A Novel Model for Weibo Reposts Prediction by Using Generic Based Segmented BPNN

Guixun Luo, Yun Liu, Zhiyuan Zhang

Key Laboratory of Communication and Information Systems,
Beijing Municipal Commission of Education Beijing Jiaotong University, China
{12111007, liuyun, zhangzhiyuan}@bjtu.edu.cn

Abstract

We proposed a novel model for predict the Weibo reposts based on segmented BPNN and genetic algorithm. First of all, we studied the characteristics of Weibo reposts, and then proposed a segmented way for building BPNN by adding momentum to analyze the reposts data. Each segmented data sets will be considered as training data (inputs and outputs) to train the neural network. Secondly, genetic algorithm is utilized to generate a best solution to build the neural network. Accordingly, we include segmentation point, hidden layers, inputs, learn rate and momentum into a chromosome to find the best genes. Finally, we conduct extensive performance evaluations on the datasets of Weibo. As a result, the proposed model can improve the prediction accuracy of Weibo reposts.

Keywords: Weibo reposts, BPNN, Genetic algorithm

1 Introduction

The past decades have witnessed the circumstances that social network services have become the most significant medium for information sharing and dissemination. In online social networks, e.g. Sina Weibo, information diffusion occurs when a user reads and shares a post of another user, then the shared post is read and shared by other users, and so on [1-3]. Tens of millions of Weibo posts are created and reposted each day in China. If a Weibo post is of great quality and interesting, it will be shared and posted by some other users. Predicting the future reposted number of a target post is the base of the research on the effects of social networks. The behavior of information exchange of online social network happens in a short time; users, who has participated in the process of forwarding a post are not independent of each other; factors that affect forwarding a post are complicated. Therefore, the research of the rules of social network information diffusion is very important. In this paper, we focus on the topic of the prediction of online social network information diffusion. According to the law of forwarding a target post in the previous period of time (with the same temporal scale), we predict the forwarding number of the same target post at a certain time point (a future time point). Theoretically, the concern of this paper is a time series prediction problem. Because the post forwarding or information diffusion is time-dependent. The forwarding number of different blog posts vary on very different temporal scales.

In order to predict the forward number of a blog post in the next time point, we use the back propagation neural network (BPNN) [4-6] to achieve this goal. The BPNN is fitted by training the network with known input/output data sets. The training paradigm finds a set of weight values that minimizes the error across the set of facts. If the network is validated, it can then be used to predict outputs based on new input values [7-8]. About the topic discussed in this paper, the repost numbers of a target blog post of previous time with a same temporal scale will be used as inputs and the forward number of the same post of the next time point (with a same temporal scale) will be used as output in the process of training the network. However, a large number of reposts of Weibo occurred during the initial period of time. Subsequently, the number of Weibo reposts will be significantly reduced. So the reposts data should be analyzed in a segmented way and the parameters of BPNN should trained in each segmented section based on reposts number and the change rules of the reposts data. In order to solve the mentioned problems, we have proposed a segmented way to analyze the reposts data and an adaptive dynamic method to learn the parameters in each segmented BPNN in this paper.

It is worth noting that the back-propagation neural network is a layered network consisting of an input layer, an output layer, and at least one hidden layer of nonlinear processing elements. There are three important elements that we should focus on when building the training network: (1) How many inputs that should be passed into the input layer (the number

*Corresponding Author: Yun Liu; E-mail: liuyun@bjtu.edu.cn
DOI: 10.3966/160792642018091905002

of inputs). (2) How many inputs that should be passed into each hidden layer. (the number of inputs of each hidden layer). (3) How many hidden layers in the network. (the number of hidden layers). In this paper, we use the genetic algorithm [9] to search a best elements combo (best solution) to build the training network. Genetic algorithms are commonly used to generate high-quality solutions to optimization and search problems by relying on bio-inspired operators such as mutation, crossover and selection. If we want to build a suitable neural network, we should find a best solution generated by genetic algorithm. In genetic algorithm which will be used in this paper, a population of candidate solutions (number of inputs, number of inputs of each hidden layer, number of hidden layer) to an optimization problem is evolved toward a best solution (the best way to build a network) when a satisfactory fitness level has been reached for the population [10-12]. In general, the final purpose of this research is to build a proper backpropagation neural network learning with time varying inputs. we have done the following work in this paper:

(1) We have proposed a segmented way to analyze the reposts data

(2) Each segmented data sets will be considered as training data (inputs and outputs) to train the neural network respectively.

(3) Genetic algorithm is utilized to generate a best solution to build each segmented neural network.

The whole framework of this paper is shown in Figure 1.

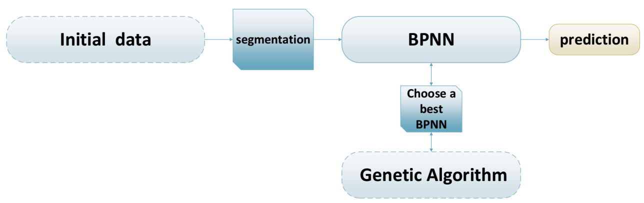


Figure 1. Framework of this paper

The rest of the paper is organized as follows: Section 2 states the proposed model, we clearly explain why and how does the proposed model works. Section 3 describes the evaluation procedure, and provides encouraging results. Finally, Section 4 gives conclusions and outlook for further research in this area.

