Research on Anti-Stealing Algorithm Based on Deep Learning

Shunjun Chen[1], Biao Long[2], Xiaoyan Wang[3], Xiaoling Wang[1], Liya Ye[1]

[1]State Grid Zhejiang Electric Power Co., Ltd. Training Center, China

[2]State Grid Zhejiang Anji County Power Supply Co., Ltd., China

[3]State Grid Hangzhou Xiaoshan Power Supply Co., Ltd., China

Abstract:

In recent years, the emergence of various smart professional hi-tech power stealing methods has made stealing behavior more hidden. Facing the increasingly severe large-scale professional power theft phenomenon, traditional anti-theft methods lack the ability to cope with high-tech power theft. This paper applies deep learning algorithms in artificial intelligence to anti-theft problems and analyzes anti-theft models. In the construction process, by constructing an anti-theft model, and proposing methods for optimizing and training the model, an anti-theft model that can predict anti-theft behavior is finally obtained.

Keywords: Artificial intelligence, Deep learning, Anti-theft model.

I. INTRODUCTION

Electric energy is a key factor in promoting my country's economic development and maintaining social stability, and it is a public service related to the lives of people in society. In order to seek personal benefits, electricity theft often occurs and continues to be repeatedly prohibited. In recent years, electricity stealing methods have shown a trend of specialization, and the main body of electricity stealing has gradually expanded from dedicated users to low-voltage users. The emergence of various intelligent professionalhigh-tech stealing methods has made the stealing behavior more concealed and formed a certain scale. Large-scale and long-term power theft is a huge damage to the normal and stable operation of the power grid and the economic benefits of the power grid company. Based on the above reasons, anti-theft

has always been one of the important tasks of power supply companies.

In the face of the above-mentioned serious electricity theft situation, we must first strengthen the technical training of employees, improve the quality of employees, and improve the workforce. Only in this way can we combat electricity theft[1]. With the development of network informatization, electricity theft analysis and judgment need to use a large amount of data generated by users' electricity consumption, introduce artificial intelligence algorithms and big data analysis technology to mine and analyze normal users' electricity consumption habits, and combine the previous identification experience and rules to make the difference For users who steal electricity, establish an anti-theft model to accurately identify users suspected of high electricity theft. In-depth research on electricity theft can improve work efficiency and benefit. Traditional anti-electricity stealing technology reforms usually improve electric energy metering devices, but today's electricity theft behavior continues to be professional and large-scale. Traditional hardware preventive measures lack the ability to deal with high-tech electricity theft, so artificial intelligence is needed. The algorithm combines big data to propose a new method.

II. OVERVIEW OF DEEP LEARNING ALGORITHMS FOR BIG DATA MODELING AND ANALYSIS

Traditional statistical analysis algorithms, data mining, and machine learning are the main methods of big data modeling and analysis[2]. Big data modeling and analysis is based on building mathematical models, learning the characteristics and the laws of big data, and obtaining useful knowledge. The characteristic is the key attribute that determines the knowledge contained in the data object.

The extraction of data features by traditional machine learning algorithms will depend more on the researcher's professional and profound understanding of the problem being studied, and whether the final model can accurately fit known problems with a high probability depends on whether the feature extraction is appropriate. Relying on manual selection of features will have certain limitations, while deep learning methods can automatically learn features.

It uses some simple but non-linear models to transform the original data into higher-level and more abstract expressions. Deep learning is to learn more useful features by establishing a machine learning model with multiple hidden layers and a large amount of training data, and ultimately improve the accuracy of classification or prediction[3]. Deep learning is a multi-layer neural network. The depth of the model structure usually has more than ten layers. It has super generalization ability and can also learn very complex functions, which is more conducive to

discovering the rich information contained in massive data. At the same time, the deep learning model has the ability to automatically learn features. The features of each layer are not manually designed, but learned from the data itself. Through layer-by-layer feature transformation, the sample features in the original space are transformed into a new feature space. Make classification and prediction easier. Perform cleaning, screening and deep mining of a large amount of data inside the power system, extract the respective data characteristics for different user categories, and use deep learning algorithms to build a data simulation model to simulate user load data, which can fully tap the deep layers hidden in the massive power data Relationship, so it has a unique advantage.

