Intelligent Automation And Soft Computing, 2020
Vol. 26, no. 3, 609–616
DOI: 10.32604/iasc.2020.013939

Research on the Automatic Extraction Method of Web Data Objects Based on Deep Learning

Hao Peng*, Qiao Li

School of Information Science and Engineering, Hunan International Economics University, Changsha 410205, China
Address: Lu GuYuan, High-tech Industrial Development Zone, Yuelu District, Changsha City, Hunan Province

ABSTRACT
This paper represents a neural network model for the Web page information extraction based on the depth learning technology, and implements the model algorithm using the TensorFlow system. We then complete a detailed experimental analysis of the information extraction effect of Web pages on the same website, then show statistics on the accuracy index of the page information extraction, and optimize some parameters in the model according to the experimental results. On the premise of achieving ideal experimental results, an algorithm for migrating the model to the same pages of other websites for information extraction is proposed, and the experimental results are analyzed. Although the overall effect of the experiment is not as good as that of the page information extraction in different websites, it is far more effective than that of using the model directly on new websites. A new method is proposed to improve the portability of the information extraction system based on machine learning technology. At the same time, the deep nonlinear learning method of the depth learning model can prove deeper features, can have a more essential description of the abstract language, and can better express and understand sentences from the syntactic and semantic levels.

KEY WORDS: **Automatic extraction, deep learning, neural network, Web data.**

1 INTRODUCTION

TWO main methods of information extraction are manual rule extraction and machine learning technology. The manual rule extraction has the advantages of high accuracy and poor portability. It strongly depends on the field of knowledge system of rule writers and manual labour. In recent years, the rapid rise of the crawler technology is like the information extraction, but no matter how developed related technologies, they cannot avoid the step of the formulating information acquisition rules for each field (Huang et al., 2013). In recent years, machine learning technology has been used to overcome the drawbacks of traditional manual rules for information extraction, such as increasing the portability of the system, establishing the knowledge base of the extraction system, and using self-expanding technology to learn from unmarked thesaurus tables (Zhang et al., 2014).

With the increasing popularity of network applications worldwide, the Web has become the largest information carrier in the field of information extraction, and search engines and the crawler technology has become the most effective way to obtain Web information. In the situation that the target information needs to be extracted more accurately, the search engine cannot work simply by querying and counting the keyword search intensively, while the crawler technology has the drawback of traditional manual rules (Landers et al., 2016). This paper uses the excellent training effect of the depth learning model to realize the intelligent Web information extraction method. The purpose is to solve the problem of low system utilization and poor portability in manual rules (Feng et al., 2014). The structure features of the neural network are used to identify the position information between the semantics of the text and the text sequence, to extract the target information (Nga and Yanai, 2014). At the same time, the machine learning technology is used to grasp the fuzzy phenomena and the regularity of the features, and to extract the target information of heterogeneous Web pages in the same domain, so as to explore new

CONTACT Hao Peng ka8259@163.com

possibilities for the transplantation of the information extraction system.

Special contributions of this paper include:

(1) The algorithm flow analysis of the deep learning model.

(2) The web data object extraction process analysis.

(3) The deep learning model is applied to the extraction of web data pairs.

The rest of this paper is organized as follows: Section 2 introduces the algorithm process of the deep learning model. Section 3 introduces the process of web data object extraction. Section 4 cites a case for analysis, and Section 5 is the summary.

2 THE INTRODUCTIONTOTHEDEEP LEARNING MODEL

THE artificial neural network contains only one hidden layer, and its training model parameters become shallow learning. However, shallow learning and training methods needs a lot of experience and skills, and it is not easy to master. In the 1980s and 1990s, researchers used random gradient descent and reverse propagation to attempt the deep web-based learning (Zhi-Jian and Sun, 2013). However, the deep neural network learning is very slow, and in practice did not achieve the desired results, since then the development of the artificial neural network slowed down. Until 2006, Hinton proposed the idea of the layer-by-layer initialization, which could effectively overcome the difficulty of training deep neural networks (Wu and Yu, 2018). The learning method based on the stochastic gradient descent and backward propagation introduces new ideas, and the depth learning method has developed rapidly.

