RESEARCH ON NODE NETWORK TRANSMISSION CAPACITY PREDICTION MODEL FOR LARGE SCALE REMOTE SENSING DATA COLLECTION

Lu Bai [1], Xiaogang Liu [2], Man Zhao [3], Zhibao Wang [2, 4 *], Guiying Shi [2]

[1] School of Computing, Ulster University, Belfast BT15 1ED, UK
[2] School of Computer and Information Technology, Northeast Petroleum University, Daqing 163318, China
[3] School of Communication and Electronic Engineering, Qiqihaer University, Qiqihaer 161003, China
[4] Bohai-Rim Energy Research Institute, Northeast Petroleum University, Qinhuangdao 066004, China

KEY WORDS: PSO, BP, prediction model, remote sensing data, network transmission capacity.

ABSTRACT:

In recent years, the use of remote sensing technology has grown exponentially in various industries such as agriculture, forestry, and urban planning. Remote sensing data collection systems rely on a network of nodes to collect and transmit data. The transmission capacity of these node networks is a critical factor in the performance and efficiency of the entire system. However, accurately predicting the transmission capacity of a node network can be a challenging task. To carry out large scale open remote sensing data collection, it is necessary to predict the network transmission capacity of nodes in the face of the difference in the execution speed of each node for various tasks. It is necessary to predict the network transmission capacity of nodes. In this research, we propose a node network transmission capacity prediction model for large scale remote sensing data collection using a combination of Particle Swarm Optimization (PSO) and Backpropagation (BP) algorithms. The proposed PSO-BP model aims to accurately predict the transmission capacity of a node network in a remote sensing data collection system. The model is tested and evaluated using a large-scale dataset and the results show that the proposed model outperforms existing models in terms of prediction accuracy. This work contributes to the field of remote sensing data collection by providing a reliable and efficient method for predicting the transmission capacity of node networks.

1. INTRODUCTION

Space remote sensing technology is entering the era of industrial applications, in which Landsat, Sentinel and other public interest satellite remote sensing data play an important role (Hemati et al., 2021; Phiri et al., 2020; Segarra et al., 2020). With the development of remote sensing technologies and advancement of deep learning-based algorithms, significant progress has been achieved in recent years in many remote sensing tasks, for example, object detection from remote sensing images (Cheng and Han, 2016; Deng et al., 2018; Li et al., 2020; Qian et al., 2020; Song et al., 2020; Wang et al., 2021; Zhang et al., 2019; Zhu et al., 2021), remote sensing change detection (Hecheltjen et al., 2014; Jensen and Im, 2007; Khelifi and Mignotte, 2020; Shi et al., 2022; Q. Wang et al., 2018; Zhang et al., 2021) and remote sensing big data (Chi et al., 2016; Deren et al., 2014; Liu et al., 2018; Ma et al., 2021, 2015; Yu et al., 2021a, 2021b). Remote sensing data user groups such as individuals, small and medium-sized enterprises are gradually increasing. In the process of data application, satellites are generally publicly released on the Internet for global users, which is limited by the service capability of the data source itself and the acquisition capability of data user nodes. Due to the constraints of data source's own service capability and data user node acquisition capability, the problems of low efficiency of large-scale public remote sensing data acquisition and low utilization rate of user acquisition nodes still exist (Lee et al., 2011). The problems such as low efficiency of large-scale public remote sensing data collection and low utilization rate of user collection nodes still exist (Wang et al., 2022).

However, there are few studies on network node transmission capacity prediction. To carry out large scale open remote sensing data collection in crowdsourcing mode, it is necessary to predict the network transmission capacity of nodes considering the difference in the execution speed of each node for various tasks. The traditional Backpropagation (BP) algorithm uses the inertia weights between the input and hidden layers and the inertia weights between the hidden layer and the output layer to establish the prediction model. The inertia weights between the hidden layer and the output layer are randomly initialized, and the learning factors are randomly generated constants. The algorithm has a strong sensitivity to these coefficients during the training process, and it is easy to fall into long iteration time and local extremes. This study analyses the possible factors affecting the transmission speed of the node network at the levels of data source, transmission medium and receiving terminal of open remote sensing data collection. A range of controllable factors were selected to establish a prediction model of network transmission capacity. For the traditional BP neural network algorithm, the inertia weights of network algorithm, it is easy to fall into local minima and other shortcomings. The Particle Swarm Optimization (PSO) algorithm with good robustness and global search capability is combined with the BP neural network algorithm to propose an improved PSO-BP algorithm. The improved algorithm includes the introduction of gradient, momentum factor, and inertia weight adjustment function. To further improve the accuracy of the algorithm in the iteration, the accuracy dynamic adjustment function is introduced, and the learning factor is also introduced into dynamic adjustment function. Finally, a comparison experiment between the

* Corresponding author

improved PSO-BP algorithm and the traditional BP algorithm is conducted on a real data set collected from Sentinel 2.

