Eng Int Syst (2025) 2: 179–188

Engineering
Intelligent Systems

Real-time Integration Technology of Large-scale Heterogeneous Data Based on Big Data and Artificial Intelligence

Jiawei Zhang*

Beijing Normal University - Hong Kong Baptist University United International College, Zhuhai 519000, Guangdong, China

In today's era of increasingly sophisticated network communication technology, large-scale heterogeneous data are emerging, and people's requirements for data integration technology are increasing. To address this challenge, many scholars have begun to combine big data with artificial intelligence to integrate large-scale heterogeneous data in real time. Since the neural network can fully take into account the characteristics of the data, it has strong upgrade ability and fault resistance, and does not need to know the real-time data integration for specific instance training in advance. Hence, it is able to deal with the real-time integration problem of large-scale heterogeneous data. In order to increase the speed and effectiveness of data integration, this study presents a neural network based on big data and artificial intelligence to assess large-scale heterogeneous data real-time integration technology. According to this study's experimental findings, the particle swarm-BP neural network has an error of about 0.020 and the BP neural network has an error of about 0.021 in terms of training results for the three techniques. The enhanced particle swarm-BP neural network technique has an inaccuracy of 0.15 to 0.2; hence, with the enhanced particle swarm-BP neural network technique, there is significantly less error. Also, the improved algorithm requires much less training time than the other two algorithms in the range of 114ms-121ms, indicating that the algorithm has better integration efficiency. Therefore, this method is very effective for the real-time integration of large-scale heterogeneous data.

Keywords: heterogeneous data, neural networks, artificial intelligence, big data

1. INTRODUCTION

In real life, large-scale heterogeneous data can pose many challenges to system users. It not only reduces the efficiency of data transportation; it also prevents users from carrying out tasks easily and smoothly. With the development of science and technology, and the emergence of big data and artificial intelligence, such problems have been solved. Moreover, big data and artificial intelligence technology also allows heterogeneous data to be efficiently classified and the required information to be extracted. Hence, big data and artificial intelligence technology can ensure the real-time integration of heterogeneous data.

The emergence of the Internet has made people's lives more and more convenient. The Internet is being widely used not only for medical care and online shopping, for instance; it also plays a huge role in the transmission of data between companies, organizations and departments. However, although the Internet makes it easier for people to obtain information, it also generates an increasing amount of heterogenous data, subsequently placing more pressure on people working with such data. Big data technology can effectively mine data and extract useful information, and the emergence of artificial intelligence has also brought

*Address for correspondence: Jiawei Zhang, Beijing Normal University - Hong Kong Baptist University United International College, Zhuhai 519000, Guangdong, China, Email: zhangjw21265@163.com

more convenience to people's lives. If the large-scale heterogeneous data facts can be integrated, tasks can be completed more efficiently. Therefore, it is imperative to examine the real-time integration technology applied to big data that is heterogeneous. The novel aspect of this paper is the use of neural networks and particle swarm optimization within the context of big data and artificial intelligence.

2. RELATED WORK

The research on heterogeneous database technology is important not only from a theoretical perspective, but also in terms of its application in computer networks. AI can usually find anomalies in data that is complex, massive, and dynamic. However, protecting the security and privacy of cloud data has become a most important and difficult-to-resolve issue. Additionally, it is critical to safeguard the security and privacy of sensitive data from unauthorized individuals attempting to access information [1]. A study found that by querying the global schema represented by the ontology, users of ontology-based data integration systems are able to efficiently retrieve data located in various sources. His aim was to establish a logical framework for data integration using ontologies that include information disclosure [2]. A study considered the possibility that, depending on the timing of a disaster, it may be difficult to obtain and understand all the relevant facts. As a result, it is important to implement an early reaction strategy and manage the response to a catastrophe. He developed a method for gathering both timely and reliable data on damage done to infrastructure [3]. The need for developing distinctive and dynamic information technology infrastructures has been emphasized by Schimpf. He focused on reliable and fast image transmission and high-speed cellular access, and proposed a technique for the real-time integration of heterogeneous data to avoid signal loss during data transmission [4]. A study studied the integration of heterogeneous data using mixed integer nonlinear programming and mixed integer linear programming methods. He introduced three innovative real-time data integration optimization methods [5]. A study have proposed methods for the real-time integration of large-scale heterogeneous data, which facilitate the timely transmission of data and improve the efficiency of people's work with data. They believe that innovative real-time integration technology should be adopted to process heterogeneous data.

In terms of technological development, the framework for data integration systems is maturing, but with the generation of massive amounts of diverse data, the heterogeneity problem associated with data sources and the integration of data has become very prominent. A study created a new framework for the integration of two sets of high-dimensional and heterogeneous data using neural networks. The researcher modeled heterogeneous random variables using exponential distributions and identified shared and unique patterns in two datasets by using structured decomposition of the underlying natural parameter matrices [6]. He investigated a class of uncertain neural networks with time-varying delays and the robust state estimation problem, and demonstrated that the building of resilient state estimators for such neural networks may be accomplished by resolving linear matrix inequalities using a new bounds technique [7]. A study proposed a simple and effective neural network supervision method. This method is used for large-scale image search and learns binary code features by minimizing an objective function defined on classification errors and other ideal datasets [8]. Deep neural networks (DNNs), according to Cheng , have gained much attention recently and have significantly improved accuracy in a variety of applications. The availability of a graphics processing unit with high computing power plays a key role in its success [9]. Scholars agree that neural networks can achieve real-time integration of heterogeneous data because they are a type of artificial intelligence technology that can effectively process large-scale data. However, these researchers have not specified how to apply a neural network to heterogeneous data real-time integration technology [10].

