

## CUD Digital Repository

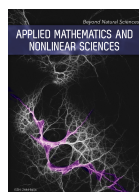
This article is licensed under Creative Commons License and full text is openly accessible in CUD Digital Repository.

### HOW TO GET A COPY OF THIS ARTICLE:

CUD Students, Faculty, and Staff may obtain a copy of this article through this link:

<https://doi.org/10.2478/amns.2021.2.00112>

Title (Article)	Cognitive computational model using machine learning algorithm in artificial intelligence environment
Author(s)	Shangyi Liu, Constantin-Viktor Spiridonidis and Mohammed Abdulrazzqa
Journal Title	<i>Applied Mathematics and Nonlinear Sciences</i>
Citation	Liu, S., Spiridonidis, C. -, & Abdulrazzqa, M. (2022). Cognitive computational model using machine learning algorithm in artificial intelligence environment. <i>Applied Mathematics and Nonlinear Sciences</i> , <a href="https://doi.org/10.2478/amns.2021.2.00112">https://doi.org/10.2478/amns.2021.2.00112</a>
Link to Publisher Website	<a href="https://doi.org/10.2478/amns.2021.2.00112">https://doi.org/10.2478/amns.2021.2.00112</a>
Link to CUD Digital Repository	<a href="http://hdl.handle.net/20.500.12519/511">http://hdl.handle.net/20.500.12519/511</a>
Date added to CUD Digital Repository	February 14, 2022
Term of Use	<a href="#">Creative Commons Attribution 4.0 International License (CC BY 4.0)</a>



# Applied Mathematics and Nonlinear Sciences

<https://www.sciendo.com>

## Cognitive Computational Model Using Machine Learning Algorithm in Artificial Intelligence Environment

Shangyi Liu<sup>1†</sup>, Constantin-Viktor Spiridonidis<sup>2</sup>, Mohammed Abdulrazzqa<sup>3</sup>

<sup>1</sup> University of Science and Technology Liaoning, Anshan 114051, China

<sup>2</sup> Department of Architecture, Faculty of Architecture and Interior Design, Canadian University Dubai, Dubai, United Arab Emirates

<sup>3</sup> Department of Business Administration, Faculty of Administration Sciences, Applied Science University, Al Eker, Kingdom of Bahrain

### Submission Info

Communicated by Juan Luis García Guirao

Received June 17th 2021

Accepted September 24th 2021

Available online December 27th 2021

### Abstract

In order to explore the application of machine learning algorithm to intelligent analysis of big data in an artificial intelligence (AI) environment, make cognitive computing meet the requirements of AI and better assist humans to carry out data analysis, first, the theoretical basis of machine learning algorithm is elaborated. Then, a cognitive computational model based on the machine learning algorithm is proposed, including the essence, principle, function, training method of deep belief network (DBN) algorithm, as well as the joint use of DBN algorithm and multilayer perceptron. Finally, the proposed algorithm is simulated. The results show that under the same parameter conditions, the accuracy rate of the DBN algorithm combined with multilayer perceptron is higher than that of the DBN algorithm; when the number of units is  $>40$ , the accuracy rate of the DBN algorithm combined with multilayer perceptron is significantly higher than that of the DBN algorithm; when the number of units is 30, the best effect can be obtained, and the error rate is  $<0.05$ , but the DBN algorithm cannot achieve this effect alone; when the number of network layers is specified as four, the error rate of the DBN algorithm combined with multilayer perceptron is  $<0.05$ , forming the optimal level. In the AI environment, the performance of the cognitive computational model based on the DBN algorithm and multilayer perceptron can reach the highest level, which makes the computer become a handy intelligent auxiliary tool for human beings.

**Keywords:** artificial intelligence; machine learning; DBN algorithm; multilayer perceptron; cognitive computational model

**AMS 2010 codes:** 92B20

<sup>†</sup>Corresponding author.

Email address: [aolingshuang@ustl.edu.cn](mailto:aolingshuang@ustl.edu.cn)

## 1 Introduction

The development of science and technology has led to the penetration of the Internet and information technology into all aspects of people's life. People's pursuit of life is no longer limited to ordinary technical means. The requirements for information products are increasingly high end and intelligent. The massive data generated by the development of intelligent systems also needs to be deeply mined in an intelligent way [1, 2]. The development of artificial intelligence (AI) and realising the intelligent analysis of big data and the mining of useful data information have become the research contents of great importance to scholars in China and foreign countries [3]. To analyse the real-time changes in big data and automatically obtain valuable data information, it is necessary to optimise and innovate the interactive analysis method of data; therefore, cognitive computing has made its appearance [4].

