Journal of University of Science and Technology Beijing
Vol.7(2000), No.4, p.269

An Improved Artificial Neural Network Model for Predicting Silicon Content of Blast Furnace Hot Metal

Bin Yao, Tianjun Yang, Xiaojun Ning

Metallurgy School, University of Science and Technology Beijing, Beijing 100083, China
(Received 1999-11-10)

Abstract: Based on the skills of initializing weight distribution, adding an impulse in a neural network and expanding the ideal of plural weights, an artificial neural network model with three connection weights between one and another neural unit was established to predict silicon content of blast furnace hot metal. After the neural network was trained in the off-line state on the basis of a large number of practical data of a commercial blast furnace and making many learning patterns, satisfactory testing and simulating results of the model were obtained.

Key word: blast furnace; silicon content; neural network

The blast furnace (BF) is a huge high-temperature reactor, and controlling hot metal temperature has a very important effect on safe performance and the quantity and quality of hot metal. Si content of hot metal produced from a BF is known to be an important parameter, because Si content is directly related to the coke rate and hot metal temperature and also decides the charge balance in oxygen steel-making converters. So the uniformity in Si content from one heat to another and the accurate prediction of Si content greatly helps to stabilize BF operation [1,2].

Since hot metal temperature and Si content are closely related to each other, a model for predicting silicon, coke rate and temperature simultaneously has to be used in an interactive mode. To some extent, they have a positive role on stabilizing BF temperature level and improving quality of pig iron. But in fact, there are very complex chemical reactions and the processes of heat and mass transfer and the liquid flow in a BF. This leads to its characteristics that the system is strongly coupling and non-linear and makes it very difficulty to develop mathematical models to predict Si content and hot metal temperature [3–5].

In this work, based on production characteristics of No.10 blast furnace with a effective volume of 2 580 m^3 and its operation practice at Anshang Iron and Steel Company in China, an improved artificial neural network (ANN) model is established to predict the silicon content, which has an improved topology structure and learning algorithm.

1 Theoretic Analysis

1.1 Structure of the neural network

An attempt was made to improve the learning speed of the back propagation algorithm and the accuracy of prediction. The skills of initializing weight distribution were introduced, and an impulse item was added to the neural network (NN) learning. On the basis of expanding the ideal of plural weights, an artificial neural network model with three connection weights between one and another neural unit was developed [5–8]. **Figure 1** shows its topology, and it has a three-layer structure. The number of input nodes is, typically, taken to be the same as the number of state variables, which mainly affect Si content of hot metal. And the output layer neurons represent Si content of hot metal. The hidden

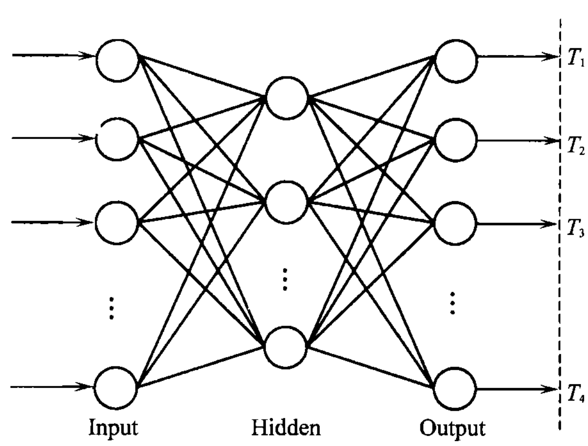


Figure 1 A three-layer feed-forward NN structure.

layer neurons are used to extract the characters of parameters and their number is varied. **Figure 2** shows the structure of three connection weights between one and another neuron.

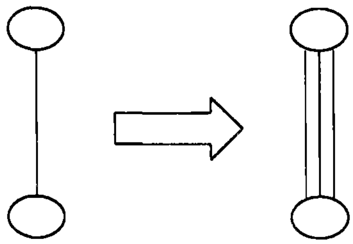


Figure 2 Three connection weights between one and another neuron.