3. HETEROGENEOUS DATA INTEGRATION METHOD BASED ON IMPROVED PSO-BP NEURAL NETWORK

The aim of data integration is to provide users with a common interface that provides access to and can handle a variety of heterogeneous data sources. At the same time, it offers a platform that can combine the results of multiple queries can minimize or even completely avoid the heterogeneity problem caused by differences in environments, operating conditions, and human factors [11]. This is in order to eliminate the conflict between heterogeneous data, achieve transparent access between different data sources, and achieve the purpose of data integration: sharing, and effective and consistent information query.

In practical applications, each node in the system does not use the same type of database, or the database versions used are not uniform due to historical reasons, which are collectively referred to as the database inhomogeneity problem [12]. Maintaining data integrity in heterogeneous databases is not simply a matter of data replication; it is essential to save heterogeneous data correctly and maintain its accuracy and consistency. The overall structure of the heterogeneous data integration system is shown in Figure 1.

As shown in Figure 1, heterogeneous data sources and wrappers corresponding to the data sources are the components of the heterogeneous data layer. The data layer not only provides the actual data storage and management functions; it also accepts the instructions of the upper layer and completes the tasks [13]. The main function of the wrapper is to provide a unified schema for the form of output data from various heterogeneous data sources. At the same time, it provides an interface for the communication between the intermediate integration layer and the data source. The interface can conceal the heterogeneity between the various data sources and complete the seamless connection between them.

The emergence of heterogeneous database technology not only reduces the cost of real-time integration technology, but also makes computing more convenient and greatly improves

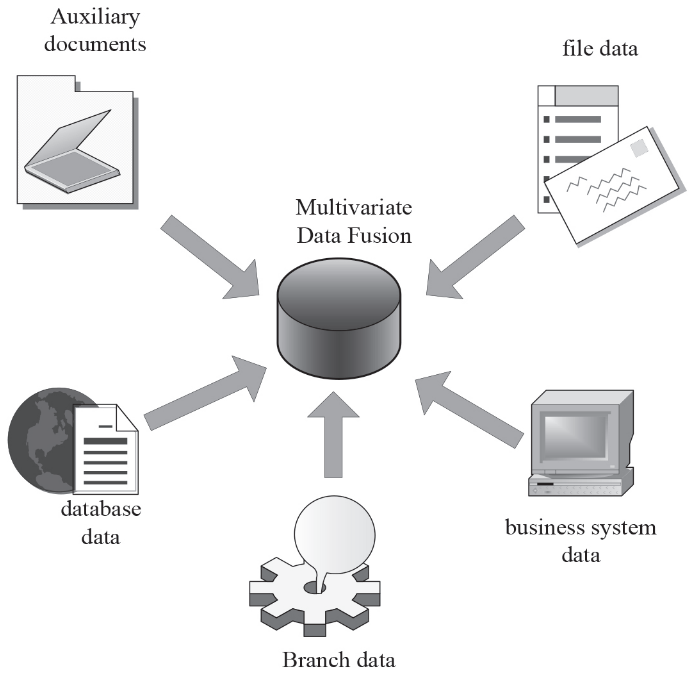


Figure 1 Overall structural framework of heterogeneous data integration system.

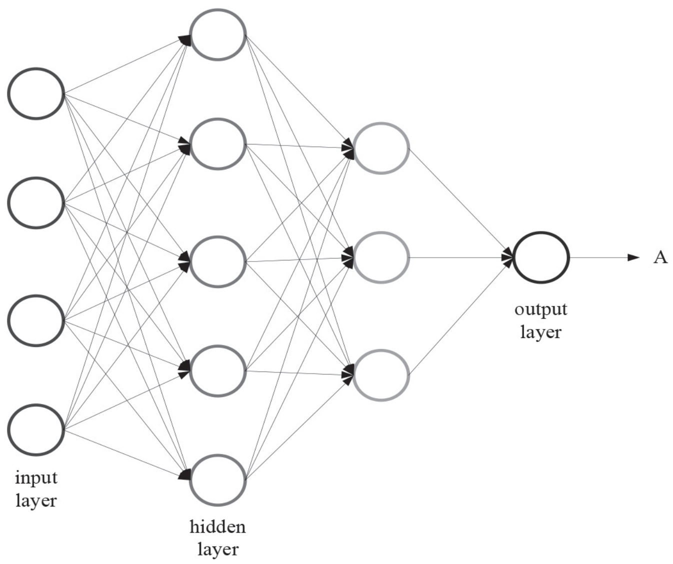


Figure 2 The structure of the BP neural network.

work efficiency. Not having to change the existing database management system and data structure is an important prerequisite for the management of real-time integration of different databases, and heterogeneous database technology makes the Internet more promising [14–15].

