LSTM Network Based Traffic Flow Prediction for Cellular Networks

Shulin Cao(✉) and Wei Liu

State Key Labs of ISN, Xidian University,
Xi'an 710071, Shaanxi, People's Republic of China
slcao_cn@stu.xidian.edu.cn, liuweixd@mail.xidian.edu.cn

Abstract. The traffic flow prediction of cellular network requires low complexity and high accuracy, which is difficult to meet using the existing methods. In this paper, we propose an long short-term memory (LSTM) network based traffic flow prediction in which we consider temporal correlations inherently and nonlinear characteristics of cellular network traffic flow data. We use Back Propagation Through Time (BPTT) to train the LSTM network and evaluate the model using mean square error (MSE) and mean absolute error (MAE). Simulation results show that the proposed LSTM network based traffic flow prediction for cellular network is superior to the stacked autoencoder network based algorithm.

Keywords: Deep learning · Long short-term memory (LSTM) · Traffic flow prediction · Cellular network

1 Introduction

With the popularity of smart phones and the upgrading of wireless communication techniques, the demand for data services increases rapidly. Thus, the resource allocation place the critical role for meeting the demand. However, the environment is changed dynamically and the resource allocation based on information at the current moment has a certain time delay, it can not satisfy the demand for resource at the current moment. So traffic flow prediction in cellular network is imperative requirement.

There are many traditional methods for predicting traffic flow in cellular network. For example, In [1], Auto-Regressive Integrated Moving Average (ARIMA), fractional ARIMA, artificial neural network (ANN), and wavelet-based predictors were used for wireless traffic prediction and analyzed computational complexity. The joint Kohonen maps and ARIMA time series models

The financial support of the program of Key Industry Innovation Chain of Shaanxi Province, China (2017ZDCXL-GY-04-02), of the program of Xi'an Science and Technology Plan (201805029YD7CG13(5)), Shaanxi, China, of National S&T Major Project (No. 2016ZX03001022-003), China, and of Key R&D Program - The Industry Project of Shaanxi (Grant No. 2018GY-017) are gratefully acknowledged.

H. Song et al. (Eds.): SIMUtools 2019, LNICST 295, pp. 643–653, 2019.
https://doi.org/10.1007/978-3-030-32216-8_63

method was proposed in [2] which is a short-term traffic prediction. A theory which aims to model the univariate traffic condition data flow as a seasonal autoregressive integrated moving average process was proposed in [3]. In wireless networks, information theory techniques are proposed in [4] for discrete sequence prediction. A multi-resolution finite impulse response neural network learning algorithm based on the maximum overlap discrete wavelet transform was proposed in [5] for network traffic prediction (real world aggregated Ethernet traffic data).

With the development of machine learning techniques, various machine learning methods are used for wireless network traffic flow prediction, the experimental results show that these methods have effectively improved the accuracy of traffic flow prediction. The joint principal component analysis and time series model was proposed in [6] to predict the fluctuation of Internet traffic in the international IP transmission network. An ANN model based on Multilayer Perceptron was proposed in [7] to predict Internet traffic flow in IP networks. Three methods for accurately predicting traffic in TCP/IP-based networks were presented in [8], which included a neural network integration methods and two adaptive time series methods (ARIMA and Holt-Winters) respectively. A short-term network traffic prediction algorithm LSVM-DTW-K based on Chaos Theory and Support Vector Machines was proposed in [9] for wired and wireless campus networks. In [10], the traffic model based on Elman-NN network was used to predict future traffic and the results showed that this method can achieve better performance. In addition, it also includes traffic flow prediction problems when incomplete data exist. There are many other methods of machine learning that have been proposed for traffic prediction, [11–13].

