Predicting Short-term Traffic Flow by Long Short-Term Memory Recurrent Neural Network

Yongxue Tian*†, Li Pan *†
*Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China
†National Engineering Laboratory for Information Content Analysis Technology, Shanghai, China
Email: {521370tyx, panli}@sjtu.edu.cn

***Abstract*—Intelligent Transportation System (ITS) is a significant part of smart city, and short-term traffic flow prediction plays an important role in intelligent transportation management and route guidance. A number of models and algorithms based on time series prediction and machine learning were applied to short-term traffic flow prediction and achieved good results. However, most of the models require the length of the input historical data to be predefined and static, which cannot automatically determine the optimal time lags. To overcome this shortage, a model called Long Short-Term Memory Recurrent Neural Network (LSTM RNN) is proposed in this paper, which takes advantages of the three multiplicative units in the memory block to determine the optimal time lags dynamically. The dataset from Caltrans Performance Measurement System (PeMS) is used for building the model and comparing LSTM RNN with several well-known models, such as random walk(RW), support vector machine(SVM), single layer feed forward neural network(FFNN) and stacked autoencoder(SAE). The results show that the proposed prediction model achieves higher accuracy and generalizes well.**

1. Introduction

As a significant part of smart city, intelligent transportation system (ITS) can effectively relieve the traffic congestion, reduce the air pollution and provide more secure traffic conditions. As a key technology in ITS, traffic flow prediction has gained more and more attention with the rapid development and deployment of ITS. The objective of traffic flow prediction is to provide accurate and timely traffic flow information for individual travelers, business sectors, and government agencies. However, the freeway transportation network is a complex system mixed with many other factors such as weather conditions, landform etc. Therefore, the short-term traffic flow is highly non-linear and stochastic, which makes it a huge challenge to be predicted accurately.

The evolution of traffic flow can be considered as a temporal and spatial process. The traffic flow of the ith observation station at the tth time interval is denoted by $f_{i,t}$. At time T, the task is to predict traffic flow $f_{i,T+1}$ at time $T+1$ based on the historical traffic flow sequence $F = \{f_{i,t}|i \in O, t = 1, 2, \ldots, T\}$, where O is the full set of observation stations. Traffic flow prediction can be classified into long-term, mid-term and short-term according to the length of prediction interval ΔT, which is defined as the interval between time T and $T+1$. For research and analysis, the Highway Capacity Manual [1] suggests to use 15-min as short-term prediction interval.

Short-term traffic flow prediction has attracted numerous attentions from worldwide researchers in the past decades. In general, the prediction methodology can be divided into two major categories, namely parametric approach and nonparametric approach. In the early period of research, parametric models such as time series models and Kalman filtering models were applied into traffic flow prediction. The model structures of parametric approaches are predetermined based on certain theoretical assumptions and the model parameters can be computed with empirical data. The most widely used parametric method is Autoregressive Integrated Moving Average(ARIMA) model, which assumes that the traffic condition is a stationary process. ARIMA method is also written as ARIMA(p,d,q), where p, d, q respectively represent the autoregressive, intrgrated and moving average polynomial orders. Levin and Tsao applied Box-Jenkins time series analysis to predict freeway traffic flow and found that ARIMA(0,1,1) model was most effective [2]. Due to the stochastic and nonlinear nature of traffic flow, parametric models cannot describe it accurately with analysis formulas. Therefore, more researchers have paid attention to nonparametric methods. In the domain of artificial intelligence based approaches, Support Vector Machine(SVM) or Support Vector Regression is considered as an effective and efficient algorithm. The essence of SVM is to map data into a high-dimensional feature space via a nonlinear relationship and then performs linear regression within this space. Castro-Neto *et al.* used Online Support Vector Regression (OL-SVR) to predict traffic flow under typical and atypical traffic conditions (such as holidays and traffic accident) [3]. Artificial Neural Network(ANN) is another popular countermeasure for traffic prediction due to its capability of handling multi-dimensional data, flexible model sturcture, strong generalization and learning ability as well as adaptability [4]. Vlahogianni *et al.* optimized the neural network with genetic approach and applied the model to short-term traffic flow prediction [5]. Lv *et al.* [6] and Huang *et al.* [7] used two deep learning models, i.e., Stacked Autoencoder (SAE) and Deep Belief Networks

978-1-5090-1893-2/15 $31.00 © 2015 IEEE
DOI 10.1109/SmartCity.2015.63

(DBN) to predict the traffic flow respectively. However, the above-mentioned nonparametric models require the length of the input historical data to be predefined and static and cannot automatically determine the optimal time lags.