2 Methodology

In this section, we introduce the method of finding the segmented BPNN of time variation, Back Propagation Neural Network and Genetic Algorithm, inspired by which we construct our model. In the final part, we will discuss the proposed model and how does the model works.

We assume that $P = \{p_1, p_2, ..., p_n\}$ stands for a set of n blog posts. For a particular blog post p_k, we set $T = \{t_1, t_2, ..., t_n\}$ as n time points on the time line, and all temporal scales are the same, i.e. t_2-t_1 = t_3-t_2 = ... = t_n-t_{n-1}. $F = \{f_1, f_2, ... f_n\}$ stands for the forward number of this blog post p_k at each time point, e.g. f_n means the target blog post p_k has been reposted f_n times during the temporal scale t_n-t_{n-1}.

2.1 Back Propagation Neural Network

A basic neural network consists of three parts: input layer, hidden layer(s), output layer. Input layer receive inputs, output layer produce outputs, hidden layers provide the interconnections between inputs and outputs. We use the forward numbers $\{f_1, f_2, ... f_{n-1}\}$ as inputs from time point 1 to n-1, f_n is used as output in the next time point n. The neural network is fitted by training the network through known inputs and outputs data. During the training we can get a set of weight values that minimizes the error across the set of known data. Finally, the network will learn the interconnections between inputs and outputs. When the network validated, the network can then be used to predict a blog post's forward number (output) based on the same blog post's previous forward number sets (new inputs).

Before we process, denote the inputs, weights and outputs of BPNN as:

$$x(t) = (f_1, f_2, ..., f_{n-1}) \in R, \quad \textbf{(1)}$$

$$y(t) = f_n \in R, \quad \textbf{(2)}$$

$$w(t)_j = (w_{1,j}, w_{2,j}, ..., w_{n-1,j}) \in R, \quad \textbf{(3)}$$

Where $x(t)$ is the input vector, we use a mass of Weibo reposts data as inputs according to a fixed time interval. $w(t)_j$ is the weight vector for j-th input of a hidden layer and $y(t)$ is the output, w_{ij} is weight value between input i of input layer and input j of hidden layer, n is the time point. In order to learn the network, we need learn the interconnections between input layers and output layers. In the process of learning, all the training data will be delivered from input layer to hidden layers and then to output layer, all data will be processed layer by layer by using error feedback mechanism, i.e. backpropagation. In the process of backpropagation, the error value between the learned output value and real sample value should be computed at first. Then the error value will be retrained to the network to make the error value converge, for more details please see the previous work talked about BPNN [13-15].

The back propagation neural networks are multi-layered. The strategy of constructing the network has a great influence on the training results [16].

The j-th input of a hidden layer is computed through the following formula:

$$in_j = \sum_i w_{ji} f_i \,, \tag{4}$$

Then the j-th output of this hidden layer is defined as:

$$o_j = g(in_j) \,, \tag{5}$$

Where g(.) is the sigmoid function:

$$g(x) = \frac{1}{1+e^{-(x+\theta)}} \,, \tag{6}$$

θ is the bias value to balance the S function. In this paper, in order to catch the relations between the input pattern and output value of hidden layer, we define θ as:

$$\theta = \varphi . \frac{\sum_i f_i}{n-1} \,, \tag{7}$$

where φ is the hyper-parameter to make the bias in a proper range. In a similar way, the final output o can be computed using the formula (X). The average error of the network can be given by:

$$E = \frac{1}{2}(g(f_n) - o)^2 \,, \tag{8}$$

Stochastic gradient descent (SGD) is utilized in the learning process to minimize the E. For more details please the references in [23]. Each weight of each layer will be updated as:

$$\Delta w_{kj} = \eta (t_k - o_k) o_k (1 - o_k) o_j \,, \tag{9}$$

$$\Delta w_{ji} = \eta \delta_j o_i \,. \tag{10}$$

2.2 Momentum Factor

The substance of building a neural network is to solve the problem about nonlinear optimization. Therefore, local convergence is the weakness when SGD is used to find a global optimal solution. There are many local minima and flat regions when computing mean square error of quadratic functions.

In order to suppress the occurrence of concussion, we define a momentum factor σ during the process of SGD to learn each weight:

$$\Delta w(n+1) = -\eta \frac{\partial E}{\partial w} + \sigma \Delta w(n) \,, \tag{11}$$

Where $0 < \sigma < 1$. When the learning process enters the flat area of the error, the weight change of the momentum term can be helpful to escape the flat area and effectively control the situation that the final result falling into local minimum value.

