

# **SOLUTIONS FOR IMPROVING TRANSPORTATION IN SOUTH AFRICA: TRAFFIC DEMAND FORECASTING OF PUBLIC BICYCLE STATION BASED ON BP NEURAL NETWORK**

**S LIN**

School of Highway, Chang'an University, Xi'an 710064, China  
Tel: (+86) 13055261977, Email: [2016901524@chd.edu.cn](mailto:2016901524@chd.edu.cn)

## **ABSTRACT**

To attain sustainable development, developing countries must focus on the expansion of public transportation systems, this is especially the case in South Africa. The forecasting of the traffic demand for public bicycles is of great significance in optimizing the deployment of vehicles and improving the efficiency of public resources utilization. This paper establishes a traffic demand prediction model using the BP neural network, namely the error back-propagation neural network. The data and network structures of the model were adjusted, and the basic parameters of the model were determined. The international data set was applied to validate the model, and the test results indicate that the BP neural network traffic demand forecasting model outweighs the traditional linear regression prediction method in Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE). Finally, the paper offers recommendations for local authorities in South Africa on how to utilize data to effectively improve public transportation system.

## **1. INTRODUCTION**

As a convenient and low-carbon public transportation method, public bicycles do not only alleviate urban traffic problems, but also meets the sustainable development strategy. Within a Chinese domestic context, the rapid development of public bicycles has created an imbalance between supply and demand of bicycle stations, which not only leads to a resource wastage but also hinders urban traffic order (Wang, 2017).

Previous scholars have carried out a series of studies on the traffic demand prediction of public bicycle stations. Parkin et al. (2008) predicted the scheduling demand of the urban public bicycle system using the logistic regression model. Kaltenbrunner et al. (2010) used a time series analysis method to predict the number of public bicycles at various rental points based on Barcelona public bicycle system operational data. Bacciu et al. (2017) used the random forest and support vector machine model in machine learning to predict short-term bicycle borrowing issues at public bicycle stations.

However, current research on public bicycle data analysis and prediction models is mainly based on mathematical optimization models and machine learning algorithms, while deep learning methods are rarely introduced and analyzed. The paper introduces BP neural network model in deep learning and studies the short-term traffic demand forecasting problem of public bicycles, which is an innovation in this field. Furthermore, China's experience of public transport management is an excellent lesson for other developing countries, such as South Africa.

## 2. METHODOLOGY

BP algorithm is one of the most effective multi-layer neural network learning methods, whose features are forward propagation of signal and backward propagation of error. By continuously adjusting the weight value of network, the final output of the network is as close as possible to the expected output to achieve the purpose of training.

### 2.1 BP neural network structure and forward propagation

The BP neural network structure consists of various layers of neurons (a total of  $l$  layers), whereby first layer is the input layer, last layer (the  $l^{th}$  layer) is the output layer, and the other layers are hidden layers (  $2^{nd}$  layer  $\sim$   $(l-1)^{th}$  layer). The perceptron combines weights in a linear combination and converts them into output signals through an activation function to create the final output result. This process is known as forward propagation, and will be analyzed as follows. The activation of the  $j^{th}$  neuron in the  $l^{th}$  layer is

$$a_j^l = \sigma \left( \sum_k \omega_{jk}^l a_k^{l-1} + b_j^l \right) \quad (1)$$

where:  $\sigma( )$  acts as the activation function,  $\omega_{jk}^l$  is the connection weight of the  $k^{th}$  neuron in the  $(l-1)^{th}$  layer to the  $j^{th}$  neuron in the  $l^{th}$  layer,  $a_k^{l-1}$  acts as the activation function of the  $k^{th}$  neuron in the  $(l-1)^{th}$  layer and  $b_j^l$  is bias of the  $j^{th}$  neurons in the  $l^{th}$  layer.