In order to achieve higher prediction accuracy, this study proposed a model called Long Short-Term Memory Recurrent Neural Network (LSTM RNN), which can capture the nonlinearity and randomness of traffic flow more effectively, as well as overcome the issue of back-propagated error decay through memory blocks, and thus shows superior capability for time series prediction with long temporal dependency. With the ability to memorize long historical data and automatically determine the optimal time lags, the LSTM RNN achieves higher prediction accuracy and generalizes well with different prediction intervals.

The rest of this paper is organized as follows. Section 2 presents the LSTM RNN architecture and builds the model for traffic flow prediction. Experiments design and results analysis are given in Section 3. Finally, Section 4 concludes the paper and proposes the future work.

2. Traffic Flow Prediction Based on LSTM RNN

Among the numerous approaches for short-term traffic flow prediction, neural networks are fairly popular due to its capability of handling multi-dimensional data, flexible model sturcture, strong generalization and learning ability as well as adaptablity. In general, neural networks can be classified into two types, namely feed forward neural network (FFNN) and recurrent neural network (RNN). Both can be applied in short-term traffic flow prediction, and the accuracy closely depends on the length of input historical traffic flow data. The traditional FFNN, whose connections do not form cycles, only maps the current input vector to output vector but cannot memorize the earlier input data and determine the optimal time lags either. When FFNN is applied to short-term traffic flow prediction, the input data must be truncated into specific length. Therefore, the prediction results are not desirable. In contrast, the RNN, which allows cyclical or recurrent connections, can in principle maps the whole historical input data to each output. The key point is that the recurrent connections allow a memory of previous inputs to persist in the network's internal state, and thereby influence the network output. However, for standard RNN architecture, the influence of a given input on the hidden layer, and therefore on the network output, either decays or blows up exponentially as it cycles around the networks recurrent connections. This effect is often referred to in the literature as the vanishing gradient problem [8]. To address this problem, Hochreiter and Schmidhuber proposed Long Short-Term Memory(LSTM) architecture [9].

The primary objectives of LSTM RNN are to model long-term dependencies and determine the optimal time lags for time series problems. These features are especially desirable for short-term traffic flow prediction, due to the lack of priori knowledge on the relationship between prediction results and the length of input historical data. The LSTM RNN architecture is composed of one input layer, one recurrent hidden layer whose basic unit is memory block instead of traditional neuron node, and one output layer. Memory blocks are a set of recurrently connected subnets. Each block contains one or more self-connected memory cells and three multiplicative units: the input, output and forget gates, which provide continuous analogues of write, read and reset operations on the cells. The multiplicative gates allow LSTM memory cells to store and access information over long periods of time, thereby mitigating the vanishing gradient problem. For example, as long as the input gate remains closed, the activation of the cell will not be overwritten by the new inputs arriving in the network, and can therefore be made available to the net much later in the sequences, by opening the output gate. Figure 1 provides an illustration of the LSTM RNN prediction model architecture with one memory block.

Suppose that the input historical traffic flow sequence is denoted as $\mathbf{x} = (x_1, x_2, \ldots, x_T)$, the LSTM RNN computes the hidden vector sequence $\mathbf{h} = (h_1, h_2, \ldots, h_T)$ and the output predicted traffic flow sequence $\mathbf{y} = (y_1, y_2, \ldots, y_T)$ by iterating the following equations:

$$h_t = \mathcal{H}(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \tag{1}$$

$$y_t = W_{hy}h_t + b_y \tag{2}$$

where the W term denote weight matrices(e.g. W_{xh} is the input-hidden weight matrix), b term denote bias vectors(e.g. b_h is hidden bias vector) and $\mathcal{H}$ is the hidden layer function, which is implemented by the following composite function:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{3}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{4}$$