Deep learning has developed rapidly, and prediction methods based on deep learning have also developed, such as in transportation and communication networks. In [14], a stacked autoencoder model (SAE) to learn general traffic flow characteristics, after extracting the traffic characteristics, then this method uses top-level logistic regression to predict traffic flow. Two different artificial neural network methods are proposed in [15], which are multilayer perceptrons and stacked autoencoder for predicting Internet traffic. An underlying deep belief network (DBN) and top-level multitask regression layer deep learning model was proposed in [16], where DBN is used for unsupervised feature learning. The deep learning models that has been used to perform traffic flow prediction that do not fully consider the correlation between time series. Recently, LSTM network has been developed and extensively used on time series prediction, such as, for TCP/IP networks, a model that combines LSTM with deep neural networks was proposed in [17], which utilized autocorrelation features to improve the accuracy of network traffic prediction. So, LSTM network takes full account of the temporal correlation of time series and can remember some of the information entered before so that to exploit the relationship between these time series and improve the prediction accuracy. In this paper, we proposed a LSTM network based traffic flow prediction for cellular network. Time series data were highly related, so we can utilize LSTM that can reserve long-term memory to learn the basic characteristics of cellular network traffic flow data in the cell.

The rest of the paper is organized as follows. We describe the system model in Sect. 2. The LSTM network based traffic flow prediction is described in Sect. 3. The simulation results of the proposed LSTM network based traffic flow prediction for cellular networks are provided in Sect. 4. In Sect. 5, conclusion is offered.

2 System Model

The system considered in this paper consists of one micro cell which serves K users as shown in Fig. 1. In this paper, we consider the uplink traffic flow data of the micro cell. The data was collected every time slot and each time slot consists of n minutes. Denote $x^{\langle t \rangle}$ as the traffic flow data at tth time slot. In this network, we want to use the collected traffic flow data $\{x^{\langle 1 \rangle}, x^{\langle 2 \rangle}, x^{\langle 3 \rangle}, \cdots, x^{\langle T \rangle}\}$ from T time slots to predict the volume of traffic flow $x^{\langle T+1 \rangle}$ at the $(T+1)$th time slot.

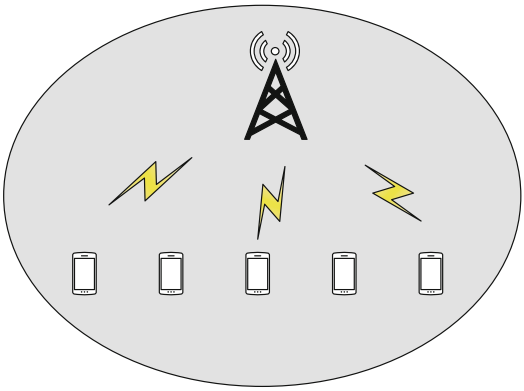

Fig. 1. A simple cellular network

3 LSTM Network Based Traffic Flow Prediction

3.1 LSTM Network

Figure 2 shows the structure of LSTM network used for predicting traffic flow. As shown in Fig. 2, we use T consecutive traffic flow data $\{x^{\langle 1 \rangle}, x^{\langle 2 \rangle}, x^{\langle 3 \rangle}, \cdots, x^{\langle T \rangle}\}$ as the input to the LSTM network to predict cellular traffic flow $x^{\langle T+1 \rangle}$ at the $T+1$ time slot. The traffic flow prediction requires T time steps at a time, and each time step corresponds to an LSTM cell. The LSTM network adopts a self-looping method in which only one data can be entered into the network at a time.

The basic component of LSTM network is the LSTM cell as shown in Fig. 3, the tth LSTM cell corresponds to the tth time step, which has the ability to

