DOI: 10.11916/j.issn.1005-9113.2018007

Route Temporal-Spatial Information Based Residual Neural Networks for Bus Arrival Time Prediction

*Chao Yang, Xiaolei Ru and Bin Hu**

(Key Laboratory of Road and Traffic Engineering of the Ministry of Education, School of Transportation Engineering, Tongji University, Shanghai 201804, China)

Abstract: Bus arrival time prediction contributes to improving the quality of public transport services. Passengers can arrange departure time effectively if they know the accurate bus arrival time in advance. We proposed a machine-learning approach, RTSI-ResNet, to forecast the bus arrival time at target stations. The residual neural network framework was employed to model the bus route temporal-spatial information. It was found that the bus travel time on a segment between two stations not only had correlation with the preceding buses, but also had common change trends with nearby downstream/upstream segments. Two features about bus travel time and headway were extracted from bus route including target section in both forward and reverse directions to constitute the route temporal-spatial information, which reflects the road traffic conditions comprehensively. Experiments on the bus trajectory data of route NO. 10 in Shenzhen public transport system demonstrated that the proposed RTSI-ResNet outperformed other well-known methods (e.g., RNN/LSTM, SVM). Specifically, the advantage was more significant when the distance between bus and the target station was farther.

Keywords: bus arrival time prediction; route temporal-spatial information; residual neural network; recurrent neural network; bus trajectory data

CLC number: U121 **Document code:** **Article ID:**

1 Introduction

1.1 Background

In the operation of public transportation system, it is vital to provide reliable and comfortable service for passengers. The key to solve the problem of bus operation and scheduling is to maintain the punctuality and stability of the service, so that passengers can effectively arrange their departure time and make informed decisions for travel choices. Meanwhile, when the normal operation is interfered, operators may take appropriate measures according to the real-time vehicle circumstances to adjust schedules and departure intervals (e.g., increase or reduce the speed, or prolong parking of some sites), so as to make the operation management more effective. The study on bus arrival time prediction method aims at improving the bus travel time prediction precision. Precisely predicted arrival time can help bus operators guarantee bus punctuality and reduce the passenger waiting time, thereby enhancing the bus attraction and competitiveness as well as promoting the intelligent transportation system (ITS) implementations.

Due to the stochastic property of traffic conditions and the subjectivity of drivers, it is difficult to accurately predict bus arrival time. Therefore, the modeling for bus arrival time prediction is not a straightforward task. However, the wide application of information technologies, such as radio frequency identification, smart card, and automatic vehicle location, has generated plenty of high-precision data and boosted bus arrival time prediction researches based on different kinds of data sources.

Received 2018-12-15.
Sponsored by the Transportation Science and Technology Planning Project of Henan Province, China (Grant No. 2019G-2-2).
Corresponding author. E-Mail: 329167869@qq.com.

1.2 Literature Review

Various emerging and sophisticated algorithms have been designed and applied to solve the problem of bus arrival time prediction. According to Ma et al.[1], traffic prediction approaches have been transformed to intelligent computing from traditional statistical models. Compared with traditional models, the machine-learning technology is more competent to process missing and extreme data, and needs little prior knowledge. Therefore, machine-learning approaches are often used to dispose multi-dimensional features with non-linear relationship, among which Artificial Neural Network (ANN) and Support Vector Machines (SVM) have achieved relatively good results for bus arrival time predictions.

1.2.1 ANN

ANN has become a popular tool in the domain of traffic prediction because of its flexible multi-layers structure, capability of handling high-dimensional data, and strong learning ability and generalization[1-2]. Ji et al.[3] established the prediction model of bus arrival time based on wavelet neural network and used the particle swarm algorithm to optimize the model parameters.

1.2.2 SVM

SVM transforms low-dimensional nonlinear problems into linear problems in a high-dimensional space problem by means of nonlinear transformation and solves high-dimensional problems by constructing kernel functions. SVM-based forecasting methods have been demonstrated with better performance than baseline in several studies[4-5].

A class of machine-learning techniques, named deep learning, is developing rapidly in recent years, which has significantly improved in object detection, visual object recognition, speech recognition, and many other domains such as genomics and drug discovery[6]. The research on using deep learning techniques to predict bus arrival time directly is scant. However, the applications on other domains especially sequence task have considerable reference significance.

1.2.3 Recurrent Neural Networks (RNNs)/Long Short-Term Memory (LSTM) Networks

RNNs have been verified to be adept at forecasting the next word in the sequence or the next character in the text[6-9]. As expounded in Ref. [6], in order to correct gradient explosion in RNNs, LSTM networks were proposed by adding special hidden units called forgotten gate and memory gate, whose natural behavior is to filter invalid ingredient and remember valid ingredient of inputs automatically for a long time. Ma et al.[1] established the LSTM network model to predict traffic speed based on coil data.