Define  $\omega^l$  as a weight matrix, each element represents a weight, and each row represents the weight of the connected layer. The activation function is rewritten in a matrix form, and the forward propagation process can be expressed as

$$a^l = \sigma(\omega^l a^{l-1} + b^l) \quad (2)$$

In situations where multiple samples are input simultaneously, the input layer is divided into  $m$  columns, and each column represents one input sample. The activation function in situation of multi-sample input can be expressed as

$$A^l = \sigma(\omega^l A^{l-1} + b^l) \quad (3)$$

where:

$$A^{l-1} = \begin{bmatrix} | & | & \dots & | \\ a^{[l-1](1)} & a^{[l-1](2)} & \dots & a^{[l-1](m)} \\ | & | & \dots & | \end{bmatrix} \quad (4)$$

### 2.2 Back propagation and training

Weights are assigned with random values at the initial establishment phase of the neural network. When the neural network learns its input-output rules based on the training data set, the network will adjust continuously according to the errors of classified weight.  $(y - \hat{y})^2$  is used to quantify the error of each prediction.

$$Error = \frac{1}{2} \sum_{\mu} (y - \hat{y})^2 = \frac{1}{2} \sum_{\mu} (y - f(\sum \omega \cdot x))^2 \quad (5)$$

where:  $y$  is the expected value of output,  $\hat{y}$  is the predictive value of output,  $\omega$  is the weight value, and  $x$  is the input value,  $f( )$  is the activation function.

Reasonable adjusting of value of error can improve the prediction accuracy of the neural network. It can be recognized from equation (5) that the value of error is related to the input value  $x$  and the weight value  $\omega$ . Since the input value is related to the original data, only the weight value can be adjusted. The paper uses the gradient descent method to adjust the weight value. The weight value is inversely proportional to the gradient of the current  $\omega$  position and used to calculate the partial derivative of the weight.

$$\Delta\omega = -gradient = -\eta \frac{\alpha Error}{\alpha\omega} = \eta(y - \hat{y})f'(h)x \quad (6)$$

where:  $h = \sum\omega \cdot x$ ,  $\delta = (y - \hat{y})f'(h)$ .  $\delta$  is an error term and can be used for arithmetical operation.  $\eta$  is the learning rate, which controls the speed of weight variation. It can be seen from the above derivation that in a complex network structures, the weight is gradually updated from the last layer to the previous network level, which is known as back propagation. The basic idea is to adjust network parameters by calculating the error between the output layer and the expected value to reduce the error.

### 3 RESULTS AND DISCUSSION

#### 3.1 Data source and processing

This paper utilizes the data of the public bicycle rental point in Washington, DC. The data set is derived from the Center for Machine Learning and Intelligent Systems, University of California, Irvine. Operational data for a hot spot in Washington, DC from January 1, 2011 to December 31, 2012, was recorded in hours, with a total of 17,379 observations recorded. The basic characteristics of the data set are shown in Table 1.

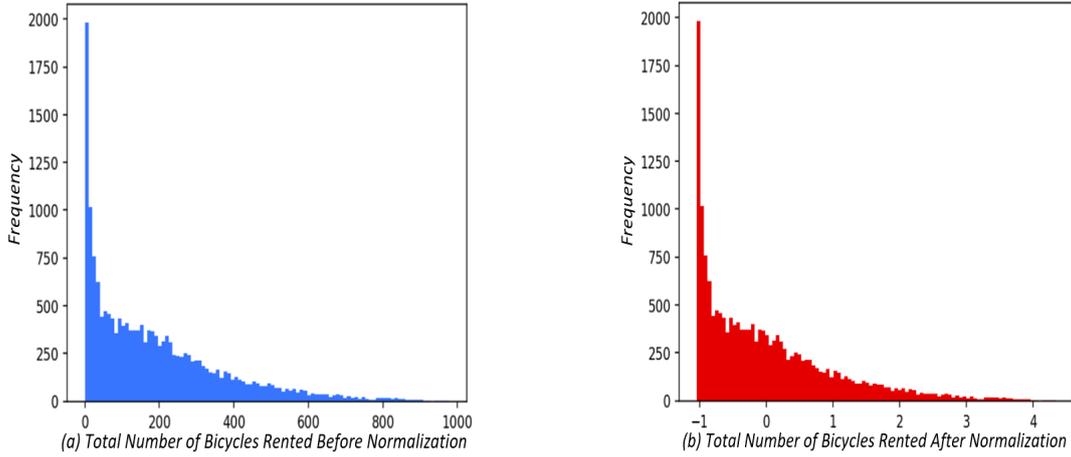
**Table 1: Main data characteristics and description**