2. NODE NETWORK TRANSMISSION CAPACITY PREDICTION MODEL

2.1 Analysis of factors affecting the transmission capacity of node networks

This paper considers the data transmission in large-scale public remote sensing data collection under the crowdsourcing model. During the collection process, data is transmitted from the public remote sensing data source through multiple transmission media to the collection node. The data transmission process mainly involves "two sources and one medium", which are the data source, the transmission medium, and the receiving terminal. The paper analyses the characteristics of the three stages of factors affecting node network transmission capacity, and Figure 1 shows the factors affecting node network transmission capacity.

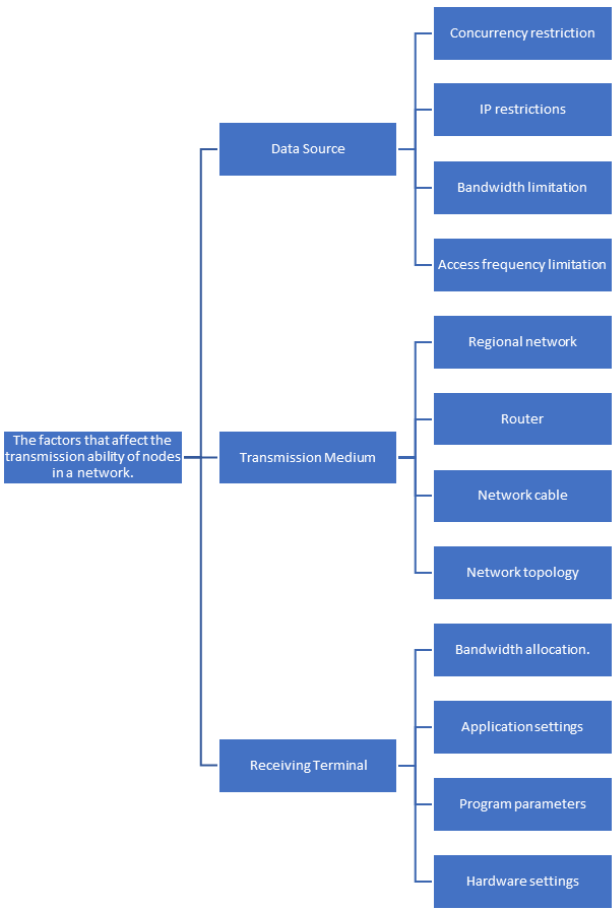

Figure 1. The factors that affect the transmission ability of nodes in a network

2.1.1 Constraint analysis based on the data source side

The main factors that affect the transmission capacity of a node network at the data source are related to the physical configuration and server parameter settings of the data source server.

In order to ensure the long-term stability and security of the data source, the number of concurrent accesses, IPs, access frequencies, access bandwidth speeds, allowed access time periods, and allowed access areas of users are usually restricted based on the physical configuration of the server node.

Through long-term collection, it has been found that when a single user frequently accesses the USGS data source for a long period of time, the user's order production speed is relatively slow, and the data download speed in the morning is much faster than that in the afternoon. The data source is closed for maintenance every Wednesday in Beijing time. User information must be provided when downloading, and the download speed is the fastest in the early morning in Beijing time.

2.1.2 Constraint analysis based on transmission medium

In the long-term practice process, it is found that the acquisition nodes located in different geographical locations also have significant differences. Therefore, it is guessed that when using the network medium for data transmission, the total network sites in different spatial regions sources differ, and the service capacity of configured network bandwidth varies from one spatial region to another. Similarly, in the same spatial region under the same network configuration environment, different physical facilities such as routers, cables, switches, similarly affect the data in transmission speed in the network medium.

To verify the effect of network topology on the transmission speed of data in the network medium, relevant experiments on relay service nodes were conducted. First, taking the public data source as the source end and the relay node as the receiving end, the relay service node host directly collected a certain amount of remote sensing data from the above-mentioned public data source, and stored it in the local physical hard disk of the relay service node. Then, taking the relay node as the source end and the various collection nodes as the receiving end, other collection nodes download data from the relay node to the local. The experimental results show that the speed of using the relay node to collect public remote sensing data is indeed much higher than that of not using the relay node, but due to the high cost of using the relay node, it is not suitable for large-scale long-term public remote sensing data collection.