1.2.4 Convolutional Neural Networks (CNN)

The powerful performance of CNN has been confirmed in many tasks where labelled samples are relatively sufficient. Zhang et al.[10] proposed a CNN-based model to predict the crowd flow in each region of a city, whose results demonstrated that the model improves accuracy significantly. Gebru et al.[11] used CNN and Google Street View to estimate the demographic makeup of the USA. Sainath et al.[12] proposed an architecture which combined the Conv with LSTM networks on a variety of large vocabulary tasks. Shi et al.[13] proposed the convolutional LSTM (ConvLSTM) for precipitation nowcasting, specifically predicting the future rainfall intensity over a relatively short period of time.

With the power of deep learning, we tried to apply this technology to bus arrival time predication in this paper. We were faced with and solved three main difficulties: a) which features are useful for bus arrival time prediction and how to extract these information from bus trajectory data; b) how to construct these features to conform to the input form of deep learning models and take advantage of the models; and c) how to adjust the model architecture to make it suitable for the problem of bus arrival time prediction.

The rest of this paper is organized as follows: Section 2 presents a formal description of the problem of bus

arrival time prediction and an analysis of the temporal-spatial correlation between features and bus travel time prediction; Section 3 reveals the construction of the proposed model and the input; Section 4 provides a case study in Shenzhen bus system and analyzes the prediction performance of the proposed model compared with baseline models; and Section 5 gives the conclusion and discussion.

2. Formulation of Bus Arrival Time Prediction Problem

Definition 1: Section s is the part of bus route between the s-th station and $s+1$-th station.

Definition 2: Shift i is the i-th running on bus route in one day.

Based on the above definition, two features are described by brace in Fig.1.

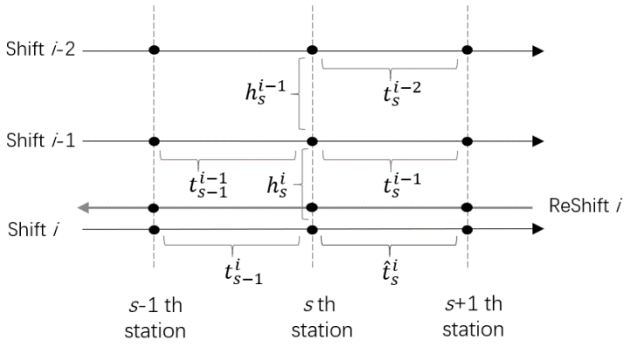


Fig.1 Symbols description in bus arrival time prediction problem

(a) t_s^i is the bus travel time of shift i in section s.

$$t_s^i = T_{s+1}^i - T_s^i \quad (1)$$

where T_s^i and T_{s+1}^i are bus arrival time of shift i at s-th and $s+1$-th stations, respectively.

(b) h_s^i is the headway of shift i at s-th station

$$h_s^i = T_s^i - T_s^{i-1} \quad (2)$$

On the premise that the truth value T_s^i has been obtained, the predict value of arrival time $\hat{T}_{s+1}^i$ can be determined according to the predict value of travel time $\hat{t}_s^i$.

$$\hat{T}_{s+1}^i = T_s^i + \hat{t}_s^i \quad (3)$$

Thus, the problem of bus arrival time prediction at a target station can be transformed to the problem of bus travel time prediction in a target section.

Yu et al.[4] suggested that the travel time of the preceding bus that has just arrived at the station can be used to reflect the traffic conditions, and the headway can be used to reflect the timeliness of the bus travel times of the preceding buses.

In our opinion, the headway and the travel time in a target section only utilize temporal information. It is easy to imagine that during evening peak periods, the bus travel time on the whole route including the target section will increase drastically. An actual observation is shown in Fig.2. The change trend of the bus travel time in the target section had correlation with its adjacent sections and even with all other sections on the whole route.

