from 600 to 1024 time were selected, and relatively clean data can be obtained.

Some heats lacking time series data underwent data augmentation using generative adversarial networks (GANs).

DESCRIPTION OF THE TRAINING AND TUNING PROCESS

The deep learning models are trained with the mean square error (MSE) as the loss function and Adam as the optimizer. The learning rate is set to 0.001, and ReduceLROnPlateau is the learning rate scheduler. The models with the highest accuracy on the validation set in the training process are automatically saved. For each target, 1,000 automatic hyperparameter optimization

Table VI.   The Data Range and Cleaning Criterion

Features	Units	Range
Set Oxygen Consumption Before TSC Test	Nm^3	≥ 3500.000
Set Oxygen Consumption Before TSO Test	Nm^3	≥ 4200.000
Target Steel Temperature	°C	1600.000–1680.000
Target [C]	pct	0.040–0.150
Hotmetal [C]	pct	3.500–5.000
Hotmetal [Si]	pct	0.010–0.800
Hotmetal [Mn]	pct	0.010–0.600
Hotmetal [S]	pct	0.001–0.350
Hotmetal [P]	pct	0.050–0.160
Hotmetal [V]	pct	0.010–0.08
Hotmetal [Ti]	pct	0.005–0.160
Hotmetal Weight	kg	90,000.000–150,000.000
Hotmetal Temperature	°C	1250.000–1450.000
Lime Weight (Before TSC)	kg	≥ 500.000
Dolomite Weight (Before TSC)	kg	≥ 100.000
Oxygen Consumption Before TSC Test	Nm^3	≥ 3500.000
Oxygen Consumption Before TSO Test	Nm^3	≥ 4200.000
TSC Thermal Arrest [C]	pct	0.010–0.760 pct
TSC Temperature	°C	1500.000–1650.000

Table VII.   Hyperparameter Boundaries and the Best Hyperparameters of SabCP

Targets	Hyperparameters and Boundaries	Best Hyperparameters for Each Task
TSO-Temp (°C)	a_1: [2, 4, 8, 16, 32, 64] n_1: [2, 4, 8, 16, 32, 64] c_1: [128, 256, 512, 1024]	$a_1 = 32, n_1 = 8, c_1 = 512$ $N_1 = 3, N_2 = 4$ number of average pooling kernels = 16 number of self-attention head = 8
TSO-C (Pct)	N_1: [1, 2, 3, 4, 5, 6] N_2: [1, 2, 3, 4, 5, 6]	$a_1 = 64, n_1 = 16, c_1 = 256$ $N_1 = 5, N_2 = 2$ number of average pooling kernels = 8 number of self-attention head = 16
TSO-P (Pct)	number of average pooling kernels: (multiscale average pooling module) [2, 4, 6, 8, 16, 32]	$a_1 = 64, n_1 = 8, c_1 = 512$ $N_1 = 4, N_2 = 3$ number of average pooling kernels = 4 number of self-attention head = 8
TSO-S (Pct)	number of self-attention head: (transformer encoder layer) [2, 4, 8, 16]	$a_1 = 32, n_1 = 16, c_1 = 512$ $N_1 = 3, N_2 = 3$ number of average pooling kernels = 8 number of self-attention head = 16

trials were performed using the Bayesian optimization algorithm. The hyperparameter boundaries and best hyperparameters of SabCP are detailed in Table VII.