Cognitive computing technology is the computer simulation of the human brain to complete the analysis of data information. Its core function is to imitate the human brain for problem analysis. The difference between cognitive computing and traditional manual computing systems is that traditional computing software is based on prescribed criteria or data information when solving problems, while cognitive computing solves unpredictable and uncertain problems in human life [5]. Cognitive computing technology greatly influences the field of AI. It stimulates the computer to have the active learning function similar to the human brain, so that the computer can become a handy auxiliary for human beings to solve problems or perform work that is not suitable for the human brain to process. Cognitive computing also has the characteristics of fast data processing and intelligent processing mode [6].

With the wide use of high-performance computers, machine learning [7] has been widely used in various fields. Machine learning can extract and learn important knowledge according to the past or current data information and thereafter use the knowledge to predict unknown events. Machine learning is generally regarded as AI. In fact, it has exceeded the scope of AI. The gradual optimisation and innovation of the machine learning algorithm can promote the rapid development of AI. Along with the optimisation and innovation of the machine learning algorithm, learning ability is also very important [8, 9]. The goal of learning is to enhance the system performance through knowledge acquisition. The core task of AI is to enable computers to help human beings to do things better, and machine learning is the most core part of these things [10]. Machine learning is a frontier subject in the field of AI. The intelligence of machine learning is that it can automatically optimise the computer algorithm according to past experience. Some scholars have proposed that if the computer system contains machine learning function, it cannot be called an intelligent system [11]. As for machine learning, Sun et al. [12] optimised the performance of deep belief networks (DBNs) in analysing large-scale data and proposed a DBN computation model based on MapReduce. At present, there are still many shortcomings in the application of machine learning algorithm in data analysis through cognitive computing. For example, the validity of massive data information in the Internet cannot be determined, which leads to great difficulties in the calculation and analysis of data information.

Based on this, in an AI environment, a cognitive computational model based on the machine learning DBN algorithm combined with multilayer perceptron is proposed. Through the analysis and training of the model network, the performance of the model can reach the highest level, and the uncertainty of massive data in related fields can be remedied, which lays an experimental foundation for the combination of machine learning and AI for efficient big data analysis.

## 2 Method

### 2.1 Cognitive computing model based on machine learning algorithm

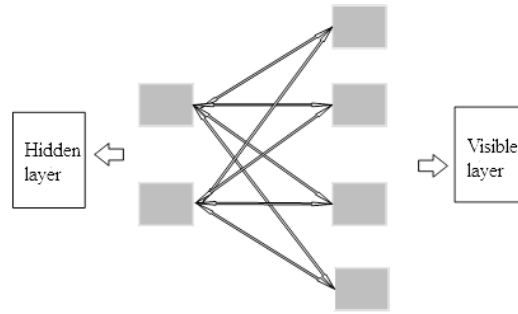
Machine learning includes three forms: supervised learning, unsupervised learning and reinforcement learning [13]. Algorithms related to machine learning include inverse linear regression, logical regression, naive

Bayes,  $K$ -nearest neighbour algorithm, support vector machine (SVM), random forest, boosting and AdaBoost [14]. Its advantage is that the same algorithm can be used to analyse and solve many different things, and the process and operation are relatively simple [15].

### 2.1.1 DBN algorithm

First, we consider the algorithm function. The DBN algorithm belongs to a kind of neural network in machine learning and can have both unsupervised learning and supervised learning functions [16]. DBN is a probability generation model, which establishes a combination distribution between observation data and tags. By training the weights between units, the neural network can train the data information with the maximum probability as the goal. DBN is used for feature recognition, data classification and probability generation. DBN algorithm has strong practicability, wide application fields and high expansion performance. It can be used in handwriting recognition, speech recognition and image processing in machine learning [17].

Second, we look at the structure and principle of the algorithm. The components of DBN are multilayer units (visible units and hidden units). Visible units can be used to receive input, and hidden units can be used for feature extraction [18]. The connection between the top two layers of units has no direction, forming the associative memory. The other neural layers belong to directed connection, and the bottom layer belongs to data vectors. A unit represents a dimension of the data vector. Further, the restricted Boltzmann machine (RBM) is a component of DBN, which can train the DBN layer by layer. The data vector can be used to judge the hidden layer in each layer. Then, this layer is used as the data vector of the unit in the next higher layer. The RBM can function independently as a clusterer. There are only two layers of units in RBM, one of which is the visible layer for training input data. The other layer is the hidden layer, which plays the function of feature detector. Figure 1 shows its structure.



