COMPUTER-AIDED CIVIL AND INFRASTRUCTURE ENGINEERING

Computer-Aided Civil and Infrastructure Engineering **33** (2018) 333–350

Prediction of Bus Travel Time Using Random Forests Based on Near Neighbors

Bin Yu

School of Transportation Science and Engineering, Beihang University, Beijing, PR China and Transportation Management College, Dalian Maritime University, Dalian, PR China

Huaizhu Wang

Transportation Management College, Dalian Maritime University, Dalian, PR China

Wenxuan Shan

School of Transportation Science and Engineering, Beihang University, Beijing, PR China

&

Baozhen Yao*

School of Automotive Engineering, Dalian University of Technology, Dalian, PR China

Abstract: *The prediction of bus arrival time is important for passengers who want to determine their departure time and reduce anxiety at bus stops that lack timetables. The random forests based on the near neighbor (RFNN) method is proposed in this article to predict bus travel time, which has been calibrated and validated with real-world data. A case study with two bus routes is conducted, and the proposed RFNN is compared with four methods: linear regression (LR), k-nearest neighbors (KNN), support vector machine (SVM), and classic random forest (RF). The results indicate that the proposed model achieves high accuracy. That is, one bus route has the results of 13.65 mean absolute error (MAE), 6.90% mean absolute percentage error (MAPE), 26.37 root mean squared error (RMSE) and 13.77 (MAE), 7.58% (MAPE), 29.01 (RMSE), respectively. RFNN has a longer computation time of 44,301 seconds for a data set with 14,182 data. The proposed method can be optimized by the technology of parallel computing and can be applied to real-time prediction.*

1 INTRODUCTION

In Intelligent Transportation Systems (ITSs) and Advanced Traveler Information Systems (ATISs), the prediction of bus travel time with reasonable accuracy is important. Travelers can efficiently arrange their schedules and reduce their waiting time, if they can obtain accurate bus arrival information, because waiting time is more significant to a person than travel time (Ben-Akiva and Lerman, 1985). The prediction of bus arrival/running time is also important for bus operators. The prediction results can provide information about the future conditions of a bus system in the short term; bus operators can adjust their bus schedules by applying a higher or lower speed in advance. The prediction of bus arrival time can reduce the waste of bus resources. For example, bus travelers can change their travel mode from bus to taxi or private car when they feel anxious about bus delays without a reasonable expected arrival time, especially for bus stops without timetables.

*To whom correspondence should be addressed. E-mail: *yaobaozhen@hotmail.com*.

DOI: 10.1111/mice.12315

The objective of the prediction of bus travel time is to forecast the travel times of buses between two locations (e.g., two bus stops). Traditional time-series models aim to capture the characteristics of bus travel time over time, which only requires travel time. In addition, Automatic Passenger Counter (APC) and Automatic Vehicle Location (AVL) data are usually employed to predict bus travel time (Shalaby and Farhan, 2003). Studies also seek relations between bus travel time and passenger flows, which are collected from APC data. In these APC-based methods, the travel times of buses are assumed to be influenced by passenger flows at bus stops. Meanwhile, AVL data is useful for prediction because an AVL system provides the current locations of buses, which can be applied to the prediction of the remaining travel. In our study, we collect the data of buses from the AVL system and use this data to measure current traffic conditions on route segments.

A variety of prediction models for bus arrival/running time have been developed over the past decades. Historical average models (Jeong, 2005; Williams and Hoel, 2003), nonparametric regression models (Chan et al., 2009; Chang et al., 2010; Park et al., 2007; Smith et al., 2002; Tam and Lam, 2009), time series models (Al-Deek et al., 1998; Thomas et al., 2010; Chien et al., 2002; Kalman, 1960; Shalaby and Farhan, 2003; Yu et al., 2010), artificial neural network (ANN) models (Adeli, 2001; Adeli and Hung, 1994), and support vector machine (SVM) models (Yu et al., 2010; Yu et al., 2006) are commonly used.

Historical average models assume that historical data are similar to real-time data and predict the running time of a particular trip by averaging over several previous trips. These models may be unreliable when real-time data differ from historical data in spatial or temporal aspects.

Nonparametric regression models are simple because of the absence of estimating parameters. In the last decade, the *k*-nearest neighbor (KNN) model was extensively employed in many fields as a nonparametric regression model. Chang et al. (2010) developed a KNN model to estimate bus travel time; the results proved that the model is effective according to the accuracy and computing time of prediction.

Time series models assume a trend with time in the data set and speculate the predicted value from the trend; thus, these models are very dependent on the similarity between historical prediction and real-time prediction. The methods usually have a short time lag when applied in real-time prediction. Kalman filtering models have the ability to accommodate traffic fluctuations with time-dependent parameters. Originating from the state-space representations in modern control theory, Kalman filter model is applied for predicting short-term traffic demand and travel times. Chien and Kuchipudi (2003) developed a path-based and a link-based model to predict bus arrival time. Shalaby and Farhan (2003) proposed a Kalman filter model to predict bus arrival time and discovered that Kalman filter model outperformed the regression and ANN (artificial neural network) models. An enhanced algorithm based on Kalman filter model was developed to predict bus arrival time and was proved more effective than the standard ANN models (Chen et al., 2004).

Artificial neural network models are widely used in transportation (Adeli and Yeh, 1990; Dharia and Adeli, 2003; Jiang and Adeli, 2004, 2005; Park et al., 1991; Wu and Adeli, 2001) for its ability to deal with complex relationships in data sets. Unlike multivariable models, ANNs can be developed without specifying the form of the function, whereas the restrictions on the multicollinearity of the explanatory variables can be neglected. Chien et al. (2002) provided two ANN models to predict transit arrival time, which are trained by link-based and stop-based data. Both of the models were integrated with an adaptive algorithm to improve prediction accuracy. Meanwhile, hybrid models with a combination of Kalman filtering and neural networks showed good results (Chen et al., 2004; Chien et al., 2002). Jeong and Rilett (2004) used a historical data-based model, regression models and artificial neural network models to predict bus arrival time and found that ANN models outperformed the others in terms of prediction accuracy. van Hinsbergen et al. (2009) combined neural networks in a committee using Bayesian inference theory. An evidence factor was used as a stopping criterion during training and as a tool to select and combine different neural networks.

SVM models are a specific type of learning algorithm characterized by the capacity control of the decision function, the use of the kernel functions and the sparsity of solutions (Cristianini and Shawe-Taylor, 2000; Vapnik, 2013; Vapnik, 1999). Yu et al. (2006) suggested that SVM model is suitable for bus arrival time prediction based on historical data of bus arrival. But Chen et al. (2004) pointed out historical data had difficulty in handling dynamic traffic conditions owing to the lack of real-time data. Then Yu et al. (2010) developed a hybrid model combining a Kalman filtering method with a SVM model, which takes latest bus arrival information/real-time data into account.

However, neural networks and SVMs are complicated because of the large number of parameters needed to be adjusted. Additionally, these algorithms tends to overfit the data (Breiman et al., 1984). Highlighted interest focuses on the emerging type of machine learning technique in recent years, such as random forests, neural network ensembles, bagging and

boosting, and so on (Ghimire et al., 2010; Gislason et al., 2006; Sesnie et al., 2008; Steele, 2000). Ensemble learning algorithms work by running a "base learning algorithm" multiple times, and forming a vote out of the resulting hypotheses (Dietterich, 2002). Ensemble learning technique might have higher accuracy because the group of classifiers performs better than only one single classifier.

Random Forest (RF) model is constructed in a random vector of the data feature space (Breiman, 2001). RF models improve the accuracy of regression without a great increase in computation complexity. Additionally, these models can explain the importance of thousands of variables (Breiman, 2001; Iverson et al., 2008). RFs are efficient and accurate compared with other machine learning models; thus, they are widely applied in different fields (Cutler et al., 2007; Genuer et al., 2010; Yang et al., 2016). Generally, RFs have shorter calculation time and the problem of multicollinearity can be ignored. RF is not sensitive to outliers and remains robust despite missing data. Meanwhile, RF models can reduce overfitting (Breiman, 2001; Friedman and Meulman, 2003) because of the random selection in features and training samples.