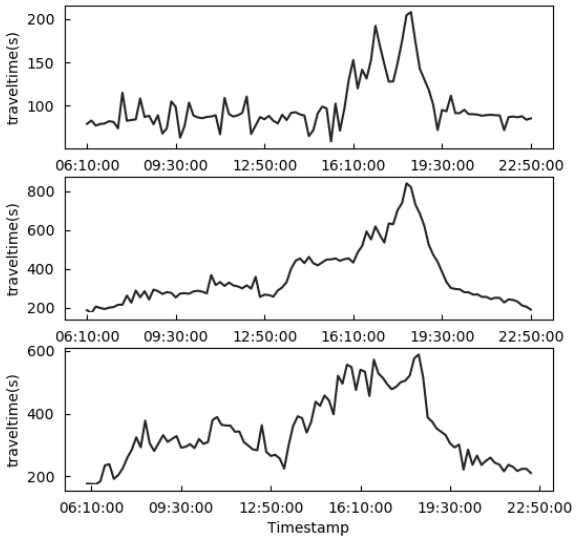


Fig.2 Bus travel time in every five minutes during route operation hours (**Top**: the upstream adjacent section in forward direction; **Center**: the target section in forward direction; **Bottom**: the target section in reverse direction)

Fig.2 shows the common change tendency in a certain extent. At 7 am, the travel time began to rise and fluctuate. Until about 4 pm, the travel time experienced a substantial increase and peaked at 6 pm. After 7:30 pm, the travel time returned to normal.

Thus, considering the spatial-temporal information, we extended to extract the two features from the whole

route. Let t^i be a set of bus travel time in continuous sections of route by shifts i.

$$\boldsymbol{t}^i = \{t_s^i | s = 1,2,\dots,S-1\}$$

where S is the total number of stations on the route.

Likewise, let h^i be a set of headway in continuous stations of route by shifts i.

$$\boldsymbol{h}^i = \{h_s^i | s = 1,2,\dots,S-1\}$$

So far, all the information that has been discussed is confined to one direction of the route.

Definition 3: Re-Shift i is the last running in the opposite direction to shift i on the route, which has passed the target section before shift i, as shown in Fig.1 by solid line with left arrows.

Although the direction is different, the majority traveling tracks of the Shift and Re-Shift are overlapping. The traffic condition of the section in reverse direction is similar to the forward direction in most time period, as shown in Fig.2. Let $\boldsymbol{t}_r^i$ and $\boldsymbol{h}_r^i$ represent a set of bus travel time and headway by Re-Shift i, respectively.

Then, the problem of bus travel time prediction can be inducted as follows:

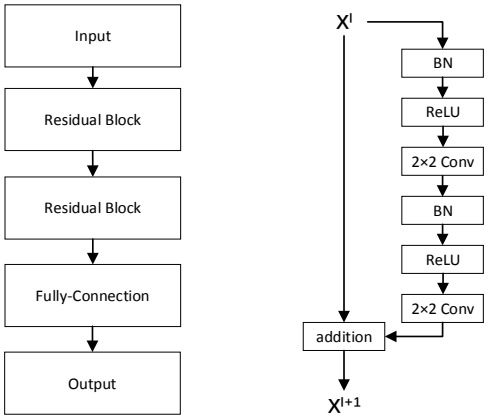
Problem 1: Given the historical observations $\{\boldsymbol{t}^i, \boldsymbol{h}^i | i = n-j, n-j+1, \dots, n\}$ and $\{\boldsymbol{t}_r^i, \boldsymbol{h}_r^i | i = n-k, n-k+1, \dots, n\}$, predict t_m^n, where n is the target shift and m is the target section.

3 Residual Networks Based on Route Spatial-Temporal Information

Fig.3 shows the architecture of RSTI-ResNet, which is a framework modeling the route spatial-temporal information.

3.1 Route Spatial-Temporal Information

A j×s matrix of bus travel time was constructed by historical observations $\{\boldsymbol{t}^i | i = n-j, n-j+1, \dots, n\}$, as shown in Fig.4. The elements in a row of the matrix are the bus travel time of each section on route generated by a certain shift. The elements in a column of the matrix are the bus travel time of each shift passing a certain section. Likewise, a $j \times s$ matrix of bus travel time was constructed by historical observations $\{\boldsymbol{h}^i | i = n-j, n-j+1, \dots, n\}$. Then, we combined these two matrixes into a tensor $\boldsymbol{X_f} \in \boldsymbol{R}^{j \times s_f \times 2}$, as shown in Fig.4, where $\boldsymbol{R}$ represents real number space.



