

Application of deep learning for division of petroleum reservoirs

Yaqiong Qin^{1,a}, Zhaohui Ye¹ and Conghui Zhang²

¹ Department of Automation, Tsinghua University, Beijing 100084, China

² China Oilfield Services Limited, Sanhe, Hebei 065201, China

Abstract. Traditional methods of dividing petroleum reservoirs are inefficient, and the accuracy of one-hidden-layer BP neural network is not ideal when applied to dividing reservoirs. This paper proposes to use the deep learning models to solve the reservoir division problem. We apply multiple-hidden-layer BP neural network and convolutional neural network models, and adjust the network structures according to the characteristics of the reservoir problem. The results show that the deep learning models are better than one-hidden-layer BP neural network, and the performance of the convolutional neural network is very close to the artificial work.

1 Introduction

In the field of petroleum well-logging, reservoirs are those that have interconnected pores and are capable of storing fluids such as water and oil [1]. The task of reservoir division is to determine the layered interface and the thickness of different types of reservoirs. In the process of petroleum exploration and development, people use specialized detection instruments and utilize physical measurement methods such as sound, electricity, magnetism and radioactivity to obtain various logging curves. At present, the artificial method of dividing reservoirs is mainly observing the characteristics of the logging curves to qualitatively identify the reservoir type and the layered interface.

The artificial method of dividing reservoirs is time-consuming and laborious, what's more, its stratification results are greatly influenced by the professional experience of the logging analysts. Therefore, it is meaningful to explore the automatic approaches to dividing reservoirs. In the early days, people's idea was to integrate expert experience into computers and build an expert system for well-logging [2]. This method has high requirements for researchers' expertise and is not good for atypical reservoirs. After that, researchers used mathematical methods to study the division of reservoirs. For example, Du W [3] proposed a stratification method based on logging curves' activity. The activity of the curve implies the degree of curve' saltation, which can help identify the interface of different layers, but due to the complex variation of logging curves, the accuracy of this method is not ideal.

With the development of artificial intelligence technology, people began to use neural networks to study the problem of reservoir partitioning. Chen Z and Zhang L [4-5] used BP neural network with one hidden layer to divide reservoirs; Sun K [6] introduced the coordinate-method of solving the optimization problem into BP

neural network, but the network still had only one hidden layer. These attempts did not significantly increase the accuracy of automatic stratification. Zhu K and Li M [7-8] utilized one-hidden-layer BP neural network to deal with the oil-gas-water reservoir classification problem. The classification accuracy is 88% or higher. However, they make judgments for a given reservoir and cannot automatically determine the layered interface. Duan et al. [9] creatively used convolutional neural network [10] to predict reservoir parameters and made good results, but it was not applied to the problem of reservoir division.

At present, deep learning technology is developing rapidly, and the deep network has more powerful expressiveness than the shallow network. However, mainstream deep learning models have not been effectively applied to the problem of reservoir division. Therefore, this paper introduces multiple-hidden-layer BP neural network (MBPNN) and convolutional neural network (CNN) into the reservoir division problem. We use mainstream deep learning optimization methods, and specially optimize the model according to the reservoir division problem. In the end, the stratification effect is better than the one-hidden-layer neural network, and the classification accuracy is over 90% on average, which can meet practical needs.

The rest of the paper is organized as follows. Section 2 introduces the pre-processing methods of logging data, the structure of multiple-hidden-layer BP neural network and convolutional neural network, and the optimization strategy. Section 3 gives the results of the experiment. Section 4 summarizes the content of the article.