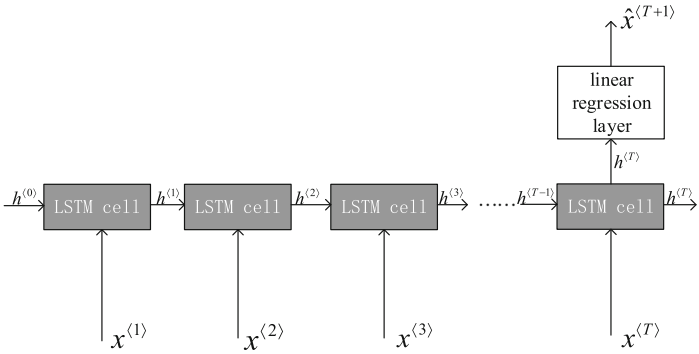

Fig. 2. The structure of LSTM network

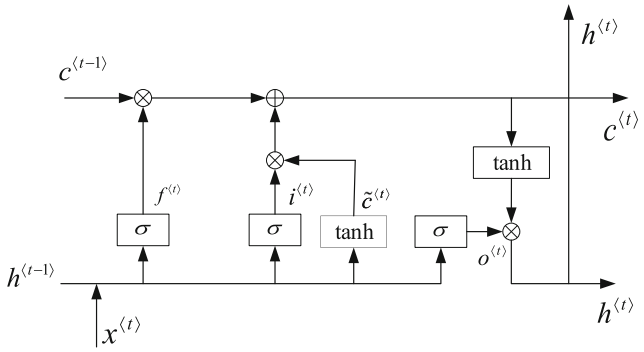

Fig. 3. The structure of LSTM cell

remove or add information to a cell's state through a well-designed structure consisting of three different kinds of gates, which are "input gate", "output gate" and "forget gate" [18]. Each gate is a feedforward network layer which consists of one hidden layer, the number of hidden layer neurons denoted as P which is called LSTM cell units. The gate is a way of letting messages pass by and overcoming the vanishing gradient and the exploding gradient. The gate has the ability to choose whether to pass the information by itself and the gate's output value is in the range of (0,1), where 1 means "completely reserved", 0 means "completely discarded". $\sigma(x)$ is the gate function, which is usually chosen as sigmoid function so that the gate function can be expressed as

$$\sigma(x) = \frac{1}{1 + \exp^{-x}} \tag{1}$$

According to Fig. 3, the input of the LSTM cell is $\boldsymbol{h}^{\langle \boldsymbol{t-1}\rangle}$, $\boldsymbol{x}^{\langle \boldsymbol{t}\rangle}$, $\boldsymbol{c}^{\langle \boldsymbol{t-1}\rangle}$, where $\boldsymbol{c}^{\langle \boldsymbol{t-1}\rangle}$ is the cell state at the $t-1$ time step, $\boldsymbol{h}^{\langle \boldsymbol{t-1}\rangle}$, the output value of cell state at the $t-1$ time step. The output value of the LSTM cell is $\hat{x}^{\langle t+1\rangle}$ at the t time step.

In LSTM network, $\boldsymbol{i}^{\langle t \rangle}$, $\boldsymbol{f}^{\langle t \rangle}$, and $\boldsymbol{o}^{\langle t \rangle}$ denote the output value of "input gate", "forget gate" and "output gate", which represented by (2), (3) and (4), respectively. The output value of the forget gate decides whether the content of the cell state at $t-1$ time step will add to the update cell state at the t time step. The output value of the input gate decides whether new candidate values $\tilde{\boldsymbol{c}}^{\langle t \rangle}$ could add to cell state, where $\tilde{\boldsymbol{c}}^{\langle t \rangle}$ is expressed as (5). Then, combine $\tilde{\boldsymbol{c}}^{\langle t \rangle}$ and $\boldsymbol{c}^{\langle t-1 \rangle}$ can get the update cell state $\boldsymbol{c}^{\langle t \rangle}$, which indicates the update contents of the cell state stored at the t time step and the cell state is expressed as (6). The output value of the output gate decides whether the updated cell state can influence the output of the LSTM cell, the output value of cell state is expressed as (7) [17].

$$\boldsymbol{i}^{\langle t \rangle} = \sigma(\boldsymbol{W_i}[h^{\langle t-1 \rangle}, x^{\langle t \rangle}] + \boldsymbol{b_i}) \tag{2}$$

$$\boldsymbol{f}^{\langle t \rangle} = \sigma(\boldsymbol{W_f}[h^{\langle t-1 \rangle}, x^{\langle t \rangle}] + \boldsymbol{b_f}) \tag{3}$$

$$\boldsymbol{o}^{\langle t \rangle} = \sigma(\boldsymbol{W_o}[h^{\langle t-1 \rangle}, x^{\langle t \rangle}] + \boldsymbol{b_o}) \tag{4}$$

$$\tilde{\boldsymbol{c}}^{\langle t \rangle} = \tanh(\boldsymbol{W_c}[h^{\langle t-1 \rangle}, x^{\langle t \rangle}] + \boldsymbol{b_c}) \tag{5}$$

$$\boldsymbol{c}^{\langle t \rangle} = \boldsymbol{i}^{\langle t \rangle} * \tilde{\boldsymbol{c}}^{\langle t \rangle} + \boldsymbol{f}^{\langle t \rangle} * \boldsymbol{c}^{\langle t-1 \rangle} \tag{6}$$