(a) RSTI-ResNet (b) Residual Block

Fig.3 Left: the overall scheme of the RSTI-ResNet; Right: Residual Block

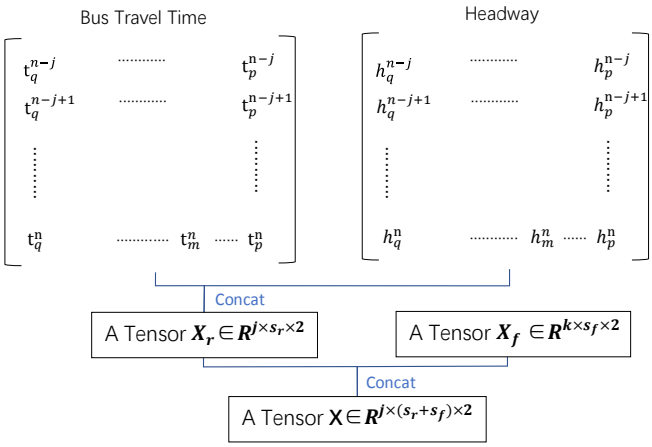


Fig.4 Constitution of route temporal-spatial information where $1 \le q \le m \le p \le S-1$ and $s_r = p - q$. If $k < j$, the missing element of X_f in the dimensionality of k is filled by 0.

The reverse direction information was constructed to a tensor $\boldsymbol{X_r} \in \boldsymbol{R}^{k \times S_r \times 2}$ in the same way by the historical observations $\{\boldsymbol{t}_r^i, \boldsymbol{h}_r^i | i = n-k, n-k+1, \dots, n\}$.

Finally, a tensor $\boldsymbol{X} \in \boldsymbol{R}^{j \times (s_r + s_f) \times 2}$ was achieved by

concatenating $\boldsymbol{X_r}$ to $\boldsymbol{X_f}$, corresponding to the top check in Fig.3(a), as an input fed to the proposed model. The rows in tensor $\boldsymbol{X}$ represent the route spatial information while the columns denote the route temporal information.

Tentatively, a fraction of values in tensor $\boldsymbol{X}$ which had not been observed were taken place by the value 0. Take the bus travel time in downstream sections of the target section (i.e., $\{t_s^n | s \geq m\}$) as an example. According to the precondition, the shift n has just arrived at station m when the bus travel time in section m would be predicted.

3.2 Structure

The structure of the proposed model framework consisting of two components, i.e., residual block and fully-connection, is inspired by the philosophy of Deep Residual Network proposed in Ref. [14].

Residual Block. In order to obtain the spatial correlation between the target section and other sections and the temporal correlation between the target shift and preceding shifts, it is necessary to design a multilayer CNN. A stack of convolutions can have further effects to capture the wide dependencies that exceed the size of their filters.

Residual learning was employed in the proposed model, which can gain accuracy from considerably increased depth. Residual learning has shown state-of-the-art results on multiple challenging recognition tasks, including object detection, image classification, localization, and segmentation[14]. Each residual block can be expressed in a general form[15] as follows:

$$\boldsymbol{x_{l+1}} = h(x_l) + F(\boldsymbol{x_l}, W_l) \tag{4}$$

where $\boldsymbol{x_l}$ and $\boldsymbol{x_{l+1}}$ are input and output of the l-th block; $h(x_l) = x_l$ is an identity mapping; W_l includes all learnable parameters in this unit; and F is the residual function, e.g., a stack of two 2×2 convolution layers with pre-activation: Batch Normalization[16] and ReLU[17], as shown in Fig.3(b).

In the convolution layer, a filter handles the input $\boldsymbol{X_l^c}$ from upper layer as

$$\boldsymbol{X_l^{c+1}} = W_l^c * \boldsymbol{X_l^c} \tag{5}$$

where * denotes the convolutional operation and $\boldsymbol{X_l^{c+1}}$ is the output of the c-th convolution in the l-th unit.

Fully Connection. The network ends with a 1-way fully-connected layer with linear. Ultimately, the output of full-connected layer is taken as the predict value of bus travel time.

Our RSTI-ResNet was trained to predict t_m^n by minimizing mean squared error between the true value and the predicted value with l2-regularizer as follows:

$$L(\hat{t}_m^n; \theta) = \|t_m^n - \hat{t}_m^n\|_2^2 + \lambda\|\theta\|_2 \tag{6}$$

where θ are all learnable parameters in the proposed model.

4 Experiments

Dataset. The trajectory data we used was taken from Shenzhen bus system, from Nov. 1st in 2015 to Apr. 30th in 2016. An all-day track of a bus running on route NO. 10 is shown in Fig.5, which presents the shape of the route. The biggest red triangle is the target station. The part of route enclosed by the dashed box is the target section.

From Fig.5, it can be seen that the target section is the 10-th section in the forward direction and the 5-th section in the reverse direction. In our experiment, nine continue shifts including the target shift, the 8-th, 9-th, 10-th sections in the forward direction while night continue re-shifts, and the 5-th section in the reverse direction were selected to provide the route temporal-spatial information, meaning $\{\boldsymbol{s_r}, \boldsymbol{s_f}, j, k\}$ is equal to $\{3,1,9,9\}$. Thus, the size of inputs to RTSI-ResNet is 9×4×2.