2 Deep learning models for reservoir division

2.1 Features and samples

^a Corresponding author: qyq16@mails.tsinghua.edu.cn

2.1.1 Feature selection

There are many kinds of logging curves. From the perspective of machine learning modeling, people can use the Principal Component Analysis method to reduce the dimension of the logging curves and select the logging curve that contributes more to the stratification. However, not every well has a full range of logging curves, and the conventional 9 curves are available in most wells. There are 9 main types of conventional logging curves: natural gamma ray (GR), spontaneous potential (SP), caliper (CAL), deep lateral resistivity (RD), shallow lateral resistivity (RS), and micro lateral resistivity (RMLL), acoustic time difference (DT), neutron (CNCF), density (ZDEN). From the perspective of artificial method, the conventional 9 curves are extremely important for dividing reservoirs. Therefore, for practical considerations, the conventional 9 curves are chosen as input features.

2.1.2 Sample selection

This paper takes the shaly sandstone reservoir wells in the marine area as the research object. For these wells, the effective reservoir is sandstone and the non-effective reservoir is mudstone. The logging data is chosen from four oil wells in a certain sea area. The depth ranges of these four wells are Well1: 950m-3376.2m, Well2: 2778.9m-3883.1m, Well3: 2768.6m-3392.3m, and Well4: 843.8m-3873.8. m. The manual interpretation results show the reservoir type at each depth point of each well. The sand layer includes the oil layer, the water layer, the oil-bearing water layer, and the oil-water layer. The paper regards all sand layers as one class and the mud layer as another.

Each depth point is a sample that corresponds to a label (sand or mud layer). This sample is a 9-dimensional data whose each dimension corresponds to a curve. In order to ensure that each type of reservoir can be effectively trained, the training set is selected according to the ratio of sand layer to mud layer.

2.1.3 Data pre-processing

Logging curves often have data missing at some depth, so the dataset should be filled with missing values first. In this experiment, quadratic polynomial interpolation is performed on the missing data to ensure uniform variation.

The units of the conventional 9 curves are different, and the values of 9 curves differ quite a bit. Therefore, it is necessary to preprocess the logging data by the method of Z-score standardization.

The Z-score standardization method standardizes each dimension of the original dataset to data with a mean of 0 and variance 1. This method does not change the probability distribution of the original dataset. The standardization formula is given by formula (1), where X is a certain dimension of the original data, μ , σ are the mean and standard deviation of X , and X^* is the standardized data. This method eliminates the effects of different dimensions.

$$X^* = \frac{X - \mu}{\sigma} \quad (1)$$

If linear-function-normalization is used to adjust the size of each dimension of data to [0, 1], some extremely large values will have a large impact on that dimension of data. For example, the value of resistivity is less than 10 at most depths, but in some tight rock layers, the value can reach tens of thousands. Linear-function-normalization results in the value of resistivity close to zero at most depths, and close to 1 only at few depths, causing the resistivity have little contribution to reservoirs division. Hence, this normalization method is not suitable. The normalization formula is given by formula (2), where X_{min} and X_{max} are the minimum values and maximum values of X , respectively, and X^* is the normalized data.

$$X^* = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2)$$

2.2 Multiple-hidden-layer BP neural network

2.2.1 Network structure

Prior to this, people used one-hidden-layer neural network to solve the classification problem, because it is difficult to train multiple-hidden-layer neural network. With the development of deep learning, people invent many strategies for training deep neural networks. We will use mainstream methods to implement modeling and training of multiple-hidden-layer neural networks.

Our network model uses the conventional 9 curves as the input of the neural network, so the input layer has 9 neurons. Since our purpose is to classify the points on each depth, the number of neurons in the output layer is 2. The setting of the hidden layer is to be tried by experiment. After trying, the number of hidden layers is determined to 3, and the number of neurons in each hidden layer is 15, 10, and 8, respectively. The structure of the model is shown in Fig. 1.