The depth feature obtained from the deep learning model better represents the essence of data. Hinton also put forward the idea of solving the difficulty of training the deep neural network model, and the deep learning has entered a decade of rapid development. Among them, outstanding achievements have been made in the computer vision, speech recognition, bioinformatics, natural language processing and other fields. In different fields and different problems, different depth learning models are different. For example, the convolution neural network has better performance in image recognition, and the recurrent neural network is more suitable to solve the problem of speech recognition (Cheng yong et al., 2014). At present, the main models of depth learning are self-coding, the convolution neural network, the depth confidence network and the recurrent neural network. The following will mainly introduce the basic model of the depth learning used in this paper, the convolution neural network, the recurrent neural network, and the long-term and short-term memory neural network.

2.1 The Convolution Neural Network Model (CNN)

The convolutional neural network (CNN) is proposed to study the receptive field of the cat visual cortex. The local receptive area is used as the input of the hierarchical structure, and each layer obtains the salient features of the data through the convolution nucleus. The weight sharing structure of the convolutional neural network reduces the number of weights, reduces the complexity of the model and avoids over fitting. The local perception and parameter sharing characteristics of the convolutional neural network makes the spatial relationship reduce, improve the training performance of the back-propagation algorithm, and helps to construct the deep complex neural network model. The CNN is the first learning algorithm to train the multi-layer network structure. Figure 1 is the convolutional neural network model. The model has two hidden layers (c1 and c2). Each layer is composed of two-dimensions, (feature map) and is composed of multiple neurons. The S1 and S2 are pooled.

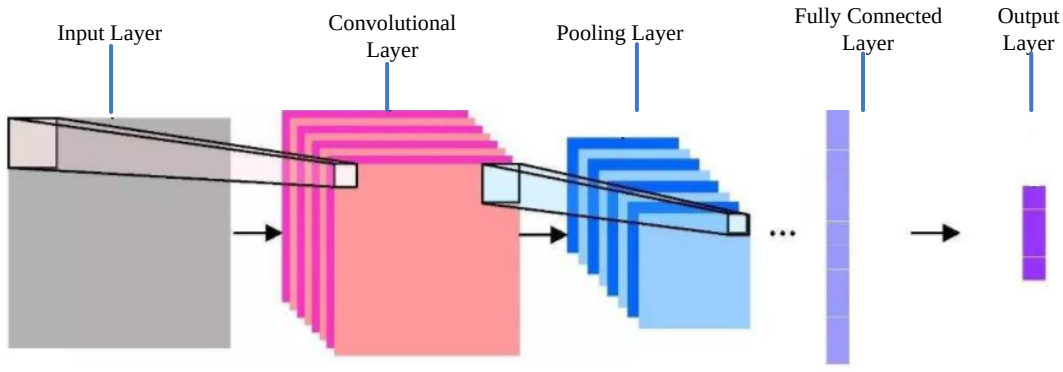

Figure 1. The Convolution Neural Network Model Diagram.

Through the convolution operation of the three convolution kernels, input X is mapped to a feature map containing the same number of convolution kernels, and S1 is obtained after pooling, then S1 is used as the input. The five feature maps of the S2 layer are generated by convolution and pooling of the five convolution nuclei. All neurons of the five maps are connected into a vector and input into the classifier to get output h (Formula 2.5).

$$h_i = \sigma\big(pooling\big(\sigma(wx)\big) + b\big) = \sigma(wH + b) \quad (1)$$

Among them, for the activation function, the commonly used nonlinear activation functions are sigmoid, tan h, etc., and w and b are model parameters.

The pooling layer is an important operation in the convolution neural network model to reduce the parameters between the convolution layers, reduce the complexity of the model, and prevent over-fitting. Pooling can be divided into the general pool, the overlapping pool, the empty Pyramid pool and so on. The most common pooling operation is the average pooling and maximum pooling.