RF has been applied in transport such as prediction of traffic flows (Hamner, 2010; Leshem and Ritov, 2007) and bus travel time prediction (Gal et al., 2015; Moreira, 2008). In Moreira's work, the travel time prediction is designed for planning purposes of mass transit companies in a relatively macroscopical aspect. That is, Moreira gave an application of three machine learning algorithms including RF, and travel times of whole trips for transit companies are predicted considering pay day impact, seasonality of the year, and so on. In general, Moreira's work focused on business transit in macroscopical aspect, which did not consider characteristics of buses and bus data, for example, bus dwell time, traffic conditions. Gal et al. (2015) combined queuing theory and machine learning to forecast the bus travel time, with the main concept of predicting travel time based on queuing theory and identifying outliers of the travel time by using machine learning. RF is one of the algorithms that was used in Gal's research for the detection of outliers in scheduled transportation. Gal et al. (2015) used the travel time of the preceding 1 bus as an estimate of the predicted one. Different from these researches, we propose a hybrid model, which combines RF and near neighbors, to forecast the travel time considering current traffic conditions both on current segment and next segments.

Near neighbors method and its extension are also used in prediction of bus travel time, namely KNN method. Main property of near neighbor regression is that the method needs few or no parameters, whose calibration will cost much time in computing. Chang et al. (2010) developed a model based on the nearest neighbor nonparametric regression using historical and current data from the AVL system. Different from the conventional nearest neighbors (KNN) method, where a certain number of samples are selected for prediction, we apply near neighbor method to calculate a weight for each sample in the selection of training set. The near neighbor method used in our article is the linear search (exact method) and compared with other search methods (e.g., K-d tree, KNN), linear search is simple and provides exact results, which can be applied with the help of cloud and parallel computation for faster computation.

The prediction of bus arrival time is usually considered in two ways. First, some researchers simultaneously consider bus running time and bus dwell time. Wall and Dailey (1999) used a combination of both AVL data and historical data to predict bus arrival time. A Kalman filter model is used to track vehicle location and predict bus travel time, where dwell time is not explicitly coped with as an independent variable. Chien et al. (2002) did not consider dwell time as input variables in their ANN model. Second, some researchers consider bus dwell time and running time separately (Jeong and Rilett, 2004; Shalaby and Farhan, 2003). In this article, bus running time and bus dwell time at stops are not separately considered and are not estimated. However, we combine them as the travel times of buses between bus stops. The bus dwell time is taken as a factor of bus travel time in our model.

In terms of model features, a lot of studies focus on the relation of bus travel time over the historical data. Some regression techniques predict the dependent variable by the formulation formed by a set of independent variables that affect travel time, which may include road and traffic conditions, weather, signals, intersections and driver characteristics (Bo et al., 2010; Patnaik et al., 2004). In our method, traffic conditions are mainly considered in model formulation owing to data availability of other factors. Compared with other factors such as weather, road conditions, and driver characteristics, traffic conditions gradually play an important role in affecting bus travel time, especially in congested cities.

The purpose of our work is to predict the bus travel time using the proposed method and analyze the performances in different cases, considering the current traffic conditions as the factors. The predicted travel time can be provided to travelers to help in decision making and can be used for bus operations. In addition, AVL systems and parallel and cooperative computing facilitate real-time prediction.

This paper aims to make two contributions. A new method for the prediction of bus travel time is proposed, that is, random forests based on the near neighbor (RFNN). RFNN involves random forests and the concept of near neighbor, in which a preselection process for training data set is posed to enhance the performance of random forests. Although the computation time of RFNN is longer, the results of RFNN show higher accuracies in mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean squared error (RMSE). RFNN makes it possible for bus operators and passengers to seek for more accurate predictions of bus arrival time. In addition, RFNN suits for large-scale data sets because of the extraction of similar samples and discard of unessential data in the entire data set. Incidentally, random forest is also applied to predict bus travel time in the manuscript, and the performance of RF is evaluated by the comparison with other methods. RF is rarely used in the field of prediction of bus travel time, especially in the prediction that considers traffic conditions as a factor.

The remainder of this article is organized as follows: Section 2 describes RFs and the proposed RFNN model. In Section 3, a numerical test with sensitivity analysis and state-of-the-art comparison are presented. Section 4 concludes the article and provides an outlook on future works.

2 METHOD

In this section, the concept of classic RFs is presented, and the proposed improvement algorithm (Random Forests based on near neighbors) is described.

2.1 Random forests

RF model is a kind of Classification and Regression Tree (CART) model and a type of ensemble learning algorithm. Considering the problem of overfitting, Breiman (2001) proposed the RF model that combines the results of multiple trees (forest) without a significant increase in computation complexity.

In decision tree (DT) learning, the term feature is commonly used, such as the independent variables in regression models. A feature is defined as the dimensions of a data set. For example, weather, road length, and traffic conditions may affect the prediction of bus travel time, which are also referred to as features in DT learning. In the procedure of training, each tree is built based on a random subset of features. For a specific data set, it has a set of features and one random subset of features is assigned to each built tree (also referred to as feature bagging). The reason for this process is the correlation of the trees. Specifically, if a few features are strong predictors for the target output, these features will be selected in many of the built trees, which cause them to become correlated (Ho, 2002). Typically, a third of the total number of features selected for each built tree is recommended. Meanwhile, each tree is trained with a random subset of the original data (Breiman, 2001). In the procedure of data selection, bootstrap sampling is employed, which enables the remaining unused subset to be used for calculating general errors. After the training (based on Information Gain) of the data set, RFs model returns the average output of all aggregations by voting. Different from the single decision tree, RFs is the combination of multiple decision trees. Each tree is an expert of classification or regression in a certain set of features. The final results of RF are obtained by the voting of all trees in RFs, which are superior to single classifier models (Liaw and Wiener, 2002).

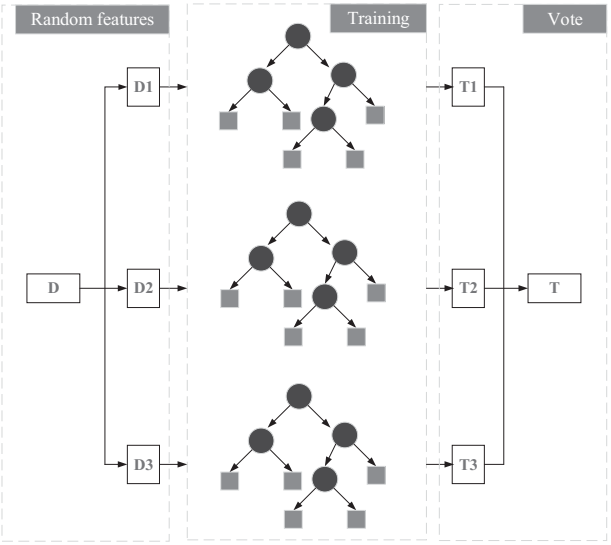

Fig. 1. Process of random forests.

For example, as depicted in Figure 1, assuming five features/independent variables ($d1, d2, \ldots, d5$) for each data, which jointly determine the value of the dependent variable (output), these five features are generated in a single tree for the regression in CART models. Nonetheless, RF model generates a forest with multiple trees where each tree generates with a random subset of the entire features ($D1, D2 \ldots$), assuming that each tree generates with three random features: $D1$ ($d1, d3, d4$), $D2$ ($d2, d3, d5$), and $D3$ ($d1, d2, d4$). Furthermore, each tree is trained based on Information Gain, and each tree majors in certain features (e.g., $D1$ experts in $d1, d3, d4$). That is, the concept of branching in RFs is to set the feature with the maximum information as the upper split feature (e.g., the upper gray circle of T1 in Figure 1). Then the prediction result is obtained according to the

vote of the forest ($T1$, $T2$, $T3$ stand for three different trees), where the average value of the outputs from all trees is commonly used for the final regression result (see Equation (1)). RF will divide the data set based on values of each feature and finally there might be several data at an end leaf. In regression, each tree outputs the mean value of those several data at one of the end leaves, and final prediction results are the mean of each tree. Note that each tree has the equivalent weight in voting.

$$H(x) = \frac{1}{T} \times \sum_{i=1}^{T} h_i(x) \tag{1}$$

where T is the number of trees in the forest and $h()$ stands for the prediction values of the ith tree.

RFs model requires a limit number of parameters (two main parameters): the number of trees in the forest (refer as n_{tree}) and the number of input variables/features used to generate each tree (refer as m_{try}). That is, n_{tree} represents the number of trees in the forests, whereas m_{try} represents the same number of features every tree in the forest contains. The influence of m_{try} means the strength of each tree and also the correlations between trees. A larger value indicates an increase in strength and correlation (Peters et al., 2008). The performance of prediction using RFs model is proved better by increasing the strength and decreasing the correlation, so the value of m_{try} is twofold (Ließ et al., 2012).