Data Preprocessing. Except the actual bus arrival time at each station, our raw bus trajectory data recorded current time, speed, longitude, latitude, etc. A set of rules was provided to judge the bus arrival time from the raw bus trajectory data.

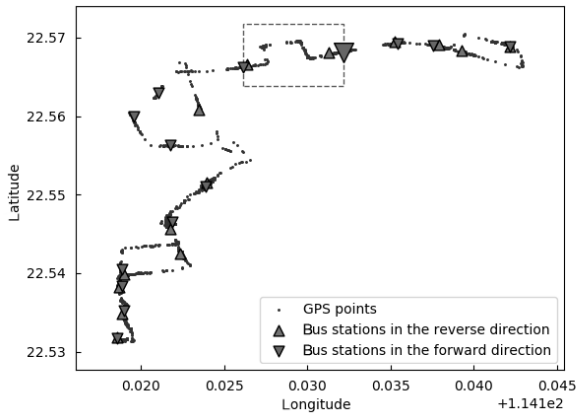


Fig.5 An all-day track of a bus running on route NO. 10 (The spots are GPS points projection; and the downward and upward triangles are the bus stations in the forward and reverse directions, respectively)

(a) Scenario One: there are one or more GPS points in the platform area, with which the value of the recorded bus speed is 0 km/h, as shown in Fig.6(a). Then the current time recorded on the GPS point, which is the closest to the station coordinate, was selected as the actual bus arrival time.

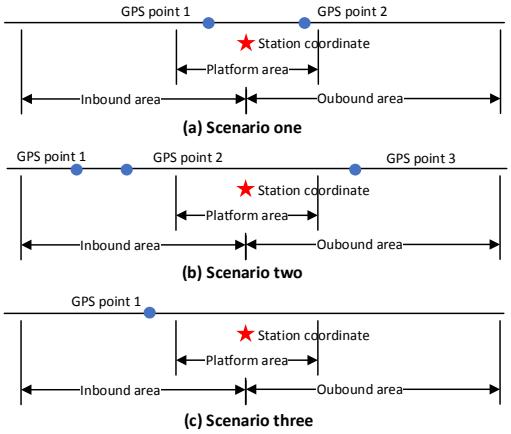


Fig.6 Scenarios of GPS points located near the bus station

(b) Scenario Two: there is no GPS point that satisfies the condition in scenario one, and one or more GPS points are in inbound area and outbound area, as shown in Fig.6(b). Then two GPS points which are the closest to station coordinate in inbound and outbound area are selected, e.g., GPS point 2 and GPS point 3 in Fig.6(b).

The process of bus entering and leaving the station can be divided into three parts, as shown in Fig.7, where $V_{GP2}(V_{GP3})$ and $t_{GP2}(t_{GP3})$ are the bus speed and current time recorded with GPS point 2(3).

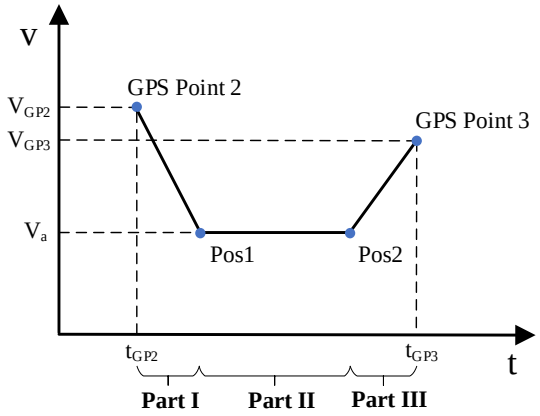


Fig.7 Process of bus entering and leaving the station

1) Part I: Inbound and deceleration. The bus at the GPS point 2 decelerates uniformly and runs ahead to an unknown position (Pos1). V_a is the bus speed at Pos1.

2) Part II: Stop or pass station with a low speed. The bus keeps speed V_a and runs to an unknown position (Pos2). The speed V_a can be equal to 0 km/h if there are passengers getting on or off then the bus stops, or it is greater to 0 km/h if there is no passenger getting on or off then the bus passes platform with a low uniform speed. If $V_a = 0\ km/h$, then Pos2 is Pos1 and $S_2 = 0km$.

3) Part III: Outbound and acceleration. The bus at Pos2 accelerates uniformly and runs ahead to the GPS point 3.

