MDPI

Article

Copper Content Inversion of Copper Ore Based on Reflectance Spectra and the VTELM Algorithm

Yanhua Fu [1], Hongfei Xie [2], Yachun Mao [3], Tao Ren [4] and Dong Xiao [2,*]

[1] JangHo Architecture College, Northeastern University, Shenyang 110169, China; fuyanhua@mail.neu.edu.cn
[2] Information Science and Engineering School, Northeastern University, Shenyang 110004, China; 1870722@stu.neu.edu.cn
[3] School of Resources and Civil Engineering, Northeastern University, Shenyang 110819, China; maoyachun@mail.neu.edu.cn
[4] Software College, Northeastern University, Shenyang 110169, China; chinarentao@163.com
* Correspondence: xiaodong@ise.neu.edu.cn

check for
updates

Abstract: Copper is an important national resource, which is widely used in various sectors of the national economy. The traditional detection of copper content in copper ore has the disadvantages of being time-consuming and high cost. Due to the many drawbacks of traditional detection methods, this paper proposes a new method for detecting copper content in copper ore, that is, through the spectral information of copper ore content detection method. First of all, we use chemical methods to analyze the copper content in a batch of copper ores, and accurately obtain the copper content in those ores. Then we do spectrometric tests on this batch of copper ore, and get accurate spectral data of copper ore. Based on the data obtained, we propose a new two hidden layer extreme learning machine algorithm with variable hidden layer nodes and use the regularization standard to constrain the extreme learning machine. Finally, the prediction model of copper content in copper ore is established by using the algorithm. Experiments show that this method of detecting copper ore content using spectral information is completely feasible, and the algorithm proposed in this paper can detect the copper content in copper ores faster and more accurately.

Keywords: copper ore; spectroscopy; extreme learning machine; VTELM; regularization

1. Introduction

Copper is one of the oldest metals discovered by human beings. About 7000 years ago, casting of bronze ware appeared in Eurasia [1]. The proven reserves of copper resources in the world are 2.478 billion tons. The top ten countries with global copper reserves are Chile, the United States, Peru, Congo (Kinshasa), Australia, China, Russia, Mexico, Canada, and Argentina [2]. With the progress of the times, the continuous innovation of science and technology and the continuous development of world industrialization, the applications of copper are more and more extensive. As shown in Figure 1, open-pit ore mining situation is commonly used.

Recovery is an important index in mineral processing technology, which reflects the level of mineral processing technology and the quality of mineral processing work. For the copper content level in copper ore, the mineral processing technology adopted is not the same. Therefore, how to quickly and accurately measure the copper content in copper ore is particularly important. There are two kinds of traditional copper ore content determination [3–5], the first is the flame atomic absorption method, which has high accuracy, but low sensitivity and it is difficult to determine non-metallic elements by this technique. The second is to use chemical reagents to detect copper ores. Although this method has the same high accuracy, it is highly affected by the chemical reagents used, and the waste liquid causes great pollution.

Therefore, how to measure copper content in copper ore stably and efficiently is a problem that must be solved in mineral processing technology, which is of great significance to reduce costs and improve efficiency. Spectral analysis technology is very mature, with the advantages of nondestructive testing, fast detection speed, high resolution, low cost. In recent years, it has become a trend to use spectral analysis instead of traditional analysis methods.

Figure 1. Open pit ore mining.

Mining in mining areas often leads to the destruction of the ecological environment, and it is extremely easy to change the nature of the soil on the surface and damage the health of surrounding residents. Therefore, the abandoned land after mining in the mining area should be reclaimed. Phytoremediation [6] can effectively absorb heavy metals or other toxic substances in the soil, thereby improving the soil. Luo [7] and others used commercial chrysanthemums to carry out heavy metal soil remediation. The results showed that after three years of soil remediation, the Cd in the soil decreased by 78.1%, and the Zn content decreased by 28.4%. In addition, the rice and green vegetables grown from the repaired soil meet the requirements of dietary safety. Saleem et al. [8] planted four kinds of jute products (HT, C-3, GC and SH) in the soil heavily polluted by copper to conduct experiments. The experiments show that C-3 and HT have the strongest survivability in copper-contaminated soil and may absorb Cu in the soil. Muro-Gonzalez et al. [9] conducted experiments under greenhouse conditions and found that *Prosopis laevigata* can grow normally under heavy metal pollution and is a potential plant for land reclamation. Chauhan and Mathur [10] proposed through experiments that sunflower can effectively remove heavy metals in contaminated soil.

Spectral analysis is a quantitative analysis method based on the principle of spectroscopy by detecting the characteristic wavelength and intensity of substances. Because it does not damage the sample during analysis, the operation is relatively simple, and the analysis speed is relatively fast, it has been widely used in many fields [11–16]. Because copper ore contains gangue minerals and other impurities, these impurities more or less affect the final results of spectral data. Therefore, the spectral data of copper ore contains many other spectral information related to the detection of copper ore content, which causes the spectral data of copper ore to have too high dimensions and redundant information. If the spectral data are directly used for modeling, the input dimension of the model will be too high, the amount of calculation will be large, and the structure will be too large. Therefore, the original spectral data of copper ore should be preprocessed, and dimensionality reduction is one of the most effective preprocessing methods for high-dimensional data. Dimension reduction can reduce the influence of noise on sample training as much as possible by removing the noise and redundant information in the data set, so as to simplify the training and prediction of machine learning models. As the data is compressed from high-dimensional to low-dimensional, the structure of the machine learning model will become simpler, the accuracy of training will increase and the time required for model training will also be greatly reduced.

Because the spectrum method can detect the sample quickly and without damage. So spectroscopy is more and more used in geology. Spectral technology can effectively identify minerals and quickly

analyze minerals to identify exploration minerals [17]. Zhou et al. [18] used the spectral information of near standard soil samples to conduct hyperspectral modeling of lead content. The results show that this method can solve the problems of complex composition in soil and weak spectral information of heavy metals, and can further use remote sensing information to monitor soil heavy metals. Zhao et al. [19] established linear regression model and partial least square regression model by using the multispectral data obtained by UAV. The results show that the multispectral data can monitor the reclamation effect efficiently and quickly. Shin [20] and others studied the spectral information of heavy metal contaminated soil, and obtained that the spectral response of soil was positively correlated with the concentration of heavy metal. Zhang et al. [21] carried out four processing methods for the spectral data of heavy metal soil, and removed the noise of spectral information. Partial least squares regression and RBF neural network are used to model. The results show that the content of heavy metals in waste land can be effectively detected by using spectral information.

Principal component analysis (PCA) [22] is a simple and effective method for data processing, compression and dimensionality reduction [23–25]. It can use a small amount of data to retain the most important characteristics of the original data. PCA uses the direction of the largest variance of the original data as the projection, because the maximum variance of the data gives the most important information contained in the data. PCA can remove useless noise and reduce a lot of computation. Therefore, we used PCA to preprocess the original spectral data of copper ore.