$$\boldsymbol{h}^{\langle t \rangle} = \boldsymbol{o}^{\langle t \rangle} * \tanh \boldsymbol{c}^{\langle t \rangle} \tag{7}$$

which the parameters $\boldsymbol{W_c}$, $\boldsymbol{W_i}$, $\boldsymbol{W_f}$ and $\boldsymbol{W_o}$ are weight matrixes corresponding to the network structure of cell state, input gate, forget gate, and output gate, the dimension of them is $P \times (P+1)$, $\boldsymbol{b_c}$, $\boldsymbol{b_i}$, $\boldsymbol{b_f}$, $\boldsymbol{b_o}$ are bias vector corresponding to the network structure of cell state, input gate, forget gate, and output gate, the dimension of them is $P \times 1$ and $\tanh(x)$ is represented as $\frac{\exp^x - \exp^{-x}}{\exp^x + \exp^{-x}}$. The $\boldsymbol{h}^{\langle t \rangle}$ can get the predicted traffic flow value $\hat{x}^{\langle t+1 \rangle}$ through the linear regression layer. Because LSTM network is a self-looping architecture and the parameters it uses for each time step are shared, we continue to train the LSTM network and update these parameters to make predictions more accurate.

For the data of sequence length T, the output of the Tth LSTM cell is also the predict traffic flow data of the LSTM network and the output of the LSTM network needs to consider the input of the previous $T-1$ time step.

3.2 Training LSTM Network

When training the LSTM network, we firstly initialize all parameters, usually to a very small number close to 0 and initialize $c^{\langle 0 \rangle}$ and $\hat{x}^{\langle 0 \rangle}$ to 0. Selecting M number of epochs during training, each epoch consists of N batches, the size of each batch is J which means that each batch contains J sequences, each sequence is $\{x^{\langle 1 \rangle}, x^{\langle 2 \rangle}, x^{\langle 3 \rangle}, \cdots, x^{\langle T \rangle}\}$. When we set the parameters we must follow that the batch multiplied by the batch size equals the number of training data set. Initialize $c^{\langle 0 \rangle}$ and $\hat{x}^{\langle 0 \rangle}$ for the end of each batch size training, the epoch indicates that the data of the training set is trained several times, and the batch indicates that the training set is divided into several parts and input into the network for training.

We use Back Propagation Through Time (BPTT) to train the network [19]. Given a set of training samples $\{x^{\langle 1\rangle}, x^{\langle 2\rangle}, x^{\langle 3\rangle}, \cdots, x^{\langle t\rangle}, \cdots, x^{\langle m\rangle}\}$. For the network, we choose the time series of length T as input, and the value of $(T+1)$th time slot as label. For every batch, the loss function is represented by (8) [20]

$$L^{\langle J\rangle}\left(\hat{x}^{\langle t+1\rangle}, x^{\langle t+1\rangle}\right) = \frac{1}{J}\sum_{t=1}^{J}\left(x^{\langle t+1\rangle} - x^{\langle t+1\rangle}\right)^2 \tag{8}$$

and the whole loss function is showed in (9)

$$L\left(\hat{x}, x\right) = \frac{1}{m}\sum_{t=1}^{m}\left(\hat{x}^{\langle t+1\rangle} - x^{\langle t+1\rangle}\right)^2 \tag{9}$$

where $\hat{x}^{\langle t+1\rangle}$ is the prediction of traffic flow at t time slot, $x^{\langle t+1\rangle}$ is the actual value at $t+1$ time slot.

$$\boldsymbol{W}_* = \hat{\boldsymbol{W}}_* - \alpha\frac{\partial L\left(\hat{x}, x\right)}{\partial \boldsymbol{W}_*} \tag{10}$$

Where α is learning rate for training LSTM network, $\hat{\boldsymbol{W}}_*$ denotes $\boldsymbol{W}_*$ at previous time step, and $\boldsymbol{W}_*$ can represent $\boldsymbol{W_c}$, $\boldsymbol{W_i}$, $\boldsymbol{W_f}$ and $\boldsymbol{W_o}$. When we adjust the parameters of this network, the objective is to minimize the whole loss function of the network.