In fact, data integration refers to integrating the same information from different sources, different forms or different attributes, and providing the same interface for the purpose of data sharing. Data integration is achieved by exchanging data between different applications, and the purpose is to solve the problem of handling distributed heterogeneous data. is the time class is stored in a binary form in the database, the display is in the form of a string, and there are various display types. Due to the different display modes of time types between different databases, the time format of the source database should be obtained before synchronization, and then the time format conversion function of the target database should be used to convert to obtain the legal time data of the target database.

3.1 BP Neural Network Model

The development of deep learning has made BP neural networks increasingly popular among academics. The BP neural network was used long before it became the center of the deep learning research as we know it today[16–18]. The BP neural network was created from the early feedforward neural network, which uses backpropagation to learn the data and connect it to the information in the hidden layer just before the hidden layer learning process. Figure 2 depicts the precise structure of the BP neural network.

As shown in Figure 2, the data integrated in the process of database integration is heterogeneous, and it is difficult to match attributes. The data in the database is not static, so it is difficult to integrate heterogeneous data in real time. Even if large-scale heterogeneous data is integrated, it takes a long time, and the emergence of neural network just solves this problem. Neural networks have many advantages, such as self-learning ability, inductive reasoning ability and random information processing ability, which can solve the disorder problem of large-scale heterogeneous data. Compared with the traditional integration technology, the real-time integration technology that integrates the neural network is the most suitable [19].

There are two ways to solve the non-negative matrix factorization, one is to use Euclidean distance, and the other is to use cross entropy. The update method adopts multiplicative iteration, and the parameters are learned alternately through the artificially designed learning rate [20]. Therefore, the objective function using Euclidean distance is as Formula 1:

$$\min \|V - WH\|_F^2 = \sum_{i,j} \left(V_{i,j} - (WH)_{i,j}\right)^2 \tag{1}$$

Using cross-entropy as the objective function, its form is as Formula 2:

$$H_{a,j} \leftarrow H_{a,j} \frac{\left(W^T V\right)_{a,j}}{\left(W^T WH\right)_{a,j}} \tag{2}$$

Obviously, when $V = WH$, the formula can take the minimum value of 0. Also, it can be seen that both W and H are convex, but not WH; so it is not easy to obtain the global optimal solution. The solution method using Euclidean distance is as Formula 3:

$$W_{i,a} \leftarrow W_{i,a} \frac{\left(VH^T\right)_{i,a}}{\left(WHH^T\right)_{i,a}} \tag{3}$$

The convergence of the function is proved by introducing an auxiliary function. Non-negative matrix factorization needs to specify the number of latent factors, and the usual practice is to set the required number of categories as the number of latent factors [21]. At present, there are two commonly used methods for the non-negative matrix decomposition of clustering: 1) use the basis matrix directly for clustering, and 2) select the column subscript of the basis matrix where the largest value is located as the category.

The definition of the relevant variables of the neural network is as follows: m and n respectively represent the weight connection between the neurons in the previous layer and the neurons in the next layer, d represents the deviation of the hidden layer, and the deviation of the output layer is expressed as Formula 4:

$$ParamNum = d * m + m * n + m + n \tag{4}$$

By continuously inputting the input data into the network, and cooperating with the hidden layer information of the previous moment, the current hidden layer data expression can be continuously obtained, which is exactly what the real task needs.

The ideal output value at the ith network output node is b_{ji}, so the minimum mean square error is Formula 5:

$$MSE = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{C} \left(b_{ji}^d - b_{ji}\right)^2 \tag{5}$$

The purpose of the BP neural network is to fit the input data in order to reduce the difference between the output and the input of the model modeling [22]. Among them, C is the dimension of the output vector and M is the overall number of samples. The hidden layer representation that results from this approach also aids in data compression, unlike what is sometimes believed, since there will be fewer hidden units in the codec layer than the dimension of the original data [23].

If the data input is too random, it is often difficult to find a mapping relationship so that the data can be reduced in dimension. But it is precisely because there is a certain distribution in the data that makes this learning meaningful, so that the data can be compressed. In the actual training process, if the cost function is smaller, it means that the output and input are more similar, and the fitting ability of the model is better. For the trained model, only the encoding part is actually needed.

The network error measure for the entire dataset is defined by Formula 6:

$$E = \sum_P E_p \tag{6}$$

A regularization term is integrated into the function to suppress the output of the hidden layer nodes, so that the average output of the hidden layer is 0. At the same time, because most of the node values are 0, the non-zero value part can be used as the output. Moreover, since biological neuron connections are unsaturated, and neurons are rarely connected to each other, it is reasonable to think that sparse representation of data will be more effective [24].

When adjusting the weights of neuron connections, only the error of the output layer can be obtained, and the errors of other layers can only be obtained through the backoff of each layer of error feedback. Although the BP network greatly simplifies network training, because of its simplification, one would apply the widely used learning method and subtly introduce the concept of "error". However, the BP algorithm also has several shortcomings. For instance, it can converge the network weights to the final solution, but cannot guarantee the global optimal solution or even the minimum value of the error hyperplane.

3.2 Improvement of BP

The enhanced BP neural network significantly improves computation accuracy and speed when compared to the gradient-based BP learning approach [25]. However, the BP neural network also has drawbacks that cannot be dealt with. The disadvantages of premature convergence are challenging for the BP neural network to overcome, despite the fact that it can handle many complex situations. It is quite difficult to modify the optimization procedure for BP neural network while employing neural networks. Therefore, a global optimization approach must be obtained in order to solve this issue; this method is highly beneficial for optimizing neural networks. The enhanced BP neural network is displayed in Figure 3.