2.1 Convolutional Neural Network

Convolutional Neural Network (CNN) is often used in various occasions such as image recognition and speech recognition. In image recognition competitions, methods based on deep learning are almost all based on CNN [4]. Convolutional neural networks, like multilayer perceptrons, can be constructed by assembling layers. The basic structure of CNN consists of an input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer. The pooling layer merges multiple adjacent nodes into one to merge similar features, thereby reducing the amount of training data. The hidden features in the project are extracted, multi-level training is performed, and dimensionality reduction processing is performed. Finally, the entire CNN is trained in a supervised manner.

2.1.1 Convolutional Layer

The main calculation in the convolutional neural network is the convolutional layer. The target task is to perform feature extraction. The vectorized text vector and the convolution sum are convolutional calculations, and the operation of extracting the data before and after the data will be obtained. The convolution result of is subjected to activation function operation, and the result of the convolution function is used as the output result of the layer. Extracting local features of input data through convolutional layer operations can effectively help reduce model training parameters, and further avoid overfitting. The convolutional layer has three characteristics: local perception, parameter sharing and multi-core convolution.

2.1.2 Pooling Layer

In the convolutional layer, the pixel level of the convolution kernel is used to accurately find the changing position of the feature in the picture. However, in order to make the model have generalization capabilities and solve the problem of various types of pictures, the target object

in the data set will not always remain in the same location, in order to alleviate the over-sensitivity of the convolutional layer to location, it also aggregates the feature maps output by the convolutional layer, which is usually used to pool the feature maps after the convolution calculation. The pooling layer can reduce the dimensionality of the extracted feature information. On the one hand, it makes the feature mapping smaller, simplifies the complexity of network calculation, and avoids over-fitting to a certain extent; on the other hand, it compresses the main features of feature extraction. Common sampling rules include mean sampling, random sampling and maximum sampling. The mean value sampling is performed by taking the average value in each subregion as a sampling result, the maximum sampling is to select the maximum value of the subregion as the sampling result, and random sampling is to assign probability to the neuron and perform sampling according to the probability.

2.1.3 Activation Function

The significance of the activation function is to preserve and map the "features of the activated neuron" through the function, that is, the activation function can add non-linear factors to the neural network, which is the key to solving the non-linear problem in the neural network. There are several commonly used activation functions: linear function, slope function, threshold function, sigmoid function, hyperbolic tangent function, etc. ReLU function, sigmoid function and tanh function will be used in this article.

The mathematical expression of the ReLU function is $f(x) = \max(0, x)$. Its advantage is that it is easy to calculate and only requires one interpretation value to get the activation value, but its disadvantage is also because of its simplicity, which makes training very fragile;

The sigmoid function is used for the output of hidden layer neurons. The value range is (0,1). It can map a real number to the interval of (0,1). Its mathematical form is:

$$S(x) = \frac{1}{1 + e^{-x}}$$

tanh is one of the hyperbolic functions, and its mathematical form is:

$$t \quad n \quad hx = \frac{sinhx}{coshx} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

2.2 Time Recurrent Neural Network

Many of the problems that artificial intelligence needs to solve are sequential, such as

natural language processing, video image processing, weather observation data, and stock trading data. Multi-layer perceptron (MLP) or convolutional neural network (CNN) can only be recognized according to the current state. When dealing with the problem of sequence of events, RNN and LSTM models are needed[5].