2.3 Genetic Algorithm

In order to build a good neural network about this paper's problem, we should answer the following questions:How many inputs we have in the input layer? How many inputs in each hidden layer? How many hidden layers in the network? We use genetic algorithm [17-20] to answer these mentioned questions by simulating natural evolution. The search space of the problem is represented as a collection of solutions. More specifically, we use the combination of number of inputs, number of inputs of each hidden layer; number of hidden layers as an individual solution. The purpose of using genetic algorithm in this paper is to find a best solution from the search space with the best "genetic material", which can answer the mentioned questions and build a perfect network. For more details, please see the previous preferences [21-22].

Initially, we have m solutions in the search space and all of the solutions will be represented by character strings. The quality of a solution is measured with fitness function, which is:

$$ft = 1 - \frac{f_n^i}{\sum_{i=1}^{m} f_n^i} \,. \tag{12}$$

The greater the fitness value, the higher the probability of being selected for inheritance. After several crossovers, inheritances and mutations, the average quality of the population will be increased, and then we can achieve a best solution to build the network.

In the scenario of the prediction of Weibo reposts, we should put the time interval, number of hidden layers, number of inputs, learning rate and momentum factor into the gene to evolve a best gene to build the network work. In addition, population size, fitness function and evolution rules are all the important part to enhance the ability of seeking a chromosome.

2.4 The Proposed Hybrid Model

The prediction algorithms of the proposed model can be described as the following steps, see Figure 2:

(1) From the initial training data sets, we segmented the initial data according to the reposts number and the change rules of the reposts data. Then each segmented data part should be treated as the initial training data separately in the next step.

(2) Each kind of pattern data is used to train the parameters in the model of BPNN. Then we can obtain a nonlinear model to predict the forwarding number of a post in the next time point.

(3) We use the genetic algorithm described in section 2.3 to to find a best plan (parameter sets) to build the neural network (Repeat step(2) until minimum loss is reached).

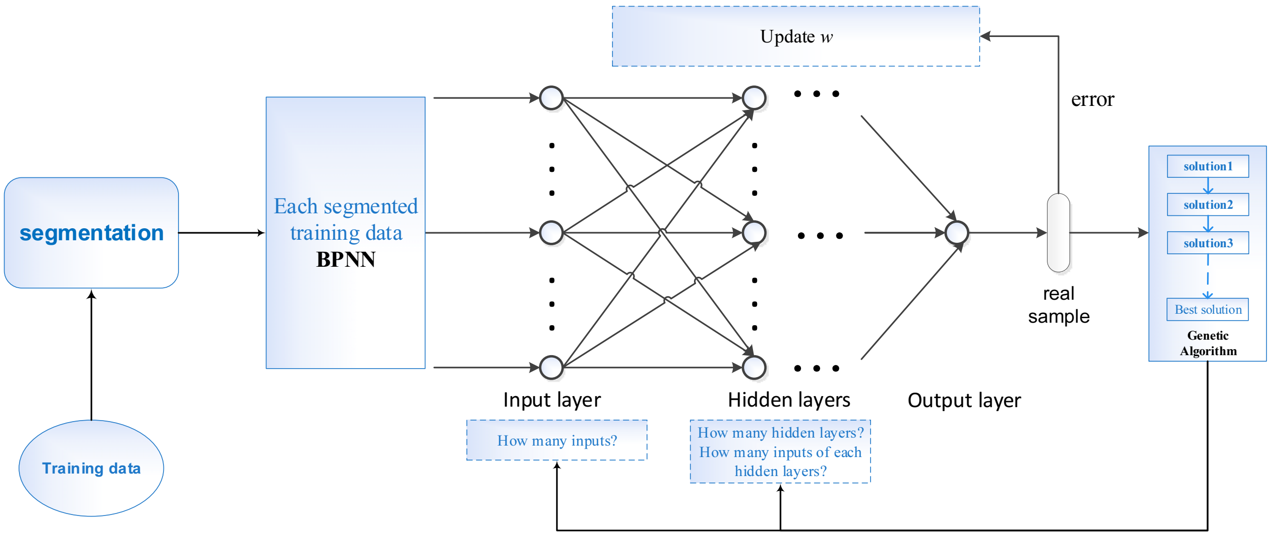


Figure 2. The proposed hybrid model

3 Evaluation

3.1 Experimental Setup

We have crawled the training data from the famous social website Sina Weibo. 70% of the initial data are random selected as the training data to predict the remainings. Population size is set as 80 to seek the best chromosome. Each chromosome contain 9 genes, No. 1 gene on behalf of the segmentation point, No. 2 to 5 genes stand for hidden layers, inputs, learn rate and momentum in the first segmented part of training data, No. 6 to 9 genes stand for hidden layers, inputs, learn rate and momentum in the second segmented part of training data. 100 generations of genetic evolution will be conducted in this paper.