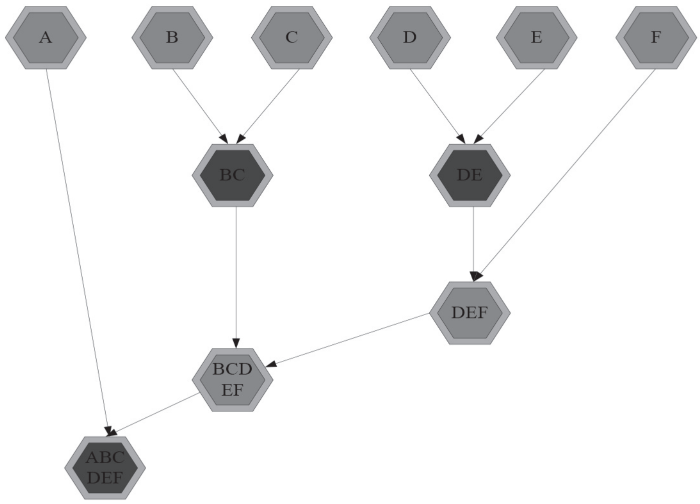


Figure 3 Improved BP neural network.

As depicted in Figure 3, the evolutionary algorithm includes particle swarm optimization (PSO).This method guides the optimization to search for the optimal value through cooperation and competition among particles in the swarm, as shown in Formula 7:

$$A_{id}\left(t+1\right)=A_{id}\left(t\right)+v_{id}\left(t\right) \tag{7}$$

By introducing PSO into the training process, one can obtain a model with stronger generalization ability, so that the output is not affected by the interference of a small amount of data, and the obtained output is also more robust.

3.3 Improved PSO-BP Algorithm

The original input is encoded into the specified feature space. The lower part is the Decoder model, which transforms the converted features into the dimension of the original data. This can be expressed by Formula 8:

$$\begin{aligned}A_{id}\left(t+1\right)=A_{id}\left(t\right)+c_1*r_1*\left(p_{id}-a_{id}\right)\\+c_2*r_2*\left(p_{gd}-a_{id}\left(t\right)\right)\end{aligned} \tag{8}$$

In addition, the PSO to be introduced learns the parameters of the distribution of the data through the neural network, which are generally the mean and variance of the Gaussian distribution. Then, the noise is resampled to obtain the hidden expression using the Gaussian formula, and then decoded back to the input to use the KL divergence to learn the parameters of the model. This model is formalized with Formula 9.

$$A_{jd}^{(t)}=p_{id}=p_{gd} \tag{9}$$

In addition, by replacing the particle hidden layer and the BP hidden layer with other networks such as BP neural network, the recurrent self-encoding network can be obtained.

$$p_r=\left(p_{r1},p_{r2},...,p_{rD}\right) \tag{10}$$

In this study, the PSO algorithm is used to learn the dimensionality reduction expression of the BP hidden layer. The sequence learning method is used whereby the data of the current moment is used to collect the data of the next moment. Then, the data items are input into the network one by one according to time, and the information in the hidden layer at the last moment represents the information of the whole sequence. Here, Formula 11 is used:

$$iff\left(a_i\right)\prec f\left(p_i\right),\,p_i=a_i;endif \tag{11}$$

In a real case, the previous hidden units are used as data for all data. Because the hidden layer is always unable to store information to determine whether the best particle exists overall, it is difficult for the last layer to represent the total number of compressed representations of global information. It is calculated with Formula 12:

$$\left|b_i^d-b_i\right|\leq\theta \tag{12}$$

where is the threshold. When $i=1,2,...,C$, the classification is considered correct; otherwise, the classification is wrong. .This method is used here to reduce the memory loss caused by the long sequence. All hidden unit information is stacked and then pooled, thereby reducing the dimensionality of a large amount of hidden unit information. In this way, data can be collected using the transformed features as shown in Figure 4.

As shown in Figure 4, there are usually two ways to improve the PSO algorithm in order to avoid the premature phenomenon: one is to always keep a specific type of particles to prevent particles from falling into the same local solution. The other is to determine when a particle is classified as a local solution. In order to increase diversity in the case of low diversity, measures are taken to scatter particles. Therefore, this paper proposes an improved particle swarm optimization algorithm. The mean squared error known as generalized mean squared error is obtained with Formula 13:

$$MSE_G={}^1/_P\sum_{i=1}^{P}\sum_{j=1}^{C}\left(b_{ji}^d-b_{ji}\right)^2 \tag{13}$$

In this way, the required hidden information layer can be obtained according to the BP neural network, and then the