Then, a model was established to obtain the running time t_1:

$$min|s_d - (s_1 + s_2)| \tag{7}$$

$$t_1 + t_2 + t_3 = t \tag{8}$$

$$s_1 + s_2 + s_3 = s \tag{9}$$

where t_i and S_i are the bus running time and distance in part i, respectively. Formula (7) makes the Pos2 close to the station coordinate as much as possible. s_d is the distance between GPS point 2 and station coordinate. Formula (8) constraints the duration in the

process of entering and leaving station $t = t_{GP3} - t_{GP2}$. Formula (8) constraints the total running distance in the process. s is the distance between GPS point 2 and GPS point 3. The relational expressions among velocity, time, and distance are not introduced here. After solving the model, the arrival time is expressed as $t_{GP2} + t_1$.

(c) Scenario Three: there are one or more GPS points only in either inbound or outbound area, as shown in Fig.6(c). It is assumed that the process that the bus decelerates uniformly from Pos1 to station coordinate if Pos1 is in inbound area, or accelerates uniformly from station coordinate to Pos1 if Pos1 is in outbound area. The arrival time is t_{Pos1} plus or subtract the running time.

According to the above rules, the actual bus arrival time was obtained from the raw bus trajectory data. The distribution of bus travel time in the target section is shown in Fig.8, where the vertical line indicates the mean value of the distribution; the mode τ, mean value μ, and standard deviation σ are equal to 240, 340.37, and 170.71, respectively. After data preprocessing, the features including bus travel time and headway that constitute the input of the model can be computed.

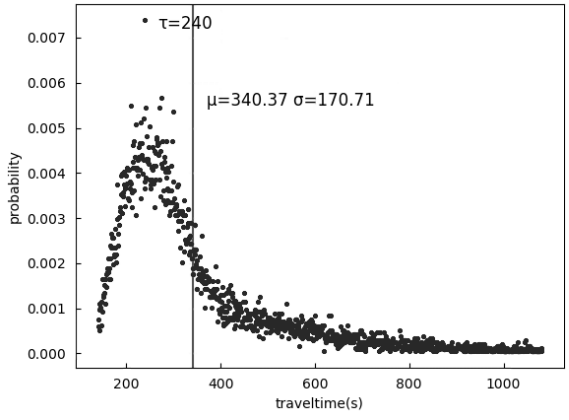


Fig.8 Distribution of bus travel time in target section

Baselines. Our RSTL-ConvNet was compared with four baselines as follows:

- **ANN/SVM:** The two features were fed only in the target section $\{t_m^i, h_m^i | i = n-8, \dots, n\}$ to an artificial neural network with four fully-connected layers/support vector machine with kernel type of radial basis function.
- **RNN/LSTM:** RNN/LSTM is famous for time series prediction, which can capture the time dependent effectively. The RNN/LSTM model architecture is two RNN/LSTM layers followed by two fully-connected layers. The same features were fed to RNN/LSTM as ANN.
- **BiRNN:** Bidirectional recurrent neural network (BiRNN) focuses on the context simultaneously, which accepts both the last moment output and next moment output of the hidden layers as input. The information about the next shifts (source of the next moment output) is unknown in our problem. The first several shifts which passed station m later than shift n in the same day of last week were regarded as the "next shifts" of shift n. The features $\{t_m^i, h_m^i, t_m^j, h_m^{j\cdot} | i = n-8, \dots, n; j = n+1, .., n+4\}$ were fed to a BiRNN model with two BiRNN layers and two fully-connected layers. $t_s^j, h_s^{j\cdot}$ are called historical future features.
- **FD-ConvNet:** The ConvNet model, whose architecture is two convolution blocks followed by one fully-connected layer, is mainly inspired by VGG nets (Simonyan et al., 2015). The convolution block is the combination of two convolution layers and one maxpooling layer. The same features were fed to FD-ConvNet as RSTI-ResNet but the reverse direction information was ignored.

Hyperparameters. In ANN/RNN/LSTM/BiRNN model, except for the bottom layer, the upper layers have 512 hidden units in order to output one predicted value. The dropout probability in recurrent layers[18] and fully-connect layers[19] are 0.5. In FD-ConvNet, the previous convolution block uses 32 filters of size 2×2 and pooling strides 2, and the latter convolution block uses 64 filters of size 2×2 and pooling strides 2×1. In

RSTI-ResNet model, the previous residual block uses 32 filters of size 2×2, and the latter residual block uses 64 filters of size 2×2 with downsampling of strides 2.

The batch size is 64. The model was trained using the asynchronous stochastic gradient descent (ASGD) optimization strategy. The learning rate is 0.001. Our data samples are about 15,000 where 80% are used for training and 20% are used for testing.

