metals

MDPI

Article

Research on Molten Iron Quality Prediction Based on Machine Learning

Ran Liu, Zi-Yang Gao, Hong-Yang Li *, Xiao-Jie Liu and Qing Lv

College of Metallurgy and Energy, North China University of Science and Technology, Tangshan 063210, China; liuran@ncst.edu.cn (R.L.); 13914187870@163.com (Z.-Y.G.); xiaojie19851003@163.com (X.-J.L.); cronaldo0119@163.com (Q.L.)

* Correspondence: hblglhy@163.com

Abstract: The quality of molten iron not only has a significant impact on the strength, toughness, smelting cost and service life of cast iron but also directly affects the satisfaction of users. The establishment of timely and accurate blast furnace molten iron quality prediction models is of great significance for the improvement of the production efficiency of blast furnace. In this paper, Si, S and P content in molten iron is taken as the important index to measure the quality of molten iron, and the 989 sets of production data from a No.1 blast furnace from August to October 2020 are selected as the experimental data source, predicting the quality of molten iron by the I-GWO-CNN-BiLSTM model. First of all, on the basis of the traditional data processing method, the missing data values are classified into correlation data, temporal data, periodic data and manual input data, and random forest, the Lagrangian interpolation method, the KNN algorithm and the SVD algorithm are used to complete them, so as to obtain a more practical data set. Secondly, CNN and BiLSTM models are integrated and I-GWO optimized hyperparameters are used to form the I-GWO-CNN-BiLSTM model, which is used to predict Si, S and P content in molten iron. Then, it is concluded that using the I-GWO-CNN-BiLSTM model to predict the molten iron quality can obtain high prediction accuracy, which can provide data support for the regulation of blast furnace parameters. Finally, the MCMC algorithm is used to analyze the influence of the input variables on the Si, S and P content in molten iron, which helps the steel staff control the quality of molten iron in a timely manner, which is conducive to the smooth running of blast furnace production.

Keywords: steel; molten iron quality prediction; data processing; machine learning

check for updates

Citation: Liu, R.; Gao, Z.-Y.; Li, H.-Y.; Liu, X.-J.; Lv, Q. Research on Molten Iron Quality Prediction Based on Machine Learning. *Metals* **2024**, *14*, 856. https://doi.org/10.3390/met14080856

Academic Editor: Corby G. Anderson

Received: 11 June 2024
Revised: 10 July 2024
Accepted: 22 July 2024
Published: 26 July 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Steel is one of the basic industries of modern society and an important pillar industry of China's national economy, and blast furnace ironmaking is a crucial link in the whole steelmaking process [1]. Blast furnace ironmaking is a complex production process of converting solid iron ore into liquid molten iron through complex and changeable physical and chemical reactions in a closed environment of high temperature, high pressure and multi-phase and multi-field coupling [2]. The quality of the output of molten iron directly affects the quality and performance of the final steel products, and also affects the production cost and production efficiency [3]. The process of blast furnace iron making needs to undergo complex and changeable physical and chemical reactions. The trace elements of the furnace charge will be integrated into the molten iron with the reaction in the blast furnace. The proportion of some elements can reflect the quality of current molten iron. Si content is an important indicator reflecting the chemical heat of molten iron [4]. Appropriate Si content in molten iron can increase the amount of iron slag, which is conducive to the removal of phosphorus and sulfur in molten iron. However, a too high Si content will lead to hot-shortness of pig iron, reducing the metal yield and increasing the production of splash [5]. Si in molten iron is one of the harmful elements in steel production.

The Si content in molten iron dissolves in the form of sulfide in molten iron, but in solid iron the solubility of sulfide will become very low, forming co-crystal particles with a low melting point. When heated to a certain temperature, these co-crystal particles will melt first, giving the product hot and brittle characteristics, which will lead to the subsequent formation of cracks during casting and rolling, affecting the quality of the steel [6]. Iron phosphorus is also a harmful element in steel production. High P content will reduce the toughness and ductility of steel, leading to an increased risk of brittle fracture. At the same time, it will increase the cold brittleness of steel, making steel prone to brittle fracture at low temperatures and reducing the use-temperature range of steel [7]. Therefore, Si, S and P content in molten iron is usually used as an indicator of the quality of molten iron.

The quality parameters of molten iron can be used to directly to evaluate the quality of molten iron and indirectly reflect the internal operating state of the blast furnace. Therefore, it is crucial to predict the quality of molten iron [8]. At present, with the advent of the era of big data and the development of artificial-intelligence technology, intelligent algorithms and machine learning have been applied by more and more experts and scholars in the prediction research of molten iron quality. Cui et al. used the GRA-FCM model to divide the comprehensive characterization parameter set of furnace temperature, and they used the GWO-SVR model to gradually predict the state parameters of each delay point to realize the dynamic prediction Si content in molten iron [9]. Zhou et al. proposed a data-driven hybrid method combining typical correlation analysis and correlation analysis, using recursive subspace identification with a forgetting factor to establish a data-driven iron temperature prediction model and a predictive controller for online prediction and control of the multivariate iron temperature index [10]. Jiang et al. proposed a novel data-driven deep-model online prediction MIQ index, which could effectively explore the potential relationship between MIQ indicators, improve the interpretability of the model and reduce the dependence on labeled samples [11]. Chen et al. built a prediction model based on grey correlation analysis and a limit-learning machine, showing that the prediction hit rate of the model reached 87% [12]. Li et al. used the wavelet-TCN combination model to predict the vanadium content in iron and summarized the measures to improve the vanadium content of iron, which provided production guidance for the steel mill staff [13].

To sum up, it is not difficult to see that prediction research on molten iron quality has gained great achievements. However, most researchers focus on improving the prediction model, ignore the importance of data processing and do not provide methods on how to adjust the quality of molten iron. In fact, data sets with more approximate real value yields for prediction are used to gain higher prediction accuracy. In order to obtain a better data set, this paper uses different methods for different types of missing values. At the same time, a new prediction model is proposed to more accurately trace the Si, S and P content in molten iron. Finally, considering that the purpose of predicting the quality of molten iron is to better guide the production, the influence of each input variable on the content of Si, S and P in molten iron is analyzed, so as to help the steel mill staff control the quality of molten iron in a timely manner.

2. Algorithm and Model

2.1. Relevant Methods for Data Processing

2.1.1. Box-Plot

The box-plot is a statistical graph used to show the dispersion of a set of data. In statistics, there are three points to a set of small to large data, in the value of the 25% position, called the lower quartile (Q1), in the 50% position, called the median (Q2) and in the 75% position on the value, called the upper quartile (Q3). In addition, the negative difference between the lower four quartile to the upper four quartile is called the Interquartile Range (IQR) value in the Q3 + 1.5IQR, called the upper bound, the Q1 − 1.5IQR, called the lower bound and higher than the upper or below the lower value, called the outliers. In the process of blast furnace data collection, a large number of outliers will occur due to equipment failure, sensor failure, operation error, etc. In this study, to filter for outliers

the data set is drawn into a box-plot, and the data beyond the upper and lower bounds are deleted.

2.1.2. The Lagrangian Interpolation Formula

The Lagrange interpolation formula is a method used for the construction of interpolated polynomials [14]. In numerical analysis, the Lagrange interpolation method can construct a polynomial through these points through known data points, so as to approximate the relationship between these data points. The advantage of this method is that it can accurately fit a polynomial through a small number of data points to facilitate further calculation and analysis of [15]. Assuming that there are n data points, $(x_0, y_0), (x_1, y_1), \ldots, (x_{n-1}, y_{n-1})$, the Lagrangian interpolation method approximates the data point by constructing a polynomial, $L(x)$. The representation of $L(x)$ is shown in Equation (1). If the interval between the data points is large or the data points are evenly distributed, the Lagrange interpolated polynomial has little fluctuation between the data points. In the collected related data set, the periodic data has the above characteristics, so the Lagrange interpolation method was used to fill in the missing values of the periodic data in this study.

$$L(x) = \sum_{i=0}^{n-1} y_i \cdot \prod_{j=0}^{n-1} \frac{x - x_j}{x_i - x_j} \quad i \neq j \tag{1}$$

2.1.3. Random Forest

Random Forest (RF) is a machine-learning method proposed by Breiman in 2001 [16]. The random forest algorithm performs model training and prediction by constructing multiple decision trees [17]. In random forest, each decision tree is trained on a randomly selected subset of features to efficiently handle high-dimensional features and large-scale data sets [18]. Multiple unrelated decision trees form a decision tree forest and take the mean value of terminal nodes as the prediction value in the regression problem [19]. The principle of the algorithm is shown in Figure 1. When filling the missing values of the associated data collected during the blast furnace ironmaking process, two or more interrelated data can be characterized by one another. For each missing value, a random forest model was trained and the target variable was the missing feature. The non-missing data in the training set were used to train the model. Missing values were used for prediction using the trained model, and for each missing value other features of the same row were used as inputs to fill the values of the missing features by the model.

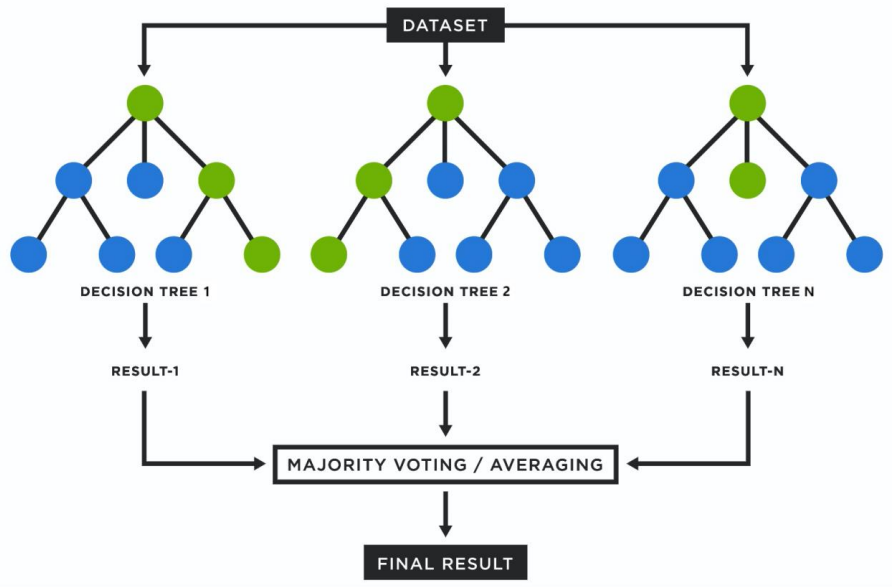

Figure 1. Structure of random forest.

2.1.4. K-Nearest Neighbor

K-Nearest Neighbor(KNN) is a machine-learning algorithm for supervised learning [20]. The basic idea of the algorithm is that, in the feature space of training set $D[(x_1, y_1), (x_2, y_2) \ldots (x_n, y_n)]$, the characteristic distance between each original sample and the sample x_k to be predicted will be calculated separately. The training sample with the highest feature similarity is selected to sum their label value, y_i, weighted to obtain the corresponding prediction result y_k; the principles of its algorithm are shown in Figure 2. Since the KNN method mainly relies on a limited number of K neighboring samples and avoids the method of discriminating class domains to determine the belonging category, the KNN method is more efficient for the set of class domains [21]. Considering that the periodic data collected in the process of blast furnace ironmaking will not change in the short term and the missing value is usually due to sensor failure, equipment maintenance or other reasons, steel operators usually use the value at the previous time point to fill the missing value. In this study, in order to process the data more objectively, the KNN algorithm can use the local neighborhood of data points to fill in the missing values of periodic data.

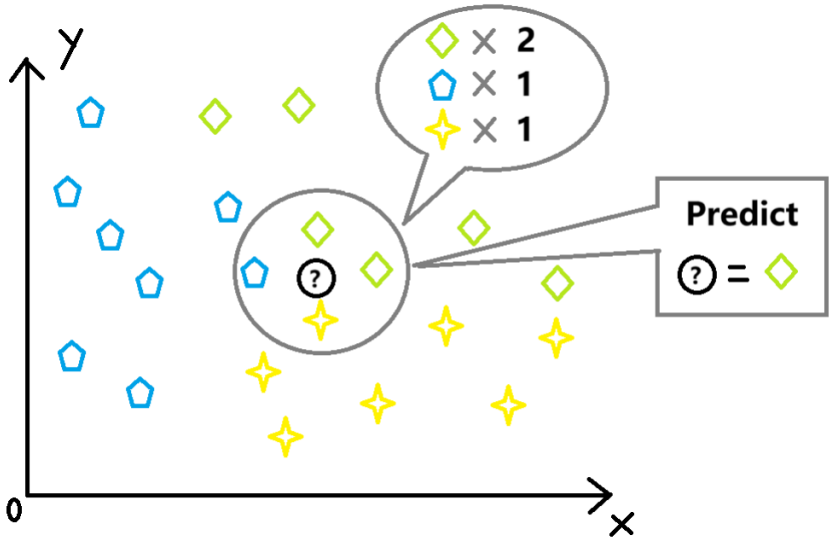

Figure 2. Principle of K-nearest neighbor.