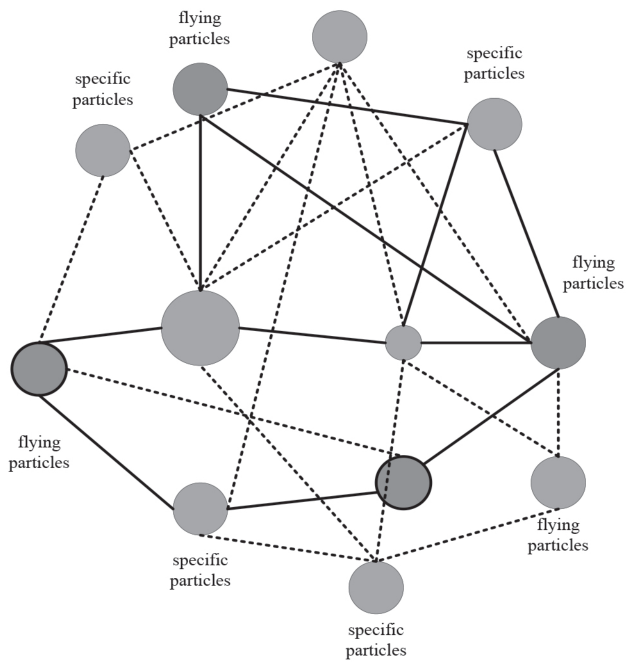


Figure 4 Improved PSO algorithm.

required transformation features can be obtained by average pooling. The transformed feature can be obtained with Formula 14:

$$O_p = F_n\left(\ldots\left(F_n\left(F_1\left(a_p w^t\right) w^2\right)\ldots\right) w^n\right) \tag{14}$$

The difference between the actual output O_p and the corresponding ideal output $a_p$is calculated.

Firstly, the algorithm inputs the parameters required by the model, including the number of hidden units, the batch size and the number of model iterations. The model parameters can then be initialized and updated iteratively:

$$\Delta w_{pq} = \alpha o_q\left(1 - o_q\right)\left(b_q - o_q\right) o_q \tag{15}$$

$$\Delta v_{hp} = \alpha o_{pk-1}\left(w_{p1}\delta_{1k} + \ldots + w_{pm}\delta_{mk}\right) \tag{16}$$

Then the trained model is used in combination with the pooling method to transform the original data:

$$E_p = \frac{1}{2}\sum_{j=1}^{m}\left(b_{pj} - o_{pj}\right)^2 \tag{17}$$

In order to facilitate the subsequent calculation, several intermediate variables are first defined to facilitate the later representation. The number of samples is ignored here, because it is usually updated in batches; that is, parameters are updated once with a batch of data, which is similar to updating parameters once with a single sample, as shown in Formula 18:

$$d_{j^*} = \sum_{j\in\{1,2,\ldots,m\}} \min\left\{d_j\right\} \tag{18}$$

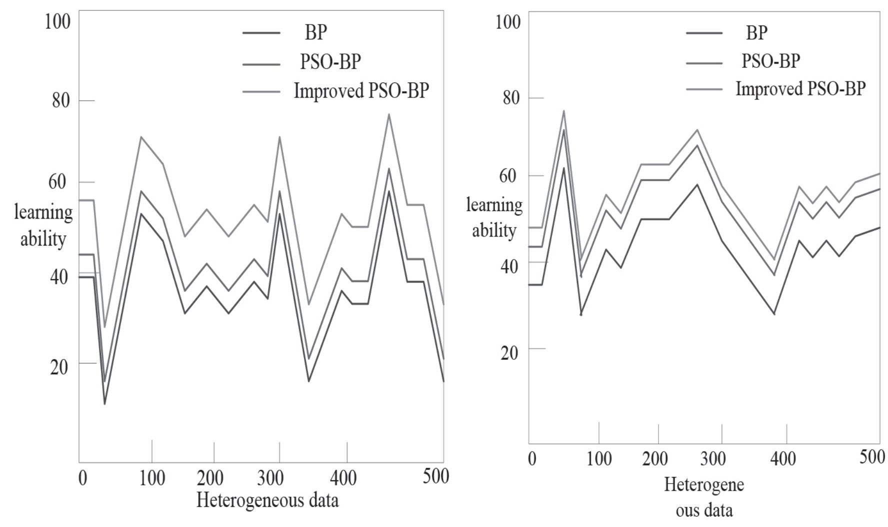
Heterogeneous data has many unique properties, which lead to more or less problems in the direct use of many existing classical BP algorithms, so it is necessary to select a suitable model for specific time series data. Although heterogeneous data has various unique properties, the general research focus is still on its heterogeneous sequence. Therefore, in order to capture this property, this study uses the BP neural network, an algorithm that trains the data according to the time occurrence. However, the traditional BP neural network is very limited in practical application due to the gradient problem, so the PSO algorithm is used to solve this problem effectively.

4. EXPERIMENT OF HETEROGENEOUS DATA REAL-TIME INTEGRATION TECHNOLOGY

4.1 Comparative Experiment of BP Algorithm, PSO-BP Algorithm and Improved PSO-BP Algorithm

The experimental data used for this study comprise 1000 sets of heterogeneous data; the experimental software is MATLAB. Before formally using the algorithm in this paper for comparative experiments, we examine the improved PSO-BP algorithm, a powerful self-learning method, and compare the self-learning capabilities of the BP algorithm, the PSO-BP algorithm and the improved PSO-BP algorithm. The experiment is carried out as depicted in Figure 5:

As shown in Figure 5, the trend of the data in the original data is very messy. Although the trends of each category are different on the whole, there is no clear change between the data; that is, there is no clear main line. However, after the feature transformation of the neural network, the trend of the data is very clear. Although it is very similar in shape, it is



a. The learning ability of the three methods under the training set data

b. The learning ability of the three methods under the test set data

Figure 5 The learning ability of the three methods under the training set data and the test set data.