**Fig. 1** RBM network structure. RBM, restricted Boltzmann machine

In Figure 1, the units (grey boxes) on the left form the hidden layer, and the units on the right form the visible layer. Each layer can be represented by a vector; each unit represents a dimension, and bidirectional connection is carried out between two layers.

In order to make the research content more general, each layer of the DBN algorithm training is a Markov random field model, and each model contains a visible layer and a hidden layer.  $U = (U_1, \dots, U_n)$  represents the variable matrix in the visible layer,  $P = (P_1, P_n)$  represents the variable matrix of the hidden layer. Based on  $(U, P)$  Gibbs distribution, the variable matrix  $P$  of the hidden layer is used to describe the marginal distribution of the variable matrix  $U$  of the visible layer.

$$S(U) = \sum_m S(U, P) = \frac{1}{K} \sum_m \exp(-\epsilon(U, P)). \quad (1)$$

In Eq. (1),  $K = \sum_{U, P} \exp(-\epsilon(U, P))$ . Therefore, the likelihood function of parameter  $\lambda$  can be defined as follows:

$$\ln L(\lambda | U) = \ln S(\lambda | U) = \ln \sum_P \exp(-\epsilon(U, P)) - \ln \sum_{U, P} \exp(-\epsilon(U, P)). \quad (2)$$

According to Eq. (2), the gradient of parameter attribute adjustment is calculated.

$$\frac{\alpha \ln L(\lambda | U)}{\alpha \lambda} = -\ln \sum_P P(P|U) \frac{\partial \in (U, P)}{\alpha \lambda} + \ln \sum_P P(P, U) \frac{\partial \in (U, P)}{\alpha \lambda}. \quad (3)$$

Here,  $\alpha$  represents the parameter adjustment value.

Each RBM contains a hidden layer and a visible layer, which represent different variables. The visible and hidden layers are not connected with itself. If the values are 0 and 1, the energy equation of the RBM can be obtained.

$$E(U, P) = -\sum_{x=1}^a \sum_{y=1}^b U_x O_{xy} P_y - \sum_{x=1}^b I_x U_x - \sum_{y=1}^a R_y P_y. \quad (4)$$

Here,  $O$ ,  $I$  and  $R$  represent the paranoid coefficients, and  $a$  and  $b$  represent the maximum ranges of the paranoid coefficient.

According to Eq. (4), it can be identified that in the energy equation, the weight should be adjusted globally and the bias coefficient should be relatively independent. If RBM is regarded as an image, the image only contains the connection between the hidden layer and the visible layer, and there is no connection inside the layer. According to Gibbs distribution and Eq. (4), the conditional probability equation can be obtained.

$$\begin{aligned} p(U_y = 1 | P) &= \sigma \left( \sum_{y=1}^a O_{xy} P_y + I_y \right) \\ p(P_y = 1 | U) &= \sigma \left( \sum_{x=1}^a O_{xy} U_x + R_y \right) \end{aligned} \quad (5)$$

where  $p$  is the probability and  $\sigma$  is the coefficient.

Then, the RBM network is trained using the contrastive divergence algorithm. After the parameter of paranoid ideation is adjusted, the learning function of parameter  $\lambda$  can be obtained.

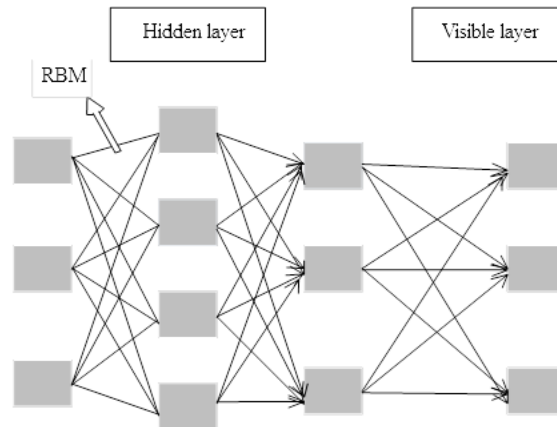
$$\lambda^{T+1} = \varphi \cdot \lambda^T + \mu \cdot \left\{ \frac{\alpha}{\alpha \lambda} \left[ \frac{1}{V} \sum_{Z=1}^V \ln L(\lambda | U_y) \right] - \beta \cdot \lambda^T \right\}, \quad (6)$$

where  $\varphi$  represents inertia factor,  $\mu$  represents the learning rate, and  $\beta$  represents the penalty factor.

The RBM training process aims to calculate the probability set that can generate many training samples. In this probability set, the training samples have the largest probability. The decisive factor of the probability combination is the weight size, so the purpose of RBM training is to find the optimal weight.