We use the stander evaluation formula Mean Square Error (MSE) to evaluate the prediction accuracy. MSE is defined as follows:

$$MSE = \frac{\sum_{\Omega_\eta}(r_{real} - \hat{\hat{r}}_{predict})^2}{|\eta|}. \quad \textbf{(13)}$$

3.2 Evaluation of Segmentation for BPNN without Genetic Algorithm

3.2.1 Real Data Analysis of Weibo Reposts

Figure 3 shows the total number of the reposts of Weibo post "*Chinese actor Yumin Chou has been married with his wife*" during 14 days. We can see that most of the behaviors of reposts happened in the period from 0 to 10^5 s. The number of reposts in this period (0 to 10^5 s) accounted for 96% of the total reposts during 14 days. In order to further accurately study the forwarding behavior, we collect the reposts number during the time period from 0s to 10^5s, see Figure 4. The red dashed line stands for the average value of the reposts the mentioned Weibo, The green dashed line stands for the median value and the blue dashed line is the median value but the duplicate reposts are not counted. More specifically, we can see that the spacing between the mentioned three lines is very large, which means the dispersion degree of reposts data is high and the stability of reposts data is poor. Although BPNN has the ability to deal with nonlinear mapping problems, the data with high dispersion degree and poor stationarity will bring the difficulties of parameter training and lead to the reduction of prediction accuracy.

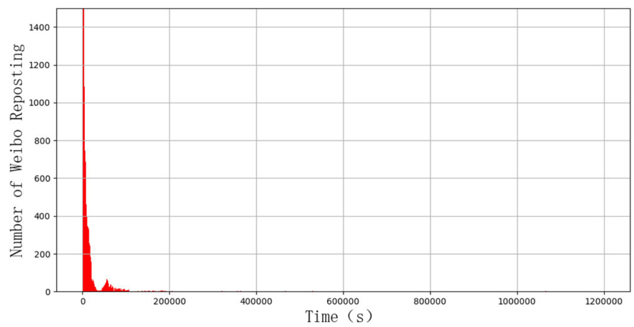


Figure 3. Statistics of Weibo reposts (during 14 days)

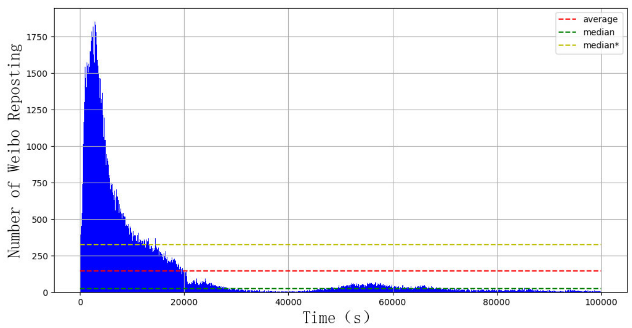
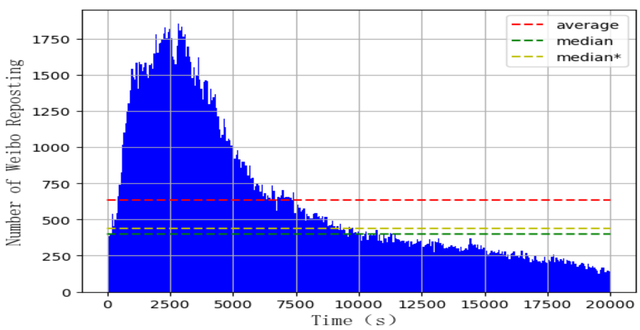


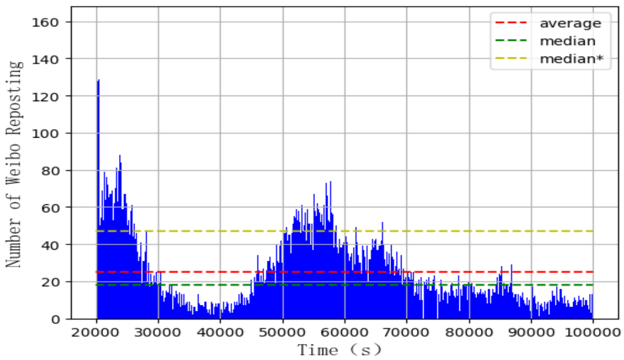
Figure 4. Statistics of Weibo reposts (from 0s to 105s)

Empirically, we can set up the segmentation points according to the time point of intersections between average repost value line and the reposts line. Let's look back Figure 4, the average repost value line and the reposts line cross at the time point 2×10^4s. Time period from 0-2×10^4s is the first phase of data, time period from 2×10^4s-10^5s is the second phase of data. We can get the conclusion from Figure 4 that the number of reposts and the range ability of reposts are both large in phase one, but in phase two, number of reposts and the range ability of reposts are both small. It means that there is a distinct difference between the two phase of reposts behavior.

Then we analyze the data of two phase separately. In Figure 5, we can see that the spacing between the mentioned three lines (average value line, the median value line and the median value line but the duplicate reposts are not counted) is very small in each phase, which means the dispersion degree of reposts data is low and the stability of reposts data is strong. It is suitable to use the same neural network model to train.