2.1.3 Constraint analysis based on the acquisition terminal

Among the many factors that affect the network transmission capability of the acquisition terminal, the node's own network bandwidth configuration plays a dominant role. Having a higher network bandwidth configuration is more beneficial for network data transmission. During the work process of the collecting terminal, when other applications occupy a large amount of network bandwidth, it seriously affects the node's ability to collect data using the network bandwidth. When the terminal uses third-party software to limit the network bandwidth of the collection client, it also affects the efficiency of data network transmission. The number of threads, task type, start time, and task length set by the terminal collection client also affect the network transmission speed of the node, and these factors are controllable factors of the collection terminal, so these controllable parameters can be used as input for subsequent network transmission ability prediction models. In addition, factors such as disk read and write speed, network card, and other physical hardware facilities on the node also affect the network transmission ability of the node.

2.2 Model parameter feature selection

To establish a model for predicting node network transmission capabilities and accurately predict the network transmission capabilities of crowd-sourced collection nodes, the factors that

affect the data in the network transmission environment were analysed in the previous section from the three stages of data source, transmission medium, and receiving terminal. Many of these factors are uncertain and not controlled by the receiving terminal, making it difficult to discover their intrinsic patterns. Therefore, they cannot be considered in the node network transmission capability prediction model. Considering the need for publicly available remote sensing data collection, and from the perspective of being long-term, stable, inexpensive, and practical, many uncontrollable factors are discarded. Without considering many data source-side factors and physical factors, a selection of controllable factors is made from the perspective of dynamic adjustments to the client application parameters of the collection terminal. Using machine learning theory, a model for predicting node network transmission capabilities is established. The model is trained using the node's actual historical collection records, enabling the prediction of the node's network transmission speed. Table 1 shows the selected features that affect network transmission capabilities.

Feature Source	Feature Name
Data source	Number of concurrencies
	Access frequency
	Access time
	Order submission time
	Order capacity
	Whether it is a working day
Transmission medium	Space area
	Networking method
	Router model
	Switch models
	Network cable type
Receiving terminal	Network bandwidth
	Network card type
	Hard drive transfer rate
	Number of task threads
	Task progress
	Task length
	Collection terminal time: hour
	Collection terminal time: minute
	Collection terminal time: second
	Whether it is a working day

Table 1. Selected features affecting network transmission capability.

3. METHODS

3.1 PSO

PSO is an algorithm with advantages of simplicity, fast convergence and easy implementation, and is widely used in scheduling optimization, data mining, model training and other aspects of swarm intelligence optimization algorithms (D. Wang et al., 2018).

The position of the i-th particle in the population in a D-dimensional search space consisting of N particles is denoted as $X_i = (x_{i1}, x_{i2}, \dots, x_{iD}), i = 1,2, \dots, N$ which are the possible solutions of the problem. The velocity of the particle also consists of a D-dimensional vector, denoted as $V_i = (v_{i1}, v_{i2}, \dots, v_{iD}), i = 1,2, \dots, N$. It determines the direction and distance of the particle moving in the population. In addition, the movement of each particle component of the velocity is bounded by the maximum limiting velocity V_m. when $V_{ij} > V_m$, $V_{ij} = V_m \ (1 \le i \le N, 1 \le j \le D)$. Similarly, the particle s displacement x_{ij} is also bounded by x_m. When $x_{ij} > x_m$, $x_{ij} = x_m \ (1 \le i \le N, 1 \le j \le D)$, the fitness value of the particle at this time can be calculated according to the objective function. To prevent the blind search of particles, usually $x_{ij} \epsilon (-x_m, x_m), v_{ij} \epsilon (-v_m, v_m)$. The equations of its velocity and displacement update during the particle iterative process are as follows.

$$\begin{cases} v_i^{k+1} = w \cdot v_i^k + c_1 r_1 [p_i - x_i^k] + c_2 r_2 [p_g - x_i^k] \\ x_i^{k+1} = x_i^k + v_i^k, i = 1,2, \dots, N \end{cases} \quad (1)$$

In Equation (1), the number of current iterations of the particle is denoted by k, and the inertia weight coefficient is denoted by w. The larger the value of inertia weight, the stronger the global search ability of the particle. c_1 and c_2 denote the learning factors. c_1 describes the influence of the particle by the individual extremes, so that the particle has global search ability and avoids getting into local solutions. c_2 represents the influence of global optimum on the particle. r_1 and r_2 are random numbers between (0,1). w, c_1 and c_2 are the three constants that jointly determine the spatial search ability of the particle. The position of the optimal fitness value calculated by the particle during the iteration process is represented by the individual extreme value, which are expressed as $p_i = (p_{i1}, p_{i2}, \dots, p_{iD}), i = 1,2, \dots, N$. The position of the optimal fitness value computed by all particles in the population during the iteration is denoted by the global extremum p_g: $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$.