Third, we move on to DBN training. DBN is a neural network based on RBM, which can be either a generation model or a discriminant model. The process of training the DBN involves the use of unsupervised layer-by-layer analysis to get the weights. Thus, the training process of DBN should be carried out layer by layer. In each layer, the hidden layer prediction should be carried out according to the data vector. Then, the hidden layer is used as the information vector for adjacent higher layers. In other words, many RBMs can be combined to form a DBN. The visible layer (input value) of the next RBM is the hidden layer (output value) of the previous RBM. During the training process, the RBM of the previous layer must be fully trained before the RBM of the next layer can be trained, which goes all the way to the end layer. Finally, the units are the essential components of the neural network, and the DBN is composed of many units and RBMs. The restricted layer of the DBN network structure comprises the visible layer and the hidden layer, and there are corresponding connections between layers. However, there is no connection between the neurons in the interior of the layer (grey boxes). By training the units of the hidden layer, the association between the higher-order data displayed in the visible layer can be mined. Figure 2 shows the composition of a DBN network.

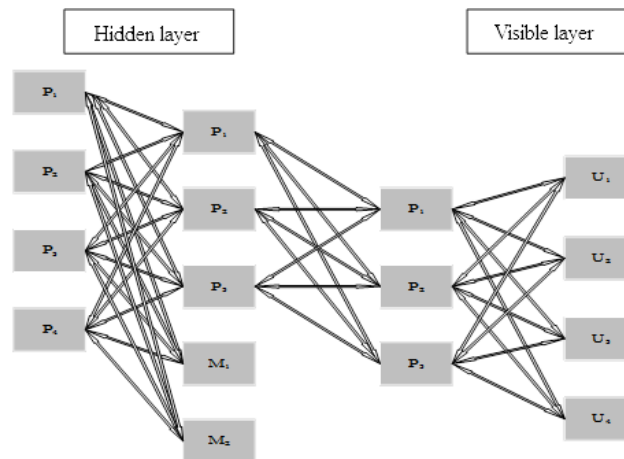
Fourth, we come to the essence of the DBN algorithm. From the perspective of unsupervised learning, the goal is to preserve the nature of the original features to the greatest extent and to reduce the dimensions



**Fig. 2** DBN network. DBN: deep belief network; RBM: restricted Boltzmann machine

of features. From the perspective of supervised learning, the goal is to keep the error rate of classification as low as possible. Irrespective of whether supervised or unsupervised learning is adopted, the DBN algorithm is essentially a process of feature learning, in other words, a process to get the best feature expression.

Fifth, we detail the DBN training process. The first RBM needs to be fully trained; its weight and offset are fixed, and its hidden units are used as the import vector of the next RBM; then, after the second RBM is fully trained, the second RBM is overlapped with the first RBM. It is necessary to repeat the above steps several times. If the data in the training set contain labels, when the top RBM is trained, the visible layers in the RBM must contain both visible units and the units representing classification labels. Both of them are trained at the same time, as shown in Figure 3 below.



**Fig. 3** DBN training framework. DBN: deep belief network

For each training data, the corresponding labelled units can be set to 1 after being opened, and the remaining ones will be closed and set to 0. The parts labelled  $M_1$  and  $M_2$  in Figure 3 are the top labels for training in RBM.

The neural unit at the top of the network evaluates and classifies tasks based on the data generated by the lower network. When the DBN is trained at the end of the network layer, it can use the feedforward neural network to slightly adjust the previous evaluation and classification according to the data information with labels. However, the processing method of this study is better than the direct use of a feedforward neural network, because the DBN has higher efficiency, and it only changes the weight parameters and then trains in a small area. Moreover, the training speed is very fast and the convergence time is relatively short.

## 2.2 Construction of the cognitive computational model

The DBN algorithm and multilayer perceptron are combined, and the cognitive computational model is designed. The last layer of the DBN is the output value of the RBM, which is used as the input value of the multilayer perceptron. Based on the input values and training samples, the linear perceptron training is carried out to obtain the decision model. In the decision model, the conclusion whether the data information is valid or not is obtained. After the DBN training results are obtained, the multilayer perceptron training is carried out. For the training of the multilayer perceptron, an error control strategy is added. This strategy is also an important part of the joint algorithm. The input value of the multilayer perceptron is set as  $C = (C_1, C_2, \dots, C_n)$ , the relative weight vector is set as  $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n)$  and the threshold is represented by  $\eta$ ; the cognitive strategy function can be expressed as follows:

$$Y = \gamma^d X + \eta. \quad (7)$$

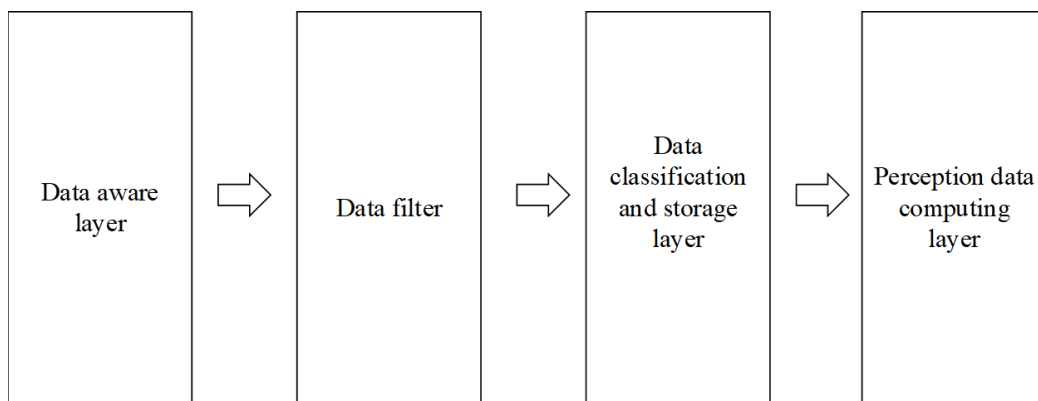
If  $y \geq 0$ , the result of the strategy is valid. If the result is  $< 0$ , the policy result is invalid.

Then, the training rules of the model weight are established.

$$Q_{(l+1)} = Q_l + \kappa(R - Y) \cdot X_Q, \quad (8)$$

where  $\kappa$  represents the learning rate parameter.

Thus, Figure 4 is a cognitive computational model based on machine learning algorithm.



**Fig. 4** Cognitive computational model based on machine learning algorithm

First, let us see the data acquisition layer. It senses the corresponding Internet data flow generated by users using computers and enjoying related services. The relative expansion of hardware functions, such as wearable devices or smart phones and other related intelligent terminals, enables the system to obtain a large volume of information. The data acquisition layer can transfer all the collected data information to the perceptual data storage layer. At the same time, in order to make the acquired perceptual data effective and reduce the excessive extra cost caused by storing invalid data information, a simple filtering strategy is set up in the data acquisition layer. Based on the properties of the data, the information is filtered to find out whether the information is the perceptual data related to the follow-up experiment. If it is, the data are passed to the data-computing layer. If it is not, it is necessary to filter out the data. Under this operation, many useless and damaged data information will not continue to occupy the system memory. Then, the filtering operation is implemented from the specified range of data information. For example, if the ambient temperature is  $-100^\circ\text{C}$ , it can be judged that there is an error in the measurement of this data information, and the data should be filtered. This kind of data-filtering strategy can greatly reduce the extra cost of the system.

Second is the data storage layer. The data brought in by a lot of hardware and associated services are generally massive. If these data are stored, most of the storage space of the system will be occupied. Therefore,



how to find a more efficient and secure method to store and process these data is an urgent problem to be solved. Therefore, the storage requirements of the storage layer are deeply optimised, i.e., we propose a data storage layer that can carry out data perception before and after. In this layer, a distributed storage system based on different attributes of the data is proposed. In the system, based on the data classifier, the obtained data are divided into static and dynamic data, and different processing methods are formulated according to the different data types. For the static dataset (time or work attributes), the distributed database is first imported and, at the same time, a static data export port is developed for the execution of the function of the data-computing layer. For the data stream with a form of perception, such as the ambient temperature in different time periods, the performance of this kind of data has strong timeliness. If the data can be analysed in the shortest time, valuable analysis of the data can be carried out in the shortest time.

Finally, for the above types of data, the processing method of this study is to transfer from the cache space directly to the perceptual computing layer, which is provided to the upper-level computing for use. However, this will cause a different problem, and the data information that is being cached will be lost quickly. For this problem, a kind of fault-tolerant recovery strategy is formulated to store a backup while the data are used, so as to ensure the fault-tolerant performance of the data. Based on this, even if the system runs abnormally, it can still export the data that has not been analysed from the backup for research and calculation after the system is repaired. The research measures of static data are very mature at present. Therefore, in the data-computing layer, the core of the research is to process the perceptual dataset efficiently.

### 2.3 Algorithm simulation

MATLAB software was used in the algorithm simulation of this research. The simulation part consists of 16 imported values, upper and lower environment sets and two types of export results. In the simulation, four kinds of input and four kinds of information of upper and lower layers are used, and the machine learning algorithm proposed in the research is used to process the same. The result is whether the imported data information is valid.