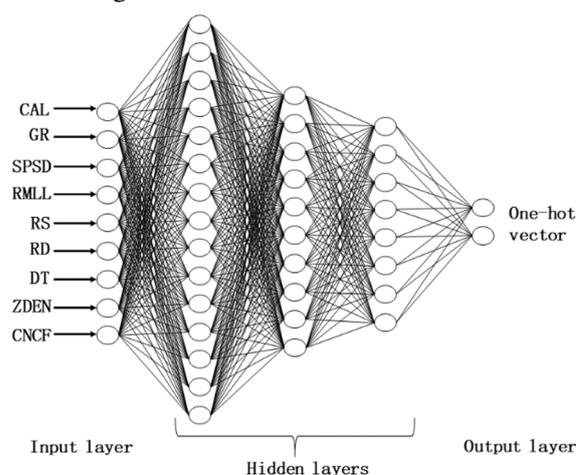


Figure 1. Multiple-hidden-layer neural network's structure

2.2.2 Optimization Strategy

A. Activation function selection

As a nonlinear classification problem, we choose cross entropy as the loss function, softmax as the activation function of the output layer, and ReLU [11] as the

activation function of other layers. ReLU is close to the linear function, which can reduce the calculated amount when calculating gradient. ReLU can help to avoid gradient vanishing and solve the convergence problem of the deep network.

B. Batch Normalization

Batch Normalization [12] is added in front of each hidden layer. Batch Normalization normalizes the input value of the activation function so that the distribution of the input values becomes a distribution of which the mean is 0 and the variance is 1. This allows the input to remain in the sensitive area of the activation function, preventing neurons in the deeper layers becoming inactive during training.

C. Back propagation algorithm

Mini-batch gradient descent method is used to implement back propagation. Each epoch, we extract a batch of samples from training set to calculate the gradient, then update weights. Adam algorithm is applied to set learning rate of the model, which is self-adaptive. Generally, its initial learning rate should be set relatively small.

D. The initialization of the weight

To initialize weights of the network, we try two ways. The first way is using Xavier initializer [13] which can automatically determine the initial size of the weight matrix by the number of input and output neurons (n_{in} and n_{out}). In this experiment, the initial weight is set to satisfy Gaussian distribution whose mean is 0 and variance is $\frac{2}{n_{in}+n_{out}}$.

The second method is using Restricted Boltzmann Machine (RBM) to perform layer-by-layer pre-training [14]. Using the weights obtained by RBM to initialize the BP neural network helps to solve the problem that the BP neural network may easily fall into the local minimum point and the convergence is too slow when randomly initialized.

E. Prevent overfitting

Dropout algorithm [15] is used to prevent overfitting. This algorithm randomly discards some of the cells in the hidden layer during training. After trying, it is beneficial if the network discards 25% of the nodes.

Early stop method is adopted during the training process. It is common that the training error will gradually decrease with time, but the error of the verification set will rise again. To avoid this phenomenon, we stop the training process when the verification set error no longer continues to drop after a certain number of iterations.

2.3 Convolutional neural network

In marine areas, the artificial rules of dividing reservoirs in shaly sandstone wells are as follows. Compared to the mudstone layer, the sandstone layer has lower natural gamma ray, higher spontaneous potential, and higher lateral resistivity (oil layer) or slightly lower resistivity (water layer). For the sandstone layer, three resistivity curves are mutually dispersed, and the neutron and density curves exhibit the characteristic of "envelope".

It can be seen that the artificial method of reservoirs division is similar to "viewing pictures", and the

hierarchical interface is determined by observing the characteristics of the nine conventional curves. CNN is a neural network specially designed to process data with a similar grid structure, and has been widely used in the field of image processing. As an analogy, we can change 9 curves of data into a two-dimensional matrix format, then use the CNN model to complete the prediction.

2.3.1 Data Processing Method

BP neural network treats each depth point as an isolated object when dealing with classification problems, so the information of adjacent points does not help to predict the type of the current point. However, as for the artificial classification, the trend of the curve is a very important basis for judgment. Therefore, when using the CNN model, we consider a depth point and several depth points adjacent to it as a whole, and predict the reservoir type of the depth point through the overall information.

We select a continuous 9 points along the depth of the well as a whole, taking the label of the midpoint as the overall label. The arrangement of the log data for each point is shown in Fig. 2. The reasons for this arrangement are as follows. Three resistivity curves of RD, RS and RMLL arranged together can help to extract the relationship between the three. CNCF and ZDEN arranged together is beneficial to extract the feature of "neutron density envelope".