For regression problems the inventors recommend $D/3$ for m_{try} (D is the number of total features/input variables) as the default value. The default value of n_{tree} is 500, which was proven; however, it is not appropriate to obtain stable results (Grimm et al., 2008). Thus, we set $n_{\text{tree}} = 1{,}000$ in this article and the value changes in *sensitivity analysis* to figure out the best value of parameters.

2.2 RFs based on near neighbors

The scale of data might be a double-edged sword for arrival time prediction. Few data usually cause a lack of necessary information and hinder the capture of inner relations with its features. Nonetheless, the error rate of the prediction model usually increases in mass training data because the large scale of data may have many data that are not strongly relevant to predictions, which may negatively influence the prediction. To improve the accuracy of the entire prediction process in this article, we propose a hybridization method of the random forests model based on the near neighbors method, which imposes a process of preselection for the training set in RFs models. That is, RFNN contains two main procedures, where the first process involves selection of the training set for the RF model from the original data set, and the second process involves the training and regression procedure of RF model.

First, the preselection process of training set for RFs is based on the concept of "near neighbors," which can also be considered as the reorganization of original data, that is, the result of the preselection is taken as the training set of RF models. Training data set is necessary and essential for machine learning algorithm, but traffic data of public buses is vast thus contains many useless or noisy data for a specific prediction of bus travel time. Therefore, we attempt to identify high-quality data/samples for a certain prediction, which may improve the quality of training set and the prediction. The "high-quality" is measured by the similarity of the training set and predicted data, and the similarity is often numerically measured by the distance of the compared ones in aspect of data features. That is, similarity between samples and the reference sample is considered to be the quality of samples when compared with the reference sample, where higher similarity indicates higher quality. In bus travel time prediction, similar conditions of traffic usually lead to similar bus running conditions (e.g., running speed) and this similarity is captured by the preselection process of training sets. In other words, if we predict the bus running time for a specific traffic condition, we might learn a lot from those similar conditions, which entails the similarity and distance between the reference sample (data to be predicted) and other training samples. The distance between the reference sample and other samples are calculated as Equation (2), where the difference will be assigned a specific weight (see Figure 2) for selection (selection probability). The result of the selection (a set of samples) will be further set as the training set of RFs model. These "near neighbors" mean samples, which are close to the reference sample (in measure of distance). A commonly used distance metric for continuous variables is the Euclidean distance, whereas the Hamming distance or correlation coefficient is common for discrete variables. The Euclidean distance is chosen in this article for distance calculation according to the type of the collected data, as shown in Equation (2).

$$\text{distance}(X_0, X_1) = \sqrt{\sum_{i=1}^{|X_0|} (X_{0i} - X_{1i})^2} \tag{2}$$

where X_0 stands for the reference sample (the sample to be predicted) and X_1 represents the sample in the training set. X_{1i} represents the ith features of sample X_1, and the difference of samples is measured by the similarity of sample features/dimensions.

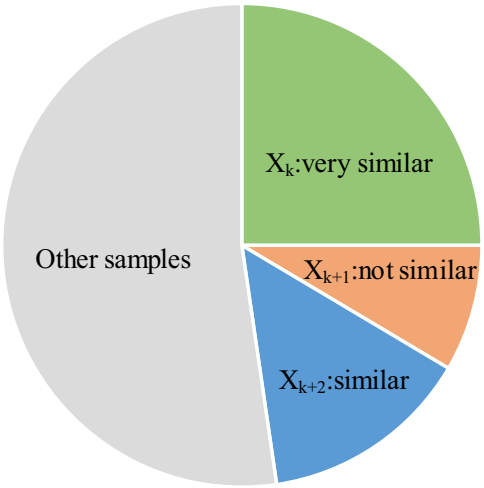

Fig. 2. Selection probability (Roulette) of each sample when comparing with the reference sample X_0. For example, X_k is a very similar sample compared with X_0 thus X_k has a higher choose probability than that of X_{k+1} and X_{k+2} in the selection of training set.

Second, the training samples for the RFs prediction model, which is a black-box regression process, are selected using the Roulette method, which is based on the selection probability (the obtained distance in the first process) of each sample. The training set of RF is selected from the original data set, which is based on a similarity-related probability. A higher selection probability is assigned to samples, which are similar to the data to be predicted (reference sample). Figure 2 shows the generation of the selection probability of samples. Results of the selection are the similar samples (similar traffic conditions) that have higher probability to be chosen in the training set from the vast sample set. However, the selected samples are not used for training of RF model directly. That is, a reselection process (bootstrap sampling) is used, which allows the remaining unused subsets to be used for calculating general errors, as commonly used in RF models. After the training of data sets, RFs model returns the average output of aggregations of trees in the forest.

The procedure of the proposed RFs based on near neighbors is depicted in Figure 3.

Bus travel time prediction between adjacent bus stops that considers current traffic conditions can be described as follows: bus travel time between bus stops (including bus running time and bus dwell time at stops) is assumed to have relations with the average bus dwell time of the current stop and the current traffic conditions on the predicted route segment and next segments. In Equation (3), $\hat{T}_{t,k}$ stands for the predicted value of bus travel time (from the start point of segment k to

```
1  Set num_data = number of initial dataset
2  Set num_training = number of training set for RFs
   model
3  //Initialization
4     Normalization of data
5     Divide the dataset into num_training for training set
6     and num_data-num_training for  test set
7  //Near neighbors (pre-selection)
8     For i = 1 to num_training
9        distance_i = Euclidean distance between data_i and
   the reference/predict sample
10       selection_probability_i =
   (max_distance-distance_i)/sum(max_distance-distance_i)
11       cumulative_probability_i =
   cumulative_probability_i-1 + selection_probability_i
12    End for
13    Select num_training samples based on Roulette
   method
14 //Random forests regression
15    Bootstrap selection for the num_training samples in
   former process
16    Calibration of parameters (ntree, mtry) in RFs
   model
17    RFNN_model = machine learning
18    prediction_result = average(prediction value of
   ntree trees based on RFNN_model)
```

Fig. 3. Procedure of the RFs based on near neighbors (RFNN).

the start point of segment $k+1$) on segment k at time t, and $c_{t,k+1}$ represents the current traffic conditions on segment $k+1$, which is the downstream segment of segment k. Two variables are used to measure traffic conditions: the average running speed $s_{t,k}$ and the speed variance $v_{t,k}$ of the segment. $\hat{T}_{t,k}$ is the combination of the running time on segment k and the dwell time $d_{t,k}$ at the end of segment k (bus stop dwell time) at time t. Current traffic conditions on segment k are measured by the average running speed and the variance of vehicles on segment k (refer to Equation (4)). $\hat{T}_{t,k}$ can be predicted by the set of variables in Equation (5), which are input variables in our machine learning algorithm: RFNN. Estimated values are used to replace the real values because of the availability of real-time data. For example, $s_{t,k}$ is not available at time t in bus data because it is rare that a bus sharply finishes the running on segment k at time t and then the average running speed on k can be calculated. Therefore, we use the average running speeds of the preceding buses, which have finished running on segment k, to approximately replace the average running speed at time t.

$$\hat{T}_{t,k} = f\left(d_{t,k}, c_{t,k}, c_{t,k+1}, \ldots\right) \tag{3}$$

$$c_{t,k} = h\left(s_{t,k}, v_{t,k}\right) \tag{4}$$

$$\hat{T}_{t,k} = g\left(d_{t,k}, s_{t,k}, v_{t,k}, s_{t,k+1}, v_{t,k+1} \ldots\right) \tag{5}$$

2.3 Model validation

The validation of the RF model can be measured in three indices: MAE, MAPE, and RMSE. The MAE measures the average magnitude of the errors in a set of forecasts, whereas the RMSE measures the average magnitude of the error. The RMSE gives a relatively high weight to large errors and is always larger or equal to the MAE. The greater difference between them, the greater the variance in the individual errors in the sample set. MAE and RMSE have the dimensions of prediction and observed values (second), whereas MAPE is dimensionless (%). The three indices are calculated for the validation data sets as follows:

$$\mathrm{MAE} = \frac{1}{n} \times \sum_{i=1}^{n} |f_i - y_i| \tag{6}$$

$$\mathrm{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \frac{|f_i - y_i|}{y_i} \times 100\% \tag{7}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n-1} \times \sum_{i=1}^{n} (f_i - y_i)^2} \tag{8}$$

where f_i is the prediction result, y_i is the observed value (real value), and n is the number of samples.