The import values are set to  $a_1$ ,  $a_2$ ,  $a_3$  and  $a_4$ , and the upper- and lower-level data are represented by  $g_1$ ,  $g_2$ ,  $g_3$  and  $g_4$ . When the corresponding value is equal to one, it represents the existence of this input value, or it can be the upper- and lower-level properties in the current environment. The context of the imported value is shown in Table 1.

**Table 1** Context of import values

Imported value	Context information
$a_1$	$g_1, g_2, g_3, g_4$
$a_2$	$g_1, g_2, g_3, g_4$
$a_3$	$g_1, g_2, g_3, g_4$
$a_4$	$g_1, g_2, g_3, g_4$

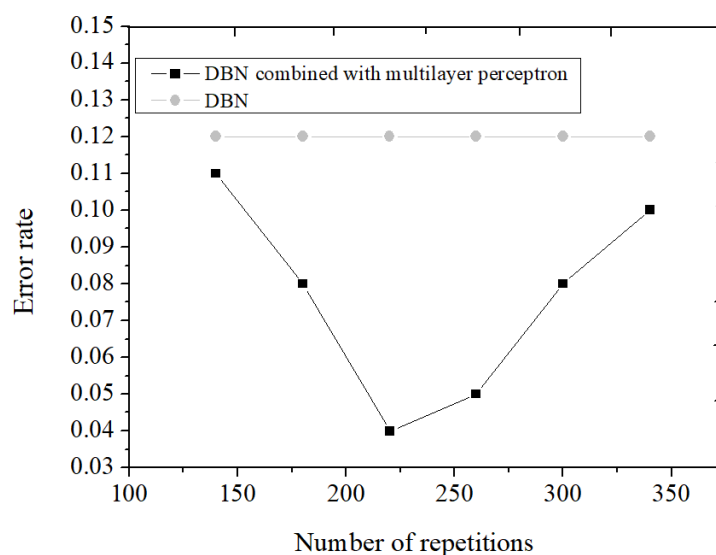
In the simulation process, the machine learning algorithm is trained by using three of the 16 derived results. However, all 16 possible data points should be used for testing. The training dataset is composed of eight derived results and 200 copies. The order of these data is disordered and used in 45 iterations. The 1,600 ( $8 \times 200$ ) units of data are divided into 16 datasets, and each set contains 100 data points. In each iteration, all the datasets are used. At the same time, other parameters, such as the DBN, the DBN algorithm and the number of units contained in the multilayer perceptron, should be considered.



### 3 Results

#### 3.1 The influence of repetition times of the training set on algorithm performance

It is considered that the number of repeated runs of the training set will have a certain impact on the accuracy of the model results. The number of units in the hidden layer is set to 50, and the number of network layers is three. From 100 to 450, the training set is optimised repeatedly every 40 increments. Figure 5 shows the results.



**Fig. 5** Relationship between model error rate and repetition number. DBN: deep belief network

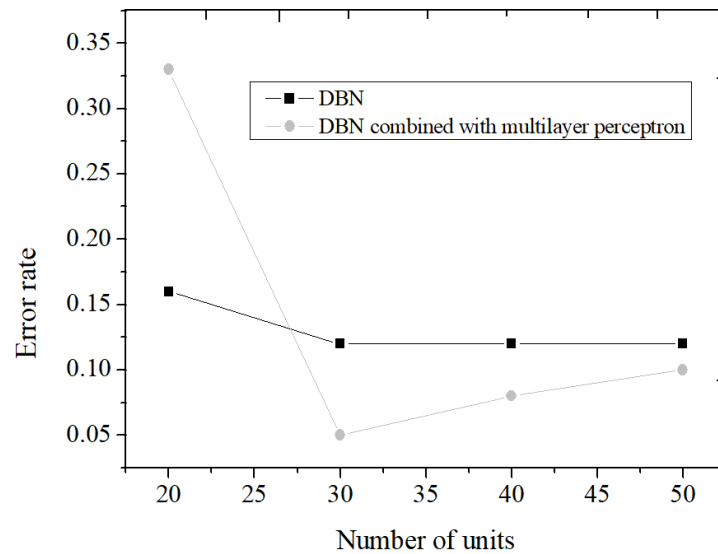
Figure 5 shows that the accuracy rate of the results obtained by the DBN algorithm combined with multilayer perceptron training according to the specified parameters is better than that of the DBN algorithm under the same parameters. Especially when the repetition number is 220, the error rate of the model is the lowest and  $<0.05$ .

#### 3.2 The influence of the number of hidden layer units on the function of machine learning algorithm

When the influence of the number of hidden layer units on the accuracy of the model results is analysed, the number of iterations of the training set is 250 and the number of network layers is three. From 20 to 50, the number of units in the hidden layer is optimised every 10 increasing units. Figure 6 shows the corresponding results obtained.