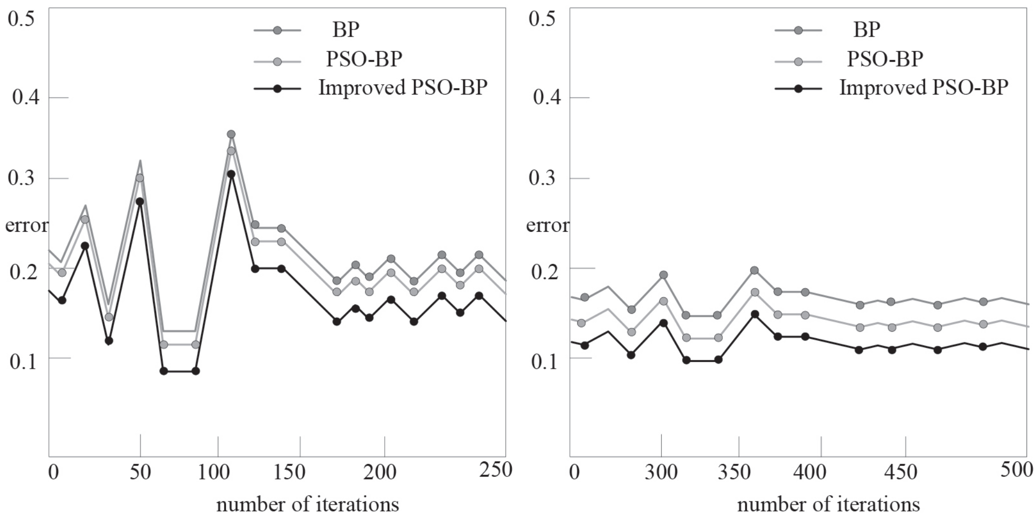


Figure 6 The error relationship between the training effect of the three algorithms and the number of iterations.

different in terms of detail. An advantage is that data can be visualized, and some rules can be easily found, which is have proven to be very valuable in previous research. Moreover, since the cluttered data becomes more regular, the algorithm used in this paper has a data preprocessing function because this will also speed up the training of the model.

The main steps of database integration are semantic integration and data integration, which involve combining multiple data models from heterogeneous databases into a single data model representation. Semantic integration is the basis of data integration. In other words, semantic heterogeneity or semantic integration is the main problem to be solved by database integration. Addressing the data integration challenges of data heterogeneity is relatively straightforward. The connection between the number of iterations and the training impact of the BP algorithm, the PSO-BP algorithm, and the improved PSO-BP algorithm are shown in Figure 6.

As shown in Figure 6, during the first half of the training, the prediction ability of the model changes greatly, indicating that the random value interferes very significantly with the neural network model. In the later stage of training, the training error of the modular neural network type begins to stabilize, and each evaluation index also tends to stabilize gradually. For different parameters, the convergence rate of the model is different, but it can be found that after 150 iterations, the model has become stable under each parameter. It can also be observed that the model parameters have little effect on the final clustering effect.

In order to better integrate the PSO algorithm into the BP neural network to achieve the best optimization effect, a particle model is established in this paper to achieve the purpose of calculating the fitness function and defining the search space. In this process, the two key points that cannot be ignored by PSO algorithm are speed and position, while the two key points that cannot be ignored by BP

Table 1 BP training results.

training times	MSE_T	MSE_G	training period
1	0.0215	0.0400	7142
2	0.0209	0.0432	7087
3	0.0213	0.0417	6983
4	0.0219	0.0458	6990
5	0.0220	0.0425	7005

Table 2 PSO-BP training results.

training times	MSE_T	MSE_G	training period
1	0.0205	0.0328	154
2	0.0211	0.0365	150
3	0.0206	0.0383	152
4	0.0202	0.0355	157
5	0.0201	0.0359	159

Table 3 Improved PSO-BP training results.

training times	MSE_T	MSE_G	training period
1	0.0153	0.0217	121
2	0.0142	0.0215	114
3	0.0148	0.0228	117
4	0.0139	0.0219	118
5	0.0125	0.0205	115

neural network learning algorithm are weight and threshold. After clearing these, the BP algorithm and PSO-BP and the improved PSO-BP are trained, as shown in Tables 1 to 3.

As shown in Tables 1 to 3, the PSO-BP and upgraded PSO-BP network require less training time than the traditional BP network.. When it reaches the precocious state, the particle will be recreated in the search space of the new PSO-BP algorithm, its position will be updated with its average position, and the convergence speed accelerates. PSO-BP has a substantially greater capacity for global optimization than standard PSO, while simultaneously having approximately the same training times and reduced mean square errors as a result of training. This means that when employed as a neural network for the learning algorithm, the extended PSO-BP algorithm outperforms both the PSO algorithm and the conventional BP approach.

4.2 Integration Efficiency and Real-time Performance of Data from Different Methods

After classifying the attributes according to various standards, comparing the training time and accuracy of the BP neural network and the improved PSO-BP attribute matching algorithm, the training time of the improved PSO-BP algorithm proposed in this paper is determined. Results are shown in Figure 7.

Figure 7 indicates that when training on the same data, the duration and training time of the improved PSO-BP algorithm are shorter than those of the conventional algorithm, and the average accuracy is significantly higher. Therefore, the experimental results fully support the enhanced PSO-BP technique for attribute matching proposed in this paper.