2.2.1 Recurrent Neural Network

The principle of RNN is to connect the output of the neuron back to the input of the neuron. This design can make the neural network have a "memory" function, as shown in Figure 1.
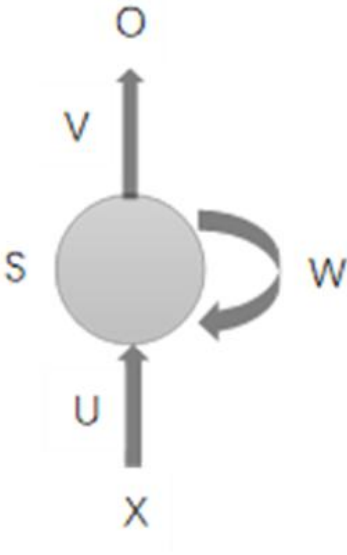

Fig 1: Schematic diagram of RNN model

Where X is the input of the neural network, O is the output of the neural network, (U, V, W) are the parameters of the neural network, S is the hidden state, which represents the "memory" of the neural network, as shown in Figure 2.
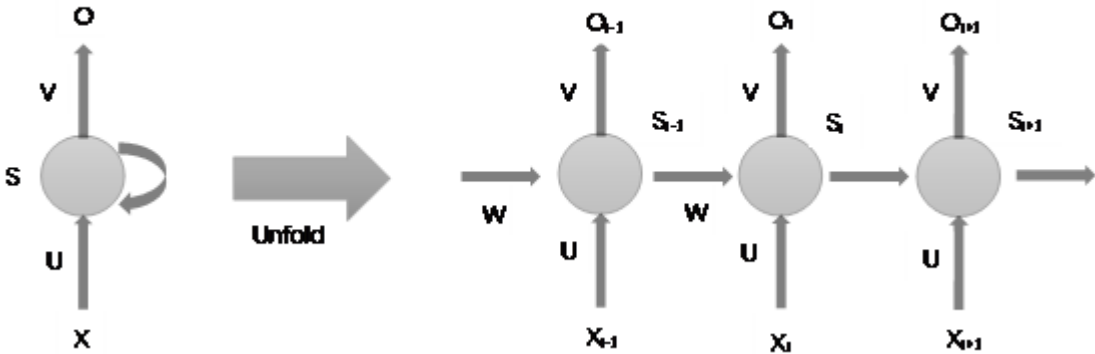
Fig 2: The principle of the RNN model

There are 3 time points in the above picture, in order: "t-1", "t", "t+1".

For example, at time t, X_t is the input of the neural network at time t, O_t is the output of the neural network at time t, (U, V, W) are the outputs of the neural network, and the W parameter is the output at time t-1, And as the input at time t, S_t is the hidden state, which represents the "memory" of the neural network. It is the input X_t after the current time point, plus the state S_{t-1} at the previous time point and the parameters of U and W The result of the joint evaluation is as follows:

$$S_t = f([U]X_t + [W]S_{t-1})$$

The f function represents the activation function, and a non-linear function is usually used, such as ReLU.