Figure 6 shows that when the number of units is  $>40$ , the accuracy rate of the DBN algorithm combined with multilayer perceptron is significantly higher than that of the DBN algorithm alone. Especially if the number of units is 30, the DBN algorithm combined with multilayer perceptron can give the best effect and can control the error rate to  $<0.05$ , which the DBN algorithm alone cannot obtain.

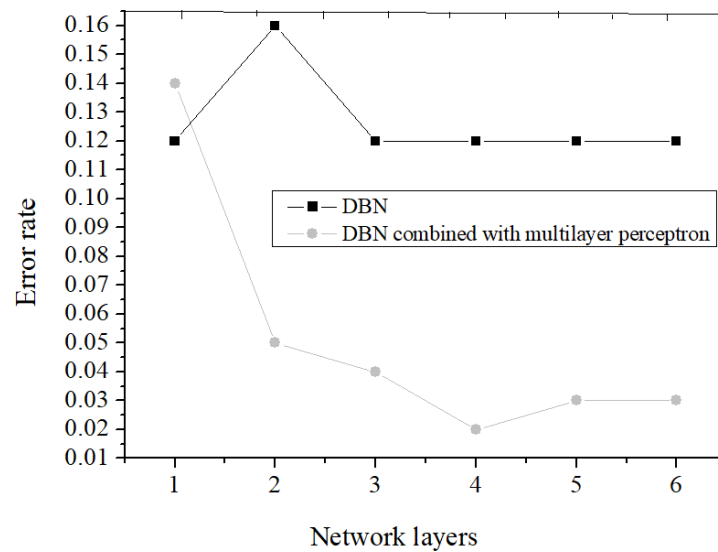
Therefore, the number of neurons in the hidden layer of the DBN network plays an important role in the accuracy of the model. When the number of neurons exceeds a certain range, the accuracy rates of the DBN network and the multilayer perceptron are basically the same. However, the accuracy rate of the combined algorithm can reach the best state, which is the effect that the DBN network and multilayer perceptron cannot achieve alone.



**Fig. 6** Relationship between model error rate and number of units. DBN: deep belief network

### 3.3 Influence of the number of network layers on algorithm performance

The number of repetitions of the training set is 300. The number of units in the hidden layer is 50. The number of network layers is changed every one increment from one to six. Each experiment is repeated 10 times, and the average value is obtained. Figure 7 shows the result.



**Fig. 7** Relationship between model error rate and number of network layers. DBN: deep belief network

Figure 7 shows that the decision accuracy of the proposed DBN algorithm combined with multilayer perceptron is higher than that of the traditional DBN algorithm on the condition that the number of network layers in the model is equal. Especially when the number of network layers is four, the accuracy of the DBN algorithm combined with multilayer perceptron is the highest. It shows that the theoretical decision accuracy of the DBN algorithm combined with multilayer perceptron can exceed 99% under the parameters specified in the study.

There are two problems when a multilayer perceptron is used. One is that the efficiency of getting the optimal solution in the training process is not high, and it is easy to fall into the local optimal solution; the other

is that the accuracy of data processing is not high. The DBN algorithm combined with the principle of multilayer perceptron can solve the above problems, and the accuracy of the results can be the highest.

## 4 Conclusion

Neural network technology can process and analyse the data flow in the era of big data more efficiently, which is also the core problem to be solved at present. Especially in the current situation, the scale of data generation is gradually increasing, and people have higher requirements for the results of data processing, which makes it difficult to find the valuable data information they need from the massive data. In this environment, people gradually use computer cognitive computing and other information technologies to complete the cognitive and problem analysis methods similar to the human brain and, finally, make accurate decisions.

According to the analysis of AI environment and the theoretical basis of machine learning algorithm, the cognitive computational model based on the DBN algorithm is proposed, and the principle and function of the DBN algorithm are analysed. At the same time, the training method for the algorithm is proposed, and the DBN algorithm and multilayer perceptron are combined to construct the cognitive computational model. Under the condition of the parameters specified, the accuracy of the results obtained by the model can be the highest, and the error rate can be controlled to  $<0.05$ . Finally, it shows that the theoretical decision accuracy of the model can exceed 99%.

However, there are still some deficiencies, such as the preprocessing method for feature selection in the cognitive computational model. In the AI environment, the data scale is very large, and the nature is diverse. A large number of feature attributes will be generated during actual research. If analysis measures are taken for all the feature attributes, it will certainly consume a lot of time, which is not in line with the current experimental research rules. If a method, which can preprocess feature attributes before data processing and analysis to exclude some feature attributes that do not affect the results of data analysis, can be developed, it can greatly enhance the data analysis efficiency of the cognitive computational model. Therefore, the development and optimisation of the feature selection method is an important research direction, which is also the next step in our follow-up research.