2.1.5. Singular Value Decomposition

Singular value decomposition is a commonly used matrix decomposition technique, which is used to decompose a matrix into the product of three matrices. The specific steps are as follows: decompose the matrix X into three matrices U, S and V^T, where U is an orthogonal matrix of m $\times$ m, whose column is called the left singular vector. S is a diagonal matrix of m $\times$ n, whose elements on its diagonal are singular values. V^T is an orthogonal matrix of n $\times$ n, whose columns are known as the right singular vector. The decomposition principle of singular values is shown in Figure 3, and the form is shown in Formula (2). From Formula (2), the original matrix X can be reconstructed by the product of its left singular vector, singular value and right singular vector [22]. In the process of data processing, there are no specific distribution characteristics of manual input data, so it is difficult to accurately fill the missing value by conventional data filling means. Therefore, in this study, the data set extracted from the automation report of blast furnace ironmaking is used as the complete data matrix to obtain the matrix X. According to the eigenvalue and eigenvector reconstruction matrix, the estimated value in the reconstruction matrix is used to fill in the missing value.

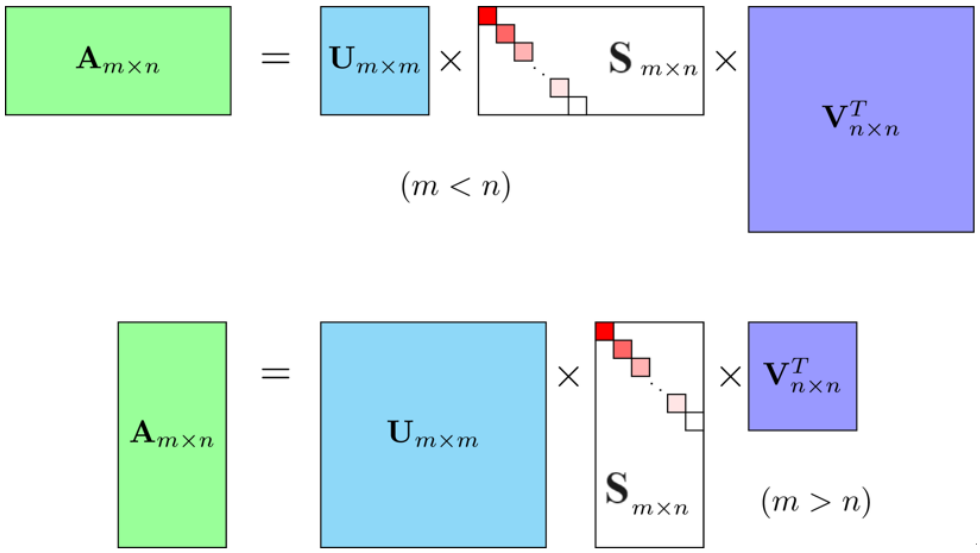

Figure 3. Principle of singular value decomposition.

$$
\begin{aligned}
&X = USV^T \\
&U = \begin{bmatrix} u_1 & u_2 & \cdots & u_m \end{bmatrix} \\
&S = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \cdots & \cdots & \ddots & \cdots \\ 0 & 0 & \cdots & \sigma_n \end{bmatrix} \\
&V = \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix}
\end{aligned}
\tag{2}
$$

where $\begin{bmatrix} u_1 & u_2 & \cdots & u_m \end{bmatrix}$ are the left singular vectors of X; these vectors are the eigenvectors of X^TX, which constitute the column of the orthogormal matrix U of m × m. $\begin{bmatrix} \sigma_1 & \sigma_2 & \cdots & \sigma_n \end{bmatrix}$ and are the singular values of X. They are the square root of the eigenvalues of X^TX and are arranged in descending order. S is a diagonal matrix of m × n. The $\begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix}$ are the right singular vectors of X, which are the eigenvectors of XX^T that constitute the column of the orthogonal matrix V of n × n.

2.1.6. Maximal Information Coefficient

Maximum Information Coefficient (MIC) is used to measure the degree of association between two variables, X and Y, the strength of linear or nonlinear, and is often used in the feature selection of machine learning [23]. The MIC idea is in view of the relationship between the two variables, which are discrete two variables in two-dimensional space, and they are representing by a scatter diagram. The current two-dimensional space in the x,y direction is respectively divided into certain interval numbers, and then the current scatter is represented on the grid, which is the joint probability calculation, thus solving the problem that the joint probability in mutual information is difficult to calculate [24]. In this study, the MIC algorithm was used to determine the time lag of the variables. A sliding window is set for each variable, which will be used to analyze the effect of the corresponding variable on the target at different time points. The change in the MIC value is observed with the window sliding; if the MIC lag peaks at a specific time, it indicates that this time node is the lag time of the variable relative to the target. The following formula is used to calculate the MIC:

$$
\begin{aligned}
I(x;y) &= \int p(x,y) log_2 \frac{p(x,y)}{p(x)p(y)} \mathrm{d}x\mathrm{d}y \\
MIC(x;y) &= \max_{a\cdot b<B} \frac{I(x;y)}{log_2 \min(a,b)}
\end{aligned}
\tag{3}
$$

In Formula (3), $p(x, y)$ is the variable between the joint probability. a, b is the number of partition grids in the x,y direction, which is essentially the grid distribution. B is the variable.

2.2. Improved Bidirectional Long- and Short-Term Memory Network Model

2.2.1. Bidirectional Long-Term and Short-Term Memory

Long-term and short-term memory (LSTM) is a recursive neural network (RNN) type, capable of learning the long-term dependencies in the data sequence [25]. Based on the RNN model, LSTM introduces gating mechanisms, such as input gate, output gate and forgetting gate, which partly avoids the problem of gradient explosion and gradient disappearance [26]. The flowchart of the LSTM prediction model is shown in Figure 4.

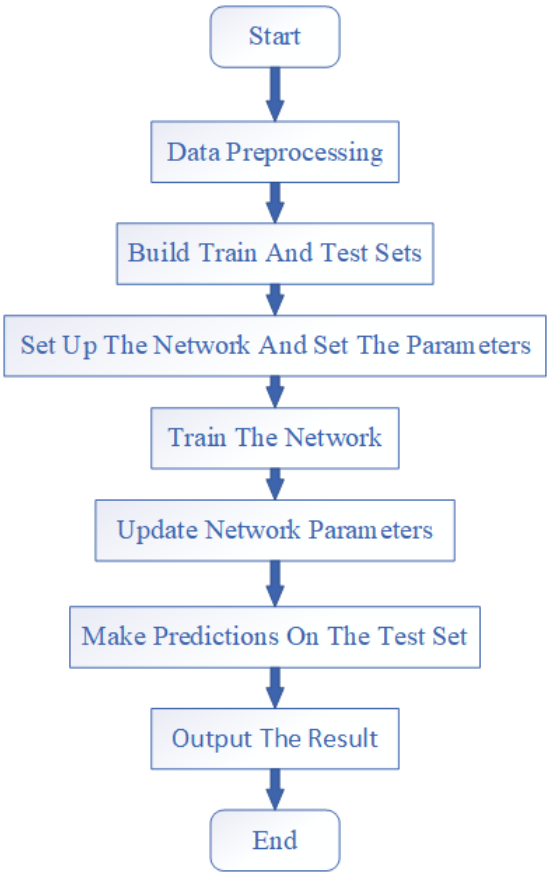

Figure 4. Process of LSTM.

Bidirectional long- and short-term memory (BiLSTM) is an extension of the traditional LSTM neural network. The BiLSTM network enhances the LSTM architecture by processing data in both forward and reverse directions [27]. This dual processing approach enables the network to acquire both past and future contextual information, enriching the model's understanding of the sequence and interpreting [28]. In BiLSTM, two separate LSTM layers are used: one processes the sequence from beginning to end, while the other processes the sequence in the opposite direction. The outputs of these two layers are then merged at each time step, usually by joining or summing, to produce a single output that reflects the past and future information [29]. The structure of its model is shown in Figure 5.

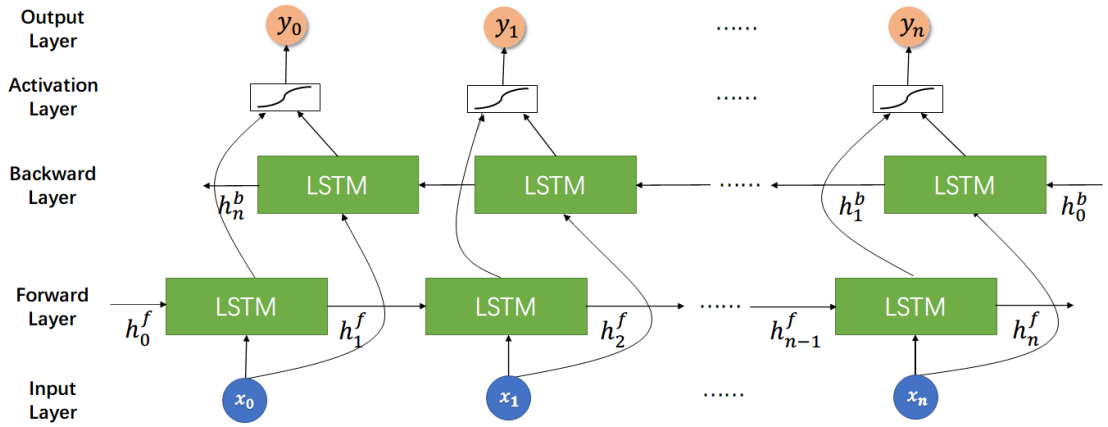

Figure 5. Structure of bidirectional long- and short-term memory.

2.2.2. Convolutional Neural Network

The Convolutional Neural Network (CNN) is a widely used neural network model for image and video processing tasks [30]. The CNN model usually consists of multiple convolutional layers, an activation function, a pooling layer and fully connected layers, trained by a back propagation algorithm [31]. During training, the CNN minimizes the loss function by adjusting the weight parameters to enable the model to better predict the target [32]. The structure of the CNN model is shown in Figure 6.

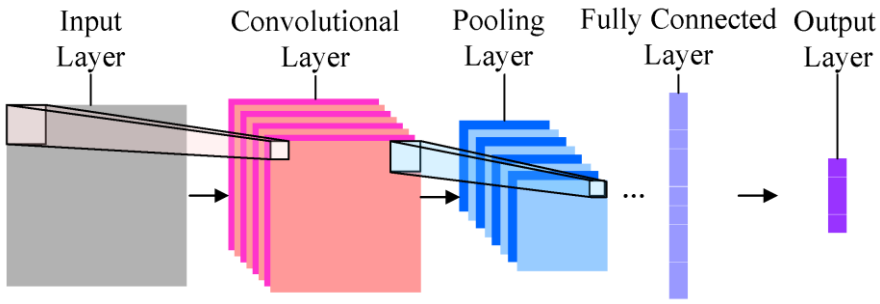

Figure 6. Structure of convolutional neural network.

2.2.3. Improved Grey Wolf Optimization Algorithm

The Grey Wolf Optimizer (GWO) algorithm has a simple parameter setting, and the principle is simple and easy to implement, but similar to other intelligent algorithms with population iteration, the GWO algorithm also has the disadvantages of low solution accuracy, and it is easy for it to fall into the local optimum [33]. The data related to blast furnace ingredients are complex, and the traditional GWO algorithm is often unsatisfactory due to premature convergence. In order to improve the optimal performance of the algorithm, an Improved Grey Wolf Optimizer (I-GWO) algorithm is proposed based on the traditional grey wolf algorithm to solve the single-objective optimization problem in this model.

(1) Tent-Sine 2-dimensional coupled chaotic map

The initialization of the population algorithm affects its search performance. With no prior information, individuals are often generated by random initialization. This strategy is sometimes useful. However, sometimes the distribution of individuals in the search domain is not uniform, which may keep individuals away from the global optimal solution, resulting in lower convergence rates. Chaos, characterized by ergodicity, randomness and regularity, is a common phenomenon in nonlinear systems, and using a chaotic variable search is obviously better than using a disordered random search. The tent chaotic map

has a simple structure, with good ergodic uniformity and a fast search speed, but there are small periods and unstable periodic points [34]. In order to avoid the tent chaotic sequence falling into small periodic points and unstable periodic points during iteration, the method of 2D-coupled sinusoidal mapping is used to improve the tent chaotic map, forcing sequences generated by the tent map to achieve a circular escape through small perturbations provided by the sinusoidal mapping, as shown in Equation (4).

$$\begin{aligned} x_{k+1} &= \begin{cases} (\lambda \frac{x_k}{a} + (1-\lambda)\sin(\pi y_k)) & x_k \in [0, a) \\ (\lambda \frac{1-x_k}{1-a} + (1-\lambda)\sin(\pi y_k)) & x_k \in [a, 1) \end{cases} \\ y_{k+1} &= \begin{cases} (\lambda \frac{y_k}{a} + (1-\lambda)\sin(\pi x_{k+1})) & x_k \in [0, a) \\ (\lambda \frac{1-x_k}{1-a} + (1-\lambda)\sin(\pi x_{k+1})) & x_k \in [a, 1) \end{cases} \end{aligned} \tag{4}$$

(2) Adaptive hunting weight coefficient

The main difference between the GWO algorithm and other heuristic algorithms is its social leadership hierarchy: the higher the rank the gray wolf has, the deeper understanding of its prey and the stronger leadership it has. This relationship plays a crucial role in group hunting during searching. However, in the original GWO hunting, the weight coefficient of the three leading wolves was the same, which obviously contrasted with the hierarchy of real wolves. Inspired by the mass update formula in the gravitational search algorithm, this study introduces Equation (5) to measure the importance of the three leading wolves as follows [35]:

$$\omega_i = \frac{3f_\gamma - f_i}{3f_\gamma - f_\alpha}, \theta_i = \frac{\omega_i}{\sum\limits_{j=\alpha,\beta,\gamma} \omega_j},\ i \in \{\alpha, \beta, \gamma\}. \tag{5}$$

In Formula (5), θ_i is the corresponding weight of each wolf. The closer the head wolf is to the prey, the higher the weight; the α wolf provides the main movement direction for the grey wolf community; the β wolf and γ wolf provide auxiliary directions to speed up the siege and attack of the prey.