1.2 Learning mechanism

In this work the weights of interconnections of the neurons for the model are decided by minimizing the error between the desired output and actual output of neural net and the process is known as training or learning of net. The minimization of the error, in other words, modification of weights, is done by using the steepest descent method of optimization. As illustrated in figure 1, during training, the network is presented through a set of correct input-output pairs called examples, then the output of every neuron in every layer respectively is established. Now assuming that u_i denotes the input sum of the i-th unit in layer k (k=2,3), that v_i denotes its output and that w_{ij} represents the connection weight between the j-th unit in layer $k-1$ and the i-th unit in layer k. w_{ij} is composed of three parts as follows:

$$w_{ij}=w_{ij}^1+w_{ij}^2+w_{ij}^3 \tag{1}$$

where w_{ij}^1, w_{ij}^2 and w_{ij}^3 represent the three weight vectors respectively, and the symbol "+" means the sum of three weight vectors including w_{ij}^1, w_{ij}^2 and w_{ij}^3. Each neuron constitutes a processing element (PE) and it is connected through various weights to the other PEs. The processing element sums the product of each input and the connection weights from the previous layer of PEs and then filters it by a nonlinear threshold function of the form

$$f(x)=\frac{1}{1+e^{-x}} \tag{2}$$

So for a multi-layer network, the output v_i is given by

$$v_i=f(u_i) \tag{3}$$

$$u_i=\Sigma w_{ij}v_j \tag{4}$$

And each input and output are given by the form of a three-weight vector as follows:

$$u_i=u_i^1+u_i^2+u_i^3=\Sigma w_{ij}^1v_j+\Sigma w_{ij}^2v_j+\Sigma w_{ij}^3v_j \tag{5}$$

$$v_i=f[(u_i^1)^2+(u_i^2)^2+(u_i^3)^2] \tag{6}$$

where u_i^1, u_i^2 and u_i^3 represent the input of the input sum of the i-th unit in layer k respectively, which are relative to the three weight vectors.

In the back propagation method, the least square error is defined by

$$E_k=\frac{1}{2}\Sigma(t_i-v_i)^2 \tag{7}$$

where t_i denotes the desired output of the i-th node, and v_i denotes the actual output of the i-th node.

Before the training of NN starts, all the weights are randomized. In the learning mode the net considers each datum set from the training patterns, one at a time and generates output. Then its output is compared with the target output of all the nodes of the output layer, and if there is any discrepancy, the error is back-propagated by changing the interconnect weights according to the following equation:

$$\Delta w_{ij}^{1/2/3}=\varepsilon\cdot\delta_i^{1/2/3}\cdot v_j \tag{8}$$

$$\delta_i^{1/2/3}=-\frac{\partial E_k}{\partial u_i^{1/2/3}} \tag{9}$$

where $\Delta w_{ij}^{1/2/3}$ denotes one of the weight vectors of Δw_{ij} and Δw_{ij} is the weight change from the i-th neuron to the j-th neuron in the next layer; ε denotes the learning rate and $\varepsilon\in[0,1]$; $u_i^{1/2/3}$, denoting one of u_i^1, u_i^2 and u_i^3, is the input vector of the hidden neuron or output neuron which is relevant to $\Delta w_{ij}^{1/2/3}$; $\delta_i^{1/2/3}$ is the error signal, which is relevant to $\Delta w_{ij}^{1/2/3}$ and $u_i^{1/2/3}$ denotes one of u_i^1, u_i^2 and u_i^3.