(a) phase one:0-2×10^4s



(b) phase two: 2×10^4s-10^5s

Figure 5. Statistics of Weibo reposts (two phases)

3.2.2 Parametric Analysis of Segmented BPNN

Based on the above data analysis, we segment the Weibo data "*Chinese actor Yumin Chou has been married with his wife*" into two phases. Each phase will be trained to predict reposts independently and then merger the prediction results of two segmented phases.

Data normalization. In order to increase computational efficiency, we use the reposts number during 60 seconds as the base unit. The characteristics of active function of BPNN require normalized data. Suppose $X(t)$, t = 1, 2, 3L as the number of reposts in each time point. $X_{\max}(t)$ is the maximal value among $X(t)$, $X_{\max}(t)$ is the minimum value among $X(t)$, then the data normalization can be computed as

$$X(t)^* = \frac{X(t) - X_{\min}(t)}{X_{\max}(t) - X_{\min}(t)}, \quad t = 1,2,3\cdots. \tag{14}$$

Hidden layer of each BPNN. There is no unified theoretical guidance for the selection of hidden layers of BPNN. Normally, BPNN with one hidden layer can approximates any function, BPNN with two hidden layers approximates any function with arbitrary precision. The increase of hidden layers will lead to a significant increase in training time. Therefore, we choose one hidden layer to build BPNN.

We use the following empirical function to choose the neuron numbers of hidden layer.

$$N_l = \frac{N_{train}}{\alpha \times (N_i + N_o)}, \tag{15}$$

where N_i stands for inputs, N_o stands for the outputs, N_{train} is the number of training data, α is the scale factor with the value between 2-8.

Inputs and outputs of each BPNN. In this section, the purpose of using BPNN is to predict the Weibo reposts. Hierarchical relationships between users, follow relations between users, the characteristics of "small world" of Weibo network will lead Weibo information exploded in short time. The input layer should contain a lot and proper inputs to capture the hidden relations behind the reasons of reposts. The reposts data should also be entered into the neural network in chronological order. In this paper, we will handle the experiment by gradually increasing the number of inputs, then the number of the final inputs are determined by analyzing the training error.

The output node is one, which stands for the reposts number of the next 60s.

3.2.3 Evaluation of Inputs and Hidden Layer

Firstly, we set a random number of input layers, and then we analyze the influence of different numbers of hidden layer nodes on the prediction accuracy of the model to determine the number of suitable hidden layer neurons. Then we change the number of input layer to find the number of suitable hidden layer nodes. The rest can be done in the same manner until we find the best combo of inputs and hidden nodes.

Figure 6 states the influences of hidden layer nodes under the situation of a certain number of input layers. It can be seen from the figure that there is no significant positive correlation or negative correlation between the number of hidden layer nodes and the final prediction accuracy. By analyzing the results, We choose 6 as the best hidden layer nodes to conduct the next experiment.

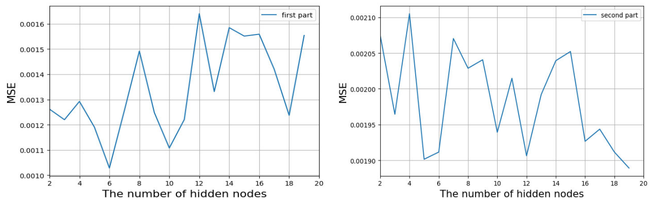


Figure 6. The influence of hidden layer nodes for accuracy of prediction

We can get the conclusion from Figure 7 that there is no significant positive correlation or negative correlation between the number of inputs and the final prediction accuracy. For the first phase, the best inputs number is 2, and for the second phase, the best inputs is 7. From the above two experimental results, the optimal inputs of different phase are different, which indicates that it is necessary to segment the initial BPNN to enhance the accuracy of prediction.

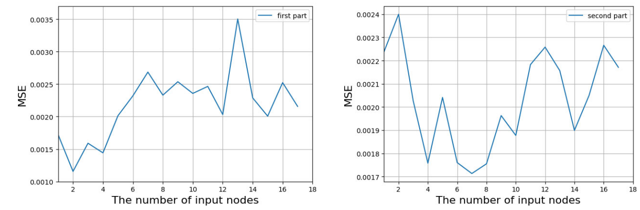
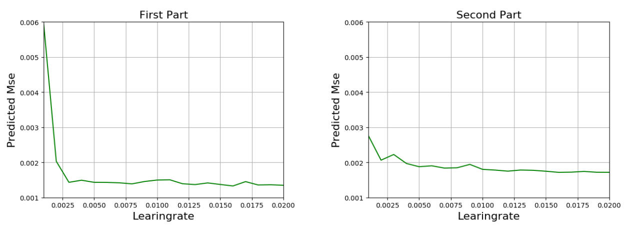


Figure 7. The influence of inputs for accuracy of prediction

3.2.4 Evaluation of Learning Rate and Momentum

For each segmented BPNN, we set the learning rate gradually increased from 0.001 to 0.02. The average results of 30 times experiments are shown in Figure 8.