In this way, nine curves of data are processed into a two-dimensional matrix format. The 9x9 rectangular frame is moved down the depth of the well in steps of 1, and the selected two-dimensional matrix block becomes a sample. For example, the matrix block from the 0th point to the 8th point in the depth direction is taken as a sample, and its label is the label of the 4th point. The matrix block from the 1st point to the 9th point in the depth direction is taken as a sample whose label is the label of the 5th point.

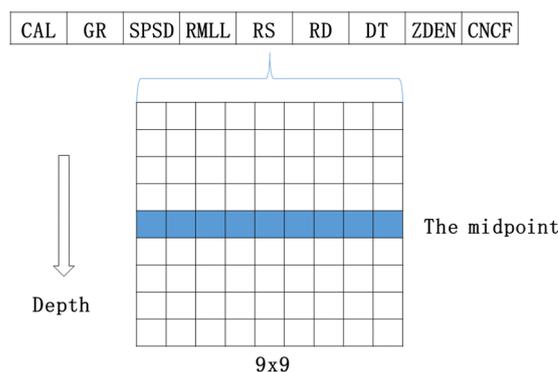


Figure 2. Data Processing Method of CNN

2.3.2 Network structure

Based on the size of the data (each well has several thousand data blocks), it is determined that two convolutional layers are enough. We choose the mainstream size of 3x3 as the convolution kernel size. Compared to another mainstream size of 5x5, 3x3 can reduce the number of parameters and is more suitable for 9x9 input. The boundary processing method of the

convolution layer and the pooling layer are zero-padded to ensure that their output and input dimensions are the same.

The first convolution layer (Conv1) has 8 different convolution kernels, and Conv1 is followed by a 2x2 max pooling layer (Pool1), so that the feature map size of the output of Pool1 is 5x5. The second convolution layer (Conv2) has 16 different convolution kernels, and Conv2 is followed by a 2x2 max pooling layer (Pool2), so that the feature map size of the output of Pool2 is 3x3. Pool2 is followed by two fully connected layers, whose numbers of neurons is 64 and 2, respectively. The concrete structure is shown in Fig. 3.

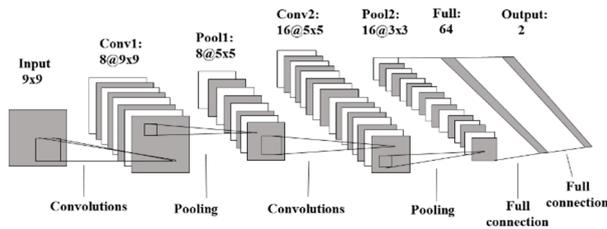


Figure 3. The structure of CNN

2.3.3 Optimization Strategy

CNN has the same settings as MBPNN in terms of activation function selection, batch normalization, backpropagation algorithm, prevention of overfitting, and so on. As for the initialization of the weight, RBM is not recommended, because CNN has the character of weight sharing which can reduce the number of network parameters and make the network easy to train.

A. Special optimization

In the evaluation of results, the classification accuracy does not fully represent the effect of reservoir division. This is because the reservoir is spatially contiguous and the data at the top and bottom interface of a reservoir is more important.

Therefore, we do a special treatment on the sample at the layered interface, adding a penalty to the loss function at that depth point, where we use a very small positive number.

From the law of log interpretation, the layered interface is generally the half-point of the curve that produces a sudden change. MBPNN takes the values of nine curves of each depth point as the research object, and it can not obtain the changing characteristics of the curves; While CNN can extract this feature and increase the prediction accuracy of the layered interface through introducing the penalty term.

3 Experiments and results

This experiment's code is based on the TensorFlow framework and the hardware is the NVIDIA GTX1050. The paper uses the data from four wells to do experiments based on two designed models above.