3 NUMERICAL TEST

3.1 Data collection and analysis

The model for bus arrival time prediction has been tested with the data sets of bus routes 232 and 249 in Shenyang, which is the capital of Liaoning Province in China. The two bus routes extend from the suburb to the center of Shenyang without timetables at each bus stop. Bus route 232 has 19 bus stops and the total distance extends 10.7 km. Meanwhile, the total travel time from origin to destination is approximately 60 minutes, with the bus frequency of 2.5 minutes. Bus route 249 has a length of about 15 km with the bus frequency of about 7 minutes and 27 bus stops. In aspect of bus running speeds, bus routes 232 and 249 have the average running speed of 15.6 km/hour and 14.9 km/hour, respectively, which indicates similar traffic conditions on the two bus routes. Seventeen out of the 18 route segments (divided by the bus stops) are set as time points for arrival time prediction in Figure 3. Similarly, 25 of the 26 route segments are set as the time points for bus arrival time prediction. Because the proposed method predicts bus travel times based on the running conditions on the next segment, the last segment of each bus route is removed from the prediction. The data set is collected on 23/02/2016–25/02/2016 (06:30–19:30) with 15,743 original data of bus arrivals for bus route 232 and 8,257 data for route 249 from the automotive vehicle location (AVL) system. Data of bus arrivals and departures at bus stops are obtained after map matching and cleaning of abnormal data. The remainder of the data set (14,182 data for bus route 232 and 7,623 for bus route 249) is divided into two subsets: 80% for training and 20% for testing. Table 1 lists the descriptive statistics of the data set and Figure 4 depicts the direction of bus routes 232 and 249, where only one direction of bus routes is selected for the case.

As depicted in Figure 5, Bus a is the bus that required predictions, whereas Bus b, …, e are the preceding buses of Bus a. The prediction process enables us to figure out the travel time (including running time and dwell time) of Bus a on Segment 1. We can obtain the running information (average speed and speed variance) of buses from the AVL systems, both on Segment 1 and Segment 2. The running information is updated when the preceding buses leave a stop. That is, the traffic conditions of Segment 1 are updated and measured with the arrival data of Bus c, which finishes travel between bus stops and just leaves for the next stop. Considering Figure 5 as an example, the bus running information is updated at time t. Arrival information of Bus c is the latest information that reflects the current traffic conditions However, the running information of Buses d and e can also be used to measure the traffic conditions on Segment 1. Because Bus c has a distance with Bus d, the main difference between the running information of the two buses is the information loss of current traffic during the "distance." In practice, a shorter interval of data updating will result in a better estimation of current traffic conditions and it is set as a 1-minute interval for updating input data considering practical application. As shown in Figure 5, we consider Bus c and Bus d as the preceding two buses of Bus a because Bus b has not finished travel on Segment 1. Meanwhile, Bus e is regarded as the preceding bus of Bus a on the next (downstream) segment.

3.2 Results

Generally there are two main parameters while using RFs: m_{try} and n_{tree}. However, the best values of m_{try} and n_{tree} for our prediction problem are unknown. Thus a searching of the parameter values should be conducted first, which aims to identify suitable value of parameters (m_{try} and n_{tree}) to predict unknown data accurately. We set the value of the two parameters to 1,000 (as previously mentioned) and $D/3$ (default value), respectively. In the next section, we discuss the sensitivity of the results when parameters' values change.

Table 1
Descriptive statistics for the collected data: route segment number, DataSize, road length, min, max, mean, and standard deviation of bus running time on each route segment. Left: 232; right: 249

	DataSize		*Length [m]*		*Min [s]*		*Max [s]*		*Mean [s]*		*SD [s]*	
Seg	*232*	*249*	*232*	*249*	*232*	*249*	*232*	*249*	*232*	*249*	*232*	*249*
1	861	310	596	320	59	89	153	142	85	117	13.28	10.87
2	769	290	703	900	101	90	271	245	143	187	22.43	28.40
3	858	309	350	488	47	87	137	149	68	116	10.98	17.41
4	851	298	980	458	107	59	250	170	177	115	31.84	19.45
5	874	296	1,000	740	163	200	657	452	306	325	78.48	45.32
6	787	310	270	517	44	100	127	243	66	174	9.82	28.90
7	894	304	605	922	96	168	283	321	135	232	21.48	30.69
8	797	306	615	615	75	115	174	228	106	157	20.22	27.92
9	794	311	670	555	162	152	514	332	282	260	61.81	36.57
10	866	324	500	1,100	107	190	304	570	164	299	21.93	84.47
11	816	288	520	607	85	145	217	347	128	239	19.91	44.05
12	858	304	1,000	385	159	73	395	147	237	106	43.55	22.49
13	862	314	485	470	84	127	211	252	124	204	21.18	59.60
14	824	324	353	285	129	124	293	256	189	185	31.27	59.53
15	870	298	322	680	77	119	172	302	107	222	13.75	37.76
16	794	302	845	740	139	134	423	267	217	200	57.86	25.51
17	807	326	510	728	89	121	220	475	119	200	15.91	54.65
18		322		399		104		387		236		67.35
19		283		680		135		293		221		52.36
20		285		228		73		204		143		36.25
21		307		920		180		332		224		45.09
22		322		445		42		120		75		31.10
23		280		357		52		208		95		30.06
24		316		650		67		136		98		24.33
25		294		760		126		290		201		35.27

The candidate input variables in the model could be divided into three aspects: average bus dwell time at the stop, the running information/conditions of the buses on the current route segment and the next (downstream) route segment. In other words, the condition (e.g., average speed and speed variance) of the current route segment reflects the current traffic conditions (e.g., congestions) and specific route segment conditions (e.g., condition of poor road pavements). For instance, one route segment with a poor pavement condition usually has slower speeds for buses. Furthermore, the condition of the next route segment could also affect the running buses on the current segment. That is, congestions on the next route segment could play a negative role in the interference on the current route segments because of traffic waves. Therefore, the traffic conditions of the current segment and only one downstream (the next) segment are selected in the case study. Traffic conditions are measured by the average running speed and the variance of average speeds, which are collected from the data of preceding buses in the AVL system. In addition, the bus dwell time at stop k is calculated and set as the average value of dwell time at stop k in the time interval of one hour. Although bus dwell time can be estimated with APC data, we use average values for our predictions owing to the lack of APC data. Specific input variables are listed in Table 2.

In Table 2, BDT stands for the average bus dwell time at the bus stop in an hour. To illustrate relative magnitudes of bus dwell time at different stops, bus dwell time is measured in proportion. That is, an index of bus dwell time at a stop is employed to represent the proportion of dwell time at a bus stop against the total dwell time of stops. It is easy to understand and calculated as

$$IBDT = \frac{1}{M} \sum_{m \in M} \frac{BDT_{ms}}{\sum_{s \in S} BDT_{ms}} \tag{9}$$

where $IBDT$ represents the index of bus dwell time at a bus stop, M is the set of buses, and BDT_{ms} stands for the bus dwell time of bus m at stop s. Figure 6 illustrates the average bus dwell time at stops among the two routes. Compared with bus route 249, route 232 has much more stops that have longer bus dwell time than the average

Fig. 4. Bus routes 232 and 249. Left for bus route 232; right: 249. The direction of the bus routes discussed in this article is from the top to the bottom.

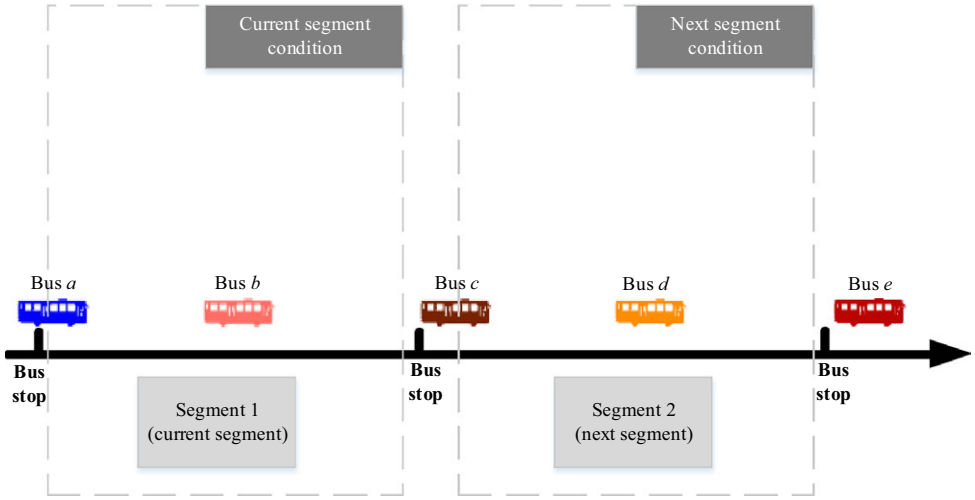

Fig. 5. Explanation of bus arrival time prediction process at time *t*.

values, which indicates larger passenger flows at these bus stops.