Evaluation Metric: Our method was measured by Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) :

$$\mathrm{RMSE} = \sqrt{\frac{\sum_i (y_i - \hat{y}_i)^2}{z}} \tag{10}$$

$$\mathrm{MAPE} = \frac{1}{z}\sum_i \frac{|y_i - \hat{y}_i|}{y_i} \tag{11}$$

where y and $\hat{y}$ are the true value and the predicted value, respectively; and z is the number of all the predicted values.

Results. Table 1 shows the RMSE and MAPE for the seven methods.

Table 1 Comparison among different methods

Method	RMSE	MAPE
ANN	81.25	19.69
SVM	79.05	18.94
RNN	77.76.	18.46
LSTM	78.08	18.00
BiRNN	78.47	18.74
FD-ConvNet	76.83	17.86
FD-ResNet	75.39	17.70
RSTI-ResNet	**74.70**	**17.38**

It was found that the RSTI-ResNet was better than the baselines, and the accuracy of the proposed model improved effectively.

(a) RNN/LSTM/BiRNN had better performance than ANN/SVM, demonstrating the effectiveness of the RNN when handling temporal information. LSTM was superior to RNN in many time sequence tasks, but they were matched in the bus travel time prediction problem, at least on the features we fed, which may not embody the long-term dependent characteristics of LSTM.

(b) BiRNN did not perform better than RNN, indicating the futility of historical future features.

(c) FD-ResNet is a ResNet model, which only utilizes the forward direction information. It had an improvement compared with RNN, suggesting the contribution of spatial information to prediction. It is helpful to learn the increase or decrease tendency in nearby sections for bus travel time prediction in target section.

(d) RSTI-ResNet had a further improvement compared with the FD-ResNet, which demonstrates that the combination of the revere direction information and the forward direction information can reflect the road traffic conditions more comprehensively.

Fig.9 shows the bus travel time predicted by the RSTI-ResNet against the true values in test set, where the black dash line is a guide line, on which the abscissa values are equal to the ordinate values.

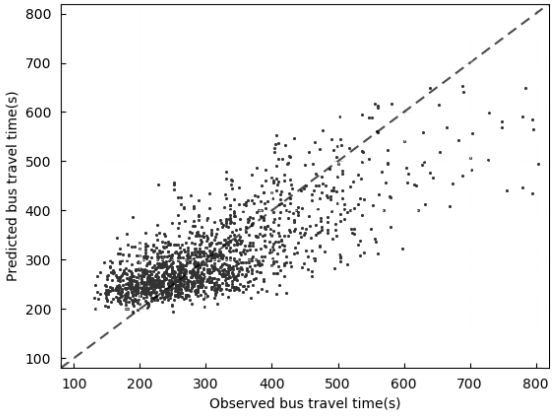


Fig.9 Predictability of the proposed model

Extension. Then, we relaxed the assumption. We tried to predict the bus arrival time at the target station T_{m+1}^n when bus has arrived at stations $m, m-1, m-2, \ldots, 2$, respectively. In our experiment, the target station was the 11-th station. Fig.10 shows the RMSE for three methods. In general, the RTSI-ResNet was still better than others. Particularly, the advantage was more significant when the distance between bus and the target station was farther, which means less short-term information.

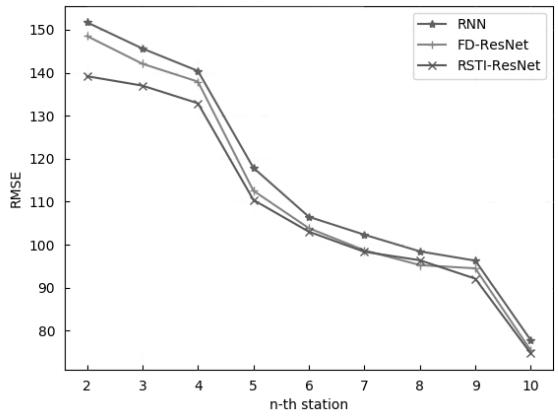


Fig.10 Predicted error in each previous station

A point on a line represents the predicted error when buses are at the *n*-th station. For example, the far-left point on the green line shows that the RMSE of the predicted value of the bus arrival time the at 11-th station is 139.19 when buses are at the 2-nd station by the method of RTSI-ResNet.

5 Conclusions

A novel route spatial-temporal information based residual network for bus arrival time prediction was proposed. A set of rules were provided to judge the actual bus arrival time from the raw bus trajectory data. The proposed model was evaluated on the route NO. 10 in Shenzhen bus systems, and the obtained results were better than other 7 baseline methods such as RNN/LSTM, ConvNet, and SVM, confirming that the model is more applicable to the bus travel time prediction problem. It demonstrates that using the spatial-temporal information of the whole route in both forward and reverse directions is effective on improving the prediction accuracy.