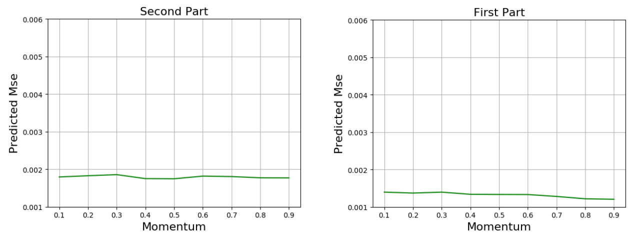


(a) (b)

Figure 8. evalution of learn rate

We can see that for the different model, the effect of learning rate is not the same. In the first segmented BPNN, each learning rate value has little effection for the final model prediction accuracy (MSE = 0.0015) expect for learning rate = 0.001. In the second segmented BPNN, all the learning rate values are result in the final prediction accuracy, that is MSE = 0.0018. We can get the conculsion that learn rate is mainly influence the training process of BPNN but has a little effect on the final model accuracy for the scenario of the prediction of Weibo reposts.

The evaluation result for the proposed momentum factor is shown in Figure 9. Figure 9 and Figure8 has a similar result: in each segmented BPNN, momentum also has a little effect on the final model accuracy for the scenario of the prediction of Weibo reposts. Adding momentum into the model is to inhibit shocks of learning process. As a single factor, although it has little effect on the model prediction, it still has the function to find a best chromosome as one of genes.



(a) (b)

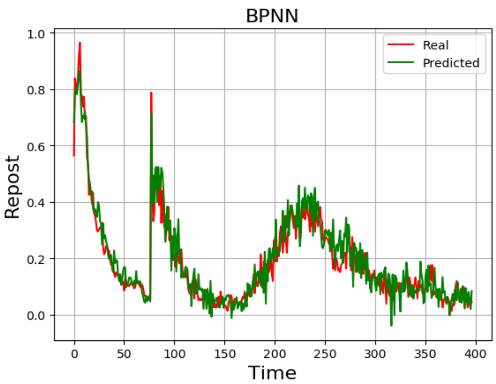
Figure 9. evalution of momentum

So far, we can achieve the best paprameters for training the segmented two phases. All the best values are summaried in Table 1.

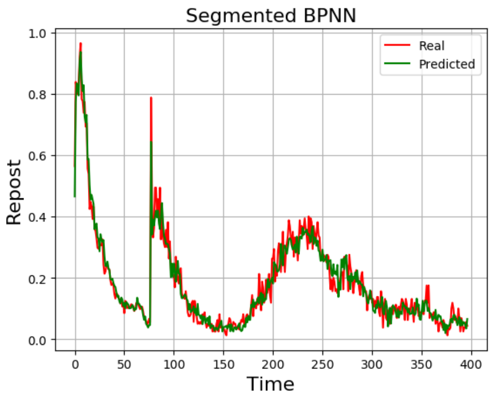
Table 1. Statistics of best paprameters values for training the segmented two phases

	inputs	Hidden layers	Learn rate	momentum
Phase 1	2	5	0.01	0.5
Phase 2	7	5	0.016	0.5

Then we compare traditional BPNN and the segmented BPNN, see Figure 10. From the figure, we can see that the best MSE for BPNN is 0.0029 but the best MSE for segmented BPNN is 0.0013. Segmented BPNN achieve a better result for prediction. The red line stands for the real value for the Weibo reposts and the green line stands for the predicted value. In Figure 10(a), the red line is nearly coverd by green line and the predicted value fluctuates greatly. In Figure 10(b), the green line and the red line are basically consistent, which means segmented BPNN achieve a better result for reposts prediction than BPNN.



(a)



(b)

Figure 10. Comparison of accuracy of reposts prediction

3.3 Evaluation of the Hybrid Model

3.3.1 Parameters Setting of Genetic Algorithm

Population size is set as 80 to seek the best chromosome. Each chromosome contain 9 genes, No. 1 gene on behalf of the segmentation point, No. 2 to 5 genes stand for hidden layers, inputs, learn rate and momentum in the first segmented part of training data, No. 6 to 9 genes stand for hidden layers, inputs, learn rate and momentum in the second segmented part of training data. 100 generations of genetic evolution will be conducted in this paper.

We use a roulette selection mechanism and optimum retention mechanism to guarantee the best generation is copied to the next generation. In this way, both the stability of the algorithm and the global convergence are added. We can get the gene exchange probability P_c and gene mutation probability P_m by using the following adaptive algorithms.

$$P_c = \begin{cases} k_1(f_{\max} - f^{'})/(f_{\max} - f_{avg}) & f^{'} > f_{avg} \\ k_3 & f^{'} < f_{avg} \end{cases}, \quad P_m = \begin{cases} k_2(f_{\max} - f)/(f_{\max} - f_{avg}) & f > f_{avg} \\ k_4 & f < f_{avg} \end{cases} \quad \textbf{(16)}$$

where $f_{\max}$ is the largest population fitness, f_{avg} is the average population fitness, $f^{'}$ is the larger fitness value of the two individuals and f is the fitness value of variated individual.