2.2 The Long-Term Memory Neural Network Model (LSTM)

In the traditional neural network, it is assumed that all inputs and outputs are independent. However, for many tasks, it is necessary to input previously calculated information. For example, to predict the next word in a sentence, it is best to know which words appear before it. The Recursive Neural Network (RNN), is shown in Figure 2, which proposes the idea of explicit modeling using temporal information. RNN has shown great success in many NLP tasks, such as the non-segmented continuous handwriting recognition and the autonomous speech recognition. In particular, the mostly used recurrent neural network (RNN) is the long and short memory neural network (LSTM), (Herumurti and Gou, 2013). The LSTM is a special type of RNN, and its essential idea and basic architecture are the same as the RNN, but the LSTM learns long-term dependency information and captures longer-term validity characteristics than the RNN.

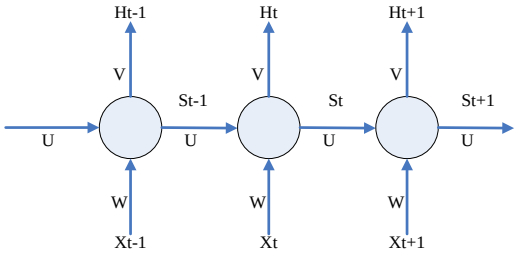

Figure 2. The Recurrent Neural Network Structure Diagram(RNN).

2.3 The Back Propagation and Gradient Descent

The back-propagation algorithm and gradient descent algorithm are the most important algorithms of neural network training algorithm. It is simple and can converge faster and better to the local optimal value. In the back propagation, the gradient vector of the error surface is calculated. This vector points to the steepest descent line from the current point, and if you move a "short" distance along it, it reduces the error (Na et al., 2015). A series of such actions (with the slow down near the bottom) will eventually find a minimum.

The back-propagation algorithm is divided into two steps: The forward propagation and weight updating. The forward propagation is the forward propagation of the input through the model, layer by layer until the output value is obtained, then the loss function is used to compare the output value with the expected output value, and the error value of the output neuron is calculated. Weight updating is to propagate the error value from the output of the model backwards and updates the weights of the neuron parameters in each layer by layer in order to minimize the loss function. If the error value is calculated, the loss function should be defined first. For example, for n samples {(x (1), y (1)] (x (n)}, the loss function is defined as:

$$E(y, y') = \frac{1}{2}|y - y'|^2 \quad (2)$$

The overall mean error values for the n samples can be expressed as:

$$E = \frac{1}{2n}\|y(x) - y'(x)\|^2 \quad (3)$$

The goal is to update the model parameters to minimize cost function E (w, b) so that the model can fit the test samples more accurately. First, all the parameters of the model need to be initialized randomly. Generally, the initial parameters are random values, which tend to zero. Then, all the parameters in the model need to be updated by optimization the algorithms, (such as gradient descent algorithm) to get the optimal solution of the objective function. The update parameters of gradient descent algorithm w and b are as follows in (4) and (5):

$$w_{ij}^{(l)} = w_{ij}^{(l)} - \alpha \frac{\partial}{\partial w_{ij}^{(l)}} J(w, b) \quad (4)$$

$$b_i^{(l)} = b_i^{(l)} - \alpha \frac{\partial}{\partial b_i^{(l)}} J(w, b) \quad (5)$$

Among them, α is the learning rate. The learning rate is the gradient descending step. If α is small, the number of convergence iterations will be higher, and the convergence speed will be slow. α alpha is larger, and may skip the local minimum, resulting in no convergence. A more appropriate learning rate depends on the experiment. The theory of deep

learning is very profound, especially the process of feedback derivation being more complicated. However, it is not very difficult to apply the theory of the in-depth learning to practical applications (He et al., 2015). There are many simple and easy-to-use open source learning frameworks to implement the in-depth learning algorithms, which makes us more focused on the algorithm research. For example, the Tensor Flow, is an open source library for in-depth learning developed by Google, and Theano, a Python-based in-depth learning package, etc.