3.2 PSO-BP Algorithm

In traditional BP neural network algorithms, the weights and thresholds are randomly generated, so the algorithm has randomness and unreliability. This paper proposes the PSO-BP neural network algorithm. First, the particles of PSO algorithm are initialised as the initialization weights of BP neural network algorithm, then iteration of PSO algorithm starts. Each iteration of PSO algorithm is followed by an execution of BP neural network algorithm. The iteration termination condition of BP neural network algorithm is determined by the prediction accuracy, which is dynamically adjusted as the number of iterations of PSO algorithm increases. The training termination condition of BP neural network algorithm becomes stricter in the later stage of PSO algorithm. After each iteration of BP neural network algorithm, the absolute value of the error is used evaluate the fitness of each particle of PSO algorithm. When the PSO algorithm satisfies its iteration termination condition, the PSO-BP algorithm also ends.

The parameters involved in the algorithm are the population size N, the dimensionality of the particles D, the number of iterations of the particles M, the inertia weight w, the learning factors c1 and c2, the maximum displacement x_m , and the maximum velocity v_m. The BP neural network algorithm uses a three-layer network topology. The value of each particle in the PSO algorithm represents the inertia weight in the BP neural network algorithm. The dimensionality of the particles D can be expressed as:

$$D = D_{in} \cdot D_h + D_h \cdot D_{out}, \quad (2)$$

In equation (2), the D_{in} , D_h, and D_{out} denote the number of neuros in the input layer, hidden layer, and output layer of the BP neural network, respectively.

In the process of predicting the transmission capacity of the nodes, six factors including the number of download task threads, task progress, task length, acquisition terminal time (in hours), acquisition terminal time (in minutes), and data source time (in hours) are used as the six neurons in the input layer. The number of neurons in the output layer is 1, and the number of neurons in the hidden layer is 10. The dimension D= 70, which indicates there are 70 inertia weights. For each particle's position $x_i = (x_{i,1}, x_{i,2}, \dots, , x_{i,D})$, $x_{i,1}$ to $x_{i,60}$ denote the inertia weights from the input layer to the hidden layer and $x_{i,61}$ to $x_{i,70}$ denote the inertia weights from the hidden layer to the output layer.

3.2.1 Improvement of inertia weights: Inertia weight (w) in Particle Swarm Optimization (PSO) represents the ability of a particle to inherit its current velocity from its previous velocity. A smaller value of w is beneficial for local search, while a larger value is beneficial for global search. To balance the global and local search abilities of particles in PSO, the inertia weight is improved by modifying the inertia weight using the following equation (3)

$$w = \frac{\left[(w_{start} - w_{end}) \cdot \cos\left(\frac{k}{M}\right) + (w_{start} + w_{end})\right]}{2}, \quad (3)$$

In Equation (3), the initial and termination values of inertia weights are denoted by w_{start} and w_{end}, respectively, and $w_{start} < w_{end}$. The current number of particle iterations is k, and the maximum number of iterations is denoted by M. Due to the changing characteristics of the cosine function, the inertia weight (w) decreases slowly during the initial and final iterations. Therefore, the algorithm has a longer global optimization time at the beginning of the iterations, effectively reducing the risk of getting trapped in a local optimum. Additionally, in the later stages of the iterations, The algorithm can perform local search with a finer granularity, allowing for more incremental and precise adjustments.

3.2.2 Improvement of learning factors: In PSO algorithm, the learning factors c1 and c2 are the acceleration factors for the individual and global best values of a particle, respectively. Similarly, to enable the particles to have a good global search ability in the early stages of the iterations and to improve the precision and convergence speed of the particles in the later stages, the values of c1 and c2 are adjusted dynamically. This approach maintains the diversity of the population in the early stages of the search and improves the search performance in the later stages. The formula for calculating c1 and c2 is as follows.

$$c_1 = c_{start} - \frac{(c_{start} - c_{end})(M - k)}{M}$$
$$c_2 = 4 - c_1 \quad (4)$$

In Equation (4), the initial and termination values of the learning factor c1 are denoted by c_{start} and c_{end}, respectively, and $0<c_{start}<c_{end}<4$. M is the maximum number of iterations, and k is the current number of iterations.

3.2.3 Adaptation function: After initializing the neural network, the absolute value of the actual output result and the predicted result of the download force of the node at the next moment as the fitness function of the PSO algorithm:

$$F(X, x_i) = |y_i - y_i'|, i = 1,2, \dots, N \quad (5)$$

In Equation (5), X is the input to the neural network which are the six input metrics mentioned earlier. y_i and y_i' are the node the actual and predicted download speed. In the iterative process, the variation of velocity and displacement of the particles uses equation (1). The inertia weights and learning factors are calculated using in equations (3) and (4), respectively.