(3) Hunting search strategy based on dimensional learning

In GWO, the four parameters α, β, γ and ω can find the optimum solution. However, it easily falls into local optimum due to lack of population diversity [36]. To avoid this problem, a novel nature-inspired algorithm, IGWO, is considered in this paper. It has three stages to operate: initializing, movement and selecting and updating.

In the initializing stage, N wolves are randomly distributed in between the limits (l_i, u_j) through Equation (6),

$$X_{ij} = l_j + rand_j[0,1] \times (u_j - l_j), i \in [l, N], j \in [l, D] \tag{6}$$

where the problem number of the dimensions is denoted by D and the total population is stored in a matrix with N number of rows and D number of columns. The fitness function is calculated by $f(x_i(t))$.

Movement stage individual hunting's are added to group hunting to improve the behavior of the wolves' hunting process, which is called Dimension Learning-based Hunting (DLH). In DLH, the behavior of individual hunting wolves learns from neighbor wolves for the updated position, $X_i(t)$. During this new learning, the updated position $X_i(t)$ is indicated by $X_{i-DLH}(t+1)$. For this individual hunting radius, $R_i(t)$ is calculated using Euclidean distance, as shown below. Moreover, it reduces premature convergence and balances the exploration and exploitation process.

$$R_i(t) = ||X_i(t) - X_{i-GWO}(t+1)|| \tag{7}$$

$$X_{i-DLH,d}(t+1) = X_{i,d}(t) + rand \times (X_{n,d}(t) - X_{r,d}(t)) \tag{8}$$

Selecting and updating stage: in this stage, the best one is selected by comparing the fitness of these two, $X_{i-GWO}(t+1)$ and $X_{i-DLH}(t+1)$, as shown in the equation below.

$$X_i(t+1) = \begin{cases} X_{i-GWO}(t+1), & f(X_{i-GWO}) < f(X_{i-DLH}) \\ X_{i-DLH}(t+1), & otherwise \end{cases} \quad (9)$$

Eventually, the iteration value is increased by one for all individual values, and the search is continued until the predetermined value is reached.

2.2.4. I-GWO-CNN-BiLSTM

Although the CNN model can convolute each time series and extract local features of the time series, it is not sensitive to time series features [37]. Therefore, the method of combining CNN and BiLSTM is adopted to make full use of the feature extraction ability of CNN and the sensitivity of BiLSTM to time series data to further improve the prediction effect of the model. Meanwhile, in order to improve the accuracy and robustness of the output of the network, the I-GWO algorithm is introduced to optimize the hyperparameters. The I-GWO-CNN-BiLSTM prediction model is finally formed, and its flow is shown in Figure 7.

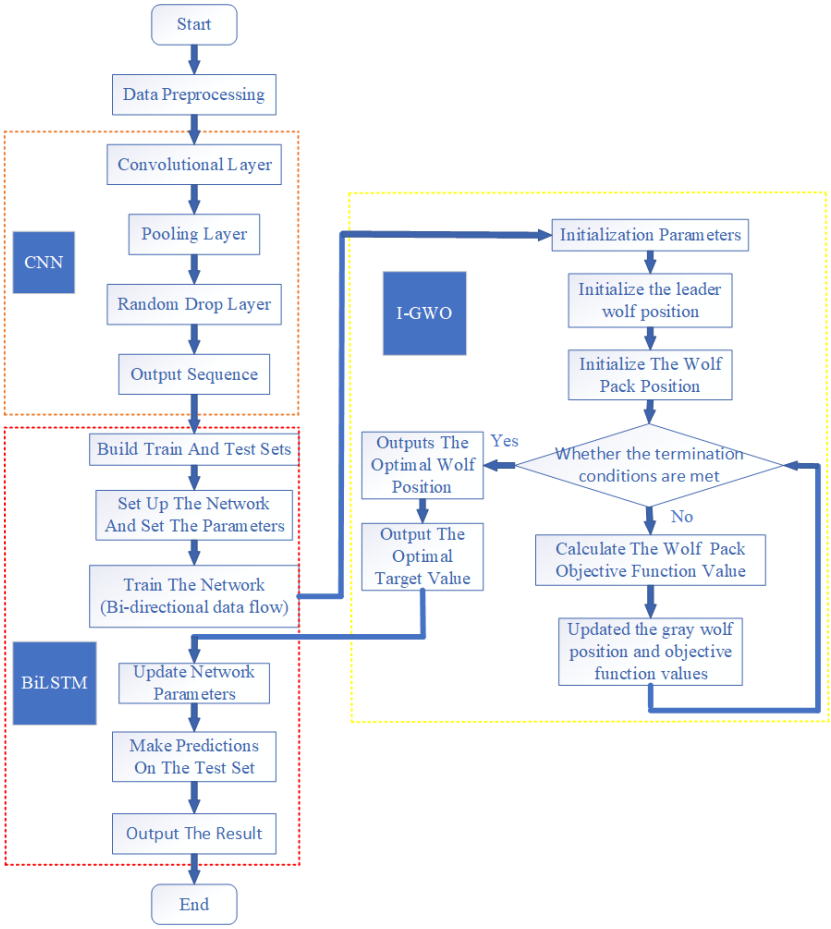

Figure 7. Process of IGWO-CNN-BiLSTM.

In the process of using the IGWO-CNN-BiLSTM model to predict the quality of molten iron, the CNN first integrates the input data, extracts features, denoises, and then passes these features as inputs to the BiLSTM network. The BiLSTM network uses its bidirectional memory ability to further process these features to capture long-term dependencies in time-series data. In the training stage of the BiLSTM network, the IGWO algorithm is used to determine the best structure of the network, including the number of LSTM layers, the

number of neurons in each layer, learning rate, etc. At the same time, the hyperparameters of the whole model are optimized, such as regularization coefficient, batch size, learning rate attenuation, etc., so as to improve the generalization ability and prediction accuracy of the model.

2.3. Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC), as a stochastic mathematical simulation method, has the advantages of strong versatility and intuitive simplicity [38]. This method is often used to simulate complex problems with random variables [39]. MCMC uses random simulation and statistical trials as means and produces a random numerical sequence that conforms to the probability distribution characteristics of the random variable, which is used as a method to conduct specific simulation tests and solve the sequence of human variables [40]. The specific steps are as follows: First, a probabilistic model is established that is consistent with the characteristics of the studied problem. Secondly, a sequence of random numbers is generated as a sampling input to the system for digital simulation trials in order to obtain a large number of simulated trial values. Finally, the results of the simulation test are processed statistically with calculated frequency and mean value, and the accuracy of the solution and settlement of the obtained problem is estimated [41]. The Markov chain assumes that the probability of a state transition at a given time depends only on its former state, which greatly simplifies the complexity of the model in the time-series model. In this study, the parameters related to Si, S and P content in molten iron were changed by some percentage, respectively, and the sensitivity was obtained by observing the change amplitude of Si, S and P content in molten iron. Since the relationship expression between the related parameters and Si, S and P content in molten iron cannot be directly obtained, it is difficult to obtain the effect of changing the related parameters on Si, S and P content in molten iron. At this time, the approximation can be simulated by using the Monte Carlo method.

3. Data Collecting and Processing

The data of this paper were obtained from a Hebei steel mill (hereinafter referred to as M steel) No.1 blast furnace production automation report from August to October 2020. The data this study required were selected from this report, including sinter SiO_2, pellet S, oxygen-rich pressure, oxygen-rich rate, cold air flow rate, hot air temperature, pressure, and 24 related parameters as input variables, and Si, P and S in molten iron as the output variable to predict the quality of the molten iron. To facilitate subsequent tabulating and mapping, the variables are renamed in English initials, as shown in Table 1.

Table 1. Abbreviation of each parameter.

Primitive Name	furnace body static pressure	thermal load	sinter P	pellet SiO_2	pellet S	top temperature northeast	top temperature southeast
Abbreviation	FBSP	TL	S P	P SiO2	P S	TTNE	TTSE
Primitive Name	top temperature northwest	top temperature southwest	oxygen enrichment stress	cold air pressure	hot air temperature	hot air pressure	top pressure
Abbreviation	TTNW	TTSW	OES	CAP	HAT	HAP	TP
Primitive Name	total soft water flow	oxygen enrichment flow	oxygen enrichment rate	cold air flow	blast kinetic energy	actual wind speed	differential pressure
Abbreviation	TSWF	OEF	OER	CAF	BKE	AWS	DP
Primitive Name	permeability index	theoretical combustion temperature	gas utilization	molten iron Si	molten iron S	molten iron P	
Abbreviation	PI	TCT	GU	MI Si	MI S	MI P	

To improve the accuracy of the prediction model, a reliable data set is crucial. However, the blast furnace data were collected and manually recorded through the sensors of each detection point, which leads to different types, time frequencies, structures and storage methods of the data. Therefore, according to the problems existing in the data, using big data technology for preprocessing, including dealing with missing values, abnormal values and other data problems, is an important guarantee to improve the data quality. The programming work of this study was done on a Legion R7000P with the operating system Windows 11, the development tool used was Matlab R2023a, and all applications were completely written by the authors.

3.1. Unified Frequency of Data

After the integration and analysis of the data collected by the No.1 blast furnace of M Steel, it can be seen that the update frequency of the data is related to the parameters. In consideration that most of the historical data stored in the database of M steel plants is hourly frequency data, the data frequency involved in the prediction model in the paper is selected as hourly frequency. The data amount of hour frequency is smaller than that of sec level and minute level frequency data, which is more in line with the field demand in the actual field application. Therefore, in order to ensure the unification of time frequency, the time frequency of data needs to be adjusted. Since the Si, S and P content in molten iron is recorded by furnace times, the time interval varies from 60 to 120 min. The data frequency was unified as follows: within 90 min was classified as the first 1 h and over 90 min was classified as the next 1 h; 989 data sets were obtained in 24 h processing.

3.2. Normalization Processing

Different types of data and the units between data are not uniform, which means that the order of data is different, so the data need to be normalized. Data normalization involves scaling the data to fall into a specific region, removing unit constraints and converting the data to dimensionless pure data. Data normalization facilitates the comparison and weighting between the parameters, accelerating the convergence rate of the model. The max–min method is easy to implement, it does not change the original information of the data, and it is easy to explain and understand. In this paper, the max–min method is used to normalize the data. The max–min method uses the maximum value and the minimum value in the data column for standardization. After standardization, the standardized value is between [0,1], and the calculation method is the difference between the data and the minimum value of the column, which is then divided by the extreme difference. The specific formula is:

$$x' = \frac{x - \min}{\max - \min} \tag{10}$$

3.3. Outlier Handling

The process of blast furnace ironmaking is complex and the production environment is harsh. External conditions such as high temperature, high pressure and high humidity can easily affect the sensitive sensing devices, which leads to interference and errors in data collection and transmission. Therefore, the outliers need to be removed to ensure the accuracy of the data. Considering that the data will be abnormal in the process of blowing out and blowing restore, the relevant data will be covered after understanding the specific time period of blowing out and blowing restore with experts from the steel mill. Data are filtered for outliers using the box-plot method, and the results are shown in Figure 8. Values beyond the upper and lower bounds of the box-plot are removed as outliers.

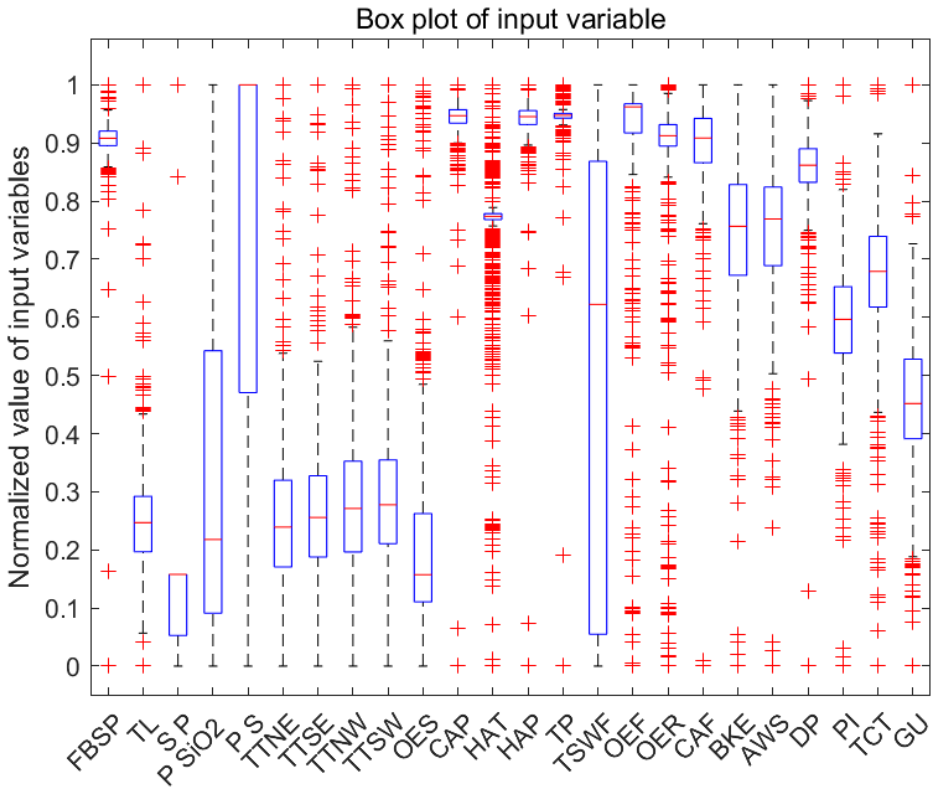

Figure 8. Results of box-plot.