3 THE WEB INFORMATION EXTRACTION MODEL AND ALGORITHM IMPLEMENTATION BASED ON DEEP LEARNING

3.1 The Overall Structure of the Information Extraction Model

THE purpose of the Web information extraction model described in this paper is to extract the target information automatically from other pages of the same type of website by labeling a small number of target information fields from the same website manually. In order to utilize the similarity between the different pages and improve the applicability and portability of the information extraction, the model uses depth learning to grasp the relevance between the different pages, and then identifies and extracts the required target fields. The overall structure of the model is shown in Figure 3.

Based on the characteristics of the information in the Web pages, this model uses the multi-level neural networks to construct the model. The Word Embedding and Softmax layers are used to represent the word vectors and classify the results of the web pages respectively. The middle two layers use the RNNF (cyclic neural network) to train the features of the inter-page text information. In view of the RNN's powerful ability in the natural language processing, the combination of the Web information extraction technology and the RNN also achieves two additional benefits, one is to omit the word segmentation step in the text processing and the other is to build the feature engineering step in the information extraction (Li et al., 2015). Because the text information in the Web pages

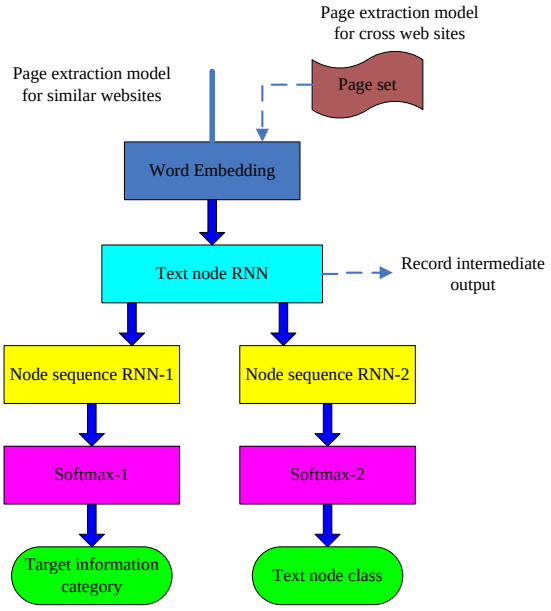

Figure 3. The Schematic Diagram of the Information Extraction Model Structure.

is usually embedded in HTML and the tag pages in small segments, most of the text in the tag pages already contains independent semantics. The model is built directly on the text nodes in the tag pages, instead of the segmenting of information in the whole page. At the same time, one of the advantages of the neural network is to be able to establish the grasp of the fuzzy relations between entities, through this relationship can be established between the descriptions of the page information relations.

For example, in a movie page, the text that follows the "directors" field is usually the director of the movie, and the text that follows the "length" field is usually the length of the movie. A good grasp of this descriptive relationship by the model saves steps of building feature engineering and directly transfers similar functions to the neural network. The overall structure of the model is divided into two parts, namely, the model for extracting the Web pages from the same website and the model for extracting pages from different websites. The two models share the first two layers of the whole network, and the latter two layers are constructed according to their respective functional requirements.

3.2 The Algorithm Implementation of the Information Extraction Model

As shown in Figure 4, the workflow of the algorithm implementation procedure is:

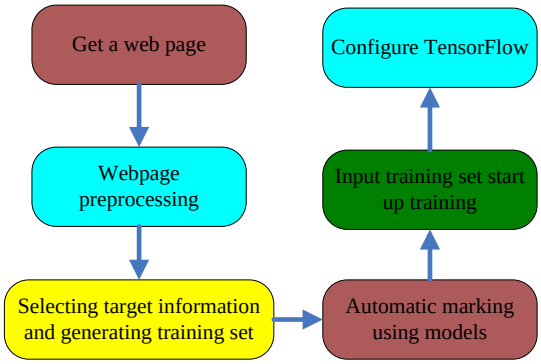

Figure 4. The Model Algorithm to Realize the Program Workflow.