Because machine learning models can process a large number of data, analyze and fit the data, and have strong generalization ability, in recent years, more and more researchers have begun to use and study machine learning, which leads to the continuous development of machine learning algorithms. Machine learning has made great progress in material science. Now more and more new materials cannot be found and designed without machine learning. Reference [26] proposes that in materials science, most of the existing experiments and computational modeling consume a lot of time and resources. Therefore, the use of machine learning to discover the properties of new materials, as well as the design and application of new materials, has been paid more and more attention. In this paper, the application of machine learning in material science is discussed in detail. Through the combination of material experiment and machine learning, a new idea is provided for the discovery of new material parameter performance. Reference [27] proposed that the discovery of new materials urgently needs to explore more advanced machine learning algorithms. Reference [28] summarized the application of machine learning in material design and material discovery of rechargeable batteries, discusses the problems of machine learning in the prediction of rechargeable battery performance, the structure and accuracy of machine learning models, and the problem of sample dimensions. Reference [29] discusses the importance of machine learning in the development and analysis of lithium sulfur batteries, and analyzes the key factors affecting lithium sulfur batteries by using machine learning method.

Extreme learning machine [30] (ELM) is a machine learning algorithm designed for single hidden layer feedforward neural network proposed by Huang in 2006. Since the algorithm randomly generates a hidden layer threshold and input weight, the algorithm does not need gradient back-propagation to adjust the weight. In this algorithm, the number of hidden layer neurons can be designed according to artificial experience or iterative optimization, and then the unique optimal output matrix can be solved by solving equations. Because of its simple structure and fast learning speed, more and more researchers begin to study this ELM algorithm deeply and use it in various fields [31–34]. Liang et al. proposed an online learning extreme learning machine (OS-ELM) by introducing an online learning mechanism into ELM [35]. This OS-ELM can update its own weights and thresholds with the training of new samples, without the need to retrain the model to obtain new weights and thresholds. Then, Huang et al. applied the integrated learning method to OS-ELM and proposed integrated online ELM (EOS-ELM) [36]. In 2016, Qu and Lang changed the single hidden layer of ELM to two hidden layers and proposed a two hidden layer extreme learning machine (TELM) [37]. In TELM, the number of neurons in the first hidden layer and the second hidden layer are the same, and the neuron connection

method is the same as that of ELM, which is that the neurons are fully connected. In this paper, the two hidden layer Extreme learning machine with variable neuron nodes (VTELM) is proposed. The number of nodes in the middle two layers can be different, and the regularization standard is used to constrain the connection matrix between the second hidden layer and the output layer, which greatly improves the generalization ability of the Extreme learning machine.

The chapters of this paper are arranged as follows: the first chapter introduces the spectral analysis, principal component analysis and copper ore detection status. The second section introduces the preparation of copper ore samples and the process of spectral testing. In the third chapter, the algorithm flow of ELM and TELM is briefly described, and a two hidden layer extreme learning machine algorithm with variable hidden layer nodes (VTELM) is proposed. In the fourth part, the VTELM model is simulated and the simulation results are analyzed. Finally, the fifth part summarizes the main conclusions of this study.

2. Data Acquisition and Processing

All the copper ore samples are from the Deerni copper deposit (Qinghai Province, P.R. China) with a total of 251 samples. For copper ore samples, we used a SVC HR-1024 portable ground spectrometer (Vista Company, city, state abbrev, USA) for spectral testing. The parameters of the spectrometer are shown in Table 1.

Table 1. Parameters of SVC HR-1024.

Spectrometer Parameters	Parameter Value
Spectral Range	350–2500 nm
Internal Memory	500 Scans
Channels	1024
Spectral Resolution (FWHM)	≤3.5 nm, 350–1000 nm ≤9.5 nm, 1000–1850 nm ≤6.5 nm, 1850–2500 nm
Bandwidth (nominal)	≤1.5 nm, 350–1000 nm ≤3.6 nm, 1000–1850 nm ≤2.5 nm, 1850–2500 nm
Minimum Integration	1 millisecond

Firstly, we clean and dry the copper ore samples collected, and then grind them. We chose to perform experiments from 10:00 to 14:00 during the day, while the weather was clear and cloudless. During the experiment, the probe of the spectrometer was perpendicular to the surface of the copper ore sample, and the distance between the probe and the sample surface should be 300 mm, the scanning time of the instrument was 1 s·times^{-1}, and the whiteboard measurement value was corrected every 10 min. In order to reduce the measurement error, we conducted 10 spectral experiments on each sample of the spectrometer for three consecutive days. At the end of the experiment, the spectral information is collected, and the average value is taken as the spectral data (973 features). After the spectral test is completed, the spectral data of copper ore samples are obtained by preprocessing the measured spectral data such as band fitting. Figure 2 shows the spectrum of a copper ore.
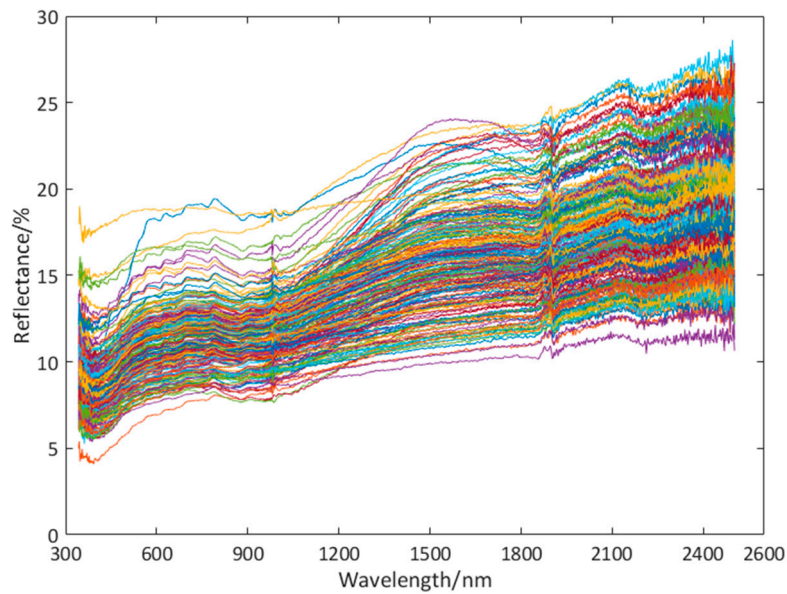
Figure 2. Spectrum of a copper ore.

3. Introduction to Neural Network

3.1. Extreme Learning Machine (ELM)

Single hidden layer Extreme learning machine is a kind of feedforward neural network. It consists of input layer, one hidden layer and output layer. For N arbitrary copper sample (x_i, t_i), where $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$, $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$, then the calculation formula of the extreme learning machine is as follows (Equation (1)):

$$f_L(x) = \sum_{i=1}^{L} \beta_i g_i(x) = \sum_{i=1}^{L} \beta_i g(\omega_i * x_j + b_i), j = 1, \dots \dots, N \tag{1}$$

where L represents the number of neurons in the hidden layer of ELM; N represents the total number of samples entering the model training; β_i represents the weight vector between the hidden layer and output layer; ω_i is a random value, which represents the connection weight between the hidden layer neuron and the input feature; g represents the activation function; b_i represents the offset vector; x_j represents the input vector.