3.2.4 Data normalisation: To improve the accuracy of the training and reduce the errors caused by the difference in the magnitude of the input values, the input data is normalized to the range of (0,1). This is done to make the input neurons more sensitive and to reduce the errors caused by the difference in the scale of the factors. The equation for normalization is as follows:

$$X_{i,norm} = \frac{X_i - X_{i,min}}{X_{i,max} - X_{i,min}}, \quad (6)$$

In equation (6), the normalized result of input neuron i is denoted as $X_{i,norm}$. $X_{i,min}$ is the minimum value of the input neuron, and $X_{i,max}$ is the maximum value of the input neuron.

3.2.5 Data inverse normalization: To facilitate the comparison of predicted results with actual results and to make it easier to calculate the fitness value of each particle, it is necessary to reverse the normalization process of the particle's predicted results. The equation for reverse normalization is as follows:

$$Y_{i,norm} = Y_i \cdot \left(Y_{i,maxV} - Y_{i,minV}\right) + Y_{i,minV}, \quad (7)$$

In Equation (7), $Y_{i,norm}$ represents the result of the i-th dimension data after de-normalization, Y_i represents the result of the i-th dimension data before de-normalization, that is, the un-normalized result after prediction, $Y_{i,maxV}$ represents the maximum value of the i-th dimension data in the actual results, and $Y_{i,minV}$ represents the minimum value of the i-th dimension data in the actual results.

3.2.6 Accuracy dynamic adjustment function: To achieve the goal of low precision (low expected error) in the early stages of BP neural network training and high precision (high expected error) in the later stages, this paper introduces a precision dynamic adjustment function to dynamically adjust the error precision of the BP algorithm during the training process. The expression for the precision dynamic adjustment function is as follows:

$$accuracy = a - b * k \quad (8)$$

In Equation (8), a is the initial accuracy which usually is 1, b is the accuracy adjustment factor, and k is the current number of iterations.

The execution flow of the PSO-BP algorithm is shown in Figure 2.

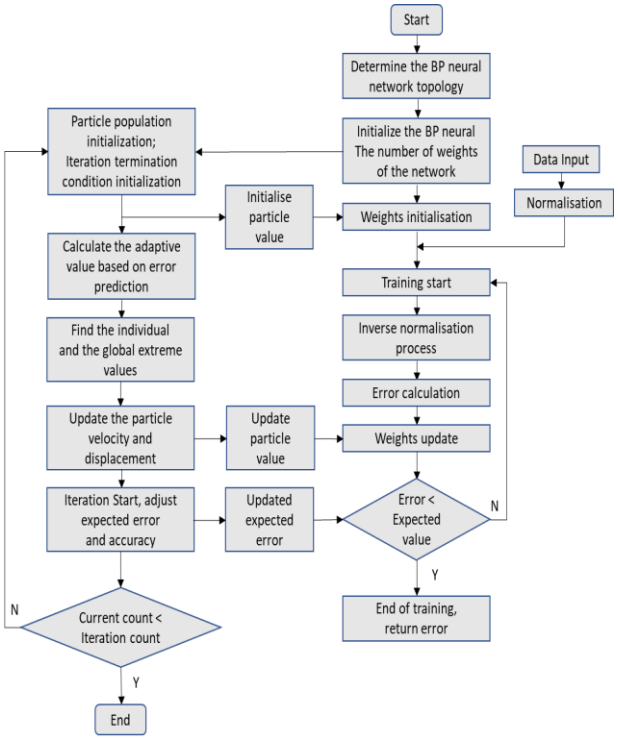

Figure 2. PSO-BP algorithm flow chart

4. EXPERIMENT AND RESULTS

4.1 Experimental parameters setting

To validate the execution results of the above proposed PSO-BP algorithm, BP, and PSO-BP algorithms were evaluated to conduct the experiments with real data sets. In Table 2, the parameters of the above two algorithms are set. The number of neurons in the input layer is 6, the number of neurons in the hidden layer is 10, and the number of neurons in the output layer is 1. The sigmoid function is used for the excitation function of the hidden layer. The inertia weights from the input layer to the hidden layer and from the hidden layer to the output layer are adjusted by combining gradient descent and momentum factor.

Algorithm	Parameters
BP	LearningRate = 0.08, Momentum factor =0.5, AvgTrainError >0.001
PSO-BP	N=50, c_{start}=3.5, c_{end}=2.5, w_{start}=1, w_{end}=0.1, b=0.2, Iter=200, v_{max}=2, $x_{i,D} \in$ [-0.5, 0.5], LearningRate =0.08, Momentum factor = 0.5, Iter=5

Table 2. Algorithm parameters setting

4.2 Experimental results analysis

To validate the execution results of the BP algorithm and the proposed PSO-BP algorithm, experiments have been done on real data sets. After cleaning and filtering the historical dataset, we collected more than 20,000 data files from a collection client of Sentinel 2 data collection. Around 20,000 data files were selected as the training set, and more than 2900 data files were selected as the test set. Table 3 lists the some of the experimental results of the tests using the two algorithms mentioned above.