## References

- [1] Tiase, V. L., Hull, S. C., Troseth, M., & Schnall, R. Development and psychometric testing of the Readiness to Engage with Patient-Facing Health Information Technology Tools (RE-PHIT) scale. *International Journal of Medical Informatics*., 2018; 118(8):1-4
- [2] Stones, S. R., Swallow, V., Majeed-Aris, R., & Hall, A. OP0243-PAREInvolvement of Children and Young People with Long-Term Conditions in the Development of Mobile APP Technology to Promote Disease Self-Management. *Annals of the Rheumatic Diseases*., 2015; 74(Suppl 2):163.3-164
- [3] Wang D, He W, Shi R. How to achieve the dual-control targets of China's CO2 emission reduction in 2030? Future trends and prospective decomposition. *Journal of Cleaner Production*., 2019; 213(3):1251-1263
- [4] Chen Y, Elenece A J, Weber G. IBM Watson: How Cognitive Computing Can Be Applied to Big Data Challenges in Life Sciences Research. *Clinical Therapeutics*., 2016; 38(4):688-701
- [5] Saibaba A K, Kitanidis P K. Fast computation of uncertainty quantification measures in the geostatistical approach to solve inverse problems. *Advances in Water Resources*., 2015; 82(9):124-138
- [6] Jordan M I, Mitchell T M. Machine learning: Trends, perspectives, and prospects. *Ence*., 2015;349(6245): 255-260
- [7] L. R. Brewster, J. J. Dale, T. L. Guttridge, S. H. Gruber, A. C. Hansell, M. Elliott, I. G. Cowx, N. M. Whitney & A. C. Gleiss. Development and application of a machine learning algorithm for classification of elasmobranch behaviour from accelerometry data. *Marine biology*., 2018;165(4):62.1-62.19
- [8] So, A. M. C., Jain, P., Ma, W. K., & Scutari, G. Nonconvex Optimization for Signal Processing and Machine Learning [From the Guest Editors]. *IEEE Signal Processing Magazine*., 2020; 37(5):15-17
- [9] Togashi K. Applications of artificial intelligence to endoscopy practice: The view from Japan Digestive Disease Week 2018. *Digestive Endoscopy*., 2019;31(3):270-272
- [10] Hou, R., Kong, Y., Cai, B., & Liu, H. Unstructured big data analysis algorithm and simulation of Internet of Things

- based on machine learning. *Neural Computing and Applications.*, 2020; 32(10):5399-5407
- [11] Sun H, Jing W, Hu B. Parallel computing of KNN Query in road network based on MapReduce. *Procedia Computer science.*, 2019; 147, 381-388
  - [12] Mao, X., Yang, H., Huang, S., Liu, Y., & Li, R.. Extractive Summarization Using Supervised and Unsupervised Learning. *Expert Systems with Applications.*, 2019; 133(11):173-181
  - [13] Kamburugamuve, S., Wickramasinghe, P., Ekanayake, S., & Fox, G. C.. Anatomy of machine learning algorithm implementations in MPI, Spark, and Flink. *Experimental Mechanics.*, 2018; 32(1):61-73
  - [14] Cao, R., Bhattacharya, D., Hou, J., & Cheng, J.. DeepQA: improving the estimation of single protein model quality with deep belief networks. *Bmc Bioinformatics.*, 2016;17(1):495
  - [15] Gou, C., Wang, K., Yao, Y., & Li, Z. Vehicle License Plate Recognition Based on Extremal Regions and Restricted Boltzmann Machines. *IEEE Transactions on Intelligent Transportation Systems.*, 2015; 17(4):1096-1107
  - [16] Tsai C Y, Huang C H. An adaptive dynamic range compression with local contrast enhancement algorithm for real-time color image enhancement. *Journal of Real-Time Image Processing.*, 2015; 10(2):255-272
  - [17] Yaman, H., Yavuz, E., Er, A., Vural, R., Albayrak, Y., Yardimci, A., & Asilkan, Ö. The use of mobile smart devices and medical apps in the family practice setting. *Journal of Evaluation in Clinical Practice.*, 2016; 22(2):290-296
  - [18] Yang M C, Chang Y H, Kuo T W. Virtual Flash Chips: Reinforcing the Hardware Abstraction Layer to Improve Data Recoverability of Flash Devices. *IEEE Transactions on Computers.*, 2016;65(9):2872-2883