Data	Description	Data	Description
instant	record index	weathersit	1: Clear, Few clouds, Partly cloudy, Partly cloudy
dteday	date		2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
yr	year (0: 2011, 1:2012)		3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
mnth	month (1 to 12)		4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
hr	hour (0 to 23)		holiday
weekday	day of the week	hum	Normalized humidity. The values are divided to 100 (max)
workingday	if day is neither weekend nor holiday is 1, otherwise is 0.	atemp	Normalized feeling temperature in Celsius. The values are divided to 50 (max)
season	season (1: spring, 2: summer, 3: fall, 4: winter)	Cnt	count of total rental bikes including both casual and registered
temp	Normalized temperature in Celsius. The values are divided to 41 (max)	registered	count of registered users
windspeed	Normalized wind speed. The values are divided to 67 (max)		
casual	count of casual		

In processing the original data, one-hot encoding was initially used for discrete categorical variables processing, including season, month, weekday, hour, and weather, and the categorical variables are vectorized. The processing results are presented in Table 2. Second, the zero-mean normalization method is used to standardize these six data characteristics, including temperature, humidity, windspeed, count of casual, count of registered users, and count of total rental bikes. The standardized data conforms to the standard normal distribution, as shown in Figure 1.

**Table 2: Category variable processing result**

Original categorical variable	Categorical variable encoded by One-hot method
Season	season 1, season 2, season 3, season 4
mntth	mntth 1, mntth 2...mntth 12
weekday	weekday_1,weekday_2...weekday_7
hr	hr_1, hr_2...hr_24
weathersit	weathersit_1, weathersit_2...weathersit_4

**Figure 1: Data normalization result**

After standardization, the raw data is converted into a pure dimensionless value, which significantly facilitates the comparison and weighting of different units and magnitude indicators. Finally, from the data (a total of 17,389), 21 days of data is selected (a total of 504) as the test set, 60 days data (a total of 1,440) as the validation set, and the rest of the data was used as the training set.

### 3.2 Algorithm implementation and parameter adjustment

The network is constructed based on the forward propagation of the signal and the back propagation of the error. The grid is initialized and initial values were then assigned to the weights, which followed a normal distribution, whose probability density function can be expressed as:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (7)$$

where: from the input layer to the hidden layer,  $\mu = 0$ ,  $\sigma = \sqrt{\text{input\_nodes}}$  (number of input nodes); from the hidden layer to the output layer,  $\mu = 0$ ,  $\sigma = \sqrt{\text{output\_nodes}}$  (number of output nodes). The activation function between input layer and hidden layer acts as the logarithmic sigmoid transfer function, whose standard form is

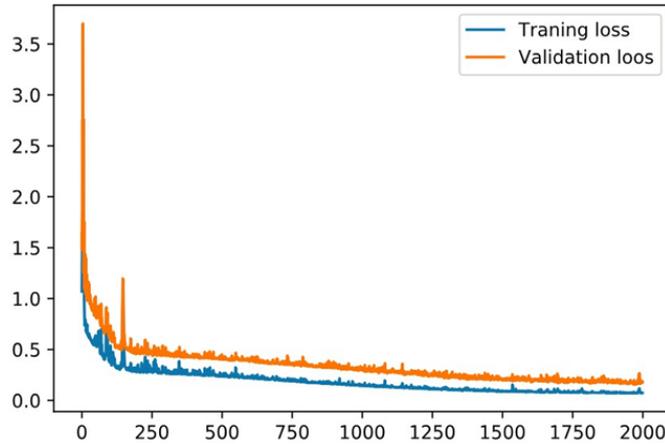
$$f(x) = \frac{1}{1 + e^{-x}} \quad (0 < f(x) < 1) \quad (8)$$

The activation function between hidden layer and output layer acts as a linear function

$$f(x) = x \quad (9)$$

The formula (5) is the error term to update the weight.

When conducting neural network unit testing, the basic data set (including data paths and data structures) is verified. The functional verification of the neural network model is then performed. Finally, the accuracy of the model operation is tested. In this experiment, the time for completing five unit tests was 0.04 s. After the unit test, the accuracy of the neural network model is guaranteed and the training process can be carried out.