2.2.2 LSTM Model Introduction

RNN is prone to the problems of gradient explosion and gradient disappearance during the training process, which will cause the gradient to gradually fail to pass down during the model training process. For example, when the gradient explosion occurs, the program will show a NaN error. The long-term dependence is that the RNN will lose the ability to learn the information connected to the remote each time the interval increases. That is, RNN has no long-term memory, and long-term short-term memory (LSTM) is specifically designed to solve the long-term dependence of RNN, as shown in Figure 3.
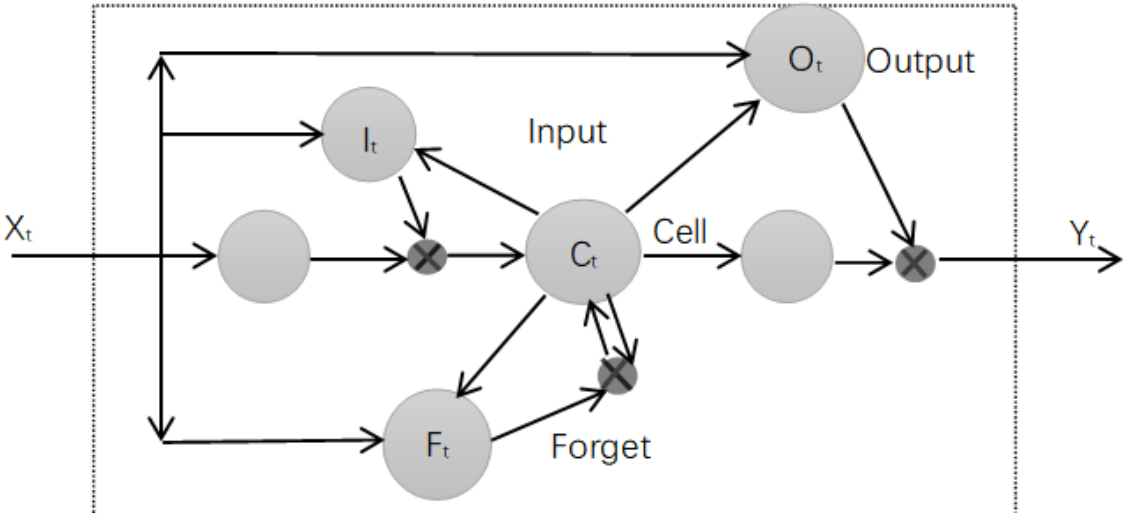
Fig 3: LSTM model structure diagram

In the LSTM neural network, each neuron is equivalent to a memory cell, where X_t and Y_t represent the input vector and output vector, respectively, and the cell is the state of the memory cell of LSTM. LSTM controls the state of memory cells through a mechanism called "gates", deleting or adding information to them.

2.3 Deep Belief Network

The DBN (deep belief network) model is a generative model. It also trains the weight parameters in neurons. It can not only identify and classify features, but also allow the entire neural network to generate training data with the greatest probability. The structure is shown in Figure 4.
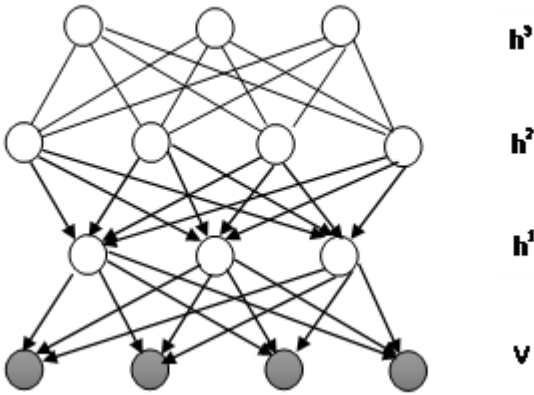
Figure 4: DBN model structure diagram

The basic component of DBN is the restricted Boltzmann machine (RBN). Its classic network structure is composed of multiple RBNs and a layer of BP. In training, pre-training is performed first, that is, each layer of RBN network is separately unsupervised training ensures that the RBN of each layer will not affect other RBNs, thereby retaining more feature information. But after pre-training, the entire DBN model does not have the ability to express input data. Therefore, fine-tuning is needed to use the classifier model as the last layer of DBN, such as BP neural network. The previous pre-training process can be regarded as the initialization operation of the weights of the classifier model, thus speeding up the training time and avoiding the local optimization of the BP neural network due to the random initialization of the parameters [6].

2.4 Adam Algorithm

Adam is a first-order optimization algorithm that can replace the traditional stochastic gradient descent process. It can iteratively update neural network weights based on training data. Adam algorithm has the advantages of easy implementation, high computational efficiency, and low storage requirements. The Adam algorithm only needs to calculate one step of the loss function, and different parameters have different learning rates. These learning rates are automatically selected by the algorithm Adam according to the first and second moment estimation of the parameter gradient.The Adam algorithm is shown in TABLE I.