No	Actual Value	BP		PSO-BP	
		Predicted Value	Error Rate	Predicted Value	Error Rate
1	976	677	0.3064	835	0.1445
2	840	854	0.0167	927	0.1036
3	832	886	0.0649	895	0.0757
4	816	848	0.0392	916	0.1225
5	848	859	0.0130	899	0.0601
6	840	867	0.0321	916	0.0905
7	920	833	0.0946	905	0.0163
8	944	842	0.1081	894	0.0530
9	920	842	0.0848	909	0.0120
10	840	799	0.0488	887	0.0560
11	824	834	0.0121	872	0.0583
12	912	843	0.0757	889	0.0252
13	976	777	0.2039	876	0.1025
14	864	833	0.0359	853	0.0127
15	808	813	0.0062	873	0.0804
16	808	798	0.0124	857	0.0606
17	912	867	0.0493	926	0.0154
18	880	888	0.0091	932	0.0591
19	832	848	0.0192	922	0.1082
20	856	858	0.0023	903	0.0549
21	920	842	0.0848	903	0.0185
22	872	876	0.0046	904	0.0367
23	824	850	0.0316	890	0.0801
24	832	853	0.0252	879	0.0565
25	832	819	0.0156	885	0.0637
26	808	845	0.0458	866	0.0718
27	896	840	0.0625	861	0.0391
28	808	833	0.0309	916	0.1337
29	816	802	0.0172	901	0.1042
30	928	838	0.0970	900	0.0302
31	816	791	0.0306	871	0.0674
32	984	824	0.1626	889	0.0965
33	840	919	0.0940	942	0.1214
34	920	909	0.0120	928	0.0087
35	856	921	0.0759	927	0.0829
36	888	907	0.0214	925	0.0417
37	920	906	0.0152	931	0.0120
38	936	898	0.0406	922	0.0150
39	832	905	0.0877	921	0.1070
40	888	905	0.0191	928	0.0450
41	816	906	0.1103	927	0.1360
42	808	909	0.1250	947	0.1720
43	936	913	0.0246	937	0.0011
44	824	899	0.0910	936	0.1359
45	920	900	0.0217	950	0.0326
46	808	882	0.0916	905	0.1200
47	968	876	0.0950	891	0.0795
48	832	754	0.0938	870	0.0457
49	824	771	0.0643	871	0.0570
50	848	765	0.0979	847	0.0012

Table 3. Selected prediction results

Figure 3 below shows the comparison of the error rates of the two algorithms for the data in Table 3. The prediction errors of the BP algorithm and the PSO-BP algorithm are very different, with the former having a large variance in the prediction results and the latter having a relatively flat prediction error. The errors of the two algorithms are mostly concentrated around 5%.

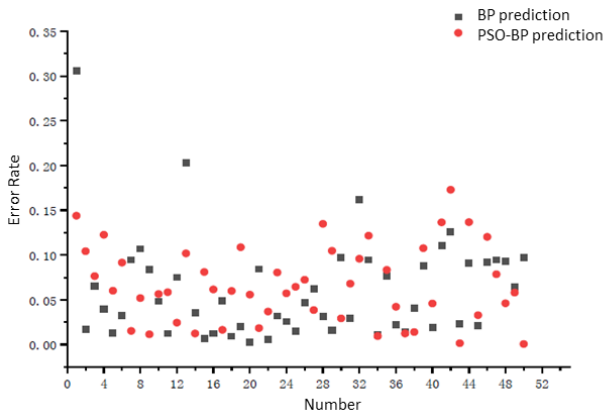

Figure 3. Error rates comparison between BP algorithm and PSO-BP algorithm

In Figure 4, the true value, and the prediction value from the two algorithms are shown. Prediction from PSO-BP algorithm is closer to the true value, while the error of the traditional BP algorithm is larger compared with PSO-BP algorithm.

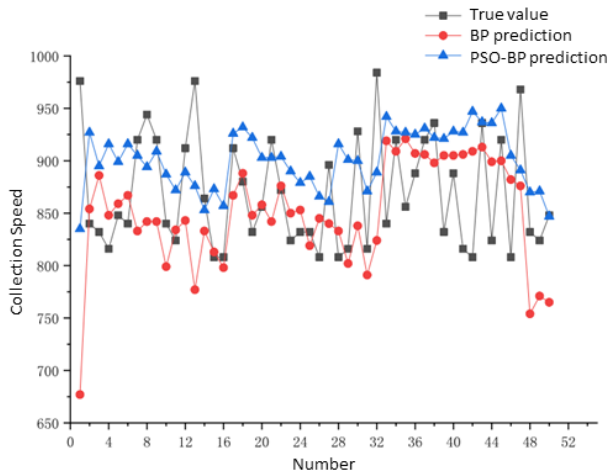

Figure 4. Results comparison between BP algorithm and PSO-BP algorithm

To further evaluate the prediction results, mean relative error and root mean square error are calculated in Table 4.