For output neurons the value of $\delta_i^{1/2/3}$ is given by

$$\delta_i^{1/2/3}=2\,u_i^{1/2/3}\,(t_i-v_i)\,v_i\,(1-v_i) \tag{10}$$

But for hidden layers, target output is known. So the weight sum of error signals of all the neurons in the succeeding layers is utilized to calculate the error signal of any neuron i in the hidden layer. The expression is as follows:

$$\delta_j^{1/2/3}=2\,u_j^{1/2/3}\,v_j\,(1-v_j)\cdot\Sigma(\delta_i^{1/2/3}\,w_{ij}^{1/2/3}) \tag{11}$$

In this work, in order to improve the convergent speed of NN, the method of an inertial emendation was adopted, and a momentum item was added to the weight change according to the following equation:

$$\Delta w(N)=\Delta w_{ij}^{1/2/3}+\alpha\cdot\Delta w\,(N-1) \tag{12}$$

where $\Delta w(N)$ denotes the revised value of the weights between input and hidden layers or between hidden and output layers in the N-th (N=1,2,3,$\cdots$) learning, $\Delta w(N-1)$ denotes the revised value of the weights between input and hidden layers or between hidden and output layers in the $(N-1)$-th learning, and α denotes the momentum factor.

Since each neuron is represented by summation function and non-linearity, the threshold of non-linearity is also calculated. The weights of these thresholds are assumed to be connected with auxiliary or bias nodes which have no input to them, but their output is always 1.0. The development of a complete neural network based on generalization involves two steps namely training and testing. During training the interconnect weights between different layers are optimized with patterns so as to obtain the lowest acceptable error. Then in the testing step, actual testing of the neural network is performed on a new set of data (*i.e.* those data were not used for training). If the results obtained during testing are not satisfactory, the training set is modified (or enlarged) so as to obtain an acceptable level of error during testing.

2 Practical Application

2.1 Input and output parameters

There are many factors, which affect the thermal state of a BF, including the physical and chemistry characteristics and the charging method, and also including the bottom parameters such as blast volume, hot blast temperature, mass of fuel, and so on. The following set of variables was finally selected to predict the Si content of hot metal in the No.10 blast furnace:

- Charge rate, batch/h;
- Blast volume, m^3/min;
- Hot blast temperature, ℃;
- Hot blast pressure, MPa;
- Top gas pressure, MPa;
- PCI, kg/t(HM);
- Blast humidity, t/h;
- Enriched oxygen volume, m^3/h;
- Top gas temperature, ℃;
- Ratio of ore to coke;
- TFe mass fraction, %;
- CO content of top gas in volumic fraction;
- CO_2 content of top gas in volumic fraction.

2.2 Making learning patterns

For modeling purposes, daily operational data in four years were analyzed. Given the frequency of gathering data and the predicted precision of the model, the data were gathered in a definite time interval and they were preprocessed using certain physical and/or empirical correlation between the input variables to the net. At the same time, the input and output variables must be smoothed in order to get ride of disturbance.

This represented a total of 400 patterns to be used for training and testing the ANN model. The data were split roughly into two thirds for training and one third for testing. Typical data are shown in **table 1**.

2.3 Application results

The 3-layer improved ANN configuration was tried out. The number of hidden layer neurons was 21. A bias neuron was also incorporated in the hidden layer to improve the net performance. The learning rate (ε) was varied between 10^{-4} and 10^{-2}, and the momentum factor (α) was varied between 0.5 and 0.9. The training re-

Table 1 Sample set of 12 data points fed to the ANN

Inputs												Output
1	2	3	4	5	6	7	8	9	10	11	12	13
7	6000	1100	0.359	0.20	18.00	2.6	6000	169	3.09	57.67	0.436	0.557
7	6020	1180	0.539	0.20	18.00	2.6	8000	129	3.36	57.87	0.445	0.730
7	6150	1120	0.359	0.20	18.00	2.4	8000	127	3.36	57.87	0.445	0.420
7	6050	1130	0.359	0.20	18.00	2.3	8000	108	3.35	57.88	0.443	0.690
7	6060	1160	0.359	0.21	18.20	2.3	8000	119	3.36	57.87	0.436	0.731
7	6010	1140	0.359	0.20	18.00	2.2	7960	104	3.36	57.87	0.443	0.756
7	6040	1160	0.359	0.20	17.65	2.4	8000	127	3.36	57.87	0.443	0.795
7	5920	1160	0.359	0.20	18.00	2.8	8000	103	3.36	57.67	0.445	0.789
6	5940	1160	0.359	0.21	18.15	2.4	8000	116	3.36	57.87	0.442	0.821
6	6000	1100	0.359	0.20	18.00	2.5	6000	134	3.09	57.67	0.436	0.777