REFERENCES

1. J. Wang, Q. Fang, W. Zhu, T. Yang, J. Wang, H. Zhang, and H. Ni: *Metall. and Mater. Trans. B.*, 2024, vol. 55, pp. 1146–55.
2. R. Zhang, J. Yang, H. Sun, and W. Yang: *Int. J. Miner. Metall. Mater.*, 2024, vol. 31(3), pp. 508–17.
3. Z.C. Xin, J.S. Zhang, J.G. Zhang, J. Zheng, Y. Jin, and Q. Liu: *Metall. and Mater. Trans. B.*, 2023, vol. 54(3), pp. 1181–94.
4. K. Feng, D. He, A. Xu, and H. Wang: *Steel Res. Int.*, 2016, vol. 87(1), pp. 79–86.
5. K. Son, J. Lee, H. Hwang, W. Jeon, H. Yang, I. Sohn, Y. Kim, and H. Um: *J. Mater. Res. Technol.*, 2021, vol. 12, pp. 555–68.
6. A. Devlin, J. Kossen, H. Goldie-Jones, and A. Yang: *Nat. Commun.*, 2023, vol. 14(1), p. 2578.
7. M. Abdulsalam, M. Jacobs, and B.A. Webler: *Metall. and Mater. Trans. B.*, 2021, vol. 52, pp. 3970–85.
8. B.A. Webler and P.C. Pistorius: *Metall. and Mater. Trans. B.*, 2020, vol. 51, pp. 2437–52.
9. T. Alatarvas, T. Vuolio, E.P. Heikkinen, Q. Shu, and T. Fabritius: *Steel Res. Int.*, 2020, vol. 91(2), p. 1900424.
10. F. He and L. Zhang: *J. Process. Control.*, 2018, vol. 66, pp. 51–58.
11. Y. Kang, M.M. Ren, J.X. Zhao, L.B. Yang, Z.K. Zhang, Z. Wang, and G.J. Cao: *Mining Metall. Sect. B*, 2024, vol. 00, p. 8.
12. L. Fang, F. Su, Z. Kang, and H. Zhu: *Processes*, 2023, vol. 11(6), p. 1629.
13. Z. Wang, J. Chang, Q.-P. Ju, F.-M. Xie, B. Wang, H.-W. Li, B. Wang, X.-C. Lu, G.-Q. Fu, and Q. Liu: *ISIJ Int.*, 2012, vol. 52(9), pp. 1585–90.
14. W. Li, Q.M. Wang, X.S. Wang, and H. Wang: *Chem. Eng. Trans.*, 2016, vol. 51, pp. 475–80.
15. R. Wang, I. Mohanty, A. Srivastava, T.K. Roy, P. Gupta, and K. Chattopadhyay: *Metals*, 2022, vol. 12(5), p. 801.
16. K.X. Zhou, W.H. Lin, J.K. Sun, J.S. Zhang, D.Z. Zhang, X.M. Feng, and Q. Liu: *J. Iron. Steel Res. Int.*, 2021, vol. 29, pp. 751–60.
17. X. Shao, Q. Liu, Z. Xin, J. Zhang, T. Zhou, and S. Li: *Int. J. Miner. Metall. Mater.*, 2024, vol. 31(1), pp. 106–117.
18. J. Bae, Y. Li, N. Ståhl, G. Mathiason, and N. Kojola: *Metall. Mater. Trans. B*, 2020, vol. 51, pp. 1632–45.
19. I.J. Cox, R.W. Lewis, R.S. Ransing, H. Laszczewski, and G. Berni: *J. Mater. Process. Technol.*, 2002, vol. 120(1), pp. 310–15.
20. Z. Liu, S. Cheng, and P.P. Liu: *High Temp. Mater. Processes*, 2022, vol. 41(1), pp. 505–13.
21. M. Wang, C. Gao, X. Ai, B. Zhai, and S. Li: *ISIJ Int.*, 2022, vol. 62(8), pp. 1684–93.
22. M. Wang, S. Li, C. Gao, X. Ai, and B. Zhai: *Steel Res. Int.*, 2023, vol. 94(7), p. 2200872.
23. C. Gao, M. Shen, X. Liu, L. Wang, and M. Chen: *Trans. Indian Inst. Met.*, 2019, vol. 72, pp. 257–70.
24. J. Schlueter, H.J. Odenthal, N. Uebber, H. Blom, and K. Morik: Proc. Iron Steel Technol, Conf., 2013, pp. 923–28.
25. J. Duan, Q. Qu, C. Gao, and X. Chen: Chinese Control Conf., 2017, pp. 4507–11.
26. Y. Shao, M. Zhou, Y. Chen, Q. Zhao, and S. Zhao: *Optik*, 2014, vol. 125(11), pp. 2491–96.
27. Y. Shao, Y. Chen, Q. Zhao, M.C. Zhou, and X.Y. Dou: *Spectroscopy Spectral Anal*, 2015, vol. 35(11), pp. 3023–27.
28. F. Jiang, H. Liu, B. Wang, and X.F. Sun: *Comput. Eng.*, 2016, vol. 42(10), pp. 277–82.
29. J.W. Guo, D.P. Zhan, G.C. Xu, N.H. Yang, B. Wang, M.X. Wang, and G.W. You: *J. Iron. Steel Res. Int.*, 2023, vol. 30(5), pp. 875–86.
30. Y. Liang, H. Wang, A. Xu, and N. Tian: *ISIJ Int.*, 2015, vol. 55(5), pp. 1035–43.
31. M. Gu, A. Xu, H. Wang, and Z. Wang: *Processes*, 2021, vol. 9(11), p. 1987.
32. V. Borisov, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, and G. Kasneci: *IEEE Trans. Neural Networks Learn. Syst.*, 2022, vol. 35(6), pp. 7499–519.