Getting a web page

The information extraction model aims at automatically extracting the target information of the same website with a similar structure by training the neural network and eliminating the trouble of writing the extraction rules for each type of website separately. In this paper, taking the movie portal website as an example, the implementation of the information extraction algorithm program for such websites is described in detail. The key information of the movie page is located in the movie profile area in the center of the page, including movie posters, movie titles, release dates, directors and actors' lists, etc. The key information in different movie pages is embedded in the HTML tags in some similar format, then it uses the tagged sample pages to train the system to obtain the ability to recognize the target information. In order to obtain a better training effect and observe and identify the accuracy of the information, this paper looked a total of 3000 movie pages from three movie websites for the experimental analysis.

Web content pre-processing

Information of the target five keys of the movie is located in the tag of the HTML and the code fragment, which is marked with a wireframe. The information in the HTML tags is too complicated, and most of them are not helpful to the identification of the target information, so the pre-processing of the web page before training can simplify the content of the page information. After the pre-processing, the information structure of the page is very clear, and it is easy to observe the location structure of the target information, such as the release date information, which follows the "premiere" node. The director and actor information follow's the "director" and "actor" nodes respectively. Because the location structure is very important for the system to judge the type of target information, the pre-processing is an important part in the process of adding ordinary web pages to the training set.

Marking target information

The only part of the system that needs to interact with the user, is the step of marking. The target information is very important. Its purpose is to tell the system, which general information is needed to obtain in the web structure, to identify such information in training. The system marks all the target information under the user's setting and adds label attributes to the node and. the HTML tag containing the target information identifies the target information. Correctly marked pages can be used as valid samples to join the training set. In order to ensure the correctness of the training results, the diversity of the sample pages should be ensured as much as possible. The pages in the sample set can basically cover all the special situations on the site such as pages with missing fields, pages with special characters such as English or numbers in some fields, unpublished movie pages, and over-dated movie pages.

Establishment of the TensorFlow Model

At present, there are many programming frameworks in the deep learning industry, such as Caffe, Torch, Theano, TensorFlow and so on. Each of these frameworks has its advantages and disadvantages, and the support for different scenarios and algorithms is different. The detailed differences between the programming frameworks are shown in Table 1.

Table 1. The Contrast of the Learning Programming Frameworks in Depth.

	Torch	TensorFlow	Caffe	Theano	TensorFlow
Language	Lua	Python	C++, Python	Python	Pythpn
Pretrained	Yes++	Inception	Yes++	Yes (Lasagne)	Inception
Multi-GPU:Data parallel	Yes	Yes	Yes	Yes	Yes
Multi-GPU:Model parallel	Yes	Yes (best)	No	Experimental	Yes (best)
Readable source code	Yes (Lua)	No	Yes (C++)	No	No
Good at RNN	Mediocre	Yes (best)	No	Yes	Yes (best)

In view of the good support of the TensorFlow system for the Python language and the RNN, this paper uses the TensorFlow as an example to construct and extract the neural network of the movie page information. Before using the TensorFlow, we must understand the following characteristics:

The TensorFlow uses a graph to represent the computing tasks.

The TensorFlow executes a graph in the context known as a Session (context).

Using Tensor to represent data.

Maintain a state through variables.

Using feed and fetch will assign or retrieve data for any (arbitrary operation).

The process of the TensorFlow system is divided into two steps: The model establishment and the execution calculation. In the process of the model establishment, the parameters of the neural network model and the selection of various model functions are

mainly set up. In this paper, the pseudo code conforming to the TensorFlow format is used to describe the process of the program implementation in the process of building the TensorFlow model.

The start training and generation model

After defining the optimization function, the whole process of constructing the TensorFlow neural network computing model is completed. Next, the training set can be inputted to train the neural network. The algorithm program in this paper takes 20 movie pages marked with movie-name, movie-date, movie-director, movie-actors and movie-length as training sets to input into the neural network. The training is executed in 100 steps. The graph shows that after 100 steps of training, the loss value (loss) has entered a very ideal range.