3.4. Missing Values Processing

Due to sensor failure, manual input errors, abnormal values being eliminated and other reasons, many missing values have been generated in the data set. Only after the missing values have been completing can they be used for the training and testing of the prediction model. In this study, in order to make the filled data closer to the true value, the data were divided into correlation data, temporal data, periodic data and manual input data, and the most appropriate method was used to fill in the missing values in the data. In order to compare the effects of different methods on filling missing values, the evaluation indexes Root Mean Square Error (RMSE), coefficient of determination (R^2) and Mean Absolute Percentage Error (MAPE) are selected; the description of evaluation indicators can be seen in Table 2.

Table 2. Description of evaluation indicators.

Evaluation Indicators	Formula	Introduction
RMSE	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$	The range of RMSE is [0, +), and the smaller the value is, the smaller the prediction error of the model is.
R^2	$R^2 = 1 - \frac{\sum_{i}^{n}(\hat{y}_i - y_i)^2}{\sum_{i}^{n}(y_i^- - y_i)^2}$	The range of R^2 is [0,1], and the closer the value is to 1, the better the model fit is.
MAPE	$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left\|\frac{\hat{y}_i - y_i}{y_i}\right\|$	The range of MAPE is [0, +), and smaller values indicate that the prediction model has a better accuracy.

In Table 2, $\hat{y}_i$ indicates the true value,y_i indicates the predicted value, y_i^- represents the mean value of the true values and n represents the total number of data points.

3.4.1. Associative Data

The correlation data refers to the data points in the blast furnace process data. Because the fluctuation of some monitoring data points is closely related to the adjustment of blast furnace equipment or auxiliary equipment of the blast furnace, the monitoring point is extremely sensitive to the change of adjustment parameters. Therefore, the characteristics of this kind of data are a strong correlation. In order to intuitively reflect the characteristics of the correlation data, taking the air permeability index and the actual wind speed as an example, the line diagram of the data fluctuation is shown in Figure 9.

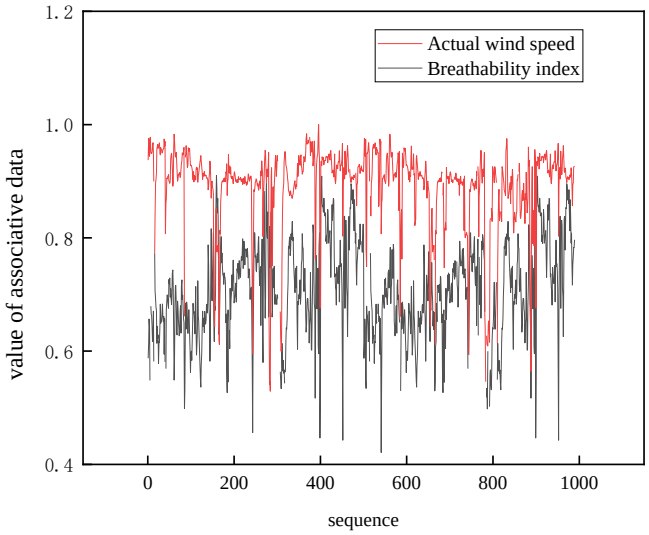

Figure 9. Comparison of the fluctuation situation of the air permeability index and the actual wind speed data.

As can be seen from Figure 9, the air permeability index has a strong synchronization with the actual wind speed data; this conclusion was confirmed by using the Pearson correlation analysis and seeing that their correlation coefficient was 0.86. The random forest algorithm learns through multiple decision trees to effectively deal with complex correlations between multiple features. When filling in the data, it often happens where some features have missing values. Random forest can effectively handle data with missing values. During the training process of each decision tree, some features are randomly selected for training, thus reducing the sensitivity to missing values. Therefore, this paper chooses to fill the missing value of association data using random forest, taking its filling of air permeability index as an example and comparing the random forest filling with the results of the AdaBoost method, as shown in Figure 10.

It can be seen from Figure 10 that the original data and the coincidence degree with the random forest algorithm is significantly higher than with the AdaBoost algorithm. In Table 3, the three evaluation indexes used to evaluate the coincidence degree of the filled data and the original data show that the random forest algorithm is better. Therefore, in this paper, the random forest algorithm is used to fill in the missing values of the correlation data, such as air permeability index and actual wind speed, cold wind flow rate and cold wind pressure and oxygen-rich flow rate.

Table 3. Comparison of results between random forest filling and AdaBoost filling.

Filling Method	RMSE	R^2	MAPE
Random Forest	0.631	0.941	18.4%
AdaBoost	1.39	0.713	33.9%

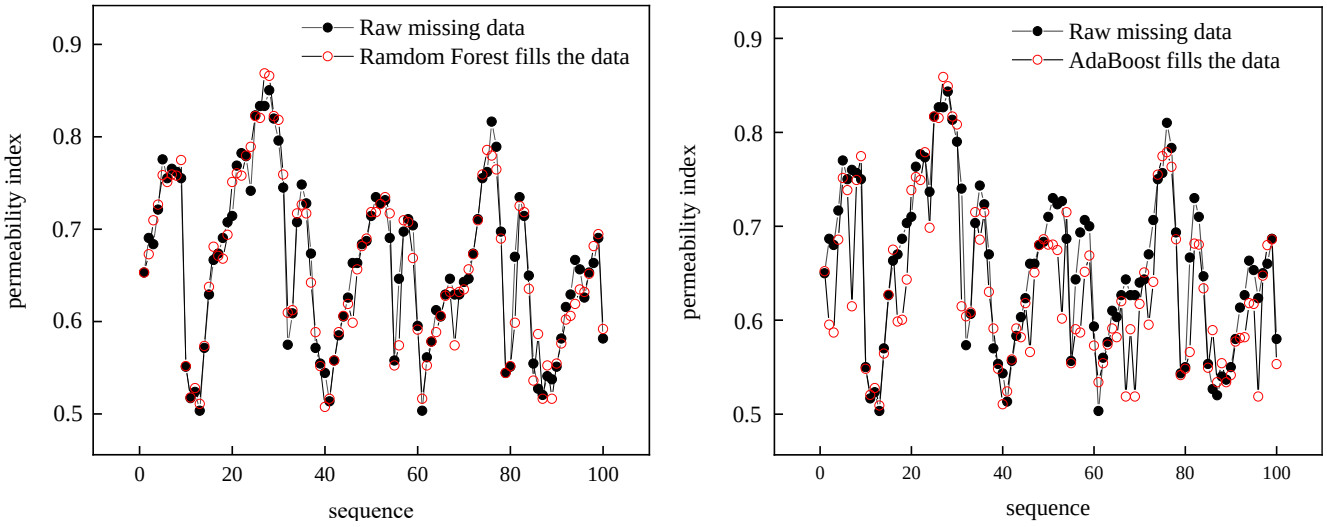

Figure 10. Comparison of results between Random forest filling and AdaBoost filling. (**a**) The effect of Random forest filling (**b**) The effect of AdaBoost filling.

3.4.2. Time Sequence Class Data

Timing sequence class data refers to the data in the blast furnace from the coupling factors of the blast furnace and the blast furnace production process of physical and chemical reactions, making the fluctuation of the data point range widen, leading to a lack of a strong correlation of related points and data fluctuations with continuity, meaning that the data before the state and the next moment state has strong connection. The main characteristics of this kind of data is that they have less instantaneous and violent fluctuations, and the continuity of the data is obvious. Taking the northeast of the top temperature as an example, the data fluctuation status is shown in Figure 11.

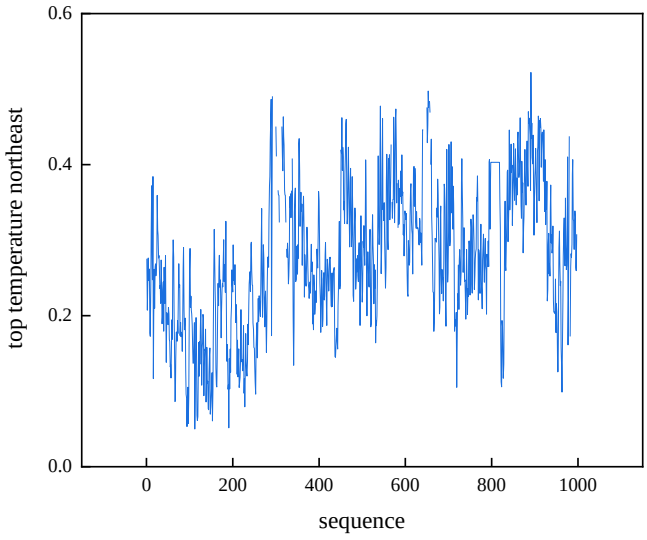

Figure 11. Top temperature northeast data volatility.

Figure 11 clearly shows that the timing data represented by the roof pressure is characterized by a small fluctuation range and strong continuity. Lagrangian interpolation can construct a polynomial function through known data points, which interpolates between data points and improves the accuracy of interpolation by increasing the number of data points. Therefore, this paper chooses to use the Lagrange interpolation method to fill in

the missing value of the temporal data, taking the missing value of the roof pressure as an example and comparing the results of the Lagrange interpolation method with the KNN filling, as shown in Figure 12.

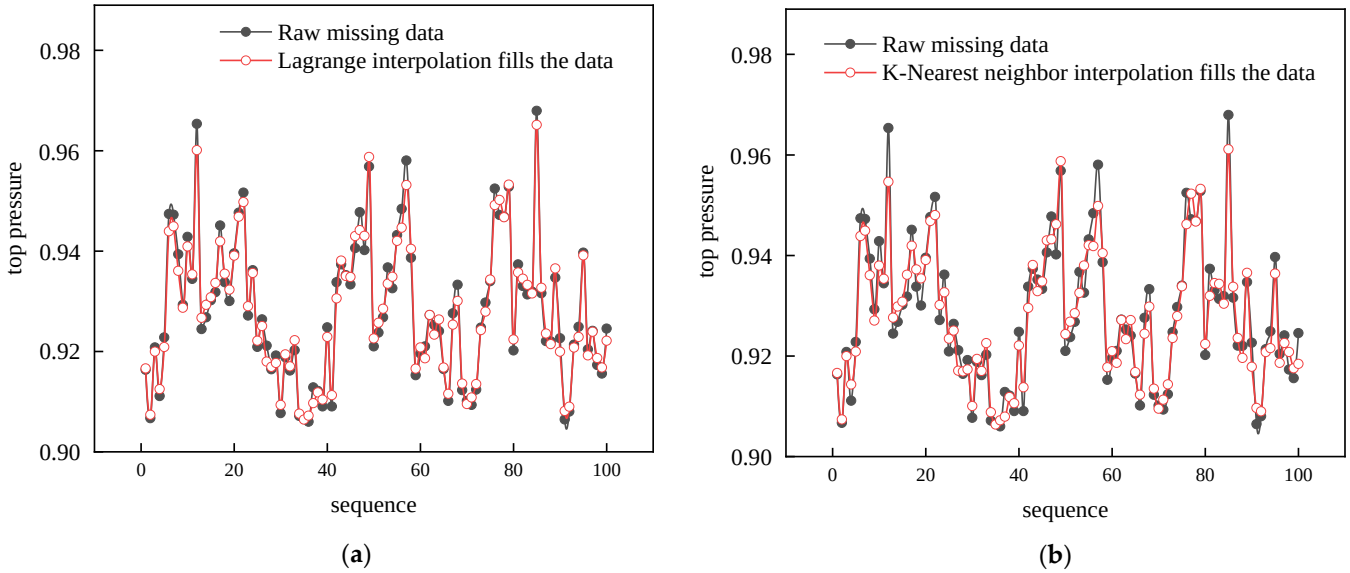

Figure 12. Comparison of results between Lagrange interpolation filling and K-nearest neighbor filling. (**a**) The effect of Lagrange interpolation filling (**b**) The effect of K-nearest neighbor filling.

It can be seen from Figure 12 that the original data and the coincidence degree with the Lagrangian interpolation is significantly higher than with the K-nearest neighbor interpolation. In Table 4, the three evaluation indexes used to evaluate the coincidence degree of the filled data and the original data show that the Lagrangian interpolation is better. Therefore, missing values of the temporal data, such as roof pressure, top temperature, total flow rate of soft water and so on, were filled using Lagrange interpolation.