TABLE I. Adam algorithm

Adam algorithm [7]

Require: The initial learning rate η, the exponential decay rate estimated by the first and second moments are β_1 and β_2, respectively, where $\beta_1, \beta_2 \in [0,1)$.
Require:θ is the initial parameter, the first and second moments are initialized to $m = 0$ and $v = 0$ respectively, and the time step is initialized to $t = 0$.

while stop criterion does not meet do

Randomly select m samples from the training set,$\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}, y^{(i)}$ is the true value corresponding to the sample $x^{(i)}$.

t←t+1

Calculate the average gradient of m samples:

$$g_t \leftarrow \frac{1}{m} \nabla_{\theta_{t-1}} \Sigma_i L\left(f_t\left(x^{(i)}; \theta_{t-1}\right), y^{(i)}\right)$$

Update of biased first moment estimation:

$$m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

Update of biased second moment estimation:

$$v_t \leftarrow \beta_2 r + (1 - \beta_2) g_t \odot g_t$$

The deviation correction of the first moment: $\widehat{m}_t \leftarrow \frac{m_t}{1 - \beta_1^t}$

Deviation correction of the second moment: $\hat{v}_t \leftarrow \frac{v_t}{1 - \beta_2^t}$

Calculate the update amount:

$$\Delta\theta_t \leftarrow -\eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \delta}}(\text{Element-level operations})$$

Parameter update: $\theta_t \leftarrow \theta_{t-1} + \Delta\theta_t$

end while

2.5 Dropout Algorithm

Deep neural networks with a large number of parameters are very powerful machine learning systems[8]. However, in such a network, overfitting is a serious problem. Moreover, the running speed of large networks is very slow, which makes it difficult to solve the overfitting problem by combining the predictions of multiple different large neural networks in the testing phase. The Dropout method can solve this problem. As shown in Figure 5.
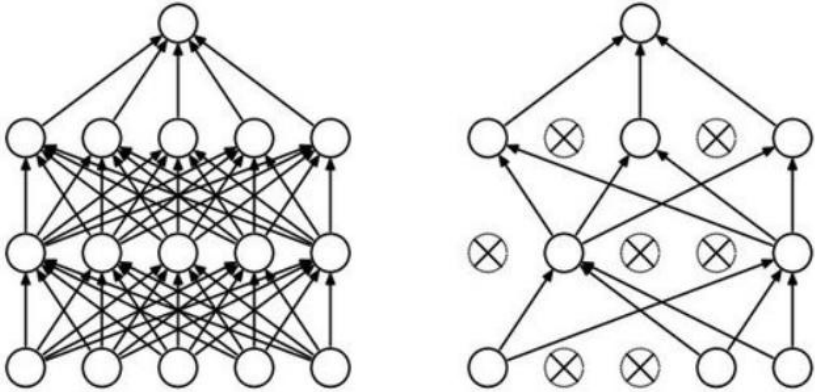

Fig 5: Standard neural network and neural network after dropout

The main idea is to randomly delete units (and corresponding connections) from the neural network during the training process, so as to prevent excessive adaptation between units. In the training process, samples are removed from networks with exponentially different "sparseness". In the testing phase, it is easy to approximate by averaging the prediction results of these sparse networks by using a single untwined network with a smaller weight. This can effectively avoid over-fitting, and can get a greater performance improvement compared to other regularization methods. Dropout technology has been proven to improve the performance of neural networks in supervised learning tasks in the fields of computer vision, speech recognition, text

classification, and computational biology, and achieve the best results in multiple benchmark data sets.