A. Test of weight initialization methods for MBPNN

For MBPNN, we choose 20% data of well1 as the training set, and the rest of the data as the test set. We train the training data for 100 times by means of Xavier

initializer and RBM pre-training, and each time it iterates about 20,000 epochs. Table 1 is the test results. It is observed that the accuracy of the two methods is almost the same, the difference is that the result of RBM pre-training is more stable. In the subsequent experiments, RBM pre-training method will be used to get the initial weight of MBPNN.

Table 1. Test accuracy of MBPNN using different methods of weight initialization.

	Max	Min	Mean
Xavier	0.9548	0.9471	0.9521
RBM	0.9525	0.9514	0.9519

B. One well training

We use 20% data of well1 as the training set, and the data of well1 to well4 as test sets respectively. Table 2 records the test accuracy of MBPNN and CNN models.

Table 2. Test accuracy of 4 wells for MBPNN and CNN with 20% data of well1 as training data.

	MBPNN	CNN
Well1	0.9515	0.9755
Well2	0.8374	0.8687
Well3	0.8229	0.8905
Well4	0.8690	0.9315

C. Multi-well mixed training

We mix three datasets which are in the same depth segment of 2778.9m-3376.2m (5973 points) from well1 to well3. All data from well1 to well4 is set as test data, respectively. Table 3 records the test accuracy of MBPNN and CNN models.

Table 3. Test accuracy of 4 wells for MBPNN and CNN with mixed data of well1-well3 as training data (the same depth).

	MBPNN	CNN
Well1	0.8251	0.9514
Well2	0.8117	0.9074
Well3	0.8736	0.8903
Well4	0.8041	0.9144

6,000 points are chosen randomly from each well of well1 to well3. All data from well1 to well4 is set as test data, respectively. Table 4 records the test accuracy of MBPNN and CNN models.

Table 4. Test accuracy of 4 wells for MBPNN and CNN with mixed data of well1-well3 as training data (randomly chosen).

	MBPNN	CNN
Well1	0.8824	0.9216
Well2	0.8414	0.8828
Well3	0.8675	0.8729
Well4	0.8753	0.9491

The above experiments show that CNN has large advantages in generalization and the test accuracy of CNN is higher.

D. Boundary error

We choose 20% data of well1 as the training set, and the rest of the data as the test set. We calculate the error between the predicted reservoir boundary and the reservoir boundary interpreted by artificial work. The average boundary error of the MBPNN method is about 0.9 m, and the average boundary error of the CNN method is about 0.4 m. Since the artificial work itself has errors, it is generally considered that the boundary error is acceptable within 0.5 m. Table 5 shows the stratification results of the three sand layers taken from the well1. There are two columns of data below each method which are the top and bottom depths (unit: m) of the sand layer. It can be seen that CNN has less boundary error than MBPNN.

Table 5. Top and bottom depths of the sand layer (unit: m).

Sand Layer	Artificial Results		MBPNN Results		CNN Results	
	top	bottom	top	bottom	top	bottom
1	970.3	976.0	968.9	975.8	969.7	975.9
2	1188.4	1197.6	1187.7	1198.4	1188.0	1197.3
3	2839.4	2918.3	2837.9	2917.8	2838.9	2917.9

We import the prediction results of CNN into the professional logging interpretation software to obtain the result map, as shown in Fig 4. In the stratified results, blue represents the sand layer. It can be seen that the stratification results of CNN are similar to the artificial results and are completely acceptable.