$$c_t = f_t c_{t-1} + i_t g(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{5}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \tag{6}$$

$$h_t = o_t h(c_t) \tag{7}$$

where $\sigma(\cdot)$ is the standard logistic sigmoid function defined in Eq.(8), $g(\cdot)$ and $h(\cdot)$ are the transformations of function $\sigma(\cdot)$ whose range are [-2,2] and [-1,1] respectively. The i,f,o and c are respectively the input gate, forget gate, output gate and cell activation vectors, all of which are the same size as the hidden vector h.

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{8}$$

$$g(x) = \frac{4}{1+e^{-x}} - 2 \tag{9}$$

$$h(x) = \frac{2}{1+e^{-x}} - 1 \tag{10}$$

There are two well-known algorithms to train the RNN model: Back Propagation Through Time(BPTT) [9] and Real Time Recurrent Learning(RTRL) [10]. This study focuses on BPTT algorithm since it is both conceptually simpler and more efficient in computation time. BPTT begins

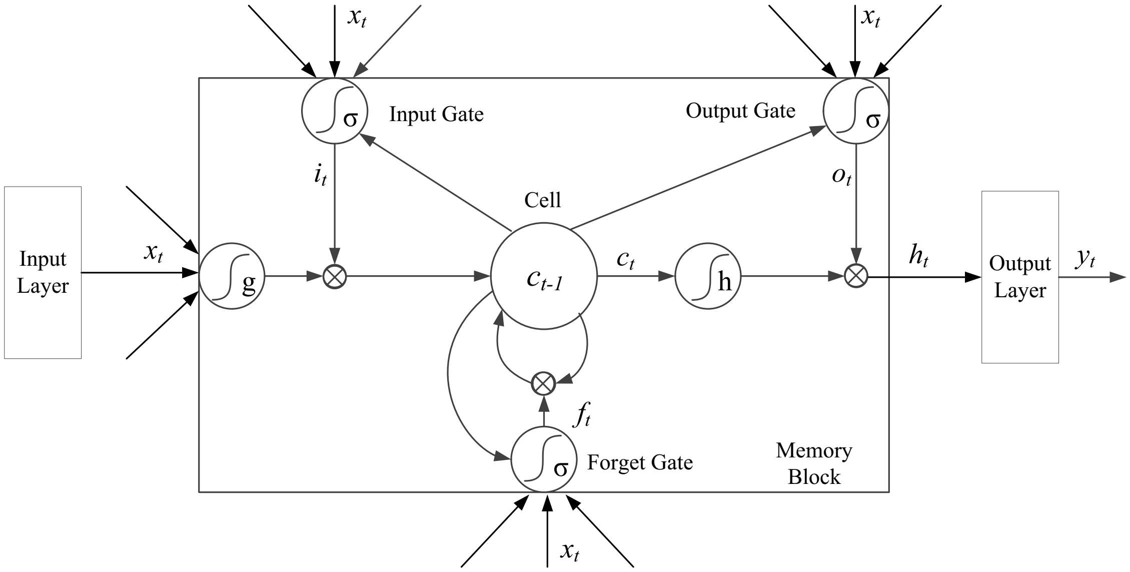

Figure 1. LSTM RNN prediction model architecture with one memory block

by unfolding a recurrent neural network into feed forward neural network through time. Like standard BP algorithm, BPTT consists of a repeated application of the chain rule. The subtlety is that, for recurrent networks, the loss function depends on the activation of the hidden layer not only through its influence on the output layer, but also through its influence on the hidden layer at the next time step. The squared error is used as the loss function, which is given by the following equation:

$$e_t = \sum (y_t - p_t)^2 \tag{11}$$

where e_t represents the error, and p_t represents the predicted traffic flow value. The detailed mathematical formulas of the BPTT are introduced in [9].