4 Simulation Results

In this section, we make the evaluation standard to calculate the accuracy of prediction. There are two evaluation standards included in this paper, which are the mean absolute error (MAE) and the Mean Square Error (MSE) [14]. Given a set of training samples $\{x^{\langle 1\rangle}, x^{\langle 2\rangle}, x^{\langle 3\rangle}, \cdots, x^{\langle t\rangle}, \cdots, x^{\langle m\rangle}\}$, the length of training set is m. The definitions of them are shown as follows respectively

$$\text{MAE} = \frac{1}{m}\sum_{t=1}^{m}\left|\hat{x}^{\langle t\rangle} - x^{\langle t\rangle}\right| \tag{11}$$

$$\text{MSE} = \frac{1}{m}\sum_{t=1}^{m}\left(\hat{x}^{\langle t\rangle} - x^{\langle t\rangle}\right)^2 \tag{12}$$

The cellular network traffic flow data set is collected from the cell which is collected every 15 min, and finally a total of 3,500 data. 80% of historical data was used for training set to train the network, and the rest of historical data was used for test set to test the performance of the network. In this paper, we use the Keras framework to build the LSTM network module [21], which is a deep learning framework and the underlying library uses theano or tensorflow. When training the LSTM network model, we found that setting the parameters

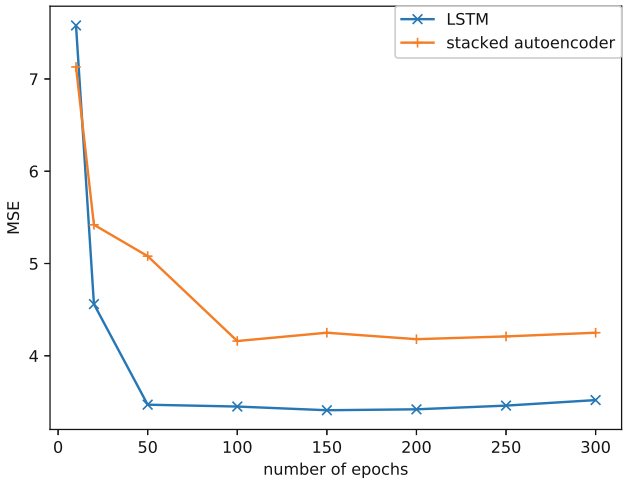

Fig. 4. MSE over epochs

$M = 300$, $N = 175$, $J = 20$, $T = 12$ and $P = 4$ to train the LSTM network will get the optimal cellular network traffic flow prediction result.

As shown in Fig. 4, LSTM has a fast convergence rate, which can achieve the best prediction accuracy when the time of epoch is 150 compared to the stacked autoencoder which need epoch 100 times to reach the current optimal prediction, so we choose $M = 150$. From the LSTM curve, it can be seen that the MSE will fluctuate a little during the training process but it will decrease at last. This is because when the new data input to the network, it may not have been trained well to make prediction. When the network is trained well, if you continue to increase the number of training epoch, it will lead to overfit, MSE will always rise.

Figure 5 shows that the variation of MSE with the number of LSTM cell units in the hidden layer in each gate. There is a sharp drop in MSE when the number of LSTM cell units changing from 1 to 4, which indicate that the prediction accuracy is gradually increasing, and the MSE reaches the minimum value when the number of LSTM cell units is 4, where the prediction effect is the best. Then the MSE starts to rise with the number of LSTM cell units increasing, while the prediction accuracy decreases, we choose $P = 4$. Because the LSTM is composed of complex nonlinear functions, the structure is complex to explain. This result can help us to choose the best local value of the cell unit number to make the best prediction for the traffic flow data.