Table 4. Comparison of results between Lagrange interpolation filling and K-nearest neighbor filling.

Filling Method	RMSE	R^2	MAPE
Lagrangian interpolation	3.92	0.982	1.84%
K-nearest neighbor interpolation	6.48	0.954	2.91%

3.4.3. Periodic Data

Periodic data refers to the data points of some non-temporal data in the blast furnace, which are counted for a certain cycle. For example, the data points of blast furnace material composition analysis, blast furnace iron output and slag output of the blast furnace are distributed in a certain range due to factors such as blast furnace volume and production capacity. Therefore, the classification of such data can be observed from the global analysis. Figure 13 is the data fluctuation of the SiO_2 content of pellet ore.

It can be seen from Figure 13 that the periodic statistical data does not have the continuity characteristics of temporal data and has more obvious periodic aggregation distribution characteristics, so it is more appropriate to choose the adjacent point mean filling method to fill the periodic statistical data. When filling in the cycle statistical data, the principle of proximity should be adopted, and the adjacent similar cycle data should be selected for backfilling. Therefore, the K-nearest neighbor algorithm can well meet the need of filling in the missing values for periodic data. In this paper, the K-nearest neighbor algorithm is selected to fill in the missing value of the periodic data, taking the missing value of SiO_2 content of pellet ore as an example and comparing the KNN algorithm with the results of the linear interpolation method, as shown in Figure 14.

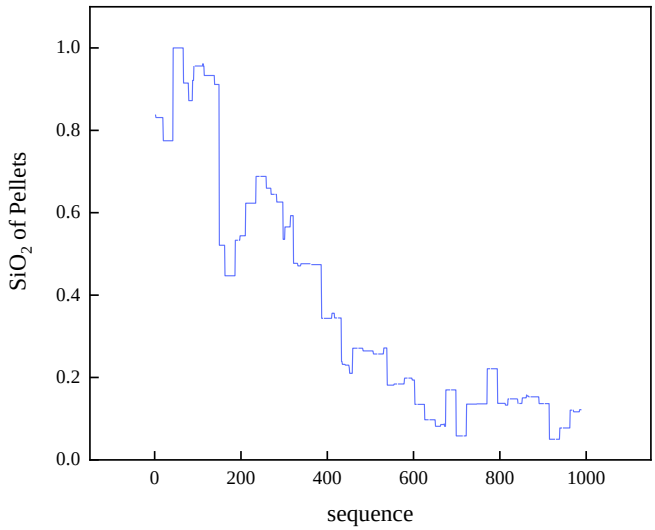

Figure 13. Fluctuation of the SiO_2 content data of the pellet.

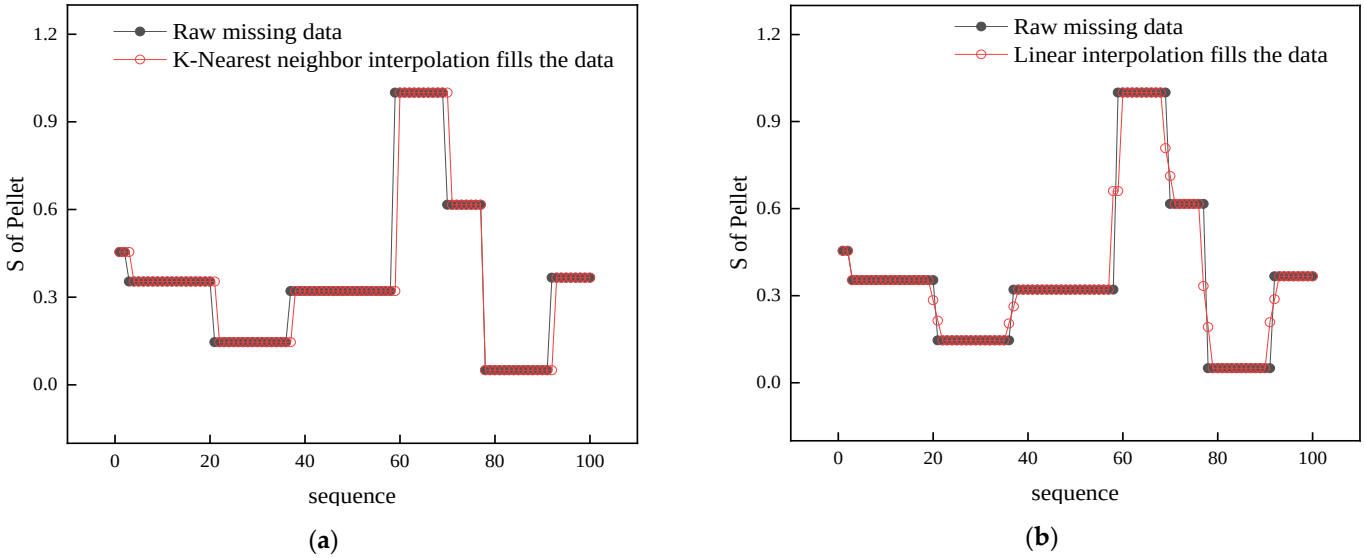

Figure 14. Comparison of the results of K-nearest neighbor filling and Linear interpolation filling. (**a**) The effect of K-nearest neighbor filling (**b**) The effect of Linear interpolation filling.

It can be seen from Figure 14 that the original data and the coincidence degree with the K-nearest neighbor interpolation is significantly higher than with the linear interpolation. In Table 5, the three evaluation indexes used to evaluate the coincidence degree of the filled data and the original data show that the K-nearest neighbor interpolation is better. Therefore, K-nearest neighbor is used to fill in periodic data, such as the P content of sinter, the SiO_2 content of pellets and the S content of pellets.

Table 5. Comparison of the results of K-nearest neighbor filling and linear interpolation filling.

Filling Method	RMSE	R^2	MAPE
K-nearest neighbor interpolation	0.131	0.592	5.02%
Linear interpolation	0.167	0.756	13.8%

3.4.4. Manual Entry of Data

Considering that the Si, S and P content parameters in molten iron are input into the system after manual testing, this situation will lead to the large spacing of missing values

of Si, S and P content in the automated report. Figure 15 is the data fluctuation of Si content in molten iron.

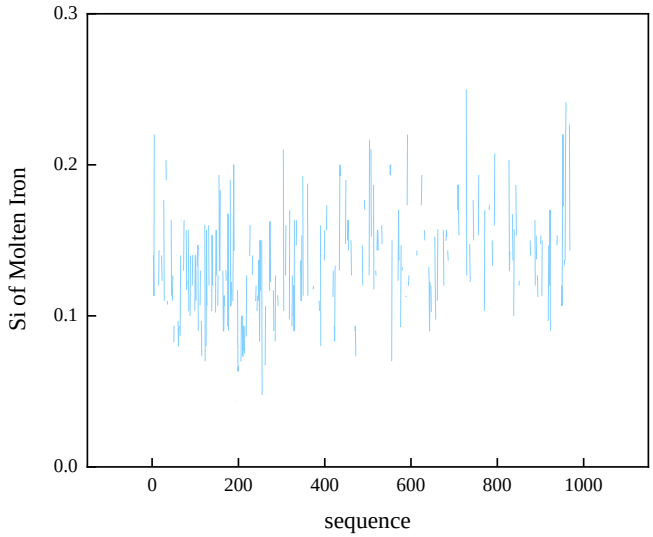

Figure 15. Fluctuation of Si content in molten iron.

It is not difficult to see from Figure 15 that it is easy to cause great errors if using the conventional data filling method. The SVD algorithm is essentially a matrix factorization technique, so it is particularly suitable to fill in the missing values stored in the matrix form. By preserving the largest singular values and the corresponding singular vectors, SVD is able to reconstruct the main structure of the data set and thus effectively fill in the missing values. In this paper, the SVD algorithm is selected to fill in the missing values of manually input data, taking the missing values of Si content in molten iron as an example, and comparing the SVD algorithm with the linear interpolation method, as shown in Figure 16.

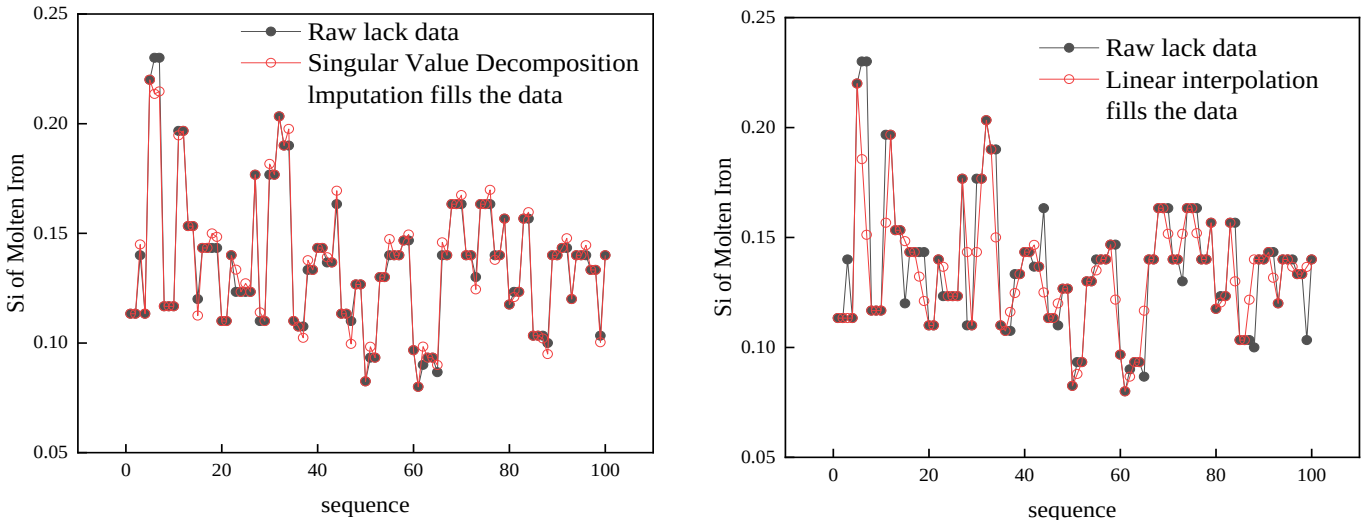

Figure 16. Comparison of the results of Singular value decomposition filling and Linear interpolation filling. (**a**) The effect of Singular value decomposition filling (**b**) The effect of Linear interpolation filling.

It can be seen from Figure 16 that the original data and the coincidence degree with the singular value decomposition interpolation is significantly higher than with the linear

interpolation. In Table 6, the three evaluation indexes used to evaluate the coincidence degree of the filled data and the original data show that the singular value decomposition interpolation is better. Therefore, this paper uses the SVD method to fill in the manual input data, such as Si, S and P content in molten iron.

Table 6. Comparison of the results of singular value decomposition filling and linear interpolation filling.

Filling Method	RMSE	R^2	MAPE
Singular Value Decomposition interpolation	0.00381	0.983	1.38%
Linear interpolation	0.0155	0.744	4.90%

3.5. Data Dimension Reduction

Considering that excessive input variables easily lead to over-fitting of the prediction model, the correlation coefficient is reduced by Pearson, and the nonlinear correlation is verified by the gray correlation method; the heat map is shown in Figure 17.

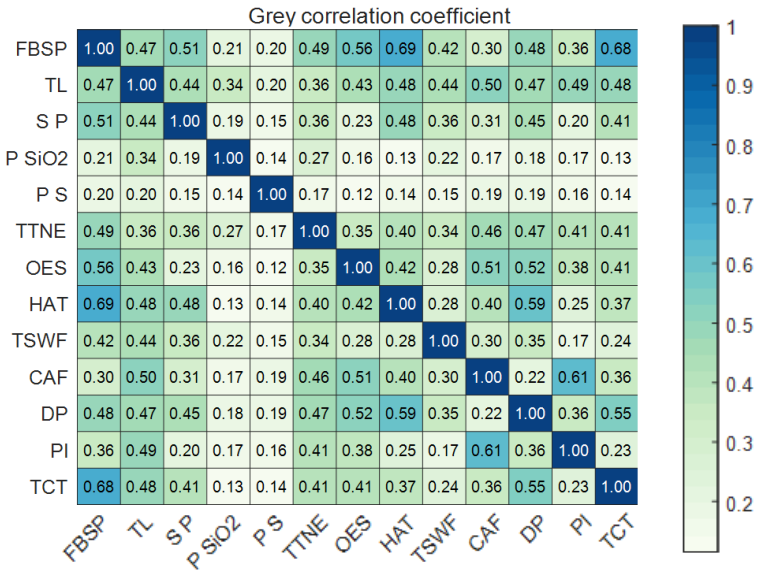

Figure 17. Heatmap.

Figure 17 shows that there is no set of variables with a correlation exceeding 0.9, indicating that another dimension reduction is not required. The input variables for the final prediction model were FBSP, TL, SP, P SiO_2, PS, TTNE, OES, HAT, TSWF, CAF, DP, PI, TCT.