The calculation process of extreme learning machine is very similar to that of standard back propagation neural network, but the output matrix of hidden layer is pseudo inverse matrix. From Formula (1), we can draw the following conclusions:

$$H\beta = T \tag{2}$$

In Equation (2):

$$H = \begin{bmatrix} g(\omega_1 * x_1 + b_1) & \bullet\bullet\bullet & g(\omega_L * x_1 + b_L) \\ \bullet & & \bullet \\ \bullet & \bullet\bullet\bullet & \bullet \\ \bullet & & \bullet \\ g(\omega_1 * x_N + b_1) & \bullet\bullet\bullet & g(\omega_L * x_N + b_L) \end{bmatrix}_{N \times L} \tag{3}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \bullet \\ \bullet \\ \bullet \\ \beta_L^T \end{bmatrix}_{L \times m} \quad T = \begin{bmatrix} t_1^T \\ \bullet \\ \bullet \\ \bullet \\ t_N^T \end{bmatrix}_{N \times m} \tag{4}$$

where: m is the number of outputs; H is the output matrix of hidden layer; T is the objective matrix of the training set.

Since the threshold of the hidden layer of ELM and the connection weight between the hidden layer and the input layer are randomly generated, the linear equation $H\beta = T$ can be solved by the least squares method:

$$\textbf{Minimize:} \|H\beta - T\| \tag{5}$$

Huang put forward two theorems on the basis of predecessors, and proved that the least square solution in the ELM model is:

$$\hat{\beta} = H^+ T \tag{6}$$

where H^+ is the Moore Penrose generalized inverse of hidden layer output matrix **H**, and the least square solution of $H\beta = T$ is unique.

3.2. Two Hidden Layer Extreme Learning Machine (TELM)

The TELM neural network is based on ELM, changing one hidden layer into two hidden layers. The number of neurons in the first layer and the first layer is the same, and the neurons in the previous layer are connected to each neuron in the next layer. During the operation of the TELM model, the connection weight between the first hidden layer and the input layer and the threshold value of the first hidden layer neuron are randomly selected, while the two hidden layer neurons The number can be obtained according to an empirical formula.

The TELM algorithm updates the output matrix of the second hidden layer by solving the connection weight of the first hidden layer and the second hidden layer and the threshold of the second hidden layer. Through calculation, the predicted output of TELM can be infinitely close to the actual output. TELM algorithm also has the advantages that ELM algorithm does not have, such as rarely falling into over fitting, updating the optimal solution and more suitable for big data processing. After algorithm analysis and actual analysis, TELM has unparalleled advantages in predictive regression and classification compared with ELM.

TELM network structure diagram is shown in Figure 3, algorithm flow chart is shown in Figure 4. In the TELM algorithm, we mainly solve the value of parameters W_1, B_1, β.
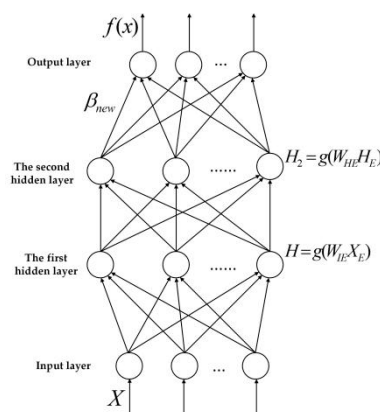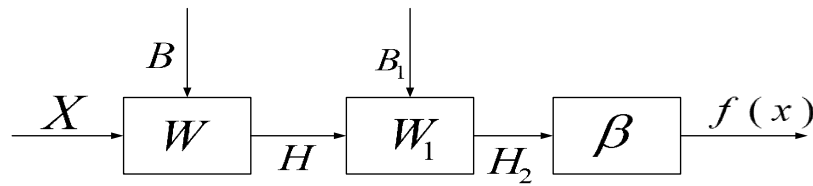

Figure 3. Network structure of TELM.

Figure 4. The work flow of the TELM.

Firstly, we assume that the training sample data set of TELM neural network is $\{X,T\} = \{x_i, t_i\}(i = 1,2,\dots,Q)$, where X represents the input feature and T represents the feature. In TELM model, the first layer of hidden layer has the same activation function and the same number of hidden layer nodes as the second layer.

Then, in the algorithm design of TELM, the two hidden layers of TELM are regarded as one hidden layer, so TELM can be regarded as ELM. We can imitate ELM to get the output matrix H as follows:

$$H = g(WX + B) \tag{7}$$

It can be seen from the workflow of the TELM that W and B are the weights and thresholds of hidden layers in ELM which are randomly initialized.

Next, by the ELM algorithm we can get matrix β:

$$\beta = H^+ T \tag{8}$$

Now, add a second hidden layer to the ELM algorithm, so that the ELM contains two hidden layers and restore it to TELM, and each hidden layer neuron is connected to each other, we can get the prediction output matrix of the second hidden layer H_1 is:

$$H_1 = g(W_1 H + B_1) \tag{9}$$

Then the real output matrix H_{1*} of the second hidden layer is:

$$H_{1*} = T\beta^+ \tag{10}$$

where β^+ is the generalized inverse of β.

Let $H_1 = H_{1*}$, which can maximize the predicted value of TELM close to the true value.

Now we assume the matrix $W_{HE} = [B_1 \quad W_1]$, so the weights W_1 and threshold B_1 of the second hidden layer can be solved as follows:

$$W_{HE} = g^{-1}(H_{1*})H_E{}^+ \tag{11}$$

where H_E^+ is the generalized inverse of matrix $H_E = [1\ H]^{\mathrm{T}}$, 1 represents a Q-dimensional vector with each element 1. $g^{-1}(x)$ is the inverse function of $g(x)$.

From Equation (11) we solve the parameters W_1 and B_1, then we can re-solve the second hidden layer prediction output H_2:

$$H_2 = g(W_1 H + B_1) = g(W_{HE}H_E) \tag{12}$$

Therefore, according to Figure 4 we can calculate β as:

$$\beta_{\text{new}} = H_2{}^+ T \tag{13}$$

Finally, we can get the final output $f(x)$ of the neural network as follows:

$$f(x) = H_2 \beta_{new} \tag{14}$$

In the calculation of β, when $\beta^T\beta$ is nonsingular, $\beta^+ = (\beta^T\beta)^{-1}\beta^T$.