In order to build the short-term traffic flow prediction model based on LSTM RNN, four key hyperparameters must be determined, including the size of input layer, the number of hidden layers, the number of hidden units(memory blocks for LSTM RNN) in each hidden layer and the size of output layer, which are denoted as nInput, nHiddens, nUnits and nOutput respectively in Table 1. The input historical data length is always equal to the size of input layer, which is defined from 1 to 12 in our model for different experiments. The number of hidden layers is assigned to 1 without deep architecture in this study, due to the fact that training LSTM RNN is quite complicated and time-consuming. Therefore, we take it as our future work. The number of units in each hidden layer is assigned in range from 5 to 40 with a step of 5. For simplicity, the units of each hidden layer are the same. The size of output layer is 1, indicating the traffic flow of next time step. Grid search method is used to obtain the optimal parameters. The internal connection of LSTM RNN is full recurrent, which is easy to implement and performs very well. A machine learning library written in Python called Pybrain [11] is appied in the implementation of the model.

TABLE 1. LSTM RNN MODEL HYPERPARAMETERS

Hyperparameters	Values
nInput	[1,12]
nHiddens	1
nUnits	{5,10,15,20,25,30,35,40}
nOutput	1

3. Experiments

3.1. Data Description and Experiments Design

In this study, data collected from the Caltrans Performance Measurement System (PeMS) [12] are applied to train and test the model. PeMS is one of the most widely used dataset in traffic flow prediction. The traffic data are collected every 30s from numerous individual detectors, which are deployed statewide in freeway systems across California. Then, the data are aggregated into 5-min interval for each detector and are accessible on the Internet for research and route guide reference. Traffic flow data have a obvious cycle of one day, and the traffic flow patterns of workdays and holidays are quite different. Due to that transportation management on workdays is more important, this study only focuses on the traffic flow prediction on workdays. In this paper, we randomly selected 30 observation stations from 6 freeways in PeMS and collected the traffic flow data on all 249 workdays in 2014. The data of the first 200 workdays are used as the training set, while the data of the later 49 workdays are used as the test set. The prediction intervals of the experiments are 15-min, 30-min, 45-min and 60-min, therefore, the raw traffic flow data from PeMS should be aggregated into the corresponding time interval. Notice that we only use the traffic flow data as the input for prediction without considering other factors, such as weather conditions, accidents, and other traffic parameters(density and speed etc.), that have a relationship the traffic flow.

Several experiments are designed to validate the effectiveness of the proposed LSTM RNN model in short-term traffic flow prediction. Four classic prediction models are chosen for comparison, which are Random Walk(RW), Support Vector Machine(SVM) [3], single hidden layer Feed Forward Neural Network(FFNN) and Stacked Autoencoder(SAE) [6]. As the baseline of our experiments, RW model is the simplest prediction model, which uses the value of current traffic flow to predict the traffic flow of next time step, denoted by equation $\hat{f}(T+1) = f(T)$. SVM model has excellent performance in both classification and regression problems through building hyperplane in the high-dimensional feature space created via kernel trick. In our experiments, Radial Basis Function(RBF) kernel is used in the SVM prediction model. As one of the most classical neural network models, single hidden layer FFNN also serves as a contrast in our experiments. Conventional BP algorithm is used to train the model. Deep neural network models are popular recently because of their excellent performance. Therefore, we also choose SAE to predict short-term traffic flow. A greedy layerwise unsupervised learning is used to pretrain the model and BP algorithm is used to do fine-tuning.

In order to prove that the proposed model is superior to the model mentioned above, three aspects of the LSTM RNN are tested via the experiments: the prediction accuracy, the memory ability of long historical data and the generalization capability with different prediction intervals. The experiments results are as follows.

3.2. Prediction Accuracy Comparison

The prediction accuracy of short-term traffic flow can be assessed by two commonly used metrics, i.e., Mean Absolute Percentage Error (MAPE) which evaluates the relative error and Root Mean Square Error(RMSE) which evaluates the absolute error. They are defined by Eq.(12) and Eq.(13).

$$\text{MAPE}(f, \hat{f}) = \frac{1}{n}\sum_{i=1}^{n}\frac{|f_i - \hat{f}_i|}{f_i} \tag{12}$$

$$\text{RMSE}(f, \hat{f}) = \left[\frac{1}{n}\sum_{i=1}^{n}\left(|f_i - \hat{f}_i|\right)^2\right]^{\frac{1}{2}} \tag{13}$$

where f is the observation value of traffic flow, and $\hat{f}$ is the prediction value.