2.6Batch Normalization Algorithm

The distribution of learning data is the feature learning essence of convolutional neural networks. When training deep neural network models, the input values of each layer of the network will continue to change with the training process. This situation is called covariate shift. When the problem of covariate shift occurs, it is necessary to reduce the learning rate of all parameters, which makes the convergence of the model more difficult. At the same time, it is difficult to converge to the optimal point, which greatly reduces the generalization ability of the network. Especially when the model uses saturated nonlinear units such as sigmoid, the covariate offset problem will cause some neuron activation values to stay in the saturation zone for a long time, and the impact will be more serious. On the other hand, the convolutional neural network training process is complicated. As long as the previous network has a small change, the change will be accumulated and amplified in the process of layer-by-layer transmission, causing the network model to become unstable. Once the input data distribution of a certain layer of the network changes, or the distribution of each batch of data is different, the network must learn to adapt to different data distributions during each iteration, which will greatly reduce the training speed of the network[9].

Therefore, this article introduces the batch normalization algorithm (Batch Normalization, BN algorithm for short). For each layer of the neural network model, the BN algorithm regularizes each element in the output vector, and normalizes the input data to the mean value 0, the standard deviation is 1.At the same time, the BN algorithm introduces learning parameters γ and β, so that the data normalization process does not affect the characteristics learned from the upper network. The normalization and forward propagation process of the BN algorithm is as follows, where μ_B is the mean value,δ_B^2 is the variance, and y_i is the output data.

$$\mu_B = \frac{1}{m}\Sigma_{i=1}^{m}x_i$$

$$\delta_B^2 = \frac{1}{m}\Sigma_{i=1}^{m}(x_i - \mu_B)^2$$

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\delta_B^2 + \varepsilon}}$$

$$y_i = \gamma \hat{x}_i + \beta \equiv BN_{\gamma,\beta}(x_i)$$

In the normalization layer, each neuron has parameters γ and β. As long as $\gamma^k = (\text{Var}[x^k])^{1/2}$ and $\beta^k = E[x^k]$ are satisfied, the data feature distribution learned from the previous layer can be restored. The neural network model added to the BN layer can use saturated nonlinear units, and can use a higher parameter learning rate, thereby reducing the difficulty of training and improving the training effect.

III. APPLICATION OF DEEP LEARNING ALGORITHM IN ANTI-STEALING MODEL

In recent years, deep learning algorithms are mostly used in image recognition, speech recognition and other fields, but there are few precedents in the field of anti-electricity theft. The design idea of the anti-electricity stealing model based on deep learning algorithm is to divide the pre-processed data into training set, verification set, and test set. Use the training set data to train the deep learning model, and observe the verification set loss function value and accuracy. The change in the rate is used to adjust the value of the hyperparameter, and finally the accuracy of the trained model is obtained with the test data set.

First carry out data preprocessing. The original data such as existing user electricity data and electricity stealing data are cleaned and formalized, and divided into three parts: training set, verification set and test set.

Secondly, use the training data set and the validation data set to build a deep learning algorithm model. The experimental analysis method is used to determine the parameter settings of the model, and then the optimal network structure of the model is determined. In this process, the number of nodes in the input layer, the number of hidden layers, and the number of nodes need to be determined[10]. When the activation function uses ReLU, the initial value of the weight uses the initial value of He; when the activation function is an S-curve function such as sigmoid or tanh, the initial value uses the initial value of Xavier. The initial value of Xavier is derived on the premise that the activation function is a linear function. Because the sigmoid function and the tanh function are symmetrical, and the vicinity of the center can be regarded as a linear function, it is suitable to use the initial value of Xavier. As the layer deepens, the bias of the activation value becomes larger, and the gradient disappears during the training process. When the initial value is the initial value of He, the breadth of the distribution in each layer is the same. The Adam method is used for parameter update. The Adam algorithm combines the ideas of momentum method and AdaGrad algorithm, uses the momentum method to move

according to the physical rules of the ball rolling in the bowl, and combines the AdaGrad algorithm for each element of the parameter. Adjust the learning rate accordingly. The hyperparameters need to be adjusted appropriately based on the verification data. Through the above methods, a basic anti-theft deep learning model can be created.