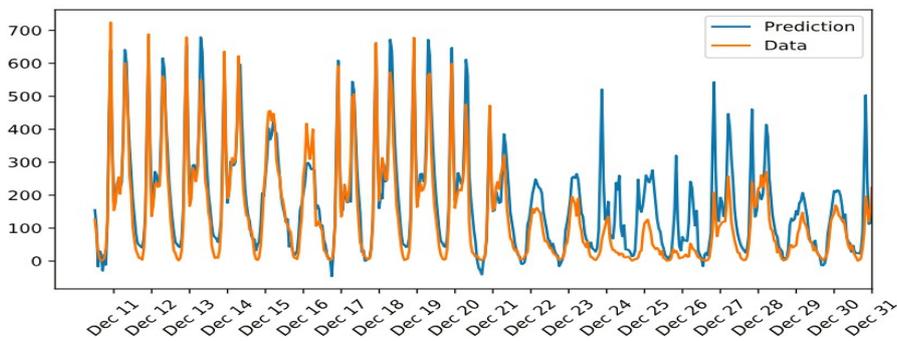


**Figure 2: Train loss and test loss of neural network**

For network training, we apply the maximum number of iterations, the learning rate, and the number of hidden nodes as 2000, 0.8 and 12 respectively. Record the training process and the training loss (train loss & test loss). The test results show that the model has a train loss of 0.082 and a test loss of 0.170. Through the training error diagram (Figure 2), it can be seen that the train loss and the test loss continue to decrease until it reaches convergence, which illustrates that the model training results are optimal.

### 3.3 Forecasting results and evaluation

Using multiple linear regression as the experimental benchmark model, and the results of BP neural network model were compared and evaluated. In order to reflect the prediction effect of the model more scientifically and intuitively, this paper uses the data of three consecutive weeks as the test set, and Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE) are used to evaluate the model. The evaluation results are shown in Table 3.



**Figure 3: Prediction results and comparison**

**Table 3: Comparison of the model prediction results**

Predictive model	Evaluation index	
	MSE%	MAPE/%
Multiple linear model	0.1423	14.3315
BP neural network	0.0879	5.2262

It can be seen from Table 3 that the BP neural network prediction model is significantly advanced compared to the multivariate linear reference model. From Figure 3, we can see that the predicted values are basically consistent with the true values, and the relative error is small. This not only validates the calculation results in Table 3, but also shows the generalization and reliability of the prediction model.

#### **4. CONCLUSIONS**

The main research and conclusions are summarized as follows: building of the neural network and completed the unit test based on the processed data, and then the model was trained in three aspects, including the number of iterations, the learning rate and the number of hidden layers. Additionally, the model prediction results were compared with the multiple linear regression benchmark model, which reveals that BP neural network model has lower MSE and MAPE and demonstrated significant superiority.

#### **5. RECOMMENDATIONS**

Based on the above discussion, recommendations for improving the public bicycle management system in South Africa are as follows:

- Improve station layout planning and supporting resources. The city-planning administration should design related projects according to the level of urban development and citizens' travel demands.
- Use information technology to improve service quality. The municipal administrative department should target some important stations in peak times, especially when the vehicles are full and no bicycles are available during rush hours. Big-data analysis should be used to dispatch vehicles in advance to maintain the balance of vehicles at each station.

#### **6. REFERENCES**

Bacciu, D, Carta, A, Gnesi, S & Semini, L, 2016. An experience in using machine learning for short-term predictions in smart transportation systems. *Journal of Logical & Algebraic Methods in Programming*, 87, 52-66.

Kaltenbrunner, A, Meza, R, Grivolla, J, Codina, J & Banchs, R, 2010. Urban cycles and mobility patterns: exploring and predicting trends in a bicycle-based public transport system. *Pervasive and Mobile Computing*, 6(4), 455-466.

Parkin, J, Wardman, M & Page, M, 2008. Estimation of the determinants of bicycle mode share for the journey to work using census data. *Transportation*, 35(1), 93-109.

Wang, Guangrong, 2017. Systematic research on bicycle-sharing development. *Journal of Chang'an University (Social Science Edition)* (2).