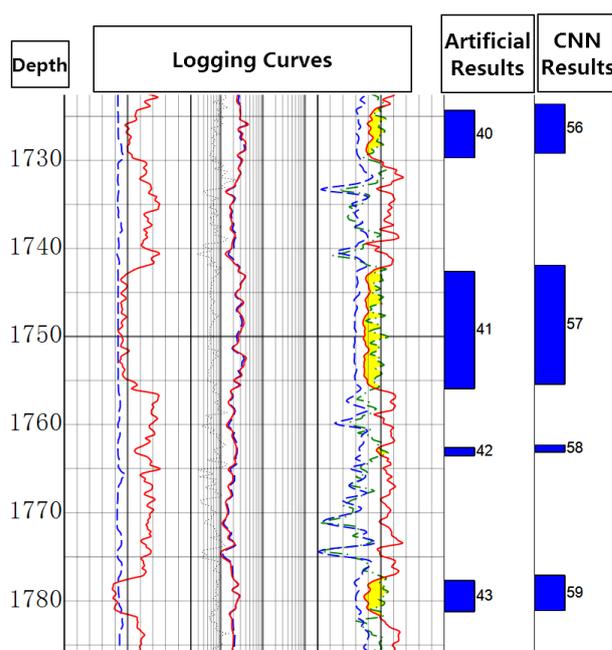


Figure 4. Layered results of CNN

4 CONCLUSIONS

This paper studies two mainstream deep learning models: multiple-hidden-layer BP neural network and convolutional neural network for reservoir division. The paper adopt the mainstream structure and optimization methods of deep learning, such as: ReLU activation function, batch normalization, Adam adaptive learning rate algorithm, dropout algorithm and so on. For the specific problem of logging, special methods are designed, such as processing the logging data into a two-dimensional matrix, adding a penalty to the loss function at the layered interface.

The experimental results show that the stratification effects of MBPNN and CNN both are good. When training data and test data are from the same well, the test accuracy can reach 95% or higher. When test data generalized to more wells, the test accuracy is also more than 82%. Because different wells may have very different data distribution, the generalization effect is related to the type of well. When training data is multi-well mixed, CNN is better than MBPNN. CNN's boundary error is less than MBPNN, and is closer to the artificial stratification result. Therefore, CNN can meet the actual stratification requirements. We think the good results obtained by our models are due to deeper structure and more suitable means to extract features of curves.

In order to further improve the accuracy of stratification, more logging data is needed for training. In addition, when the data of various reservoirs such as oil layer, water layer, oil-water layer, dry layer and so on are sufficient, we can achieve multi-classification, not just separate sand and mud layers. The method of dealing with multi-classification problems is similar.

References

1. Hong Y. Logging Principles and Comprehensive Interpretation. China university of petroleum press, 1993.(in Chinese)
2. Yi G, Lv J, et al. Expert system of petroleum reservoir evaluation. Well Logging Technology, 1986, 10(1):1-8. (in Chinese)
3. Du W, Activity division method of well-logging curves. Coal Geology of China, 1991, 3(3):83-88. (in Chinese)
4. Chen Z. Application of Artificial Neural Network for Oil-layer Recognition. Petroleum Geology & Oilfield Development in Daqing, 1994(3). (in Chinese)
5. Zhang L, Wang L, Dividing fractured carbonate reservoirs with neural networks. Compound oil and gas field, 1995(2):43-45.(in Chinese)
6. Sun K, Zhou Y. Improved neural network algorithm and its application in oil layer recognition. China Petroleum Machinery, 2004, 32(3):28-29.(in Chinese)
7. Zhu K, Song H, Gao J, et al. Application of Evolutionary Neural Networks for Well-logging Recognition in Petroleum Reservoir. Seventh International Conference on Computational Intelligence and Security. IEEE Computer Society, 2011:362-366.
8. Li M, Zhang J. Application of neural network technique for logging fluid identification in low resistance reservoir. Sixth International Conference on Natural Computation. IEEE, 2010:163-166.
9. Duan Y, Li G, Sun Q. Research on convolutional neural network for reservoir parameter prediction. Journal on Communications, 2016(s1):1-9.
10. LéCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, 86(11):2278-2324.
11. Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. International Conference on Artificial Intelligence and Statistics. 2011:315-323.
12. Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 2015:448-456.
13. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. Journal of Machine Learning Research, 2010, 9:249-256.
14. Hinton.G.E et al. A fast learning algorithm for deep belief nets. Neural Computation,vol.18,2006,1527-1554.
15. Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors. Computer Science, 2012, 3(4):págs. 212-223.