Prediction using the training model

After training, the model can be used to predict and identify the target information of the input web page. First, 20 movie pages in the training set were used to test the recognition effect. In the recognition process, the tag information was ignored, and the type of target information was judged according to the content and context of the text node. The results show that the recognition accuracy of the model is more than 98% for the pages in the training set. Only one movie director and time information of the other 20 movie pages are wrong, and all the target information of the other pages are successfully identified. The recognition training set page cannot reflect the validity of the model information extraction. From the United States, the Tuan Network non-training set of 100 film pages detect the recognition effect, and after testing, the recognition accuracy from the non-training focus page reaches more than 95%.

4 ANALYSIS OF THE EXPERIMENTAL RESULTS

THE precision rate and recall rate are two metrics widely used in the information retrieval and statistical classification to evaluate the quality of the information retrieval results. The precision rate is the ratio of the number of relevant documents retrieved to the total number of documents retrieved, which measures the precision of the retrieval system. The recall rate is the ratio of the number of relevant documents retrieved and the number of all relevant documents in the document library, which measures the recall rate of the retrieval system.

The definition of accuracy is as follows:

$$\text{Accuracy rate} = \frac{\text{Number of correct information extracted}}{\text{Number of information extracted}} \tag{6}$$

The definition of recall is as follows:

$$\text{Recall rate} = \frac{\text{Number of correct information extracted}}{\text{Number of information in samples}} \tag{7}$$

Neither the precision nor the recall rate can fully evaluate the statistical results, and sometimes contradictory situations occur. Therefore, F-Measure is needed to consider the results comprehensively, that is, the weighted harmonic means of the precision and the recall rate.

$$\mathrm{F} = \frac{(\alpha^2+1)P*R}{\alpha^2(P+R)} \tag{8}$$

When parameter α is 1, this index is called the F1 value (Fl-Score).

4.1 The Effect of the FRNN Structure Model on Results

In the process of the algorithm implementation and other experimental result statistics, this paper uses the LSTMCell. In fact, the performance of the different RNN models in training are very different. In order to test the effect of the different RNN types on the experimental results, we take three main cell models in the TensorFlow system to build the RNN for testing. The training sample set of the model is still 20 tagged movie pages from the Popular Review Network, and the test set is 1000 movie pages from the Popular Review Network. The experimental results are shown in Table 2 and Figure 5.

Among them are:

The LSTM corresponds to the LSTM model of the RNN, and the implementation class in the TensorFlow is tf.nnLSTMCell.

The GRU corresponds to the GRU model of the RNN and implements the class,tf.nn.GRU Cell.

Basic corresponds to the LSTM model that implements only the basic functions and implements the class tf.nn.BasicLSTMCell.

As seen from Table 2 and Figure 5, the LSTM model is the best training model for the information extraction of the film pages described in this paper. After the number of training sets reaches 20, the overall accuracy of the information extraction reaches more than 95%, while the other two models are less effective than the LSTM model.

In this paper, the model migration of the cross-site information extraction of the film pages is to do an experimental analysis. The experiment used the popular Review Network in the training of the model migrated to the United States and the Guevara's film pages. The pages of the migration site are 5, 10, 20, 40 and 80 groups, respectively, to observe the results, the experimental results are shown in Figure 6.

Table 2. The Statistical Results of the Different RNN Structure Models.

LSTM	f1	0.980231	0.915349	0.995923	0.961596	0.934188	0.962438
	p	1.001002	1.000103	1.001002	0.916771	0.868254	0.968795
	r	0.952859	0.939682	0.989782	1.010023	1.010023	0.956166
GRU	f1	0.799346	0.917361	0.638771	0.724319	0.991937	0.773453
	p	1.001002	1.000103	0.759463	0.671456	0.992929	0.799188
	r	0.661596	0.839682	0.719358	0.788194	1.100201	0.739781
Basic	f1	0.956385	0.881577	0.642259	0.137137	0.694655	0.699119
	p	0.979499	0.965656	0.534921	0.577778	0.529342	0.812438
	r	0.933441	0.900201	0.794925	0.083928	1.012031	0.697814