3.3. Two Hidden Layer Extreme Learning Machine Algorithm with Variable Hidden Layer Nodes (VTELM)

The structure of the VTELM neural network is similar to the TELM structure. The difference is that the neuron nodes of each hidden layer of VTELM can be different from each other. In the process of VTELM model operation, because the connection weights of the first hidden layer and the input layer, the connection weights of the second hidden layer and the first hidden layer, as well as the threshold value of each hidden layer neuron are random values, we only need to set the nodes of the first layer hidden layer and the second layer hidden layer. The VTELM algorithm solves the connection weight between the output layer and the second hidden layer, which can make the final output of the VTELM neural network tend to the actual desired output result. And in this algorithm, only the neuron nodes of the first hidden layer and the second hidden layer need to be artificially set, and the optimal solution can be obtained without setting other parameters.

The network structure of the VTELM neural network is shown in Figure 5, and the algorithm flow can be represented in Figure 6. It can be seen from Figure 6 that VETLM only needs to solve the value of β to get the optimal output.
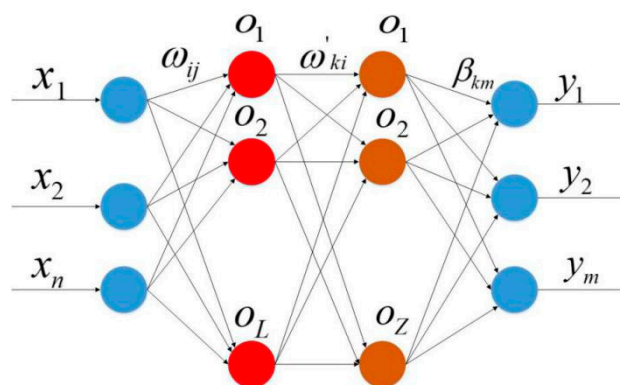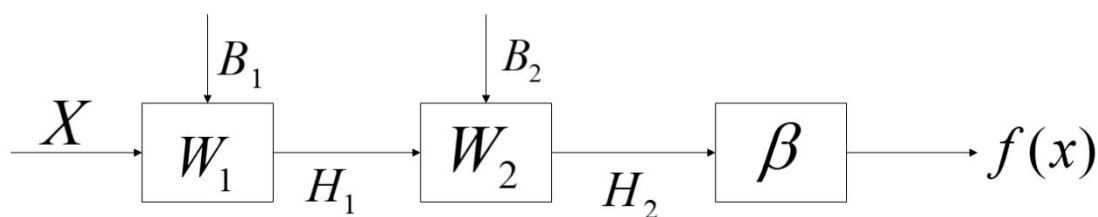

Figure 5. Network structure of VTELM.

Figure 6. The work flow of the VTELM.

In Figure 5, $\{x_1, x_2, \ldots, x_n\}$ represents the input characteristics of training samples, ω_{ij} represents the connection weight between the j-th neuron node in the input layer and the i-th neuron node in the first hidden layer; ω_{ki} represents the connection weight value between the k-th neuron node in the second hidden layer and the i-th neuron node in the first hidden layer; β_{km} refers to the connection between the k-th neuron node in the second hidden layer and the m-th neuron node in the output layer weight; $\{y_1, y_2, \ldots, y_m\}$ represents the output characteristics of training samples.

In VTELM algorithm, in order not to lose generality, suppose the connection weight ω_1 between the first hidden layer and the input layer and the neuron threshold b_1 of the first hidden layer are set as follows:

$$
w_1 = \begin{bmatrix} \omega_{11} & \omega_{12} & \bullet\bullet\bullet & \omega_{1n} \\ \omega_{21} & \omega_{22} & \bullet\bullet\bullet & \omega_{2n} \\ \bullet & \bullet & & \bullet \\ \bullet & \bullet & \bullet\bullet\bullet & \bullet \\ \bullet & \bullet & & \bullet \\ \omega_{l1} & \omega_{l2} & \bullet\bullet\bullet & \omega_{ln} \end{bmatrix}_{l\times n} \quad b_1 = \begin{bmatrix} b_1 \\ b_2 \\ \bullet \\ \bullet \\ \bullet \\ b_l \end{bmatrix}_{l\times 1}
\tag{15}
$$

Let the connection weight ω_2 between the first hidden layer and the second hidden layer and the threshold b_2 of neurons be set as:

$$
w_2 = \begin{bmatrix} \omega_{11} & \omega_{12} & \bullet\bullet\bullet & \omega_{1l} \\ \omega_{21} & \omega_{22} & \bullet\bullet\bullet & \omega_{2l} \\ \bullet & \bullet & & \bullet \\ \bullet & \bullet & \bullet\bullet\bullet & \bullet \\ \bullet & \bullet & & \bullet \\ \omega_{z1} & \omega_{z2} & \bullet\bullet\bullet & \omega_{zl} \end{bmatrix}_{z\times l} \quad b_2 = \begin{bmatrix} b_1 \\ b_2 \\ \bullet \\ \bullet \\ \bullet \\ b_z \end{bmatrix}_{z\times 1}
\tag{16}
$$

The connection weight β between the second hidden layer and the output layer is:

$$
\beta = \begin{bmatrix} \beta_{11} & \beta_{12} & \bullet\bullet\bullet & \beta_{1m} \\ \beta_{21} & \beta_{22} & \bullet\bullet\bullet & \beta_{2m} \\ \bullet & \bullet & & \bullet \\ \bullet & \bullet & \bullet\bullet\bullet & \bullet \\ \bullet & \bullet & & \bullet \\ \omega_{z1} & \beta_{z2} & \bullet\bullet\bullet & \beta_{zm} \end{bmatrix}_{z\times m}
\tag{17}
$$

Firstly, the number of training samples is Q, the input eigenmatrix is X, and the expected output eigenmatrix is Y. There are n input features in the sample input, so there are n neuron nodes in the input layer of VTELM; There are m output features in the sample output, so there are m neuron nodes in the output layer of VTELM; Then assume that there are l neurons in the first hidden layer and z neurons in the second hidden layer, and the activation functions of the two layers are the same.

$$
X = \begin{bmatrix} x_{11} & x_{12} & \bullet\bullet\bullet & x_{1Q} \\ x_{21} & x_{22} & \bullet\bullet\bullet & x_{2Q} \\ \bullet & \bullet & & \bullet \\ \bullet & \bullet & \bullet\bullet\bullet & \bullet \\ \bullet & \bullet & & \bullet \\ x_{n1} & x_{n2} & \bullet\bullet\bullet & x_{nQ} \end{bmatrix}_{n\times Q} \quad Y = \begin{bmatrix} y_{11} & y_{12} & \bullet\bullet\bullet & y_{1Q} \\ y_{21} & y_{22} & \bullet\bullet\bullet & y_{2Q} \\ \bullet & \bullet & & \bullet \\ \bullet & \bullet & \bullet\bullet\bullet & \bullet \\ \bullet & \bullet & & \bullet \\ y_{m1} & y_{m2} & \bullet\bullet\bullet & y_{mQ} \end{bmatrix}_{m\times Q}
\tag{18}
$$