SC2 indicates speed_current_2buses, that is, the average of the average running speed of the preceding two buses on the current segment, whereas VC2 represents the speed variance of the two buses on the current segment and SN1 denotes speed_next_1bus on the next segment. All these parameters are set as the input variables in the basic scenario (S0). The bus travel time is tested on the test data set (20% of the whole data) 10 times, and each time of prediction will output different values owing to the preselection based on near neighbors and bootstrap selection of training set in RF. In Figures 7 and 8, three lines are depicted to represent

Table 2
Input variables

Input variable			*Symbol*
Bus dwell time at stop			BDT
Running condition of the buses on the current route segment	Preceding 1 bus	Average speed	SC1
	Preceding 2 buses	Average speed	SC2
		Speed variance	VC2
	Preceding 3 buses	Average speed	SC3
		Speed variance	VC3
Running condition of the buses on the next route segment	Preceding 1 bus	Average speed	SN1
	Preceding 2 buses	Average speed	SN2
	Preceding 3 buses	Speed variance	VN2
		Average speed	SN3
		Speed variance	VN3

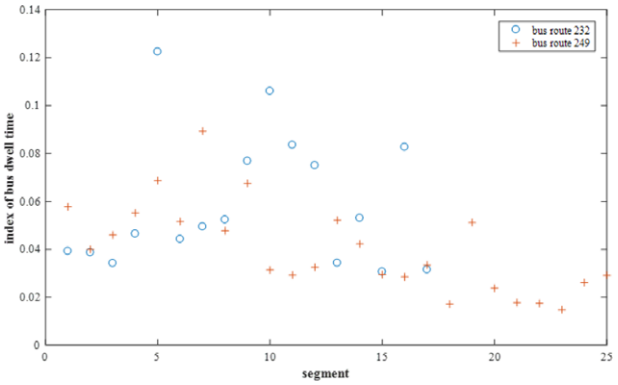

Fig. 6. Index of bus dwell time at segments.

the absolute deviations of 50 seconds. From the figures, it can be found that RFNN performs well because most prediction has an absolute deviation smaller than 50 seconds in Figure 7. As shown in Figure 8, the results show larger deviations of the prediction for bus route 249 than the deviations of the prediction for bus route 232. Many predictions of RFNN provide considerably accurate results because of the proper detection of similar traffic conditions. Note that in RFNN method, similarity is primarily measured by traffic conditions on segments; traffic conditions on a certain segment are assumed to be similar with the traffic conditions on other segments. That is, the similar samples in our pre-

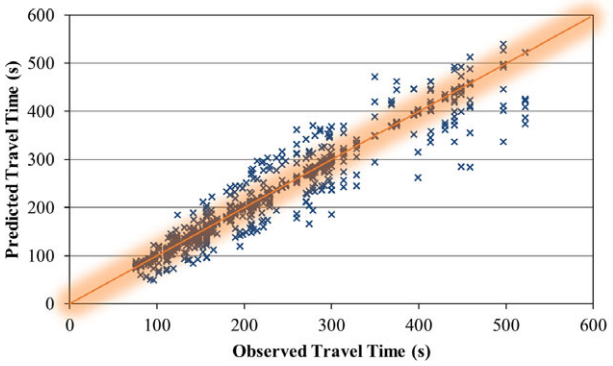

Fig. 7. Prediction results of S0 compared with the observed data of bus route 232. One hundred observed samples are randomly selected from the test data set and each observed sample is predicted by RFNN for 10 times.

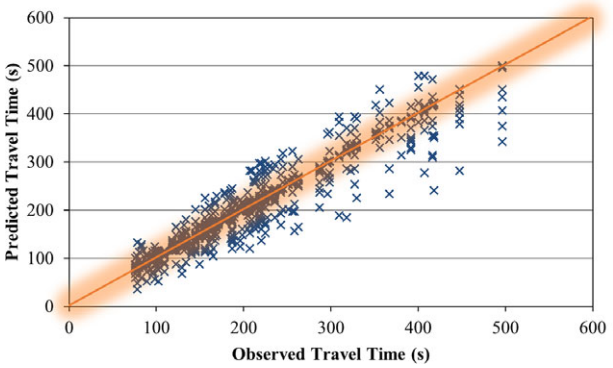

Fig. 8. Prediction results of S0 compared with the observed data of bus route 249. One hundred observed samples are randomly selected from the test data set and each observed sample is predicted by RFNN for 10 times.

selection process are selected from the data of not only a specific segment, but also other segments with similar traffic conditions. This assumption is motivated by the fact that similar traffic conditions on different segments usually lead to similar running conditions (e.g., running speed) and the quality of detecting the right current traffic conditions directly affects prediction performance. Figure 8 shows weaker identification of current traffic conditions as a result of longer bus frequencies and longer data updating owing to fewer buses, which is employed for estimating current traffic conditions. The results also imply that a significant amount of data will support the identification of similar conditions (data set of route 232 is almost twice the size of the data set of route 249).

In this case, the frequency of bus routes 232 and 249 is 2.5 minutes and 7 minutes, respectively. Assuming that the condition/information of the preceding buses

Table 3
Results of the 10 scenarios with 10 runs. Left for bus route 232 and right for 249

			MAE [s]		*MAPE [%]*		*RMSE [s]*	
Scenario	*Description*	*Input variable*	*232*	*249*	*232*	*249*	*232*	*249*
S0	Using all input variables: average speed and speed variance of the preceding 1, 2, and 3 buses on the current segment and the next segment	All	13.65	13.79	6.90	7.60	26.37	29.04
S11	Using the average speed and speed variance of the preceding 1 bus on the current segment and that of the preceding 1 bus on the next segment	BDT,SC1,SN1	15.24	15.06	7.85	8.63	30.96	30.70
S12	Current segment: 1 bus; next segment: 2 buses	BDT,SC1,SN2, VN2	15.06	14.50	7.69	8.15	32.23	30.68
S13	Current segment: 1 bus; next segment: 3 buses	BDT,SC1,SN3, VN3	15.48	14.13	7.85	7.91	32.66	29.49
S21	Current segment: 2 buses; next segment: 1 bus	BDT,SC2,VC2, SN1	19.28	17.93	10.90	11.21	38.77	36.44
S22	Current segment: 2 buses; next segment: 2 buses	BDT,SC2,VC2,SN2, VN2	17.73	17.19	10.10	10.88	37.00	35.82
S23	Current segment: 2 buses; next segment: 3 buses	BDT,SC2,VC2,SN3, VN3	17.77	17.39	10.16	10.90	33.37	32.24
S31	Current segment: 3 buses; next segment: 1 bus	BDT,SC3,VC3, SN1	22.65	22.01	13.41	14.29	44.27	42.86
S32	Current segment: 3 buses; next segment: 2 buses	BDT,SC3,VC3,SN2, VN2	21.07	20.67	12.51	13.33	39.32	38.09
S33	Current segment: 3 buses; next segment: 3 buses	BDT,SC3,VC3,SN3, VN3	21.30	20.49	12.71	13.26	38.11	36.33

reflects the condition of traffic, we use nine additional scenarios to obtain information (average speed and speed variance) about how many preceding buses could obtain a better prediction accuracy. It seems that the preceding one bus usually has a stronger relation with the current bus (need to be predicted), whereas the average running time of the preceding two buses or three buses has a weaker relation due to a long time interval (refer to the bus frequency/headway) between adjacent buses, which may result in a low-accuracy estimation of the current traffic conditions. From Table 3, the results of bus route 249 with frequency of 7 minutes show a significant distinct trend in which a longer headway (lower frequency) between adjacent buses tends to yield lower accuracy compared with the results of bus route 232. The loss in accuracy is attributed to the data collection and updating method in our article. That is, the interval of the data update and bus headway (frequency) have a combined influence on data availability. A longer headway results in fewer buses on the entire bus route and has a smaller number of bus arrival information during a determined time range, which contributes to a lack of reflection of current traffic conditions. Meanwhile, the availability of