Figure 2 shows the 15-min interval traffic flow comparison of observation values collected from No.1111531 observation station on I8-E freeway in California and prediction values obtained by LSTM RNN model. Intuitively, the prediction results are fairly good, and most of the fluctuations are captured. Detailedly, the prediction results of the five models and their optimal model parameters are listed in Table 2, in which nInput represents the input size of the model, nHiddens represents the number of hidden layers and nUnits represents the number of units in each hidden layer. As we can see from Table 2, compared with other four models, both MAPE and RMSE of LSTM RNN are lowest. The two metrics of LSTM RNN and SVM are really close, especially for MAPE, whose values are 6.49% and 6.58% respectively. Nevertheless, the optimal input sizes of SVM and LSTM RNN are 8 and 1 respectively, which means that the SVM model is much more complicated. When complexity of SVM is similar to that of LSTM RNN, i.e., the input size of SVM is 1, its MAPE and RMSE are 10.32% and 81.66 respectively. These results are far from those of LSTM RNN. Summing up the above analysis, LSTM RNN model takes its advantages of memorizing long historical data and can achieve higher prediction accuracy even if the model is quite simple. Therefore, the proposed model is effective in short-term traffic flow prediction.

Then, we investigate the effect of hidden units on the prediction results, as illustrated in Figure 3. From Figure 3, we can see that the MAPE and RMSE fall with the increase of hidden units when the number of hidden units is less

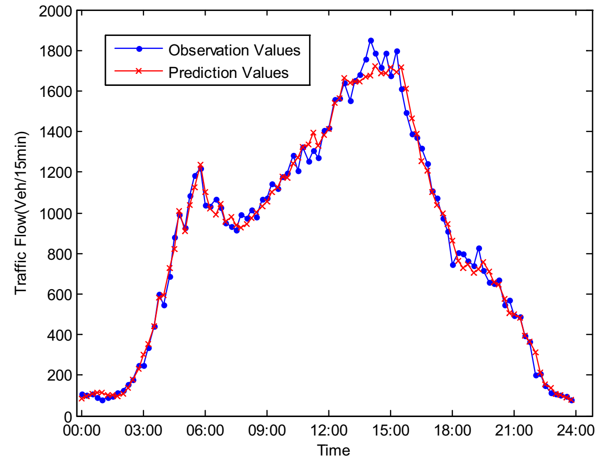

Figure 2. Comparison of observation and prediction traffic flow

TABLE 2. 15-MIN INTERVAL PREDICTION RESULTS

Models	Metrics		Optimal Parameters		
	MAPE(%)	RMSE	nInputs	nHiddens	nUnits
RW	10.39	81.79	1	NA	NA
SVM	6.58	53.10	8	NA	NA
FFNN	9.72	63.38	7	1	30
SAE	7.63	55.84	10	3	40
LSTM RNN	**6.49**	**50.94**	1	1	20

than 20, and remain stable or slightly rise up when the number of hidden units is equal or greater than 20. It can be concluded that the prediction preformance is closely related to the complexity of LSTM RNN. When the model is too simple, prediction results are poor. When the model becomes more complicated, the prediction error declines more slowly or even slightly grows.

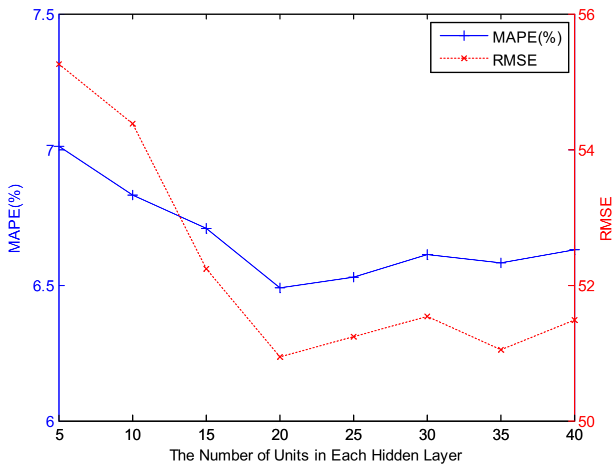

Figure 3. Prediction results with the change of hidden units

Note that, in order to alleviate the influence of random initialization for the model, we conducted each experiment for three times, and used the optimal results to make comparisons. In addition, the prediction results of other observation stations and freeways are similar.