Then, in the algorithm design of VTELM, the connection weight ω_1 between the first hidden layer and the input layer and the neuron threshold b_1 of the first layer are initialized randomly, then the output matrix H_1 of the first hidden layer is:

$$
H_1 = [g(\omega_1 * x) + b_1]
\tag{19}
$$

The weights ω_2 of the connection between the second hidden layer and the first layer and the threshold value b_2 of the second layer neurons are initialized randomly. Then, H_1 is used as the input matrix of the second hidden layer to calculate the output matrix H_2 of the second hidden layer:

$$
H_2 = [g(\omega_2 * H_1) + b_2]
\tag{20}
$$

Therefore, parameter β is:

$$H_2\beta = T \tag{21}$$

It can be concluded that:

$$\overset{\wedge}{\beta} = H_2^+ T \tag{22}$$

In order to prevent the output from overfitting, improve the robustness and generalization performance of the network, and make the network more stable, a regularization term is added to the solution process β to constrain [38,39]. It can be expressed as:

$$\underset{\beta}{\mathrm{argmin}}E(W) = \underset{\beta}{\mathrm{argmin}}(\frac{1}{2}\|\beta\|^2 + \frac{1}{2}\lambda\|\varepsilon\|^2), \text{ s.t.}\sum_{i=1}^{Z}\beta_i g(\omega_i\bullet x_j + b_i) - t_j = \varepsilon_j, j = 1,2,\ldots Q \tag{23}$$

In Equation (23), the sum of squares of $\|\varepsilon\|^2$ error represents empirical risk; $\|\beta\|^2$ represents structural risk, which originates from the principle of maximizing marginal distance in statistical theory. By using Lagrange equation, the conditional extremum problem of Equation (23) can be transformed into unconditional extremum problem:

$$l(\beta, \varepsilon, \alpha) = \frac{1}{2}\|\beta\|^2 + \frac{\lambda}{2}\|\varepsilon\|^2 - \sum_{j=1}^{Z}\alpha_j(g(\omega_i\bullet x_j + b_i) - t_j - \varepsilon_j)z \tag{24}$$

The simplified Equation (24) is:

$$l(\beta, \varepsilon, \alpha) = \frac{1}{2}\|\beta\|^2 + \frac{\lambda}{2}\|\varepsilon\|^2 - \alpha(H_2\beta - T - \varepsilon) \tag{25}$$

$\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_N]; \alpha_1 \in R^m(j = 1,2,\ldots N)$ stands for the Lagrange multiplier. Let the gradient of Equation (25) be 0:

$$\begin{cases} \frac{\partial l}{\partial \beta} = \beta^T - \alpha H_2 = 0 \\ \frac{\partial l}{\partial \varepsilon} = \lambda\varepsilon^T + \alpha = 0 \\ \frac{\partial l}{\partial \alpha} = H_2\beta - T - \varepsilon = 0 \end{cases} \tag{26}$$

If the input feature N is larger than any layer of neuron nodes, that is, $N > L$ or $N > Z$, it can be obtained by Equation (26):

$$\beta = (\frac{I}{\lambda} + H_2^T H_2)^{-1} H_2^T T \tag{27}$$

If the input feature N is less than any layer of neuron nodes, that is, $N < L$ or N V$< Z$, it can be obtained by Equation (26):

$$\beta = H_2{}^T(\frac{I}{\lambda} + H_2 H_2^T)^{-1} T \tag{28}$$

If the input feature n is equal to the neuron node of each layer, that is, $N = L = Z$, then:

$$\beta = H_2^+ T \tag{29}$$

where H_2^+ is the Moore Penrose generalized inverse of H_2. The methods of calculating the Moore-Penrose generalized inverse of a matrix include: full rank decomposition method, inverse matrix method, singular value decomposition method (SVD). When $H_2^T H_2$ is nonsingular $H_2^+ = (H_2^T H_2)^{-1} H_2^T$, or when $H_2 H_2^T$ is nonsingular $H_2^+ = H_2^T (H_2 H_2^T)^{-1}$.

Then the output of VTELM network is obtained as follows:

$$y = H_2\beta \tag{30}$$

The basic steps of the final VTELM algorithm are:

(1) Firstly, we assume that the given training sample set is $\{X,T\} = \{x_x,t_i\}(i = 1,2,\dots ,Q)$. Hidden layer chooses the most appropriate activation function;

(2) Randomly select values for the weight ω_1 and threshold B_1 of the first hidden layer, and calculate the output matrix H_i of the first hidden layer by $H_1 = [g(\omega_1 \times x)+b_1]$;

(3) Randomly select values for the weight ω_2 and threshold B_2 of the second hidden layer, and calculate the output matrix H_2 of the second hidden layer by $H_2 = [g(\omega_2 \times H_1)+b_2]$;

(4) Solve the connection weight β of the second layer and the output layer, and compare the input features (N) and number of neuron nodes size (L,Z);

(5) If $N < L$ or $N < Z$, then $\beta = H_2^T(\frac{I}{\lambda} + H_2H_2^T)^{-1}T$;

(6) If $N > L$ or $N > Z$, then $\beta = (\frac{I}{\lambda} + H_2^TH_2)^{-1}H_2^TT$;

(7) If these two values are equal: $\beta = H_2^+T$;

(8) Calculate the final output of the VTELM algorithm: $f(x) = g\{\omega_2\bullet[g(\omega_1\bullet x) + b_1] + b_2\}\beta$.

4. Experimental Results and Discussion

4.1. Processing of Copper Ore Spectral Data

The spectral data obtained by the copper ore sample through the spectral test has 973 dimensions. Due to the high dimensionality of copper ore and information redundancy, the network scale will be too complicated, the network training accuracy will decrease and the modeling effect will be worse. Therefore, principal component analysis (PCA) is used to simplify the spectral data, and the cumulative contribution rate can reach 99.8% when the principal component dimension is 15, and then the 15 dimensional principal component is used as the input of the network.

Figure 7 shows the cumulative contribution rate of the first three principal components and Figure 8 shows the spatial distribution of the first three principal components. Therefore, the initial data is compressed from 973×241 to 15×241, which reduces the interference of redundant information, improves the operation speed of the network and improves the accuracy of the model.
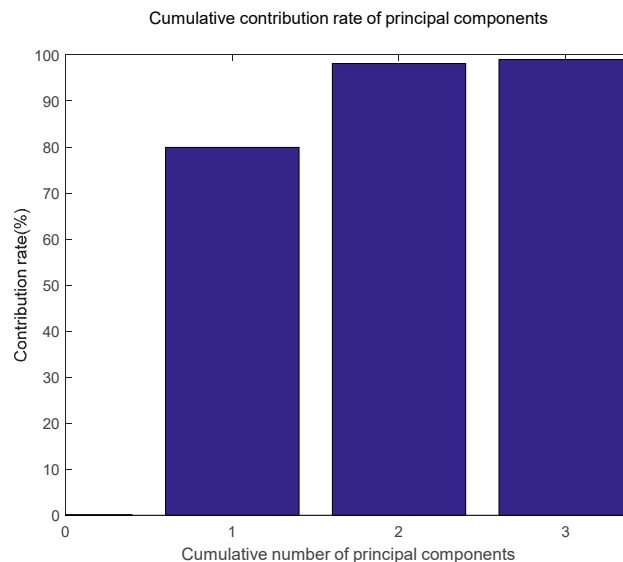