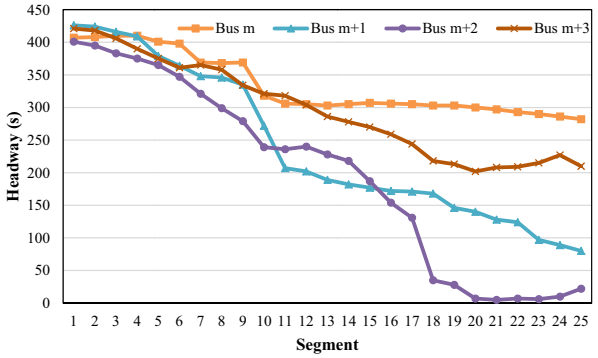

Fig. 9. Headway of buses for bus route 249 at different segment. Bus m is the first bus, and $m + 1$ etc. are the following buses. For example, plot of bus $m + 2$ is the headway between bus $m + 2$ and its preceding bus (bus $m + 1$). Furthermore, headway is calculated by the gap between the departure times of two sequential buses at stops.

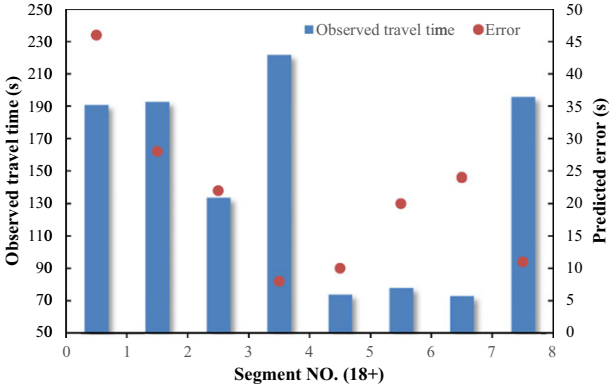

Fig. 10. Observed travel time and predicted error of segment 18–25 for bus $m + 2$. The prediction of RFNN is conducted for 10 times and the average of predicted values are used to calculate errors.

real-time data caused by the interval of the data update is distinct.

Nonetheless, if the running information of the preceding one bus is emphasized, we may get a low-accuracy estimation when unexpected events act on the preceding one bus (e.g., car accidents). What about the average running conditions of the preceding 2 or 3 buses? Therefore, 10 scenarios (with one basic scenario S0) are set to describe the cases in which the running conditions of different numbers of preceding buses are employed. The input variables and results of the 10 scenarios with 20 runs are listed in Table 3. In Table 3, S0 is the basic scenario, and all input variables listed in Table 2 are applied into the RFNN model, whereas other variables are different scenarios with diverse input variables. For example, S23 indicates that the input variables are (1) the average bus dwell time at stop on the current segment, (2) the average running conditions of the preceding two buses on the current segment, and (3) the average conditions of the preceding three buses on the next segment in RFNN model. The input variables can also be described as average bus dwell time (BDT), the average running speed of the preceding two buses (SC2) and the speed variances (VC2) on the current segment; the average running speeds of the preceding three buses (SN3) and the speed variances (VN3) on the next segment.

To evaluate the performance of the proposed method when addressing bus bunching, which indicates large gaps in headway, we detect and choose a real case from bus route 249. A bus bunching occurs between segment 18 and segment 25, especially for bus $m + 1$ and $m + 2$ in Figure 9. Serious bus bunching occurs between bus $m + 1$ and $m + 2$, whereas bus $m + 3$ weakens this deterioration. Therefore, the data of bus $m + 2$ is removed from the training set and added into the testing data set to test the performance of RFNN when facing up with bus bunching.

The results of bus $m + 2$ are depicted in Figure 10, where the predicted values of bus $m + 2$ from segment 18 to 25 exceed the observed values. The reason for these positive errors is the short headway between bus $m + 1$ and bus $m + 2$, in which generates fewer waiting passengers and shorter bus dwell times. However, the short headway (bus bunching) will prevent the real-time data updating for the predicted bus (bus $m + 2$). That is, the real-time data of bus $m + 1$ is not updated in time because it might not finish the running on the current segment. Therefore, the values of input variables for bus $m + 2$ are also not updated and the predicted values are larger than the observed values, which underestimates the changes in the number of waiting passengers owing to shorter headway, especially for the bus stops with a larger variety in waiting passengers.

Traffic congestions usually cause larger variances of bus travel time. Accuracy of the prediction method in the condition of traffic congestion is essential to evaluate the performance. Figure 11 illustrates the performance of RFNN for the prediction of buses during morning peak hours on route 232. The four bus stops (segments) with largest average bus dwell times on bus route 232 are depicted, that is segments 5, 10, 11, and 16. Segments 11 and 16 of bus route 232 have larger prediction errors (with the MAPEs of 8% – 9%). Traffic conditions (e.g., congestions) contribute to the larger error rates partly, which can be also found in segments 5 and 10 (with MAPEs about 7.5 %). Another reason for the higher error rates of segments 11 and 16 than segments 5 and 10 is the similarity between segments

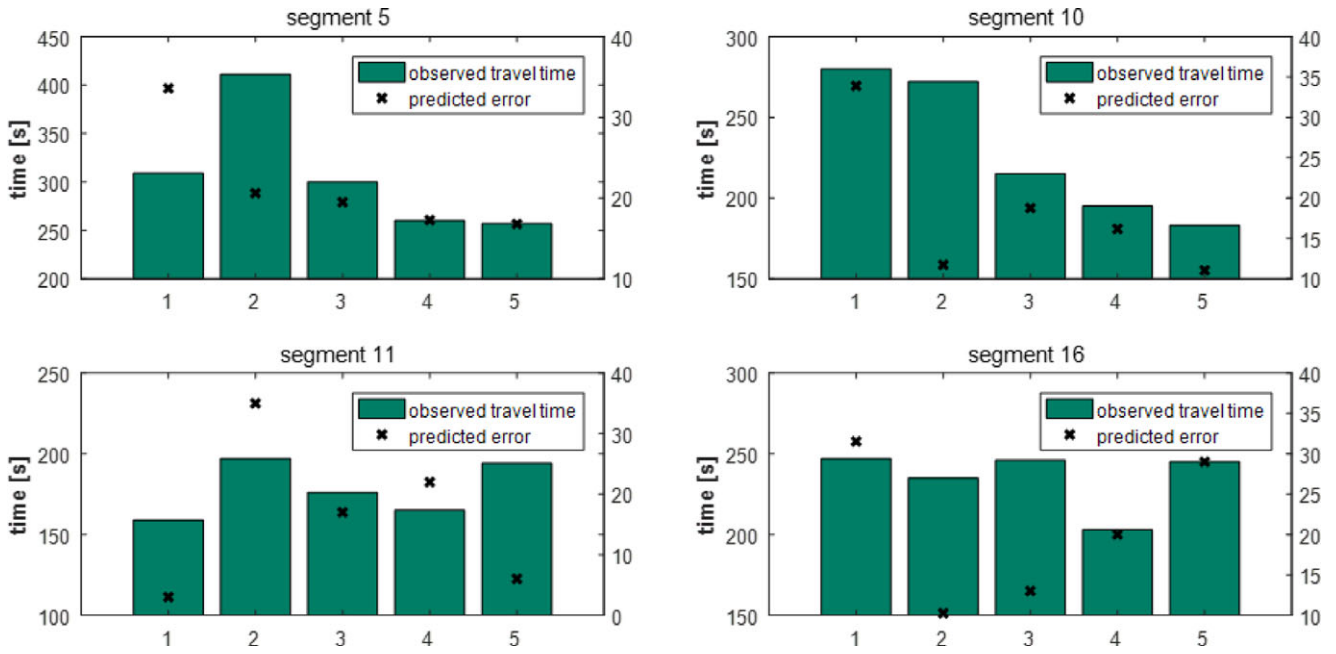

Fig. 11. Observed travel time and predicted error of segments 5, 10, 11, and 16 (four segments with the longest bus dwell times) for bus route 232. The prediction of RFNN is conducted for 10 times and the average of predicted values are used to calculate absolute errors. Five data of each segment are extracted from the testing data set.