33. L. Grinsztajn, E. Oyallon, and G. Varoquaux: *Adv. Neural. Inf. Process. Syst.*, 2022, vol. 35, pp. 507–20.
34. B. Lim and S. Zohren: *Phil. Trans. R. Soc. A*, 2021, vol. 379(2194), p. 20200209.
35. H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.A. Muller: *Data Min. Knowl. Disc.*, 2019, vol. 33(4), pp. 917–63.
36. W. Rawat and Z. Wang: *Neural Comput.*, 2017, vol. 29(9), pp. 2352–449.
37. H.T. Thai: *Structures*, 2022, vol. 38, pp. 448–91.
38. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, Q. Ma, Ye, and Y.T. Liu: *Adv Neural Inf. Process. Syst.*, 2017, vol. 30.
39. L. Prokhorenkova, G. Gusev, A. Vorobev, A.V. Dorogush, and A. Gulin: *Adv. Neural Inf. Process. Syst.*, 2018, vol. 31.
40. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, K. Łukasz, and I. Polosukhin: *Adv. Neural Inf. Process. Syst.*, 2017, vol. 30.
41. Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J.: arXiv preprint arXiv:2211.14730, 2022.
42. Oord, A. V. D., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Nal, K., Andrew. S. & Kavukcuoglu, K.: arXiv preprint arXiv:1609.03499, 2016.
43. J. Li, V. Lavrukhin, B. Ginsburg, R. Leary, O. Kuchaiev, J. M. Cohen, H. Nguyen, and R. T Gadde Jasper: In *Proceedings of Interspeech*, 2019.
44. W. Han, Z. Zhang, Y. Zhang, J. Yu, C.-C. Chiu, J. Qin, A. Gulati, R. Pang, and Y. Wu.: In *Proceedings of Interspeech*, 25 Oct 2020.
45. Y. Yu, Y. Zhang, Z. Song, Z. Song, and C. Tang: *Eng. Appl. Artif. Intell.*, 2023, vol. 126, p. 107079.
46. Zhum Z., and Soricut, R.: In *Proceedings of the 59th annual meeting of the association for computational linguistics*, 2021, vol. 1, pp. 3801–15.
47. T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin: *Int. Conf. Mach. Learn.*, 2022, vol. 162, pp. 27268–86.
48. H. Wang, J. Peng, F. Huang, J. Wang, J. Chen, and Y. Xiao: In *International Conference on Learning Representations*, 2022.
49. W. Hua, Z. Dai, H. Liu, and Q. Le: In *International Conference on Machine Learning*, 2022, pp. 9099–9117.
50. H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang: *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35(12), pp. 11106–15.
51. H. Wu, J. Xu, J. Wang, and M. Long: *Adv. Neural. Inf. Process. Syst.*, 2021, vol. 34, pp. 22419–30.
52. M. Liu, A. Zeng, M. Chen, Z. Xu, Q. Lai, L. Ma, and Q. Xu: *Adv. Neural. Inf. Process. Syst.*, 2022, vol. 35, pp. 5816–28.
53. A. Zeng, M. Chen, L. Zhang, and Q. Xu: *Proc. AAAI Conf. Artif. Intell.*, 2023, vol. 37(9), pp. 11121–28.
54. Gong, Z., Tang, Y., & Liang, J.: arXiv preprint arXiv:2310.00655, 2023.
55. T. Chen and C. Guestrin: In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–94.
56. S.Ö. Arik and T. Pfister: *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35(8), pp. 6679–87.
57. R. Shwartz-Ziv and A. Armon: *Inf. Fusion*, 2022, vol. 81, pp. 84–90.
58. C. Gao, M. Shen, X. Liu, L. Wang, and M. Chu: *Complexity*, 2019, vol. 2019(1), p. 7408725.
59. X. Wang, M. Han, and J. Wang: *Eng. Appl. Artif. Intell.*, 2010, vol. 23(6), pp. 1012–18.
60. L. Yang, H. Liu, and F. Chen: *Chemometrics Intell. Lab. Syst.*, 2022, vol. 231, p. 104679.
61. R. Zhang, J. Yang, S. Wu, H. Sun, and W. Yang: *Steel Res. Int.*, 2023, vol. 94(5), p. 2200682.
62. K.X. Zhou, W.H. Lin, J.K. Sun, J.S. Zhang, D.Z. Zhang, and X.M. Feng: *J. Iron. Steel Res. Int.*, 2022, vol. 29, pp. 751–60.
63. C. Shi, S. Guo, B. Wang, Z. Ma, C.L. Wu, and P. Sun: *Ironmaking Steelmaking*, 2023, vol. 50(7), pp. 857–66.