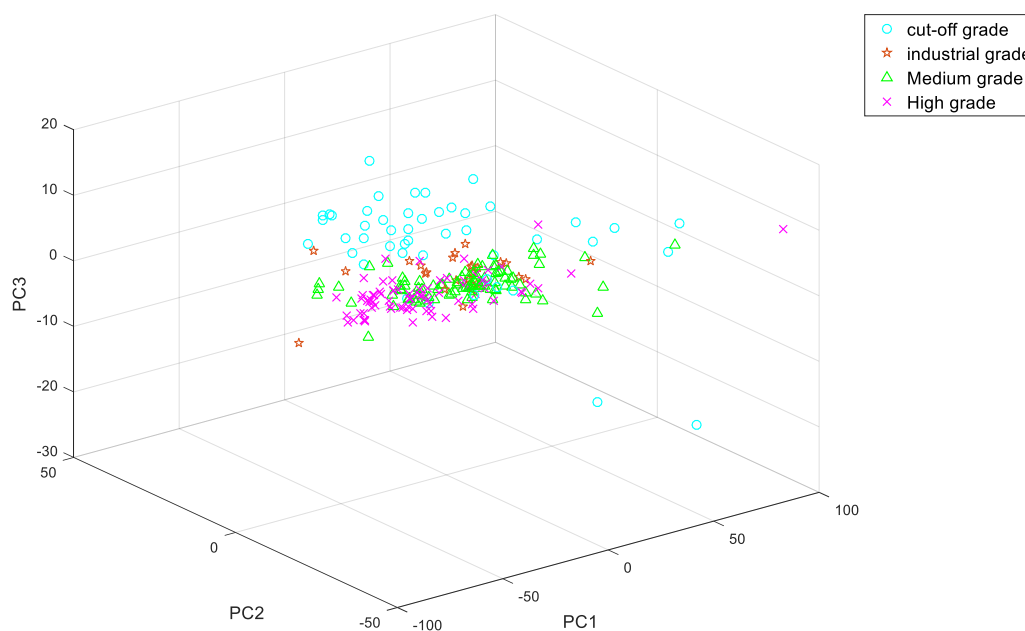**Figure 7.** Cumulative contribution rate of principal component.

Figure 8. Spatial distribution map.

4.2. Neural Network Comparison

First, this paper used ELM, BP, and RBF to establish inversion models of the copper content in copper ore. Because of the small number of samples collected, in order to maximize the use of 251 sets of data to test different models, our three models have adopted cross validation method for 10 times of copper ore cross validation. Table 2 shows the test results of the three models, and compares the models in terms of time consumption (s), root mean square error (RMSE) and coefficient of determination (R^2).

Table 2. Copper content detection models based on different neural networks.

Model Type	Time Consumption (s)	R^2	RMSE
BP	0.202432	0.62688	0.15404
ELM	0.025085	0.62834	0.13653
RBF	0.062342	0.13653	1.6936

In Table 2, it can be seen that the root mean square error of ELM is 0.13653, which is the lowest of the three models. The root mean square error of BP is 0.15404, which is close to that of ELM, but the training time of BP is 12 times that of ELM. Therefore, the ELM model is most suitable for copper ore content detection.

4.3. Comparison of Copper Content Detection Models of Different Models

By comparing the experimental results of BP, ELM and RBF neural networks, we choose to use ELM to detect copper ore content. However, due to the limitation of ELM, VTELM model is proposed in this paper. In order to verify the superiority of VTELM, this paper uses ELM, TELM and VTELM to establish the inversion model of copper content in copper ore, and the experiment is simulated in MATLAB r2016a environment. Finally, by comparing the root mean square error, prediction time and coefficient of determination of the prediction set, the performance of the copper content detection model was evaluated.

Figure 9 shows the single optimal cross validation results of Cu content models of ELM, TELM and VTELM for copper ores. Table 3 lists the performance of ELM, TELM and VTELM from the coefficient of determination (R^2), root mean square error (RMSE), time consumption (S), and the number of hidden layer nodes. The results of specific analysis are as follows:
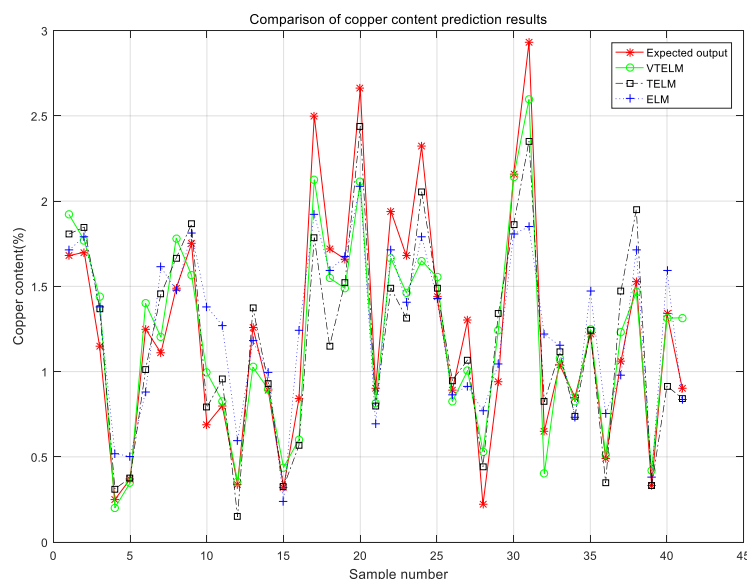
Figure 9. Experimental results of different models.

Table 3. Results of copper content detection model test.

Model Type	R^2	RMSE	S	Number of Hidden Layer Nodes
ELM	0.74822	0.12112	0.020792	11
TELM	0.83589	0.075211	0.135268	48
VTELM	0.88309	0.055629	0.027361	46/137

First of all, from the test results, compared with ELM and TELM, VTELM has the smallest root mean square error, which shows that the output value of VTELM is closer to the true value, so it can detect the Cu content in copper ore more accurately;

Secondly, compared with ELM and TELM, VTELM model has higher coefficient of determination, which indicates that VTELM has better goodness of fit and can be closer to the real value.

Third, the training time of TELM was significantly longer than that of ELM and VTELM, while the training time of ELM and VTELM was similar. This is because VTELM does not need to solve the connection weights of the two hidden layers and the threshold of the second hidden layer, and the time consumed is mainly used to solve the output matrix β.