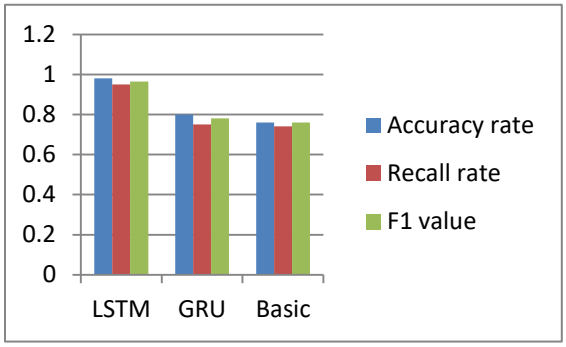

Figure 5. The Effects of Different RNN Structures Results.

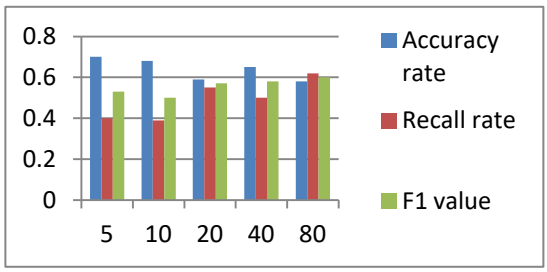

Figure 6. The Extraction Effect of the Comment Network Model Migration to the US Group Web Page.

Figure 6 is a statistical diagram of the experimental results of the migrating the model of the page extraction from popular reviews to the Metro. The graph shows that the accuracy of the page extraction has been maintained at an ideal level throughout the experiment, partly because the accuracy of the horizontal clustering of the page set nodes used in the experiment is higher. The recall rate is not ideal at the beginning, and gradually increases with the increase of the number of page sets until the accuracy is flat. When the number of page sets reaches 80, the overall migration recognition accuracy rate reaches 60%.

4.2 The Influence of the Optimization Algorithm Results

In the depth learning programming system, there are many kinds of optimization algorithms to optimize the learning rate, so that the network with the fastest number of training to achieve the best, is to prevent over-fitting. Different optimization algorithms will produce different optimization results, because of the different scenarios. In this paper, several main optimization algorithms supported by the Tensor Flow are tested and compared. In the experiment, Adam, Adagrad, Adadelta and RMSProp are used to optimize the output of the neural network. The sample set and test set are still taken from the movie page of the Popular Review Network, and the other parameters are selected the same as the algorithm implementation program described in the previous chapter.

When the input training set starts training, the loss value calculated by the model is at 500, and gradually decreases with the increase of the training steps. In the four experiments, the optimization rate of each optimization algorithm for the loss value is shown in Figure 7.

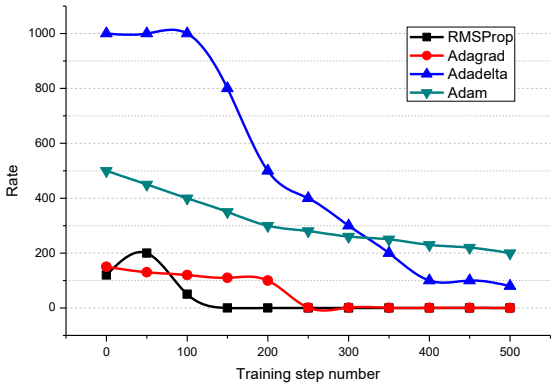

Figure 7. The Training Speed of the Different Optimization Methods

Figure 7 shows that the optimization effect of the model is the best when using the RMSProp algorithm and it keeps the fastest optimization rate from the beginning of the training. When the training reaches 400 steps, it reaches the lowest value, and the optimization effect is far greater than other optimization algorithms. However, the optimization effect of the Adadelta algorithm is very poor, even when the training steps reach 500 steps. The loss value is still not ideal, so it is not suitable for the algorithm model in this paper. The training effect of the Adagrad algorithm is slightly better than the Adadelta algorithm, but it is also not suitable for the algorithm model in this paper. After several steps of training, the Adam algorithm achieved a certain optimization effect,

but the effect is still far inferior to the RMSProp algorithm.