11 and 16. Segments 11 and 16 have similar dwell times (mentioned in Figure 6) at stops, and the running speeds of buses during congestions are similar for all buses. The two reasons lead to a negative effect on the detection and selection of similar samples in RFNN. However, the results also indicate that RFNN has an acceptable prediction accuracy during peak hours at bus stops with large dwell times.

From the results in Table 3, we find that the accuracy of bus route 249 (MAPE) is lower than the accuracy of bus route 232 in all scenarios, which is resulting from the higher frequency of bus route 249 and the weaker reflection of the current traffic conditions due to the data updating method in this article. Although certain MAE and RMSE of route 249 are smaller than the MAE and RMSE of 232, note that the MAE and RMSE is usually related to the characteristics of the data. In these two cases, bus route 249 has more segments with a lower travel time in each segment, which causes smaller MAE and RMSE. The best performance of RFNN appears when all input variables are selected in the model training, which implies the strength of RF in discovering inner relations among many factors/features. Moreover, the prediction results of {S11, S12, S13} are better than the prediction results of {S21, S22, S23}, whereas {S21, S22, S23} is better than {S31, S32, S33}. This trend implies that less average running information on current segments are of significance for MAE, MAPE, and RMSE because traffic conditions change rapidly and an average of several preceding buses will weaken the reflection and evaluation of current traffic conditions. The running condition of the preceding one bus on current segments could reflect the traffic conditions better compared with the average conditions of two or three buses, at least in these two cases. The problem is the information delay (see Section 3.1, Data collection and analysis), which will result in a worse reflection of traffic conditions. Therefore, scenarios of one bus on the current segment show better performance owing to its up-to-date reflection of traffic conditions, whereas average of two or three buses on current segment weakens the effect of current traffic conditions. Nevertheless, fewer buses on current segments might not always perform well in all cases. For example, if the preceding one bus breaks down during bus operation, the current traffic conditions cannot be implied by this bus, and the average of the preceding two or more buses will weaken the negative effects of the breakdown.

In terms of the running conditions on next segments, there seems a trend that more buses on next segments could enhance the accuracy of predictions. For example, the trend in {S31, S32, S33}, implies that in this case, the average running conditions of more preceding buses shows the common traffic conditions of the next segment, which probably have a long-term interference on the buses that run on the current segment.

Table 4
Prediction results against different n_{tree} and m_{try} with 10 runs. Left for bus route 232 and right for bus route 249

		MAE [s]		*MAPE [%]*		*RMSE [s]*	
n_{tree}	m_{try}	*232*	*249*	*232*	*249*	*232*	*249*
500	1	19.58	19.05	10.53	11.33	35.29	33.85
500	2	17.12	16.12	9.08	9.45	31.87	29.74
500	3	15.73	14.59	8.07	8.27	30.07	27.81
500	4	15.44	14.41	7.59	7.88	29.68	27.00
1,000	1	19.54	18.37	10.63	10.90	35.10	33.15
1,000	2	16.59	15.61	8.90	9.31	31.22	29.24
1,000	3	15.34	14.68	7.93	8.41	29.49	27.68
1,000	4	13.65	13.79	6.90	7.60	26.37	29.04
1,500	1	19.62	18.32	10.54	10.94	35.42	32.80
1,500	2	15.97	15.42	8.65	9.25	30.48	29.04
1,500	3	15.40	14.36	7.95	8.26	29.43	27.17
1,500	4	15.27	14.35	7.58	7.82	28.67	26.98

The performance of S0 is better than most of the scenarios with acceptable accuracy (MAE, MAPE, and RMSE). Therefore, if the interrelations among input variables are unknown, the use of all these variables in RFNN is sometimes reasonable. Actually, RFNN and RF models generate a forest with many trees, and each tree performs well (experts) in certain features (input variables). The results of RF models consider all of these trees and their own skilled features. Thus, the results of using all input variables without variables screening can be accepted as good quality of results.

In summary, the results for bus routes 232 and 249 have a similar trend in aspect of the input variables and prediction accuracy. Because the traffic conditions of the two bus routes are similar because the average running speeds of buses on the routes have a small difference (0.7 km/h), the main difference of the prediction accuracy between these two bus routes is caused by the availability of real-time data resulting from diverse headways (bus frequencies). MAPE of bus route 249 tends to be higher than the MAPE of route 232 in most scenarios because of the lack of real-time traffic information. RFNN seems to show a better performance in accuracy when the current traffic conditions are updated with new running data of buses.

3.3 Sensitivity analysis

We perform a sensitivity analysis of the two main parameters (n_{tree} and m_{try}) in RF and RFNN models, which, as mentioned earlier, may have a significant effect on the performance of RFNN. In the basic scenario S0, n_{tree} and m_{try} are set to 1,000 and 4, respectively. In this section, we attempt to evaluate the interference of the parameters on the accuracy of the prediction.

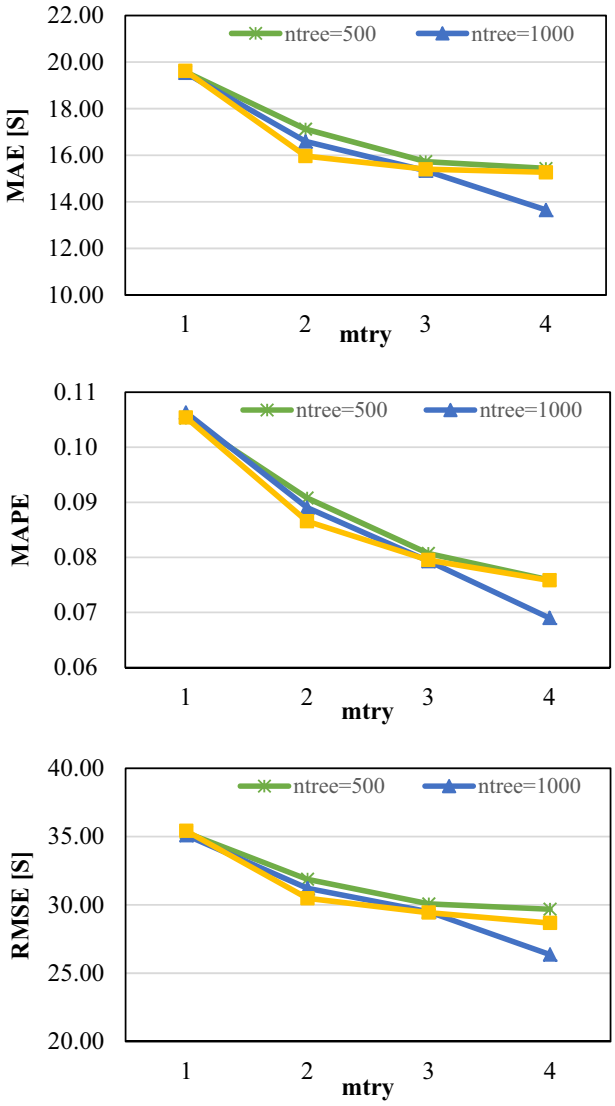

Fig. 12. MAE, MAPE, and RMSE of prediction results of bus route 232 against n_{tree} and m_{try}

The results show that a high value of m_{try} leads to better performance of RFNN especially for MAPE (see Table 4 and Figure 12), because m_{try} determines the strength of each individual tree and a large m_{try} increases this strength (Peters et al., 2008). The best results of MAPE and MAE are obtained when $n_{tree} = 1{,}000$ and $m_{try} = 4$, where $m_{try} = 4 \approx D/3$ (default value). Furthermore, the value of n_{tree} seems to have a weak interference on the prediction results, with the largest deviation of 1.78 seconds, 0.69%, and 3.32 seconds on MAE, MAPE, and RMSE, respectively, for bus routes 232 and 249. The smallest value of RMSE of

Table 5
Results of travel time prediction of five methods. Left for bus route 232, right for 249. Computation is conducted on the computer with dual-core 3.2 GHz processor and 4 GB RAM, and the computation time of RF and RFNN is collected for one time computation. Indeed, the two methods are computed 10 times

	MAE [s]		*MAPE [%]*		*RMSE [s]*		*Computation time [s]*	
Method	*232*	*249*	*232*	*249*	*232*	*249*	*232*	*249*
LR	31.30	31.61	16.41	18.05	46.77	44.93	104	42
KNN	32.66	31.35	17.33	17.77	48.47	45.24	25,216	3,715
SVM	21.09	21.28	11.16	12.37	31.24	30.33	7,112	2,405
RF	16.13	16.41	8.24	8.76	30.61	30.35	1,241	634
RFNN	13.65	13.77	6.90	7.58	26.37	29.01	44,301	6,286