Note: The input (No.1 to 12) and output (No.13) variables used are defined below. 1—Charge rate, batch/h; 2—Blast volume, m^3/min; 3—Hot blast temperature, ℃; 4—Hot blast pressure, MPa; 5—Top gas pressure, MPa; 6—PCI, kg/t(HM); 7—Blast humidity, t/h; 8—Enriched oxygen volume, m^3/h; 9—Top gas temperature, ℃; 10—Ratio of ore to coke; 11—TFe mass fraction TFe in %; 12—Volumic fraction of CO_2 in top gas; 13—Desired Si content.

sults after 10^4 iterations are shown in **figure 3** and the relative errors for most of the training data are within ±5%. The net was tested through another 50 sets of data, and the results of testing are shown in **figure 4**. Assuming that the predicting errors were within ±0.05% ($|\Delta w_{Si}| \leq 0.05$, where Δw_{Si} denotes the mass fraction change of Si between the predicted and practical values), the number of hitting the target was 43. So here the hit ratio is 86%.

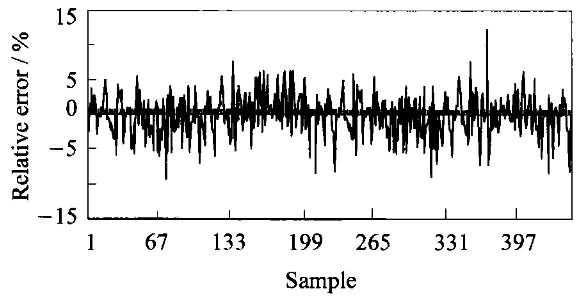
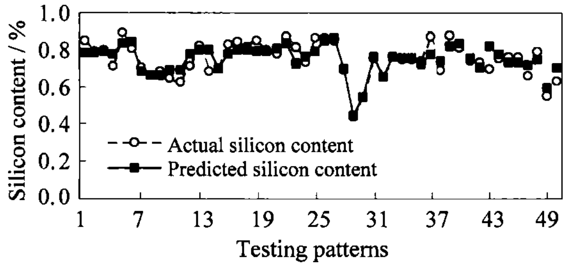


Figure 3 Relative error after training.



Figfure 4 Comparison between the predicted and the practical Si content.

3 Conclusions

In order to improve ANN learning speed and shorten learning time, the skills of initializing weight distribution and adding an impulse in the neural network were adapted. The neural network with three connection weights between one and another neural unit was developed to improve the learning speed of the back propagation algorithm. The neural network simulation of predicting Si content showed that the general performance of ANN is satisfactory, even when trained with a limited number of patterns.

Some inherent problems are always associated with the use of neural networks. For example, the drawback of the back propagation learning algorithm is that the global minimum of E can not be perfectly reached during learning. And if the error function is multi-modal in the weight space, then it is possible that learning will get stuck up in local minimum. Despite all these problems, ANN is emerging as an important tool to simulate those complex processes in the iron-making industry.

References

[1] H. J. Bachhofen: The Application of Modern Process Control Technology in Hot Metal Production. [in] *Ironmaking Conference Proceedings*. 1991, pp.703-708.

[2] Yasuo NIWA: *ISIJ International*, 31(1991), No.5, p.487.

[3] M. Marios: *Modeling, Identification and Stable Adaptive Control of Continuous-Time Nonlinear Dynamical Systems Using Networks*. ACC/WA2, 1992, pp.36-40.

[4] M. Sakurai: *Iron & Steelmaker*, 16(1989), No.11, p.59.

[5] Yasuo NIWA: *ISIJ International*, 30 (1990), No.2, p.111.

[6] Yhu-Jen Hwu: *Steel Research*, 67(1996), No.2, p.59.

[7] H. Singh, B. Deo: *Iron & Steelmaker*, 22 (1995), No.10, p.85.

[8] A. Datta: *Steel Research*, 65(1994), p.466.