In order to provide users with a unified query interface to conceal the heterogeneity of the underlying data sources, technology enabling the real-time integration of heterogeneous data has emerged. This technology allows users to concentrate on solving their own problems and complete tasks efficiently without having to worry about the heterogeneity of the data model of each basic data source. In practice, the frequency of database updates and the complexity of the fields determine the time interval for planning activities. Test results show that if the database is rarely updated, the schedule time can be set to 1–2 minutes, as shown in Figure 8.

As shown in Figure 8, as the file size increases, the time required for real-time integration increases exponentially. This is because the synchronization of data needs to go through two processes: writing to an external file and writing to the database, which is much more time-complex than ordinary data types. However, it can be seen that although the integration time of the improved PSO-BP algorithm is also increasing, the speed is relatively slow compared to that of other algorithms. Furthermore, the synchronization efficiency is also affected by factors such as dynamic memory size. If the amount of data is too large, the parsing of XML

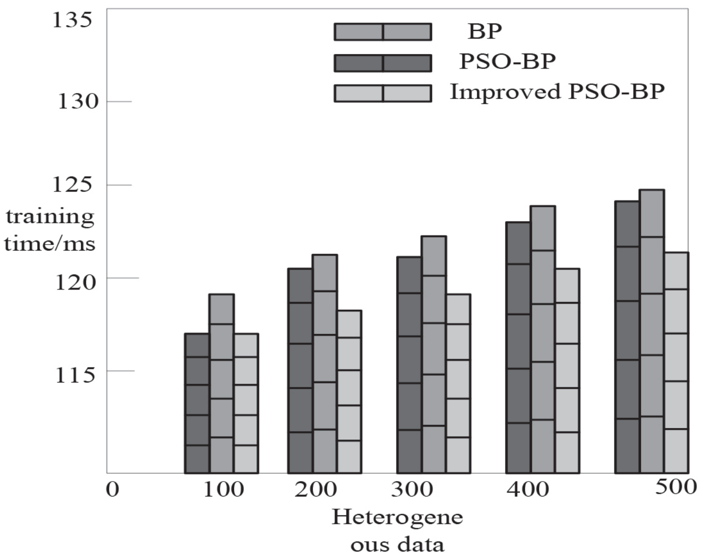


Figure 7 Training time of the improved PSO-BP algorithm.

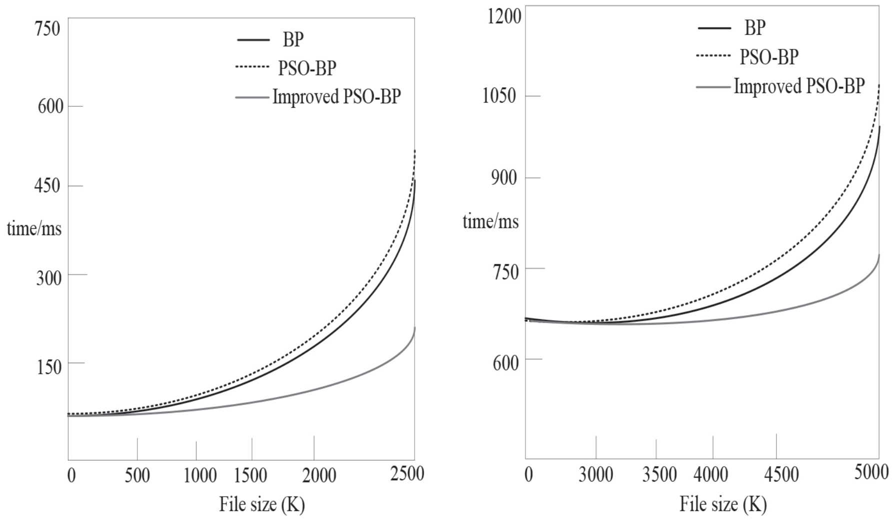


Figure 8 Real-time integration time of the three algorithms at different file sizes.

documents and the binary stream transmission will take a long time, and errors may occur due to memory exhaustion.

5. CONCLUSIONS

With the development of science and technology, the Internet has become popular, and people can work, study and shop anytime and anywhere through the Internet of Things. However, this has led to the generation of large-scale heterogeneous data. The traditional data integration technology is slow, inefficient, and cannot meet the needs of modern users. In order to address the BP neural network's susceptibility to changes in the external environment, this study proposed integrating the PSO algorithm into the BP neural network algorithm. This also enhanced the neural network's capacity for self-learning. This can also increase the BP neural network's productivity and speed when processing heterogeneous data. The relevant comparative analysis was performed to confirm that the upgraded PSO-BP algorithm has a greater real-time integration efficiency compared to the other two methods. According to the findings, the enhanced algorithm's training and testing times are longer than those of the BP neural network and the PSO-BP algorithm whether in the training set or the test set. In terms of real-time integration of heterogeneous data, the integration efficiency of the improved algorithm is also greater than that of the other two algorithms. Therefore, it makes sense to apply it to real-time data integration technology. However, this study has one major shortcoming: Only two algorithms are analyzed. There are many algorithms in artificial intelligence and big data, and other algorithms should be considered in future work.

REFERENCES

1. Kumar P S, Parthiban L. Scalable Anomaly Detection for Large-Scale Heterogeneous Data in Cloud Using Optimal Elliptic Curve Cryptography and Gaussian Kernel Fuzzy

C-Means Clustering. Journal of Circuits, Systems and Computers, 2020, 29(05):37–38.