In view of the excellent effect of the RMSProp algorithm in the experiment, this paper uses the RMSProp function as the optimization algorithm in the algorithm implementation and other experimental tests. When the optimization function is selected, the optimal learning rate is also specified. In the gradient descent optimization algorithm, the learning rate plays an important role in the training effect. Therefore, in this paper, the RMSProp algorithm is tested with different learning rates in the experiment.

5 CONCLUSION

THIS paper establishes a neural network structure model based on depth learning for the Web information extraction. The model extracted more than 95% of the information from other pages of the same Web site after training with a smaller number of training sets for a more fixed structure of Web pages. At the same time, after partial modification of the model, the page information extraction between the different websites in the same field also achieved a certain precision. In the process of the algorithm implementation, the influence of several main parameters in the model on the accuracy of the overall information extraction is analyzed in detail, and the algorithm and model are optimized through experimental results. In this paper, according to the characteristics of the text categorization and information extraction, the corresponding depth neural network models are proposed, and the models are integrated into the question answering system, and good results were achieved. The algorithm model of this paper can also be combined with the crawler technology to develop an intelligent crawler system that automatically crawls similar web pages in vertical domain.

6 REFERENCES

Chengyong, W. U., Chen, S., Chongyi, E., Zang, P., Chen, K., & Guijuan, Q. I. (2014). Automatic extraction of arid region forest belt based on cbers-02b data. Journal of Arid Land Resources & Environment, 28(4), 123-128.

Feng, Y., Jia, D., & Wang, H. (2014). Pfime: parallel automatic deep web data extraction based on hadoop. Journal of Computational Information Systems, 10(9), 3863-3870.

He, L., Shen, G., Li, F., & Huang, S. (2015). Automatic extraction of reference gene from literature in plants based on texting mining. Int J Data Min Bioinform, 12(4), 400-416.

Herumurti, D., & Gou, K. (2013). Automatic urban road extraction on dsm data based on fuzzy art, region growing, morphological operations and radon transform. Proc Spie, 8892(24), 7080-7084.

Huang, W. G., Zhu, M., Yin, W. K., & Automation, D. O. (2013). Web information automatic extraction based on dome tree and visual feature. Computer Engineering, 39(10), 309-312.

Landers, R. N., Brusso, R. C., Cavanaugh, K. J., & Collmus, A. B. (2016). A primer on theory-driven web scraping: automatic extraction of big data from the internet for use in psychological research. Psychol Methods, 21(4), 475-492.

Li, B., Ling, Z. C., Zhang, J., & Wu, Z. C. (2015). Automatic detection and boundary extraction of lunar craters based on lola dem data. Earth Moon & Planets, 115(1-4), 59-69.

Na, I. S., Le, H., Kim, S. H., Lee, G. S., & Yang, H. J. (2015). Extraction of salient objects based on image clustering and saliency. Pattern Analysis & Applications, 18(3), 667-675.

Nga, D. H., & Yanai, K. (2014). Automatic extraction of relevant video shots of specific actions exploiting web data. Computer Vision & Image Understanding, 118(1), 2-15.

Wu, K., & Yu, Y. (2018). Automatic object extraction from images using deep neural networks and the level-set method. Iet Image Processing, 12(7), 1131-1141.

Zhang, J., Duan, M., Yan, Q., & Lin, X. (2014). Automatic vehicle extraction from airborne lidar data using an object-based point cloud analysis method. Remote Sensing, 6(9), 8405-8423.

Zhi-Jian, X. U., & Sun, L. (2013). Web content automatic extraction based on data enrichment region. Computer Engineering, 39(9), 192-195.

7 DISCLOSURE STATEMENT

No potential conflict of interest was reported by the authors.

8 NOTES ON CONTRIBUTORS

Hao Peng received Master of Engineering andassociate professor. Graduated from Central South University in 2008. Research interests include web data mining and deep learning.

Qiao Li. Received Master of Engineering and associate professor. Graduated from the Central South University in 2010. Research interest includedata mining and algorithm, and internet of things.