3.6. Determine the Time Lag

As a typical process production enterprise, the stable production of blast furnace smelting is a continuous process, relying on the cooperation of various departments. From the transportation of raw fuel in the field, to the charging of the furnace, to the transmission of sensor information of various departments, and then to the occurrence of various physical and chemical reactions in the furnace, production is carried out according to established technological processes. There is continuity in the changes in the process, but due to the complexity of the blast furnace production itself the operation state of the blast furnace is stable at the previous moment and the operation state of the blast furnace begins to fluctuate at the next moment. The state at this time will have an impact on the subsequent state, and there is obvious data timing. In order to better predict Si, S and P content in molten iron, the corresponding lag time of each input variable needs to be determined in advance. The MIC algorithm can better analyze the specific lag time of each variable,

taking the process of determining the Si content in molten iron as an example. The lag time of each variable is finally confirmed, as shown in Table 7.

Table 7. Time lag analysis.

Variable	1t	2t	3t	4t	5t	6t	7t	8t	Lag Time
FBSP	**0.166**	0.148	0.151	0.146	0.150	0.155	0.154	0.152	1
TL	**0.146**	0.136	0.134	0.145	0.142	0.145	0.140	0.145	1
S P	0.198	0.198	**0.199**	0.197	0.198	0.198	0.196	0.196	3
P SiO_2	0.212	0.217	0.218	0.219	0.222	0.225	0.234	**0.240**	8
P S	0.112	0.107	0.108	0.115	0.112	0.117	0.126	**0.132**	8
TTNE	0.156	0.145	0.144	0.145	0.147	**0.160**	0.152	0.151	6
OES	0.182	0.188	0.201	0.195	**0.205**	0.201	0.202	0.185	5
HAT	**0.151**	0.140	0.131	0.130	0.136	0.125	0.139	0.122	1
TSWF	0.224	**0.228**	0.213	0.217	0.216	0.217	0.216	0.219	2
CAF	0.162	0.169	0.170	0.184	**0.193**	0.181	0.185	0.170	5
DP	**0.159**	0.151	0.145	0.135	0.137	0.150	0.152	0.142	1
BI	0.178	0.183	**0.191**	0.174	0.167	0.176	0.175	0.157	3
TCT	0.149	0.127	**0.151**	0.124	0.121	0.139	0.148	0.123	3
MI Si	0.315	0.166	0.138	0.151	0.126	0.123	0.129	**0.116**	8

From Table 7, the specific lag time of each variable can be determined according to the influence of different time delays on Si content in molten iron, the time node with the greatest influence of each variable on the Si content in molten iron has been bolded in the table (In addition to the effect of Si content on itself). The above process is repeated to obtain the different lag times of each variable for Si, S and P content in molten iron, as shown in Table 8.

Table 8. The time lag for each parameter.

Content in Molten Iron	FBSP	TL	S P	P SiO_2	P S	TTNE	OES	HAT
Si	1	1	3	8	8	6	5	1
S	5	7	5	7	7	8	4	1
P	4	7	3	2	8	1	7	4
Content in Molten Iron	**TSWF**	**CAF**	**DP**	**BI**	**TCT**	**MI Si**	**MI S**	**MI P**
Si	2	5	1	3	3	8	/	/
S	1	5	8	8	5	/	8	/
P	1	4	6	8	8	/	/	8

4. Results and Discussion

4.1. Quality Prediction of Molten Iron

The application of intelligent algorithms and machine-learning-related technology to predict the quality of molten iron can enable the field staff to obtain the changes of molten Si, S and P content in the blast furnace in advance, so as to provide the basis for the subsequent adjustment. In this paper, 989 production data of an M Steel 1 blast furnace from August to October 2020, using the I-GWO-CNN-BiLSTM model for prediction, were selected as the source of experimental data, as follows: FBSP, TL, SP, P SiO2, PS, TTNE, OES, HAT, TSWF, CAF, DP, BI and TCT as input variables. Mi Si, MI S and MI P as output variables. In order to verify the superiority of the I-GWO-CNN-BiLSTM model, the evaluation indexes RMSE, R^2 and MAPE were selected.

4.1.1. Model Training

In this paper, 70% of the 989 data sets were selected to train the IGWO-CNN-BiLSTM model, which was compared with the traditional LSTM prediction model. In order to facilitate the reader to compare the data with those in the actual production, the prediction results

were inversely normalized. The prediction result pairs are shown in Figure 18 and the performance pairs are shown in Table 9.

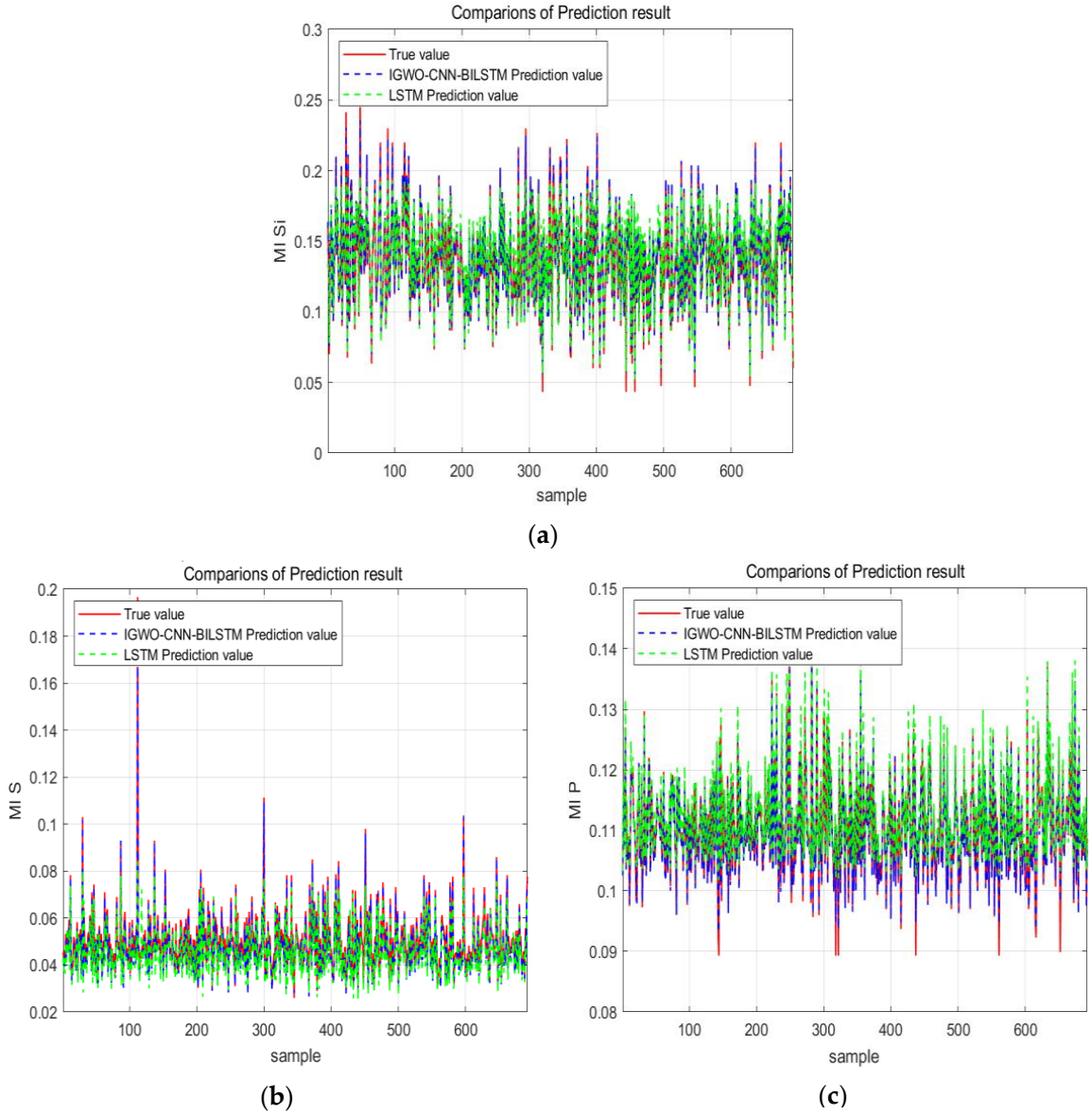

Figure 18. Comparison of the prediction results at model training. (**a**) the model prediction result of MI Si (**b**) the model prediction result of MI S (**c**) the model prediction result of MI P.

Table 9. Comparison of model performance.

Content in Molten Iron	I-GWO-CNN-BiLSTM			LSTM		
	RMSE	R^2	MAPE	RMSE	R^2	MAPE
MI Si	0.00231	0.995	1.35%	0.00977	0.916	5.81%
MI S	0.00248	0.962	4.58%	0.00714	0.691	9.33%
MI P	0.000887	0.987	0.534%	0.00299	0.859	2.06%

From Figure 18, it can be seen that the prediction results of the I-GWO-CNN-BiLSTM model can basically cover the true value, while the prediction results of the LSTM model can only cover part of the true value or beyond the range of the true value. It can be seen from Table 4 that, when predicting the Si, S and P content in molten iron, The RMSE values

of the I-GWO-CN-BiLSTM model decrease by 0.00746, 0.00466 and 0.00211 compared with the LSTM model. The R^2 value of the I-GWO-CNN-BiLSTM model increases by 0.0797, 0.272 and 0.128, respectively, compared with the LSTM model. The MAPE values of the I-GWO-CNN-BiLSTM model decrease by 4.46%, 4.75%, and 1.53%, respectively, compared with the LSTM model. In conclusion, the I-GWO-CNN-BiLSTM model has a superior performance and enables a better prediction of molten iron mass.

4.1.2. Model Testing

In this paper, the remaining 30% of 989 data sets is selected as the test set of the I-GWO-CNN-BiLSTM model for testing the model and comparing it with the traditional LSTM prediction model. In order to facilitate the reader to compare the data in the actual production with the prediction results, the prediction results were inversely normalized. The prediction results are shown in Figure 19, the prediction errors are shown in Figure 20 and the performance pairs are shown in Table 10.

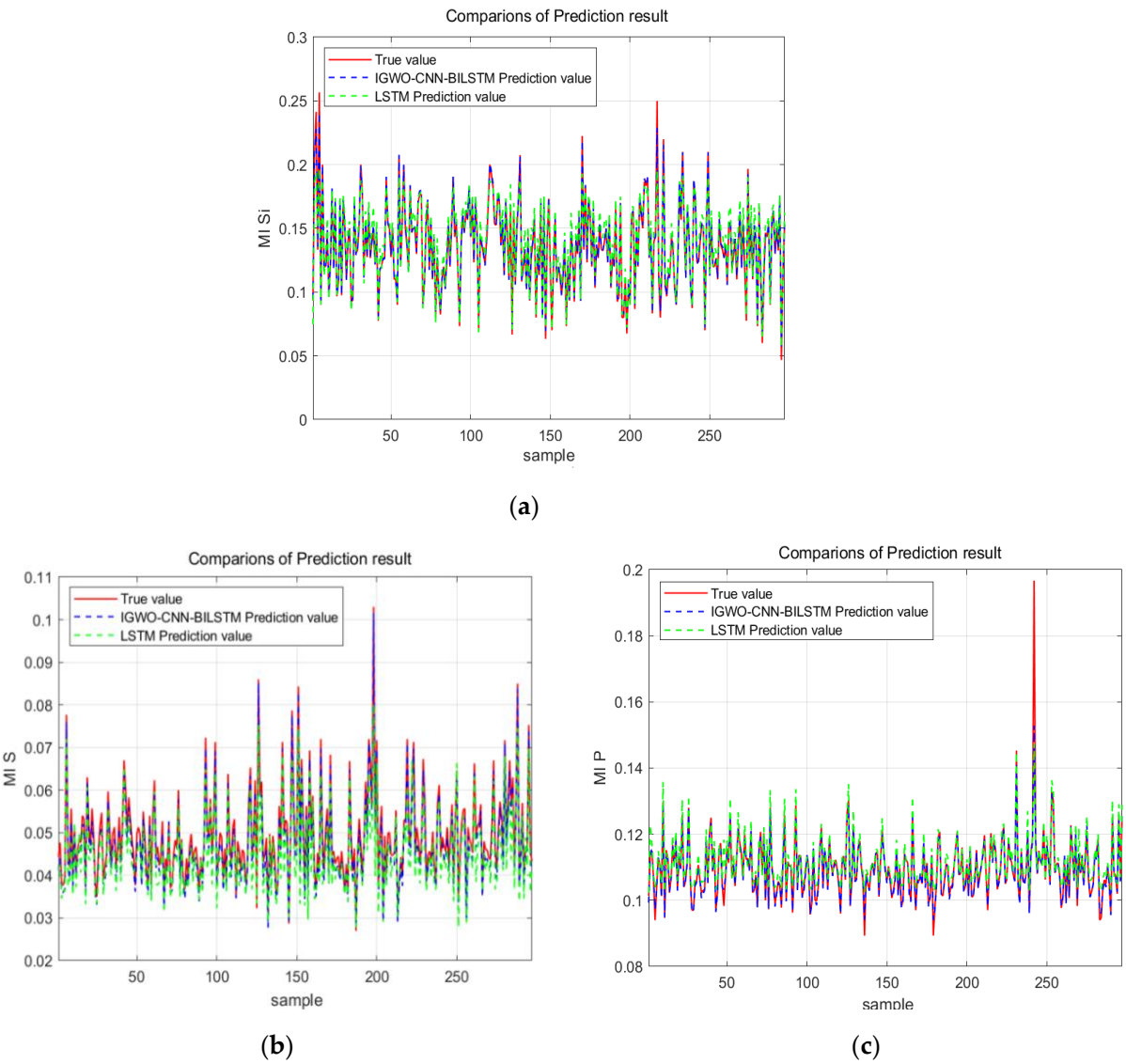

Figure 19. Comparison of the prediction results at model testing. (**a**) the model prediction result of MI Si (**b**) the model prediction result of MI S (**c**) the model prediction result of MI P.