Finally, for hidden layer nodes, ELM, TELM and VTELM all adopt a trial method, that is, hidden layer nodes iterate from 0 to 200 to find the best hidden layer nodes. For the copper ore Cu content detection model, ELM needs 11 hidden layer nodes, TELM needs 48 hidden layer nodes, and for VTELM, the first layer hidden layer nodes are 46, the second layer hidden layer nodes are 137. The more nodes in the hidden layer, the stronger the nonlinearity can be expressed by the neural network, which can describe the nonlinear characteristics of the fitting objective function more accurately, but the generalization performance will be reduced [40,41]. However, VTELM requires the most hidden layer nodes, and the generalization ability does not decrease. The results show that VTLEM has the highest accuracy and the strongest generalization ability.

In summary, VTELM has stronger generalization ability than TELM and ELM, the model is more accurate, and the calculation is more simple. Table 4 shows the detection method, chemical analysis method and instrument analysis method of VTELM based on copper ore spectral data, and compares them from three aspects of accuracy, time consumption and cost. It can be seen that the detection accuracy of the instrumental analysis method is low, and the chemical detection method has a high accuracy, but the cost is high and the time is long. The copper content detection method in copper ore proposed in this paper based on spectral data and VTELM algorithm has the advantages of short

time-consuming, low cost, and high prediction accuracy, which can meet the needs of copper ore content identification.

Table 4. Comparison of detection methods.

Test Method	Detection Accuracy (%)	Time Consumed (h)	Cost Detection (yuan)
Instrument testing	73	3	About 400
Chemical method	99	70	About 21,000
VTELM	98.4	3	About 300

In summary, the copper content detection model based on spectral data proposed in this paper is an improved two hidden layer extreme learning machine algorithms with variable hidden layer nodes, which has the optimal output matrix β and the weight matrix and hidden layer nodes of each hidden layer. Its generalization ability is better than ELM and TELM, its running speed is faster than TELM, and its coefficient of determination is the highest. Compared with the traditional instrument analysis method, this method is simple to operate, high in accuracy, fast in detection speed and low in cost.

5. Conclusions

In view of the shortcomings of traditional copper ore detection methods, this paper proposes a copper content detection model based on spectral data and VTELM, and validates the model with collected copper ore samples. First, we collected 241 copper ore samples, used chemical analysis to detect the copper ore to obtain the accurate copper content in the copper ore, and then performed a spectral test on it to obtain the initial spectral data. Since the initial spectral data has a large dimension and contains other useless information, PCA is used to reduce the dimension. Then, BP, ELM, and RBF were modeled separately using the reduced-dimensional spectral information. The modeling results showed that ELM has the fastest running speed and the smallest root mean square error. Due to the limitations of the ELM model, and in order to further improve the generalization ability of the model and reduce the difference between the predicted output of the model and the actual output, this paper proposes the VTELM model and conducts modeling experiments on ELM, TELM, and VTELM. By comparing the three models of ELM, TELM, and VTELM, VTELM has the highest coefficient of determination, the smallest error, and the running time is much lower than that of TELM, almost the same as ELM. Compared with the traditional methods, the copper content inversion model based on the reflectance spectrum and VTELM algorithm has the advantages of high accuracy, high reliability and fast detection speed. This paper provides a new idea for the detection method of the copper content in copper ore.

References

1. Radivojevic, M.; Rehren, T.; Pernicka, E.; Sljivar, D.; Brauns, M.; Boric, D. On the origins of extractive metallurgy: New evidence from Europe. *J. Archaeol.* **2010**, *37*, 2775–2787. [CrossRef]
2. Jiang, S.Q. Distribution of copper resources in the world. *World Nonferr. Met.* **2018**, *2*, 1–3.
3. Zhou, C.Y.; Qu, W. Rapid Determination of Copper and Iron in Ore Leaching Solution by Iodometry. In Proceedings of the 3rd International Conference on Chemical Engineering and Advanced Materials (CEAM 2013), Guangzhou, China, 6–7 July 2013.

4.   Gao, W. Determination of Copper, Lead, Zinc, Cobalt and Nickel in Ore by Flame Atomic Absorption Spectrophotometry. *World Nonferr. Met.* **2019**, *04*, 171–172.

5.   Shi, C.Y.; Xie, S.A.; Jia, J.P. The Study of a New Method to Determine Copper Ion by Square-Wave Voltammetry-Extraction Iodometry at the Liquid/Liquid Interfaces. *J. Autom. Methods Manag. Chem.* **2008**, *2008*, 453429. [CrossRef]

6.   Song, U. Improvement of soil properties and plant responses by compost generated from biomass of phytoremediation plant. *Environ. Eng. Res.* **2020**, *5*, 638–644. [CrossRef]

7.   Luo, F.; Hu, X.F.; Oh, K.; Yan, L.J.; Lu, X.Z.; Zhang, W.J.; Yonekura, T.; Yonemochi, S.; Isobe, Y. Using profitable chrysanthemums for phytoremediation of Cd- and Zn-contaminated soils in the suburb of Shanghai. *J. Soils Sediments* **2020**, *20*, 4011–4022. [CrossRef]

8.   Saleem, M.H.; Rehman, M.; Kamran, M.; Afzal, J.; Noushahi, H.A.; Liu, L.J. Investigating the potential of different jute varieties for phytoremediation of copper-contaminated soil. *Environ. Sci. Pollut. Res.* **2020**, *27*, 30367–30377. [CrossRef]

9.   Muro-Gonzalez, D.A.; Mussali-Galante, P.; Valencia-Cuevas, L.; Flores-Trujillo, K.; Tovar-Sanchez, E. Morphological, physiological, and genotoxic effects of heavy metal bioaccumulation inProsopis laevigatareveal its potential for phytoremediation. *Environ. Sci. Pollut. Res.* **2020**, *32*, 40187–40204. [CrossRef]

10.   Chauhan, P.; Mathur, J. Phytoremediation efficiency of Helianthus annuus L. for reclamation of heavy metals-contaminated industrial soil. *Environ. Sci. Pollut. Res.* **2020**, *27*, 29954–29966.

11.   Mao, Y.C.; Xiao, D.; Cheng, J.P.; Jiang, J.H.; Le, B.T.; Liu, S.J. Research in magnesite grade classification based on near infrared spectroscopy and ELM algorithm. *Spectrosc. Spectr. Anal.* **2017**, *37*, 89–94.

12.   Wang, N.; Zhang, L. Beer Freshness Detection Method Based on Spectral Analysis Technology. *Spectrosc. Spectr. Anal.* **2020**, *7*, 2273–2277.