3.3. The Memory Ability of LSTM RNN

The excellent performance of LSTM RNN for short-term traffic flow prediction mainly benefits from the memory ability of LSTM RNN. For purpose of verifying the ability of LSTM RNN to memorize long historical data, the MAPE and RMSE of each model with the change of historical data length are compared. The input historical data length ranges from 1 to 12 in our experiments with 15-min prediction interval, i.e., the past three hours. Note that the input historical data length is always equal to the input size of each model. The five models' MAPE and RMSE are illustrated in Figure 4 and Figure 5. The trends of both are similar. For RW model, the valid input size is always 1, therefore, the curves of MAPE and RMSE are approximately straight lines parallel to the horizontal axis . For SVM, FFNN and SAE model, with the increase of input data length, MAPE and RMSE decrease rapidly at the very start. When the input data length reaches 5, the decline rates of both metrics slow down. When the length reaches 8, MAPE and RMSE almost remain unchanged. It means that, for 15-min predictuon interval, the traffic flow in the past one hour has a great impact on the current traffic flow, and impact of the last two hours is retatively small, while the past three hours has little influence on the current traffic flow. It is in accordance with our common senses. Nevertheless, for LSTM RNN model, both MAPE and RMSE are relatively stable at a low level, only with minor fluctuations. As we can see from the figures, although the input data length is 1, LSTM RNN can also achieve pretty good traffic flow prediction results. It can be inferred that the model can memorize the earlier inputs through the recurrent connected memory blocks. Therefore, we can conclude that is proper to model long-term dependencies and determine the optimal time lags dynamically, which leads to the desirable results of short-term traffic flow prediction.

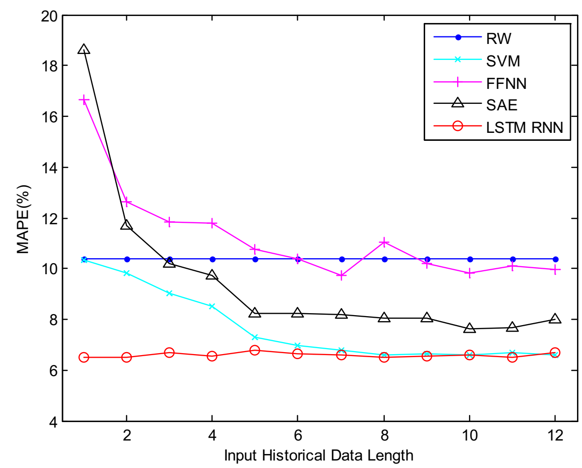

Figure 4. Five models' MAPE with the change of historical data length

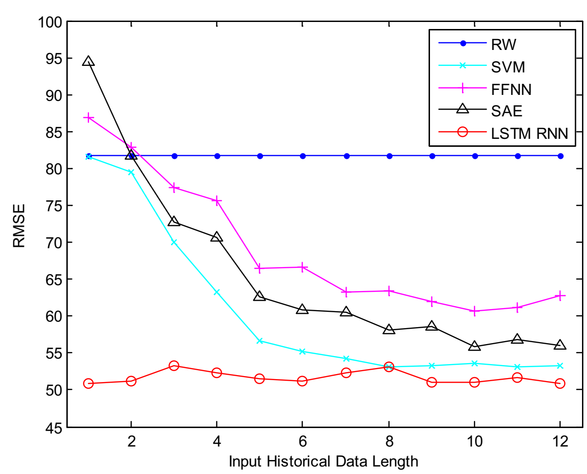

Figure 5. Five models' RMSE with the change of historical data length

3.4. The Generalization Capability of LSTM RNN

The generalization capability is an important evaluation criterion for a prediction model. In this study, we evaluated