According to Figure 19, we can judge the prediction effect of the two models by observing the coverage of the predicted value of IGWO-CNN-BiLSTM (blue dotted line)

and the predicted value of LSTM (green dotted line) on the true value (red solid line). In Figure 19a, the predicted values of IGWO-CNN-BiLSTM and LSTM evenly matched the true coverage at the [0.05,0.18] interval, but only the predicted value of IGWO-CNN-BiLSTM effectively covers the true values at the [0.18,0.25] interval. In Figure 19b, the predicted value of IGWO-CNN-BiLSTM was able to effectively cover the true value at the interval of [0.02,0.11], while the predicted value of LSTM was mostly beyond the true range at the interval of [0.02,0.05], and it failed to cover the true value at [0.05,0.11]. In particular, there is no prediction value of LSTM at the [0.08,0.11] interval. In Figure 19c, most of the predicted values of LSTM cannot effectively cover the true value at the [0.09,0.11] interval and are beyond the range of the true value at the [0.11,0.14] interval. Although the predicted value of IGWO-CNN-BiLSTM cannot effectively cover the true value at the [0.15,0.2] and [0.09,0.095] intervals, the coverage still exceeds LSTM for the true value. In conclusion, I-GWO-CNN-BiLSTM is significantly better for the tracking of Si, S and P content in molten iron than the LSTM model.

Table 10. Model performance comparison.

Content in Molten Iron	I-GWO-CNN-BiLSTM			LSTM		
	RMSE	R^2	MAPE	RMSE	R^2	MAPE
MI Si	0.00271	0.989	1.41%	0.0106	0.905	5.89%
MI S	0.00229	0.952	4.61%	0.00539	0.734	8.85%
MI P	0.00271	0.921	0.652%	0.00459	0.775	2.63%

As can be seen in Table 10, when predicting the Si, S and P content of molten iron, The RMSE index of the I-GWO-CNN-BiLSTM model is better than the RMSE index of the LSTM model, and is close to 0, which indicates that the I-GWO-CNN-BiLSTM model has a high prediction accuracy. The R^2 index of the I-GWO-CNN-BiLSTM model is better than the R^2 index of the LSTM model, and is close to 1, which indicates that the I-GWO-CNN-BiLSTM model has a high degree of fit. The MAPE index of the I-GWO-CNN-BiLSTM model is better than the MAPE index of the LSTM model, and is close to 0, which indicates that the I-GWO-CNN-BiLSTM model has low error. In conclusion, the use of the I-GWO-CNN-BiLSTM model can accurately predict the quality of molten iron, so as to provide important data support for steel mill staff and provide important guarantees for blast furnace production.

In the above, we found that the IGWO-CNN-BiLSTM model has better prediction accuracy, but it still has prediction errors. In fact, it is inevitable that the prediction models will produce errors. The specific reasons for this are as follows: Firstly, some of the data in the data set used to train the prediction model were obtained from filling in and were not all real data from the steel mill. Secondly, too much selection of features can easily lead to overfitting of the model, while too little selection of features can easily lead to underfitting of the model, and it is difficult to achieve perfect feature selection. Thirdly, the model used cannot well capture the intrinsic correlation between the independent variables and the dependent variables. Finally, training the model with too little data can also lead to large prediction errors. However, the prediction errors can not only be reduced by using the superior prediction model but can also be reduced by using the data set that more closely fits the actual value. In this paper, the accurately processed data and the roughly processed data in Section 3.4 were used to train the same model (IGWO-CNN-BiLSTM) and the evaluation index was used to evaluate the performance of the model, and the comparison results are shown in the table below.

It can be seen from Table 11 that the model trained by the accurately processed data has a higher level of accuracy in predicting Si, S and P content in molten iron. In order to more intuitively reflect the gap between the predicted value and the true value, the error of training the IGWO-CNN-BiLST model by accurately processed data for predicting Si, S and P content in molten iron is plotted in Figure 20.

Table 11. Comparison of model performance after training using accurate and rough processed data.

Content in Molten Iron	Accurate Processing			Rough Processing		
	RMSE	R^2	MAPE	RMSE	R^2	MAPE
MI Si	0.00271	0.989	1.41%	0.00467	0.965	1.89%
MI S	0.00229	0.952	4.61%	0.00369	0.937	5.85%
MI P	0.00271	0.921	0.652%	0.00349	0.913	0.932%

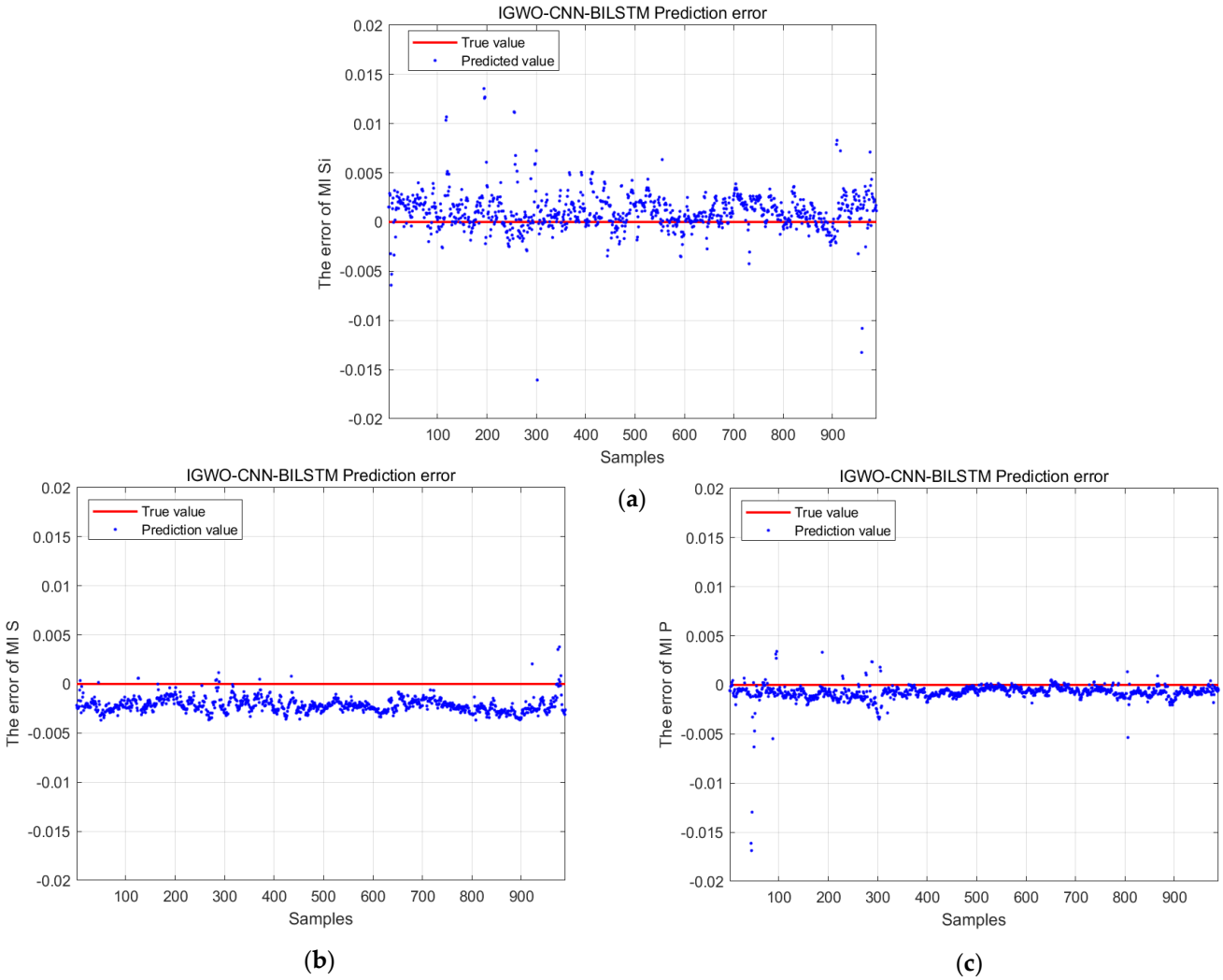

Figure 20. Error of the prediction. (**a**) The error of the MI Si prediction (**b**) The error of the MI S prediction (**c**) The error of the MI P prediction.

As can be seen from Figure 20, when using the I-GWO-CNN-BiLSTM model, the error is concentrated in the range of [−0.05,0.05] and [−0.05,0]. For P content in molten iron, the error is concentrated in the range of [−0.03,0.01]. It is not difficult to see that this study ensures that the prediction error is controlled within a small range through reasonable feature selection, more refined data processing, a superior prediction model, and an appropriate amount of data for the training and testing of the model, which makes the guiding significance of this study for the actual production more significant.

4.2. Parameter Regulation

After the accurate prediction of the quality of molten iron, if the expected target of the steel mill is not reached, it is necessary to regulate the Si, S and P content in molten

iron in a timely manner to ensure that the final qualified products can be obtained. In the molten iron mass prediction model, the effects of different input variables on the Si, S and P content in molten iron vary. Because the MCMC algorithm can conduct sensitivity analysis through random sampling, there is no need to repeat the model, with obvious advantages in computational efficiency. By using the sensitivity of the parameters, the parameter increases by 5% for the influence of the Si, S and P content of molten iron, as shown in Figure 21, and the specific values, as shown in Table 12.

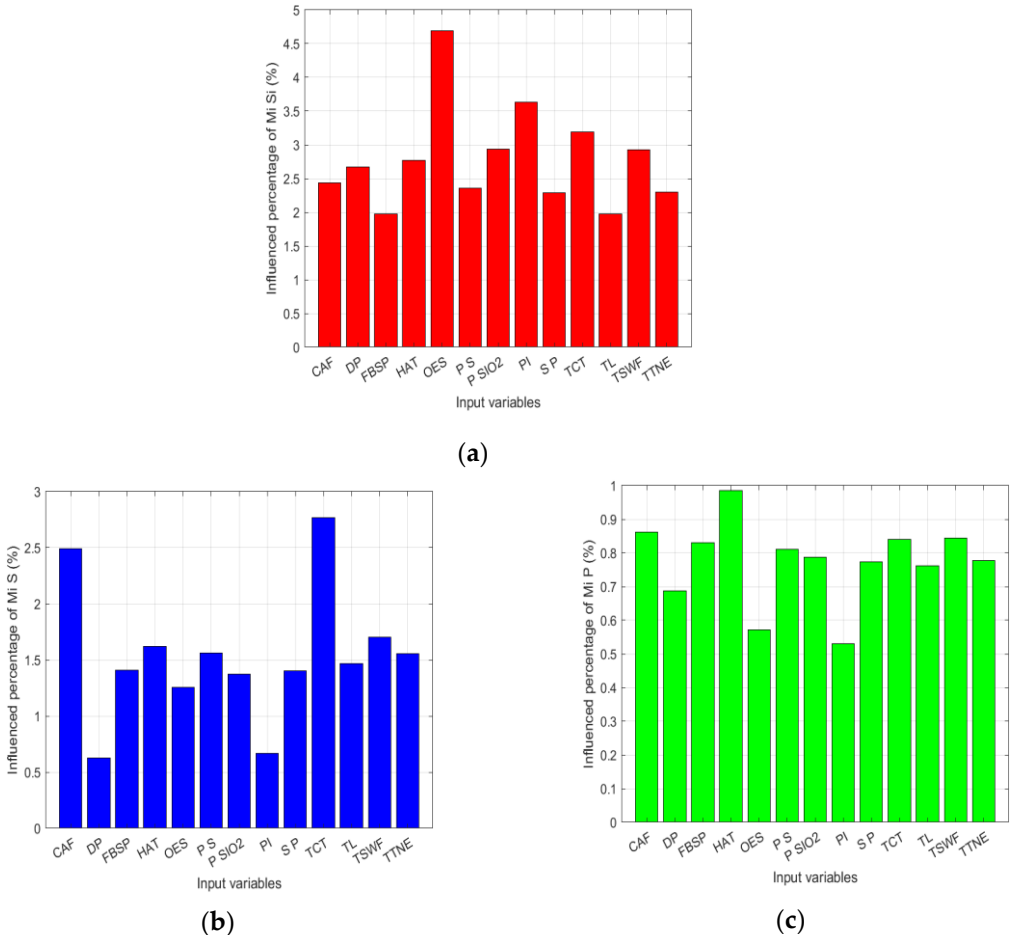

Figure 21. The percentage effect of a 5% increase of each input variable on the Si, S and P content in molten iron. (**a**) The percentage effect of the variable on MI Si (**b**) The percentage effect of the variable on MI S (**c**) The percentage effect of the variable on MI P.