13.   Mortet, V.; Gregora, I.; Taylor, A.; Lambert, N.; Ashcheulov, P.; Gedeonova, Z.; Hubik, P. New perspectives for heavily boron-doped diamond Raman spectrum analysis. *Carbon* **2020**, *168*, 319–327. [CrossRef]

14.   Yang, R.Y.; Qiao, T.Z.; Pang, Y.S.; Yang, Y.; Zhang, H.T.; Yan, G.W. Infrared Spectrum Analysis Method for Detection and Early Warning of Longitudinal Tear of Mine Conveyor Belt. *Measurement* **2020**, *165*, 107856. [CrossRef]

15.   Xiao, D.; Liu, C.M.; Le, B.T. Detection method of TFe content of iron ore based on visible-infrared spectroscopy and IPSO-TELM neural network. *Infrared Phys. Technol.* **2019**, *97*, 341–348. [CrossRef]

16.   Le, B.T.; Mao, Y.C.; He, D.K. Coal analysis based on visible-infrared spectroscopy and a deep neural network. *Infrared Phys. Technol.* **2018**, *93*, 34–40. [CrossRef]

17.   Percival, J.B.; Bosman, S.A.; Potter, E.G.; Peter, J.M.; Laudadio, A.B.; Abraham, A.C.; Shiley, D.A.; Sherry, C. Customized Spectral Libraries for Effective Mineral Exploration: Mining National MIneral Collections. *Clays Clay Miner.* **2018**, *66*, 297–314. [CrossRef]

18.   Zhou, M.; Zou, B.; TU, Y.L.; Xia, J.P. Hyperspectral Modeling of Pb Content in Mining Area Based on Spectral Feature Band Extracted from Near Standard Soil Samples. *Spectrosc. Spectr. Anal.* **2020**, *7*, 2182–2187.

19.   Zhao, Y.L.; Zheng, W.X.; Xiao, W.; Zhang, S.; Lv, X.J.; Zhang, J.Y. Rapid monitoring of reclaimed farmland effects in coal mining subsidence area using a multi-spectralUAV platform. *Environ. Monit. Assess.* **2020**, *7*, 474. [CrossRef]

20.   Shin, J.H.; Yu, J.; Wang, L.; Kim, J.; Koh, S.M.; Kim, S.O. Spectral Responses of Heavy Metal Contaminated Soils in the Vicinity of a Hydrothermal Ore Deposit: A Case Study of Boksu Mine, South Korea. *IEEE Trans. Geosci. Remote Sens.* **2019**, *6*, 4092–4106. [CrossRef]

21.   Zhang, S.W.; Shen, Q.; Nie, C.J.; Huang, Y.F.; Wang, J.H.; Hu, Q.Q.; Ding, X.J.; Zhou, Y.; Chen, Y.P. Hyperspectral inversion of heavy metal content in reclaimed soil from a mining wasteland based on different spectral transformation and modeling methods. *Spectrochim. Acta Part A Mol. Biomol. Spectrosc.* **2019**, *211*, 393–400. [CrossRef]

22.   Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 37–52. [CrossRef]

23.   Parida, A.K.; Maity, K.; Ghadei, S.B. Optimization of hot turning parameters using principal component analysis method. *Mater. Today Proc.* **2020**, *22*, 2081–2087. [CrossRef]

24.   Machidon, A.L.; Del Frate, F.; Picchiani, M.; Machidon, O.M.; Ogrutan, P.L. Ogrutan, Geometrical Approximated Principal Component Analysis for Hyperspectral Image Analysis. *Remote Sens.* **2020**, *12*, 1698. [CrossRef]

25. Sell, S.L.; Widen, S.G.; Prough, D.S.; Hellmich, H.L. Principal component analysis of blood microRNA datasets facilitates diagnosis of diverse diseases. *PLoS ONE* **2020**, *15*, e0234185. [CrossRef] [PubMed]

26. Liu, Y.; Zhao, T.L.; Ju, W.W.; Shi, S.Q. Materials discovery and design using machine learning. *J. Mater.* **2017**, *3*, 159–177. [CrossRef]

27. Xiong, Z.; Cui, Y.X.; Liu, Z.H.; Zhao, Y.; Hu, M.; Hu, J.J. Evaluating explorative prediction power of machine learning algorithms for materials discovery using k-fold forward cross-validation. *Comput. Mater. Sci.* **2020**, *171*, 109203. [CrossRef]

28. Liu, Y.; Guo, B.R.; Zou, X.X.; Li, Y.J.; Shi, S.Q. Machine learning assisted materials design and discovery for rechargeable batteries. *Energy Storage Mater.* **2020**, *31*, 434–450. [CrossRef]

29. Kilic, A.; Odabaşı, Ç.; Yildirim, R.; Erogluss, D. Assessment of critical materials and cell design factors for high performance lithium-sulfur batteries using machine learning. *Chem. Eng. J.* **2020**, *390*, 124117. [CrossRef]

30. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: A new learning scheme of feedforward neural networks. In Proceedings of the 2004 IEEE international joint conference on neural networks, Budapest, Hungary, 25–29 July 2004; Volume 2, pp. 985–990.

31. Zong, W.; Huang, G.B. Face recognition based on Extreme learning machine. *Neurocomputing* **2011**, *74*, 2541–2551. [CrossRef]

32. Huang, G.B.; Zhou, H.; Ding, X.; Zhang, R. Extreme learning machine for regression and multiclass classification. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2011**, *42*, 513–529. [CrossRef]

33. Huang, Z.Y.; Yu, Y.L.; Gu, J.S.; Liu, H.P. An efficient method for traffic sign recognition based on Extreme learning machine. *IEEE Trans. Cybern.* **2016**, *47*, 920–933. [CrossRef] [PubMed]

34. Kang, F.; Liu, J.; Li, J.J.; Li, S.J. Concrete dam deformation prediction model for health monitoring based on extreme learning machin. *Struct. Control Health Monit.* **2017**, *24*, e1997. [CrossRef]

35. Liang, N.Y.; Huang, G.B. A Fast and accurate online sequential learning algorithm for feedforward networks. *IEEE Trans. Neural Netw.* **2006**, *17*, 1411–1423. [CrossRef] [PubMed]

36. Lan, Y.; Soh, Y.C.; Huang, G.B. Ensemble of online sequential Extreme learning machine. *Neurocomputing* **2009**, *72*, 3391–3395. [CrossRef]

37. Qu, B.Y.; Lang, B.G. Two-hidden-layer Extreme learning machine for regression and classification. *Neurocomputing* **2016**, *175*, 826–834. [CrossRef]

38. Feng, T. Imputing Missing Data in Large-Scale Multivariate Biomedical Wearable Recordings Using Bidirectional Recurrent Neural Networks with Temporal Activation Regularization. In Proceedings of the 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, 23–27 July 2019; pp. 2529–2534.

39. Girosi, F.; Jones, M.; Poggio, T. Regularization theory and neural networks architectures. *Neural Comput.* **1995**, *7*, 219–269. [CrossRef]

40. Wang, R.B.; Xu, H.Y.; Feng, Y. Research on Method of Determining Hidden Layer Nodes in BP Neural Network. *Comput. Technol. Dev.* **2018**, *28*, 31–35.

41. Wang, J.; Xue, F. Self-adaptive Nonlinear Approximation Algorithm of RBF Neural Network. *Mod. Electron. Tech.* **2011**, *34*, 141–143.