TABLE 3. PREDICTION RESULTS COMPARISON OF THE MODELS WITH DIFFERENT INTERVALS

Models	15-min		30-min		45-min		60-min	
	MAPE(%)	RMSE	MAPE(%)	RMSE	MAPE(%)	RMSE	MAPE(%)	RMSE
RW	10.39	81.79	14.57	217.32	20.70	435.92	26.86	731.51
SVM	6.58	53.10	6.16	102.12	6.98	175.35	7.36	252.21
FFNN	9.72	63.38	11.65	122.77	11.82	203.36	12.98	285.84
SAE	7.63	55.84	7.61	109.66	8.58	179.39	8.97	254.57
LSTM RNN	**6.49**	**50.94**	**5.80**	**95.80**	**6.37**	**159.44**	**6.25**	**233.01**

the generalization capability with different prediction intervals of LSTM RNN through a series of experiments, also compared with other four models. The prediction intervals used are 15-min, 30-min, 45-min and 60-min and the prediction results are showed in Table 3. From Table 3, we can see that the MAPE and RMSE of RW model rise rapidly with the increase of input historical data length. So, RW model is quite sensitive to prediction interval and of bad generalization capability. The other four models are based on machine learning, and relatively stable with different prediction intervals. Nevertheless, the MAPE and RMSE of LSTM RNN are all lowest among the four machine learning based models with different prediction intervals, which demonstrate the excellent generalization capability of LSTM RNN.

4. Conclusion

In this paper, a model called LSTM RNN is proposed for short-term traffic flow prediction. LSTM RNN is able to memorize long historical input data and automatically determine the optimal time lags, which is desirable in short-term traffic flow prediction. In order to validate the effectiveness of LSTM RNN model, the data collected from PeMS are applied in our designed experiments and comparisons are made with other four classic prediction models, i.e., RW, SVM, FFNN and SAE. The experiments results show that both MAPE and RMSE are lowest with different prediction intervals, which prove that the proposed model can achieve higher accuracy and generalize well. This study mainly considers temporal influence on traffic flow, but does not take other factors into account, such as spatial impact from neighbor observation stations, weather conditions etc. Therefore, other complicated factors will be considered in the future work for purpose of higher prediction accuracy.

Acknowledgments

The authors would like to appreciate the funding support from National Key Technology Research and Development Program of China (2014BAG01B02) and the data resources from Caltrans Performance Measurement System.

References

[1] "Highway capacity manual," *Transportation research board, National Research Council, Washington, DC*, vol. 113, 2000.

[2] M. Levin and Y.-D. Tsao, "On forecasting freeway occupancies and volumes (abridgment)," *Transportation Research Record*, no. 773, 1980.

[3] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert systems with applications*, vol. 36, no. 3, pp. 6164–6173, 2009.

[4] M. Karlaftis and E. Vlahogianni, "Statistical methods versus neural networks in transportation research: Differences, similarities and some insights," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 3, pp. 387–399, 2011.

[5] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Optimized and meta-optimized neural networks for short-term traffic flow prediction: a genetic approach," *Transportation Research Part C: Emerging Technologies*, vol. 13, no. 3, pp. 211–234, 2005.

[6] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, no. 2, pp. 865–873, 2015.

[7] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 5, pp. 2191–2201, 2014.

[8] A. Graves *et al.*, *Supervised sequence labelling with recurrent neural networks*. Springer, 2012, vol. 385.

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] F. Chang, L.-C. Chang, H.-L. Huang *et al.*, "Real-time recurrent learning neural network for stream-flow forecasting," *Hydrological Processes*, vol. 16, no. 13, pp. 2577–2588, 2002.

[11] T. Schaul, J. Bayer, D. Wierstra, Y. Sun, M. Felder, F. Sehnke, T. Rückstieß, and J. Schmidhuber, "Pybrain," *The Journal of Machine Learning Research*, vol. 11, pp. 743–746, 2010.

[12] "Caltrans, performance measurement system (pems)," 2015. [Online]. Available: http://pems.dot.ca.gov