It can be seen from the chart that oxygen-rich pressure has the greatest impact on Si content in molten iron. By increasing the oxygen-rich pressure by 5%, the Si content in molten iron can be increased by 4.69%. The permeability index is due to Si content in molten iron, second only to oxygen-rich pressure, but the permeability index does not belong to the operation parameter and cannot be directly adjusted. The theoretical combustion temperature has the greatest influence on the S content in molten iron. By increasing the theoretical combustion temperature by 5%, the S content in molten iron can be increased by 2.77%. The influence of cold air flow on the S content in molten iron is second only to the theoretical combustion temperature. By increasing the cold air flow by 5%, the S content in molten iron can be increased by 2.49%. The hot air temperature has the greatest influence on the P content in molten iron. Increasing the hot air temperature by 5% can increase the P content in molten iron by 0.984%. The influence of cold air flow on the P content in hot water is second only to the hot air temperature. By increasing the cold air flow by 5%, the P content in hot water can be increased by 0.861%. According to

the prediction results, the operators of the steel plant can adjust the operating parameters according to the influence of the variables on the Si, S and P content in molten iron, so as to improve the quality of molten iron and achieve stable and efficient production.

Table 12. The percentage value of a 5% increase of each input variable on Si, S and P content in molten iron.

Content in Molten Iron	FBSP	TL	S P	P SiO_2	P S	TTNE	OES
MI Si	1.97	1.98	2.291	2.94	2.36	2.30	4.69
MI S	1.41	1.47	1.40	1.37	1.56	1.56	1.26
MI P	0.830	0.762	0.773	0.787	0.811	0.776	0.571
Content in Molten Iron	**HAT**	**TSWF**	**CAF**	**DP**	**PI**	**TCT**	
MI Si	2.77	2.93	2.43	2.67	3.63	3.19	
MI S	1.62	1.70	2.49	0.626	0.668	2.77	
MI P	0.984	0.844	0.861	0.687	0.530	0.841	

5. Conclusions and Outlook

(1) Different from the existing research results, in this study the blast furnace data were divided into correlation data, temporal data, periodic data and manual input data according to different characteristics, and different methods were used to fill in different types of missing data in the data processing process, which makes the final data set more practical, so that the prediction model trained with the data set can better track the actual value.

(2) In this paper, we combine CNN with the BiLSTM model and use the I-GWO algorithm to optimize the hyperparameters to finally form the I-GWO-CNN-BiLSTM model. The I-GWO-CNN-BiLSTM model is used to predict the Si, S and P content in molten iron and compare it with the traditional LSTM model. The results show that the I-GWO-CNN-BiLSTM model is better than the LSTM model in the evaluation indexes R^2, RMSE and MAPE and has higher prediction accuracy and can effectively predict the quality of molten iron.

(3) In this paper, the MCMC algorithm is used to conduct the sensitivity analysis of each input variable and obtains the effect of each variable increasing by 5% on Si, S and P content in molten iron. The results indicate that, in each variable, oxygen-rich pressure, theoretical combustion temperature and hot air temperature, respectively, have the greatest influence on Si, S and P content in molten iron. When the quality of molten iron is not up to the standard, the steel mill staff can make timely adjustments on this basis. In subsequent research, automatic control-related technologies can be added in this link to realize the closed loop of "prediction-analysis-regulation" completed by machines instead of manually.

Author Contributions: R.L., conceptualization, project management; Z.-Y.G., investigation, writing—original draft; H.-Y.L., methodology, writing—review and editing, formal analysis; X.-J.L., data management, supervision; Q.L., resources, software. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (52004096), Hebei Natural Science Foundation (E2024209101) and Hebei Natural Science Foundation Youth Project (E2024209053).

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, S.; Feng, W.; Zhao, J.; Zhao, Z.; Liu, X.; Liu, R.; Lyu, Q. Collaborative optimization model of blast furnace raw materials and operating parameters based on intelligent calculation. *ISIJ Int.* **2024**, *64*, 1229–1239. [CrossRef]
2. Yang, Y.; Li, D.; Guo, L.; Wang, Z.; Guo, Z. Numerical simulation of the gasification-reduction coupling process in the innovative multi-generation system. *Appl. Therm. Eng.* **2020**, *168*, 114899. [CrossRef]
3. Li, J.; Hua, C.; Qian, J.; Guan, X. Low-rank based Multi-Input Multi-Output Takagi-Sugeno fuzzy modeling for prediction of molten iron quality in blast furnace. *Fuzzy Sets Syst.* **2021**, *421*, 178–192. [CrossRef]
4. Meng, L.; Liu, J.; Liu, R.; Li, H.; Zheng, Z.; Peng, Y.; Cui, X. Prediction of Silicon Content of Hot Metal in Blast Furnace Based on Optuna-GBDT. *ISIJ Int.* **2024**, *64*, 1240–1250. [CrossRef]
5. Yuan, M.; Zhou, P.; Li, M.; Li, R.-F.; Wang, H.; Chai, T.-Y. Intelligent multivariable modeling of blast furnace molten iron quality based on dynamic AGA-ANN and PCA. *J. Iron Steel Res. Int.* **2015**, *22*, 487–495. [CrossRef]
6. Zhou, P.; Zhang, R.; Liang, M.; Fu, J.; Wang, H.; Chai, T. Fault identification for quality monitoring of molten iron in blast furnace ironmaking based on KPLS with improved contribution rate. *Control Eng. Pract.* **2020**, *97*, 104354. [CrossRef]
7. Li, J.; Hua, C.; Yang, Y.; Guan, X. A novel MIMO T–S fuzzy modeling for prediction of blast furnace molten iron quality with missing outputs. *IEEE Trans. Fuzzy Syst.* **2020**, *29*, 1654–1666. [CrossRef]
8. Zhang, Z.; Zhang, R.; Zhou, P. Long Short-Term Memory Parameter Optimization Based on Improved Sparrow Search Algorithm for Molten Iron Quality Prediction. *Metals* **2024**, *14*, 529. [CrossRef]
9. Cui, Z.; Yang, A.; Wang, L.; Han, Y. Dynamic prediction model of silicon content in molten iron based on comprehensive characterization of furnace temperature. *Metals* **2022**, *12*, 1403. [CrossRef]
10. Zhou, P.; Yuan, M.; Wang, H.; Chai, T. Data-driven dynamic modeling for prediction of molten iron silicon content using ELM with self-feedback. *Math. Probl. Eng.* **2015**, *2015*, 1–11. [CrossRef]
11. Jiang, K.; Jiang, Z.; Xie, Y.; Pan, D.; Gui, W. Prediction of multiple molten iron quality indices in the blast furnace ironmaking process based on attention-wise deep transfer network. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–14. [CrossRef]
12. Chen, W.; Kong, F.; Wang, B.; Li, Y. Application of grey relational analysis and extreme learning machine method for predicting silicon content of molten iron in blast furnace. *Ironmak. Steelmak.* **2019**, *46*, 974–979. [CrossRef]
13. Li, H.; Li, X.; Liu, X.; Bu, X.; Chen, S.; Lyu, Q.; Wang, K. Prediction of the vanadium content of molten iron in a blast furnace and the optimization of vanadium extraction. *Separations* **2023**, *10*, 521. [CrossRef]
14. Sauer, T.; Xu, Y. On multivariate Lagrange interpolation. *Math. Comput.* **1995**, *64*, 1147–1170. [CrossRef]
15. Sharma, S.; Chakraborty, S.; Saha, A.K.; Nama, S.; Sahoo, S.K. mLBOA: A modified butterfly optimization algorithm with lagrange interpolation for global optimization. *J. Bionic Eng.* **2022**, *19*, 1161–1176. [CrossRef]
16. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
17. Sekulić, A.; Kilibarda, M.; Heuvelink, G.B.M.; Nikolić, M.; Bajat, B. Random forest spatial interpolation. *Remote Sens.* **2020**, *12*, 1687. [CrossRef]
18. Antoniadis, A.; Lambert-Lacroix, S.; Poggi, J.M. Random forests for global sensitivity analysis: A selective review. *Reliab. Eng. Syst. Saf.* **2021**, *206*, 107312. [CrossRef]
19. Schonlau, M.; Zou, R.Y. The random forest algorithm for statistical learning. *Stata J.* **2020**, *20*, 3–29. [CrossRef]
20. Zhang, S. Challenges in KNN classification. *IEEE Trans. Knowl. Data Eng.* **2021**, *34*, 4663–4675. [CrossRef]
21. Shokrzade, A.; Ramezani, M.; Tab, F.A.; Mohammad, M.A. A novel extreme learning machine based kNN classification method for dealing with big data. *Expert Syst. Appl.* **2021**, *183*, 115293. [CrossRef]
22. Levinson, J.; Esteves, C.; Chen, K.; Snavely, N.; Kanazawa, A.; Rostamizadeh, A.; Makadia, A. An analysis of svd for deep rotation estimation. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 22554–22565.
23. Lin, G.; Lin, A.; Gu, D. Using support vector regression and K-nearest neighbors for short-term traffic flow prediction based on maximal information coefficient. *Inf. Sci.* **2022**, *608*, 517–531. [CrossRef]
24. Guo, Z.; Yu, B.; Hao, M.; Wang, W.; Jiang, Y.; Zong, F. A novel hybrid method for flight departure delay prediction using Random Forest Regression and Maximal Information Coefficient. *Aerosp. Sci. Technol.* **2021**, *116*, 106822. [CrossRef]
25. Moghar, A.; Hamiche, M. Stock market prediction using LSTM recurrent neural network. *Procedia Comput. Sci.* **2020**, *170*, 1168–1173. [CrossRef]
26. Nosouhian, S.; Nosouhian, F.; Khoshouei, A.K. A review of recurrent neural network architecture for sequence learning: Comparison between LSTM and GRU. *Preprints* **2021**, *07*, 0252.
27. Liu, B.; Song, C.; Wang, Q.; Wang, Y. Forecasting of China's solar PV industry installed capacity and analyzing of employment effect: Based on GRA-BiLSTM model. *Environ. Sci. Pollut. Res.* **2022**, *29*, 4557–4573. [CrossRef] [PubMed]
28. Yang, M.; Wang, J. Adaptability of financial time series prediction based on BiLSTM. *Procedia Comput. Sci.* **2022**, *199*, 18–25. [CrossRef]
29. Kulshrestha, A.; Krishnaswamy, V.; Sharma, M. Bayesian BILSTM approach for tourism demand forecasting. *Ann. Tour. Res.* **2020**, *83*, 102925. [CrossRef]
30. Bhatt, D.; Patel, C.; Talsania, H.; Patel, J.; Vaghela, R.; Pandya, S.; Modi, K.; Ghayvat, H. CNN variants for computer vision: History, architecture, application, challenges and future scope. *Electronics* **2021**, *10*, 2470. [CrossRef]
31. Kattenborn, T.; Leitloff, J.; Schiefer, F.; Hinz, S. Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. *ISPRS J. Photogramm. Remote Sens.* **2021**, *173*, 24–49. [CrossRef]

32. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 1–74.
33. Nadimi-Shahraki, M.H.; Taghian, S.; Mirjalili, S. An improved grey wolf optimizer for solving engineering problems. *Expert Syst. Appl.* **2020**, *166*, 113917. [CrossRef]
34. Liu, T.; Yang, H.; Yu, J.; Zhou, K.; Jiang, F. A global harmony search algorithm based on Tent Chaos Map and Elite Reverse Learning. In Proceedings of the 2022 14th International Conference on Advanced Computational Intelligence (ICACI), Wuhan, China, 15–17 July 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 320–325.
35. Hashemi, A.; Dowlatshahi, M.B.; Nezamabadi-Pour, H. Gravitational Search Algorithm: Theory, Literature Review, and Applications. In *Handbook of AI-Based Metaheuristics*; CRC Press: Boca Raton, FL, USA, 2021; pp. 119–150.
36. Diab, A.A.Z.; Abdul-Ghaffar, H.I.; Ahmed, A.A.; Ramadan, H.A. An effective model parameter estimation of PEMFCs using GWO algorithm and its variants. *IET Renew. Power Gener.* **2022**, *16*, 1380–1400. [CrossRef]
37. Lu, W.; Li, J.; Wang, J.; Qin, L. A CNN-BiLSTM-AM method for stock price prediction. *Neural Comput. Appl.* **2021**, *33*, 4741–4753. [CrossRef]
38. Luengo, D.; Martino, L.; Bugallo, M.; Elvira, V.; Särkkä, S. A survey of Monte Carlo methods for parameter estimation. *EURASIP J. Adv. Signal Process.* **2020**, *2020*, 1–62.
39. Nijkamp, E.; Hill, M.; Han, T.; Zhu, S.-C.; Wu, Y.N. On the anatomy of mcmc-based maximum likelihood learning of energy-based models. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 5272–5280. [CrossRef]
40. Fabreti, L.G.; Höhna, S. Convergence assessment for Bayesian phylogenetic analysis using MCMC simulation. *Methods Ecol. Evol.* **2022**, *13*, 77–90. [CrossRef]
41. Zhao, T.; Peng, H.; Xu, L.; Sun, P. Statistical landslide susceptibility assessment using Bayesian logistic regression and Markov Chain Monte Carlo (MCMC) simulation with consideration of model class selection. *Georisk Assess. Manag. Risk Eng. Syst. Geohazards* **2024**, *18*, 211–227. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.