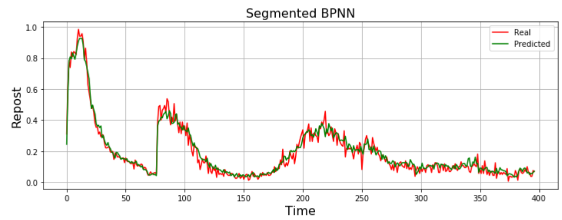
3.3.2 Evaluation of the Segmented BPNN with Genetic Algorithm

After 300 times generations, we obtain the best gene values for the final best chromosome, see Table. 2.

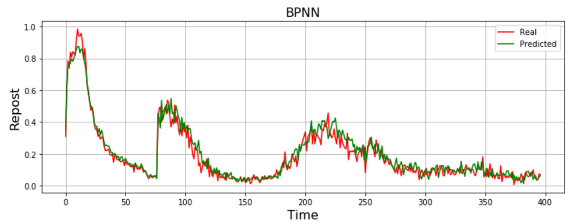
Table 2. Statistics of the best paprameters values for training the hybrid models

Segmentation point	Segmented BPNN 1				Segmented BPNN 2			
G1	G2	G3	G4	G5	G6	G7	G8	G9
0.1	**6**	**1**	**0.08**	**0.8**	**4**	**9**	**0.02**	**0.9**

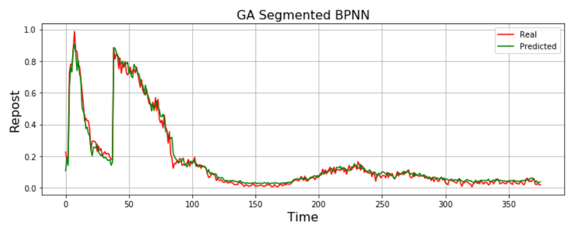
Then we use the gene values to build the propose neural network. In this section, we compare the prediction accuracy results under three kinds of BPNN, they are the regular BPNN (see Figure 11(a)), the segmented BPNN without genetic process (see Figure 11(b)) and the genetic-based segmented BPNN (Figure 11(c)). Compare to regular BPNN and segmented BPNN, the proposed GA- segmented BPNN's prediction curve is more closer to the real data curve. Moreover, the prediction accuracy of regular BPNN is 0.002, for segmented BPNN, prediction accuracy is 0.0013 and for GA segmented BPNN, MSE is 0.0007. Therefore, the proposed model enhance the prediction accuracy for Weibo reposts prediction.



(a)



(b)



(c)

Figure 11. Evalution of three kinds of BPNN

4 Conclusion

Through the real data analysis we found that the behavior of Weibo reposts has the characteristics of explosive growth in the early stage. The distribution of Weibo reposts in the statistical interval is extremely unbalanced. More specifically, the spacing between the average value of the reposts, the median value and the median value but the duplicate reposts are not counted is very large, which means the dispersion degree of reposts data is high and the stability of reposts data is poor. Although BPNN has the ability to deal with nonlinear mapping problems, the data with high dispersion degree and poor stationarity will bring the difficulties of parameter training and lead to the reduction of prediction accuracy. BPNN is used as the basic model to make prediction, but the reposts data is divided into two parts for building BPNN through analyzing the combination of average reposts value and median value. In order to find the best solution to build BPNN and enhance the ability of Weibo reposts prediction, we use genetic algorithm to search the global optimal solution. The genetic-based segmented BPNN has been evaluated on the datasets of Weibo. The results have shown that, the proposed model can improve the prediction accuracy of Weibo reposts.

Acknowledgements

This work has been supported by the Fundamental Research Funds for the Central Universities, No. 2018RC007.

References

[1] G. X. Luo, Y. Liu, Q. A. Zeng, S. M. Diao, F. Xiong, A Dynamic Evolution Model of Human Opinion as Affected by Advertising, *Physica A: Statistical Mechanics & Its Applications*, Vol. 414, pp. 254-262, November, 2014.

[2] M. Sattari, K. Zamanifar, A Cascade Information Diffusion based Label Propagation Algorithm for Community Detection in Dynamic Social Networks, *Journal of Computational Science*, Vol. 25, pp. 122-133, March, 2018.

[3] Y. Chen, C. Liang, D. Cai, Understanding WeChat Users' Behavior of Sharing Social Crisis Information, *International Journal of Human-Computer Interaction*, Vol. 34, No. 4, pp. 356-366, January, 2018.

[4] S. Kak, Neural Networks and Artificial Intelligence, *Information Sciences*, Vol. 70, No. 1-2, pp. 1-3, May, 1993.

[5] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning Internal Representations by Error Propagation, in: D. E. Rumelhart, J. L. McClelland (Eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, MIT Press, Cambridge, MA, 1986, pp. 318-362.

[6] S. J. Hanson, D. J. Burr, Minkowski-r Back-Propagation: Learning in Connectionist Models with Non-Euclidian Error Signals, *in Neural Information Processing Systems (NIPS 1987)*, Denver, CO, 1987, pp. 348-357.