Algorithm	Mean relative error	Root mean square error
BP	0.0925	0.1163
PSO-BP	0.0595	0.0703

Table 4. Evaluation of algorithm prediction results

From Table 4, the mean relative error and root mean square error of PSO-BP algorithm are smaller than those of BP algorithm, which indicates that the PSO-BP algorithm has better prediction results. Tables 5 show the number of test sets in different error rate ranges for each of the two algorithms. The proposed PSO-BP algorithm achieves better accuracy comparing with traditional BP algorithm.

Algorithm	Error <= 5%	Error <= 15%	Error > 5%
BP	1004	2371	567
PSO-BP	1302	2908	30

Table 5. Error statistics on test set

5. CONCLUSION

This paper proposed an improved PSO-BP algorithm by introducing the accuracy dynamic adjustment function and the learning factor adjustment function. Finally, a comparison experiment between the improved PSO-BP algorithm and the BP algorithm is conducted on the real data set to verify the feasibility of the prediction model and the improved PSO-BP algorithm.

ACKNOWLEDGEMENTS

This work was supported in part by TUOHAI special project 2020 from Bohai Rim Energy Research Institute of Northeast Petroleum University under Grant HBHZX202002 and project of Excellent and Middle-aged Scientific Research Innovation Team of Northeast Petroleum University under Grant KYCXTD201903.

REFERENCES

Cheng, G., Han, J., 2016. A survey on object detection in optical remote sensing images. ISPRS J. Photogramm. Remote Sens. 117, 11–28. https://doi.org/10.1016/j.isprsjprs.2016.03.014

Chi, M., Plaza, A., Benediktsson, J.A., Sun, Z., Shen, J., Zhu, Y., 2016. Big Data for Remote Sensing: Challenges and Opportunities. Proc. IEEE 104, 2207–2219. https://doi.org/10.1109/JPROC.2016.2598228

Deng, Z., Sun, H., Zhou, S., Zhao, J., Lei, L., Zou, H., 2018. Multi-scale object detection in remote sensing imagery with convolutional neural networks. ISPRS J. Photogramm. Remote Sens. 145, 3–22. https://doi.org/10.1016/j.isprsjprs.2018.04.003

Deren, L., Liangpei, Z., Guisong, X., 2014. Automatic analysis and mining of remote sensing big data. Acta Geod. Cartogr. Sin. 43, 1211.

Hecheltjen, A., Thonfeld, F., Menz, G., 2014. Recent advances in remote sensing change detection–a review. Land Use Land Cover Mapp. Eur. Pract. Trends 145–178.

Hemati, M., Hasanlou, M., Mahdianpari, M., Mohammadimanesh, F., 2021. A Systematic Review of Landsat Data for Change Detection Applications: 50 Years of Monitoring the Earth. Remote Sens. 13, 2869. https://doi.org/10.3390/rs13152869

Jensen, J.R., Im, J., 2007. Remote sensing change detection in urban environments. Geo-Spat. Technol. Urban Environ. Policy Pract. Pixels 7–31.

Khelifi, L., Mignotte, M., 2020. Deep Learning for Change Detection in Remote Sensing Images: Comprehensive Review and Meta-Analysis. IEEE Access 8, 126385–126400. https://doi.org/10.1109/ACCESS.2020.3008036

Lee, C.A., Gasster, S.D., Plaza, A., Chang, C.-I., Huang, B., 2011. Recent Developments in High Performance Computing

for Remote Sensing: A Review. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 4, 508–527. https://doi.org/10.1109/JSTARS.2011.2162643

Li, K., Wan, G., Cheng, G., Meng, L., Han, J., 2020. Object detection in optical remote sensing images: A survey and a new benchmark. ISPRS J. Photogramm. Remote Sens. 159, 296–307. https://doi.org/10.1016/j.isprsjprs.2019.11.023

Liu, P., Di, L., Du, Q., Wang, L., 2018. Remote sensing big data: Theory, methods and applications. Remote Sens. 10, 711.

Ma, X., Wang, Z., Bai, L., Xu, B., Gao, J., Wen, B., Tao, J., 2021. Implementation of a Federated Large-Scale Remote Sensing Data Sharing Platform, in: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. Presented at the IGARSS 2021 - 2021 IEEE International Geoscience and Remote Sensing Symposium, IEEE, Brussels, Belgium, pp. 5771–5774. https://doi.org/10.1109/IGARSS47720.2021.9554247

Ma, Y., Wu, H., Wang, L., Huang, B., Ranjan, R., Zomaya, A., Jie, W., 2015. Remote sensing big data computing: Challenges and opportunities. Future Gener. Comput. Syst. 51, 47–60. https://doi.org/10.1016/j.future.2014.10.029

Phiri, D., Simwanda, M., Salekin, S., Nyirenda, V.R., Murayama, Y., Ranagalage, M., 2020. Sentinel-2 data for land cover/use mapping: A review. Remote Sens. 12, 2291.