2. Benedikt M, Grau B C, Kostylev E V. Logical Foundations of Information Disclosure in Ontology-Based Data Integration. Artificial Intelligence, 2018, 262(SEP.):52–95.
3. Shiraishi M, Ashiya H, Konno A, Morita K, Kataoka S. Development of Real-Time Collection, Integration, and Sharing Technology for Infrastructure Damage Information. Journal of Disaster Research, 2019, 14(2):333–347.
4. Schimpf B, Deanda K, Severenuk D A. Integration of Real-Time Electronic Health Records and Wireless Technology in a Mobile Stroke Unit. Journal of Stroke and Cerebrovascular Diseases, 2019, 28(9):2530–2536.
5. Luan X, Wang Y, De Schutter B. Integration of real-time traffic management and train control for rail networks - Part 1: Optimzation problems and solution approaches. Transportation Research Part B: Methodological, 2018, 115(SEP.):41–71.
6. Li G, Gaynanova I. A General Framework for Association Analysis of Heterogeneous Data. The Annals of Applied Statistics, 2018, 12(3):1700–1726.
7. He H, Gang F, Jinde C. Robust state estimation for uncertain neural networks with time-varying delay. Journal of Jishou University (Natural Sciences Edition), 2019, 19(8):1329–1339.
8. Yang H F, Lin K, Chen C S. Supervised Learning of Semantics-Preserving Hash via Deep Convolutional Neural Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(99):437–451.
9. Cheng Y, Wang D, Zhou P, Zhang T. Model Compression and Acceleration for Deep Neural Networks: The Principles, Progress, and Challenges. IEEE Signal Processing Magazine, 2018, 35(1):126–136.
10. Q. Ji. Lossless Compression Algorithm of Multimedia Data Based on Artificial Intelligence, Engineering Intelligent Systems, 2022, 30(1): 23–33.
11. Atzmon M, Maron H, Lipman Y. Point Convolutional Neural Networks by Extension Operators. ACM Transactions on Graphics, 2018, 37(4CD):71.1–71.12.
12. Cao P, Xia W, Ye M. Radar-ID: human identification based on radar micro-Doppler signatures using deep convolutional neural networks. IET Radar, Sonar & Navigation, 2018, 12(7):729–734.
13. Venieris S I, Alexandros K, Christos-Savvas B. Toolflows for Mapping Convolutional Neural Networks on FPGAs: A Survey and Future Directions. ACM Computing Surveys, 2018, 51(3):1–39.
14. [Khosravi P, Kazemi E, Imielinski M. Deep Convolutional Neural Networks Enable Discrimination of Heterogeneous Digital Pathology Images. EBioMedicine, 2018, 27(C): 317–328.
15. Yin P, Zhang S, Lyu J, Osher S, Qi Y, Xin J. BinaryRelax: A Relaxation Approach for Training Deep Neural Networks with Quantized Weights. Siam Journal on Imaging Sciences, 2018, 11(4):2205–2223.
16. Wang L F, Wu H , Liu D Y, Boutat D, Chen YM. Lur'e Postnikov Lyapunov functional technique to global Mittag-Leffler stability of fractional-order neural networks with piecewise constant argument. Neurocomputing, 2018, 302(AUG.9):23–32.
17. Zhao, Y., Bai, C., Xu, C., & Foong L K Efficient metaheuristic-retrofitted techniques for concrete slump simulation. Smart Structures and Systems, 2021, 27(5): 745–759.
18. Ye, Y. & Zhang, H. (2025). The Innovation and Transfer of Computer Science and Technology in Recent and Modern Railway Transportation. International Journal of Intelligent Information Technologies (IJIIT), 21(1), 1–17.
19. X. Li, G. Duan and F. Tian. Massive Real-Time Data Mining Algorithm for a Multimedia Database, Engineering Intelligent Systems, 2022, 30 (2): 93–103.
20. Esquembri S, Nieto J, Ruiz M. Methodology for the implementation of real-time image processing systems using FPGAs and GPUs and their integration in EPICS using Nominal Device Support. Fusion Engineering & Design, 2018, 130(MAY): 26–31.
21. Artale G, Cataliotti A, Cosentino V. Real-Time Power Flow Monitoring and Control System for Microgrids Integration in Islanded Scenarios. IEEE Transactions on Industry Applications, 2019, 55(99):7186–7197.
22. Xu, Z., Jain, D.K., Neelakandan, S. et al. Hunger games search optimization with deep learning model for sustainable supply chain management. Discov Internet Things 3, 10 (2023).
23. Lakshminarayanan V, Chemudupati V, Pramanick S K. Real-Time Optimal Energy Management Controller for Electric Vehicle Integration in Workplace Microgrid. Transportation Electrification, IEEE Transactions on, 2019, 5(1):174–185.
24. Bai X, Wang Y, Zhao J. A Novel Long-Time Coherent Integration Method for Moving Target Detection. Journal of Beijing Institute of Technology, 2021, 30(4):340–351.
25. Pinto M, Zorn K C, Tremblay J P, Dallaire F, Aubertin K, Marple E et al. Integration of a Raman spectroscopy system to a robotic-assisted 1–025001.10.