After the basic model is built, the model can be further optimized by comparing the loss value and accuracy curve of the training set and the validation set. When the model is trained, the training data can be well fitted, but there is a big gap between the accuracy and loss value of the verification data and the training data, that is, the model lacks sufficient generalization ability, indicating that the model has overfitting. The main reason for over-fitting is that the model has a large number of parameters and thus has a strong expressive force, but the training data is small. At this time, the amount of data can be appropriately increased or the complexity of the model can be reduced. At the same time, the weight attenuation can be used, that is, the norm of the weight is added to the loss function to suppress overfitting. When the model is very complex, dropout can be used to suppress over-fitting, that is, a certain proportion of neurons are randomly selected from some hidden layers during the training process and discarded. These discarded neurons will not participate in the signal transmission again to achieve the effect of integrated learning. In order to increase the accuracy rate, a batch normalization algorithm can be used. The idea is to adjust the activation value distribution of each layer so that it has an appropriate breadth, and normalize it in mini-batch units. This method can make learning fast, reduce the dependence on initial valuesduring model training, and suppress overfitting. By adding the above methods, an anti-electricity stealing deep learning model can be obtained that effectively avoids overfitting, can quickly converge, and has a high accuracy rate.

Finally, use the test set to calculate the accuracy of the model, and view the confusion matrix to analyze the proportion of misjudgments.

IV. CONCLUDING REMARKS

This article introduces the application of the deep learning algorithm model in the classification of anti-electric theft, and analyzes the construction process of the anti-electric theft model. As a method that is about to be widely used in the engineering field, deep learning algorithms need to be further applied in the field of anti-electricity theft and in-depth research, to make full use of the advantages of deep learning algorithms in solving practical problems in the field of anti-electricity theft, save the cost of anti-electricity theft, and improve Power system service capacity.

REFERENCES

[1] Songxue Hou. Research and application of Hubei electric power company antielectricity stealing laboratory. (2016) International Society for Information Technology and Engineering. Proceedings of 2016 6th International Conference on Mechatronics, Materials, Biotechnology and Environment(ICMMBE 2016). Informationization International Society for Engineering and Engineering: Computer Science and Electronic Technology International Society

[2] Information Technology - Data Analytics; (2020) Studies in the Area of Data Analytics Reported from Azerbaijan National Academy of Sciences (Privacy-preserving deep learning algorithm for big personal data analysis). Computers, Networks & Communications8

[3]M.S. Suchithra,Maya L. Pai. (2020) Improving the prediction accuracy of soil nutrient classification by optimizing extreme learning machine parameters. Information Processing in Agriculture7(1)

[4]Liu Shuangjun,Yin Yu,Ostadabbas Sarah. (2019) In-Bed Pose Estimation: Deep Learning With Shallow Dataset. IEEE journal of translational engineering in health and medicine7

[5]Bo Zhao, Jiashi Feng, Xiao Wu, Shuicheng Yan. (2017) A Survey on Deep Learning-based Fine-grained Object Classification and Semantic Segmentation. International Journal of Automation and Computing14(02):119-135

[6]Li Dandan, Ge Bingyu, Huang Wenwen, Xie Lei, Qian Shengqi. (2019) Application of DBN Deep Learning Algorithm in Anti-Stealing System. Telecommunications Science35(02):113-117

[7] Zhang Hui. (2018) Research and improvement of optimization algorithms in deep learning. Beijing University of Posts and Telecommunications

[8]Dong Yu,Jinyu Li. (2017) Recent Progresses in Deep Learning Based Acoustic Models.IEEE/CAA Journal of Automatica Sinica4(03):396-409

[9]Biao Chang,Tong Xu,Qi Liu,En-Hong Chen. (2018) Study on Information Diffusion Analysis in Social Networks and Its Applications.International Journal of Automation and Computing15(04):377-401

[10]Fei Wei,Liu Cong,Hu Sheng. (2020) Research on swarm intelligence optimization algorithm.The Journal of China Universities of Posts and Telecommunications27(03):1-20+30