Qian, X., Lin, S., Cheng, G., Yao, X., Ren, H., Wang, W., 2020. Object detection in remote sensing images based on improved bounding box regression and multi-level features fusion. Remote Sens. 12, 143.

Segarra, J., Buchaillot, M.L., Araus, J.L., Kefauver, S.C., 2020. Remote sensing for precision agriculture: Sentinel-2 improved features and applications. Agronomy 10, 641.

Shi, K., Bai, L., Wang, Z., Tong, X., Mulvenna, M.D., Bond, R.R., 2022. Photovoltaic Installations Change Detection from Remote Sensing Images Using Deep Learning, in: IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium. Presented at the IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium, IEEE, Kuala Lumpur, Malaysia, pp. 3231–3234. https://doi.org/10.1109/IGARSS46834.2022.9883738

Song, G., Wang, Z., Bai, L., Zhang, J., Chen, L., 2020. Detection of oil wells based on faster R-CNN in optical satellite remote sensing images, in: Notarnicola, C., Bovenga, F., Bruzzone, L., Bovolo, F., Benediktsson, J.A., Santi, E., Pierdicca, N. (Eds.), Image and Signal Processing for Remote Sensing XXVI. Presented at the Image and Signal Processing for Remote Sensing XXVI, SPIE, Online Only, United Kingdom, p. 17. https://doi.org/10.1117/12.2572996

Wang, D., Tan, D., Liu, L., 2018. Particle swarm optimization algorithm: an overview. Soft Comput. 22, 387–408. https://doi.org/10.1007/s00500-016-2474-6

Wang, Q., Zhang, X., Chen, G., Dai, F., Gong, Y., Zhu, K., 2018. Change detection based on Faster R-CNN for high-resolution remote sensing images. Remote Sens. Lett. 9, 923–932. https://doi.org/10.1080/2150704X.2018.1492172

Wang, Z., Bai, L., Liu, X., Chen, Y., Zhao, M., Tao, J., 2022. Dynamic Task Scheduling in Remote Sensing Data Acquisition from Open-Access Data Using CloudSim. Appl. Sci. 12, 11508. https://doi.org/10.3390/app122211508

Wang, Z., Bai, L., Song, G., Zhang, J., Tao, J., Mulvenna, M.D., Bond, R.R., Chen, L., 2021. An Oil Well Dataset Derived from Satellite-Based Remote Sensing. Remote Sens. 13, 1132. https://doi.org/10.3390/rs13061132

Yu, Z., Wang, Z., Bai, L., Chen, L., Tao, J., 2021a. Remote Sensing Inversion of PM10 Based on Spark Platform, in: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. Presented at the IGARSS 2021 - 2021 IEEE International Geoscience and Remote Sensing Symposium, IEEE, Brussels, Belgium, pp. 1685–1688. https://doi.org/10.1109/IGARSS47720.2021.9554323

Yu, Z., Wang, Z., Bai, L., Chen, L., Tao, J., 2021b. Parameter Optimization on Spark for Particulate Matter Estimation, in: 2021 Workshop on Algorithm and Big Data. Presented at the WABD 2021: 2021 Workshop on Algorithm and Big Data, ACM, Fuzhou China, pp. 9–13. https://doi.org/10.1145/3456389.3456406

Zhang, J., Wang, Z., Bai, L., Song, G., Tao, J., Chen, L., 2021. Deforestation Detection Based on U-Net and LSTM in Optical Satellite Remote Sensing Images, in: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. Presented at the IGARSS 2021 - 2021 IEEE International Geoscience and Remote Sensing Symposium, IEEE, Brussels, Belgium, pp. 3753–3756. https://doi.org/10.1109/IGARSS47720.2021.9554689

Zhang, S., He, G., Chen, H.-B., Jing, N., Wang, Q., 2019. Scale adaptive proposal network for object detection in remote sensing images. IEEE Geosci. Remote Sens. Lett. 16, 864–868.

Zhu, M., Wang, Z., Bai, L., Zhang, J., Tao, J., Chen, L., 2021. Detection of industrial storage tanks at the city-level from optical satellite remote sensing images, in: Bruzzone, L., Bovolo, F., Benediktsson, J.A. (Eds.), Image and Signal Processing for Remote Sensing XXVII. Presented at the Image and Signal Processing for Remote Sensing XXVII, SPIE, Online Only, Spain, p. 33. https://doi.org/10.1117/12.2600008