[7] R. Hecht-Nielsen, *Theory of the Backpropagation Neural Network*, Harcourt Brace & Co, 1992.

[8] J. Tarigan, Nadia, R. Diedan, Y. Suryana, Plate Recognition Using Backpropagation Neural Network and Genetic Algorithm, *Procedia Computer Science*, Vol. 116, pp. 365-372, May, 2017.

[9] D. E. Goldberg, Genetic Algorithms in Search, *Optimization & Machine Learning*, Addison-Wesley Longman Publishing Co., 1989.

[10] S. Starke, N. Hendrich, S. Magg, J. Zhang, An Efficient Hybridization of Genetic Algorithms and Particle Swarm Optimization for Inverse Kinematics, *IEEE International Conference on Robotics and Biomimetics*. Qingdao, China, 2016, pp. 1782- 1789.

[11] F. P. Such, V. Madhavan, E. Conti, J. Lehman, K. O. Stanley, J. Clune, *Deep Neuroevolution: Genetic Algorithms Are a Competitive Alternative for Training Deep Neural Networks for Reinforcement Learning*, arXiv.org, 2018.

[12] M. Marzouk, A. Abubakr, Decision Support for Tower Crane Selection with Building Information Models and Genetic Algorithms, *Automation in Construction*, Vol. 61, pp. 1-15, January, 2016.

[13] Haviluddin, R. Alfred, Daily Network Traffic Prediction Based on Backpropagation Neural Network, *Australian Journal of Basic and Applied Sciences*, Vol. 8, No. 24, pp. 164-169, December, 2014.

[14] Haviluddin, R. Alfred, A Genetic-based Backpropagation Neural Network for Forecasting in Time-series Data, *International Conference on Science in Information Technology*, Yogyakarta, Indonesia, 2016, pp. 158-163.

[15] Purnawansyah, Haviluddin, Comparing Performance of Backpropagation and RBF Neural Network Models for Predicting Daily Network Traffic, *Makassar International Conference on Electrical Engineering and Informatics*, Makassar, Indonesia, 2014, pp. 166-169.

[16] Rohmatulloh, Marimin, Fuzzy Logic An Artificial Neural Network For Quality Improvement of Black Tea, *Jurnal Teknologi Dan Industri Pangan*, Vol. 18, No. 2, pp. 96-101, 2007.

[17] D. E. Goldberg, *Genetic Algorithm in Search, Optimization and Machine Learning*, Addison Wesley, 1989.

[18] K. A. D. Jong, W. M. Spears, D. F. Gordon, Using Genetic Algorithms for Concept Learning, *Machine Learning*, Vol. 13, No. 2-3, pp. 161-188, November, 1993.

[19] A. A. Freitas, A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery, in: A. Ghosh, S. Tsutsui, (Eds.), *Advances in Evolutionary Computing*, Springer, Berlin Heidelberg, 2003, pp. 819-845.

[20] S. García, A. Fernández, J. Luengo, F. Herrera, A Study of Statistical Techniques and Performance Measures for Genetics-based Machine Learning: Accuracy and Interpretability, *Soft Computing*, Vol. 13, No. 10, pp. 959-979, August, 2009.

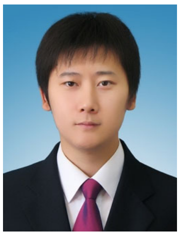
[21] F. Q. Jia, Z. W. He, Z. J. Tian, Z. B. Chen, H. C. Wang, The

Application of Genetic Algorithm Backpropagation Neural Network Model on the Prediction and Optimization of Wastewater Treatment System, *Advanced Materials Research*, Vol. 838-841, pp. 2525-2531, November, 2014.
[22] J. N. D. Gupta, R. S. Sexton, Comparing Backpropagation with a Genetic Algorithm for Neural Network Training, *Omega*, Vol. 27, No. 6, pp. 679-684, 1999.
[23] Z. Y. Zhang, Y. Liu, G. D. Xu, G. X. Luo, Recommendation using DMF-based Fine Tuning Method, *Journal of Intelligent Information Systems*, Vol. 47, No. 2, pp. 233-246, October, 2016.

Biographies

Guixun Luo received his Bachelor's degree in Beijing Jiaotong University in China, in 2010. Currently, he is a Ph.D. student in the School of Electronics and Information Engineering at Beijing Jiaotong University. His current research fields include data mining and social network.

Yun Liu received her Ph.D. degree in Electronic Engineering from Beijing Jiaotong University in China. Currently, she is an IET Fellow, a Professor and the director of the Department of communication and Information System, and the director of the Key Laboratory of Communication and Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University. Her current research interests include Telecommunication, Computer Networks, Network Security, Intelligent Transportation System, Social Dynamics, etc. Dr. Liu has edited many books and published over 200 papers and book chapters, as well as participated in many international academic activities, including the organization of several international conferences.



Zhiyuan Zhang received his Bachelor's degree in University of International Relations in China, in 2012. Currently, he is a Ph.D. student in the School of Electronics and Information Engineering at Beijing Jiaotong University. His current research fields include recommender systems and data mining.