In order to reflect the performance of LSTM in traffic flow prediction, we do a comparative experiment with a stacked autoencoder network. We use the same data set, the same data distribution, and the optimal parameters adjustment. After training network convergence, we get the perform comparison of the LSTM network and stacked autoencoder network, as shown in Table 1. According to

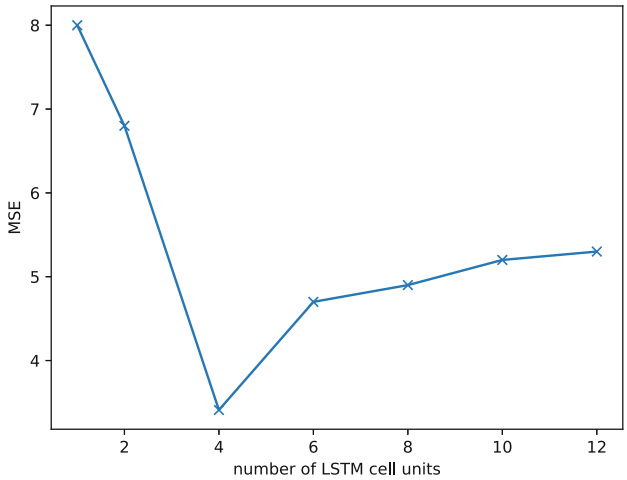

Fig. 5. MSE changes with the number of LSTM cell units in the hidden layer in each gate

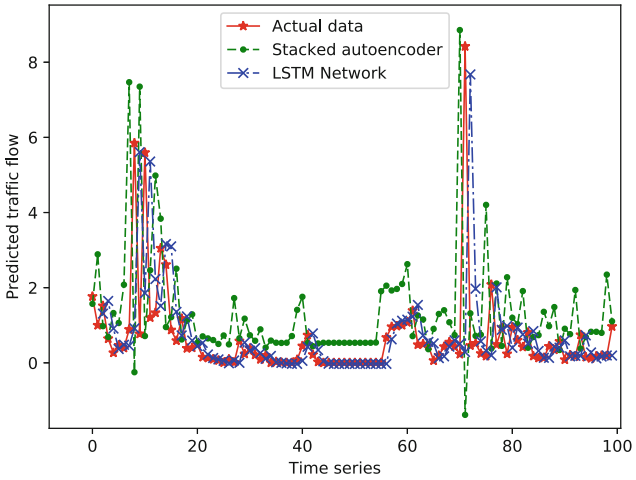

Fig. 6. Training data prediction

Table 1. Performance comparison of the LSTM network and Stacked autoencoder

Model	MAE	MSE
LSTM network	1.43	3.40
Stacked autoencoder	2.23	4.16

the results of comparison, we can make the conclusion that the LSTM network improve the accuracy of the cellular network traffic flow prediction.

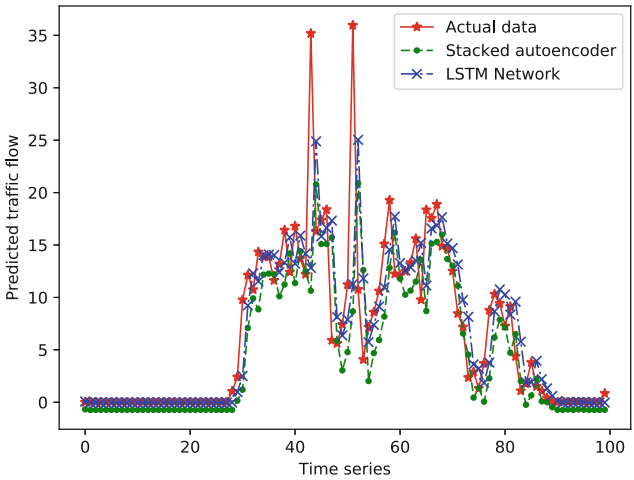

Fig. 7. Testing data prediction

Figure 6 shows the prediction of training data. As can be seen from the Fig. 6, the actual data we collected is periodic at most of the time and meets the data requirements of the LSTM network. The traffic flow prediction results of the LSTM network more closely resemble the actual data compared to stacked autoencoder network.

Figure 7 shows the partial prediction data of the testing data. As can be seen from the Fig. 7, our trained LSTM network gains high performance on the test set, indicating that LSTM network has a good generalization capability.

5 Conclusion

In this paper we proposed a deep learning based traffic flow prediction model. We extract a time series from the real-time traffic flow of the cellular networks. Then we train a deep learning model called LSTM network using these time series. Since the traffic flow at the current time is highly correlated with the previous time, the LSTM network is quite suitable for the prediction. And our simulation results showed that the proposed LSTM network gain significant performance.