In our experiment, we tried to deepen the proposed model using four residual blocks and extracted features from 11 sections on the route NO. 10, meaning the size of input was extended to 9×11×2. On one hand, the RMSE declined obviously under 45 s in validation set, and on the other, the RMSE raised beyond 100 s in the test set. The model was overfitting. It was considered that the number of our training samples (about 12,000) was not large enough to cover the various complex scenarios in high dimensional characteristic. Bus arrival time prediction on a target route and any other routes are two different tasks because the data of any two routes may be in a different feature space while there is an obvious correlation between them. Considering the correlation, we will use the data of other bus routes to help train the model by transfer learning[20] in the future, which will enhance the performance of learning by avoiding the problem of train samples insufficient in target routes.

References

[1] Ma X, Tao Z, Wang Y, et al. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. Transportation Research Part C: Emerging Technologies, 2015, 54:187-197. DOI: 10.1016/j.trc.2015.03.014.

[2] Karlaftis M G, Vlahogianni E I. Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. Transportation Research Part C: Emerging Technologies, 2011, 19(3): 387-399. DOI: 10.1016/j.trc.2010.10.004.

[3] Ji Y J, Lu J W, Chen X S, et al. Prediction model of bus arrival time based on particle swarm optimization and wavelet neural network. Journal of Transportation Systems Engineering and Information Technology, 2016, 16(3): 60-66. (in Chinese)

[4] Yu B, Lam W H K, Tam M L. Bus arrival time prediction at bus stop with multiple routes. Transportation Research Part C: Emerging Technologies, 2011, 19(6): 1157-1170. DOI: 10.1016/j.trc.2011.01.003.

[5] Yu B, Ye T, Tian X M, et al. Bus travel-time prediction with a forgetting factor. Journal of Computing in Civil Engineering, 2012, 28(3): 06014002. DOI: 10.1061/(ASCE)CP.1943-5487.0000274.

[6] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015, 521(7553): 436-444. DOI: 10.1038/nature14539.

[7] Wlodarczak P, Soar J, Ally M. Multimedia data mining using deep learning. Proceedings of 2015 Fifth International Conference on Digital Information Processing and Communications (ICDIPC). Piscataway: IEEE, 2015: 190-196. DOI: 10.1109/ICDIPC.2015.7323027.

[8] Sutskever I, Martens J, Hinton G. Generating Text with Recurrent Neural Networks. https://www.cs.utoronto.ca/~ilya/pubs/2011/LANG-RNN.pdf, 2019-04-12.

[9] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and Their Compositionality.https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf, 2019-04-12.

[10] Zhang J, Zheng Y, Qi D. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. https://www.aaai.org/ocs/index.php/AAAI/AAAI17/paper/viewFile/14501/13964, 2019-04-12.

[11] Gebru T, Krause J, Wang Y, et al. Using deep learning and Google Street View to estimate the demographic makeup of neighborhoods across the United States. Proceedings of the National Academy of Sciences, 2017, 114(50): 13108-13113. DOI: 10.1073/pnas.1700035114.

[12] Sainath T N, Vinyals O, Senior A, et al. Convolutional, long short-term memory, fully connected deep neural networks. Proceedings of 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Piscataway: IEEE, 2015: 4580-4584. DOI: 10.1109/ICASSP.2015.7178838.

[13] Shi X J, Chen Z, Wang H, et al. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. http://de.arxiv.org/pdf/1506.04214, 2019-04-12.

[14] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE, 2016: 770-778. DOI: 10.1109/CVPR.2016.90.

[15] He K, Zhang X, Ren S, et al. Identity mappings in deep residual networks. Leibe B, Matas J, Sebe N, et al. Computer Vision – ECCV 2016. Lecture Notes in Computer Science. Springer, Cham, 2016, 9908: 630-645. DOI: 10.1007/978-3-319-46493-0_38

[16] Ioffe S, Szegedy C. Batch normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. https://arXiv preprint arXiv:1502.03167, 2019-04-12.

[17] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 2012, 25(2): 1097-1105. DOI: 10.1145/3065386.

[18] Zaremba W, Sutskever I, Vinyals O. Recurrent Neural Network Regularization. https://arXiv preprint arXiv:1409.2329, 2019-04-12.

[19] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 2014, 15(1): 1929-1958.

[20] Pan S J, Yang Q. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(10): 1345-1359. DOI: 10.1109/TKDE.2009.191.