From a practical aspect, with the real-time traffic flow as inputs, the output of the LSTM network will benefit greatly in resource allocation for the service provider. In addition, the high accuracy of the LSTM network traffic flow prediction in the proposed scheme ensures an engaging user experience.

References

1. Feng, H., Shu, Y.: Study on network traffic prediction techniques. In: International Conference on Wireless Communications, NETWORKING and Mobile Computing, pp. 1041–1044 (2005)
2. Voort, M.V.D., Dougherty, M., Watson, S.: Combining kohonen maps with arima time series models to forecast traffic flow. Transp. Res. Part C Emerg. Technol. **4**(5), 307–318 (1996)
3. Williams, B.M., Hoel, L.A.: Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results. J. Transp. Eng. **129**(6), 664–672 (2003)
4. Katsaros, D., Manolopoulos, Y.: Prediction in wireless networks by Markov chains. Wirel. Commun. IEEE **16**(2), 56–64 (2009)
5. Alarcon-Aquino, V., Barria, J.A.: Multiresolution FIR neural-network-based learning algorithm applied to network traffic prediction. IEEE Trans. Syst. Man Cybern. Part C **36**(2), 208–220 (2006)
6. Babiarz, R., Bedo, J.-S.: Internet traffic mid-term forecasting: a pragmatic approach using statistical analysis tools. In: Boavida, F., Plagemann, T., Stiller, B., Westphal, C., Monteiro, E. (eds.) NETWORKING 2006. LNCS, vol. 3976, pp. 110–122. Springer, Heidelberg (2006). https://doi.org/10.1007/11753810_10
7. Chabaa, S., Zeroual, A., Antari, J.: Identification and prediction of internet traffic using artificial neural networks. J. Intell. Learn. Syst. Appl. **2**(3), 147–155 (2010)
8. Cortez, P., Rio, M., Rocha, M., et al.: Multi-scale Internet traffic forecasting using neural networks and time series methods. Expert Syst. **29**(2), 143–155 (2012)
9. Liu, X., Fang, X., Qin, Z., et al.: A short-term forecasting algorithm for network traffic based on chaos theory and SVM. J. Netw. Syst. Manag. **19**(4), 427–447 (2011)
10. Wang, J., Wang, J., Zeng, M., et al.: Prediction of internet traffic based on Elman neural network. In: Control and Decision Conference, CCDC 2009, Chinese, pp. 1248–1252 (2009)
11. Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. IEEE Trans. Neural Netw. **5**(2), 157–166 (2002)
12. Wang, H., Hu, D.: Comparison of SVM and LS-SVM for regression. In: International Conference on Neural Networks and Brain, pp. 279–283 (2005)
13. Chen, Y., Yang, B., Meng, Q.: Small-time scale network traffic prediction based on flexible neural tree. Appl. Soft Comput. **12**(1), 274–279 (2012)
14. Lv, Y., Duan, Y., Kang, W., et al.: Traffic flow prediction with big data: a deep learning approach. IEEE Trans. Intell. Transp. Syst. **16**(2), 865–873 (2015)
15. Oliveira, T.P., Barbar, J.S., Soares, A.S.: Multilayer perceptron and stacked autoencoder for internet traffic prediction. In: Hsu, C.-H., Shi, X., Salapura, V. (eds.) NPC 2014. LNCS, vol. 8707, pp. 61–71. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44917-2_6
16. Huang, W., Song, G., Hong, H., et al.: Deep architecture for traffic flow prediction: deep belief networks with multitask learning. IEEE Trans. Intell. Transp. Syst. **15**(5), 2191–2201 (2014)
17. Zhuo, Q., Li, Q., Yan, H., Qi, Y.: Long short-term memory neural network for network traffic prediction. In: International Conference on Intelligent Systems and Knowledge Engineering (ISKE), pp. 1–6 (2017)

18. Gers, F.A., Schmidhuber, J.: Recurrent nets that time and count. In: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium, vol. 3, pp. 189–194 (2000)
19. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
20. Kang, D., Lv, Y., Chen, Y.Y.: Short-term traffic flow prediction with LSTM recurrent neural network. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–6 (2017)
21. Vidnerova, P., Neruda, R.: Evolving keras architectures for sensor data analysis. In: 2017 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 109–112 (2017)