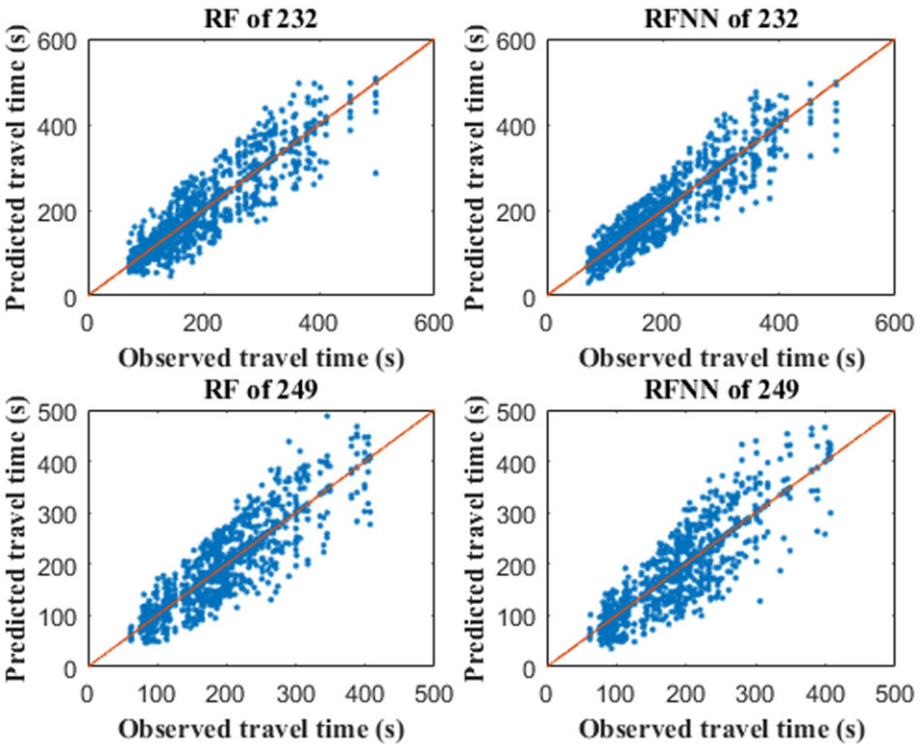

Fig. 13. Prediction results of S0 compared with the observed data. Two hundred observed samples are randomly selected from the test data set and each observed sample is predicted by RFNN and RF for 10 times.

bus route 249 occurs in $n_{\text{tree}} = 1{,}500$, but in general, the setting of parameters in S0 performs well. The results also imply that parameter calibration is necessary for better performance of the prediction in each case.

3.4 Comparison

The results of RFNN are compared in this section with other four main methods: Linear regression (LR); KNN; SVM; and classic random forest (RF). The results are listed in Table 5. These four methods have their strengths in prediction. LR is a classic and prevalent method for prediction due to its ability of analysis. KNN is a well-known nonparameter regression method that does not require the calibration of parameters. In machine learning methods, SVM has been proven to achieve higher accuracy of regression, especially for small-scale of data. Furthermore, the proposed RFNN is also compared with these typical methods, as well as RF method.

For a fair comparison among different methods, the training set and test set are randomly selected from the original data set but are set the same for the five methods. In addition, the results of RF and RFNN change

in each prediction because of the randomness in selecting training sets. Thus, both methods are computed 10 times to evaluate their performance. As shown in the table, RFNN has the highest accuracy, whereas LR model performs worst probably because of the interrelations of the input variables. The input variables (independent variables) in LR are set the same as RFNN and other methods, which indicates that all input variables in Table 2 are employed without considering the problem of interrelations among the independent variables. All input variables are selected because the selection of proper input variables or independent variables for different cases every time is complicated. The performance of each method in obtaining the inner relations among a number of input variables for high-quality prediction can be evaluated from this setting. Other methods are also set to use all variables in Table 2 for fair comparison in accuracy and computation time. The parameter k in KNN model in this article varies from 1 to 4 and is finally set to 3 according to the comparison of accuracy. When $k = 1$ or 2, the accuracy (MAE, MAPE, and RMSE) of KNN is worse than the accuracy when $k = 3$. Therefore, we adopt the best results ($k = 3$) for the further comparison with other methods. The computation time of KNN is considerable so that values of k are not larger than four. The values of main parameters in SVM are obtained using a 10-fold cross validation (McLachlan et al., 2005) and a grid search, where the main parameters are set to $c = 2^5, \varepsilon = 0.2$ using the core function of epsilon-SVR for regression.

For a fair comparison with RFNN, RF model is established with the same parameter values as RFNN, that is, $n_{\text{tree}} = 1,000, m_{\text{try}} = 4$. RFNN has better prediction results than classic RF model owing to the preselection process in RFNN because their values of parameters and input variables are equivalent. The preselection based on near neighbors method for the training set has a positive effect on the prediction accuracy. As shown in Figure 13, RFNN performs better in larger scale of data set (bus route 232) because more data could provide more similar samples for strengthening the process of preselection. For bus route 249, the better performance of RFNN is not clear (similar RMSE with RF) in the figure due to the lack of enough similar and helpful data. However, the preselection in RFNN also supports for the accuracy in MAE and MAPE.

In terms of CPU time, LR has the shortest running time, whereas the computation time of RFNN and KNN model is significantly longer owing to the time-consuming process of measuring distance/similarity in these two methods. The other two black-box methods (RF and SVM) have respectively shorter times than KNN and RFNN. Although RFNN needs a high occupation of CPU time, higher accuracy is possible. With the help of parallel computing and cloud computing, RFNN model can be applied to the real-time prediction of bus travel or arrival time based on the AVL system. Although LR model is direct and can be analyzed compared with black-box methods, the accuracy of LR is limited by the selection of independent variables in the model, and the inner relations between independent variables cannot be neglected. In contrary, RFNN model can handle cases with a larger number of input variables. With the development of ITS and ATIS systems, additional types of data and influence factors can be collected for bus travel time prediction, and machine learning methods (e.g., RF, RFNN) could also perform well with vast potential factors and provide high-quality predictions.

4 CONCLUSIONS

This article focuses on the prediction of bus travel time because it is vital to helping passengers decide departure times to bus stops and reducing anxiety of waiting passengers. RF model has a good performance in nonlinear regression and experts in coping with high-dimension variables or data. Few studies discuss the application of RFs in the prediction of bus travel time. Therefore, we propose a RF-based method that combines RF and near neighbors method. A preselection process of training set is conducted to extract similar samples from the entire data set, which can reduce computation time and eliminate negative effects of noisy and useless data, especially for a large-scale data set. Running data of buses, which contains location and time, on bus routes 232 and 249 in Shenyang are collected and extracted from the GPS data of buses. Then, the bus travel time (running time and dwell time) between adjacent bus stops is calculated based on these data and subsequently employed for model training in the proposed RFNN. The information about bus travel time is applied to measure current traffic conditions for the consideration of data availability and privacy. We consider the traffic conditions of both the current segment and the next segment as the input variables of the RFNN model. In a numerical test, we analyze the influence of the main parameters in RFNN and the effectiveness of the current traffic condition. To be specific, we discuss how many buses' running information can better reflect the current traffic conditions and improve the prediction accuracy. It seems that this number changes in different cases because of bus frequency, generally, availability of real-time data. Finally, we compare the RFNN method with four typical methods in bus travel time prediction. The results show that RFNN has a better performance in accuracy but not

computation time. With the help of parallel computing technology and better performance of computers, the long computation time does not pose a problem considering the high accuracy in prediction. More transit data can be collected from various equipment nowadays, and machine learning methods (including black-box methods) might be more proper to detect the relations between vast factors and bus travel time than conventional methods, although the analysis of these relations is not always feasible. The method proposed in the article can be applied to the prediction of bus travel time and can be easily extended to estimate bus arrival time at each bus stop based on current traffic conditions.

Our method can also be supported by APC data to enhance the estimation of bus dwell time at bus stops. Combined with the technology of parallel computing and cloud computing, reducing computation time to a low level and providing real-time prediction by setting a shorter interval of data updating is possible. In this article, only bus running data are employed for prediction. Further study will consider factors such as weather and numbers of waiting passengers. The average value of running speed can be changed to a weighted average speed where closer buses have larger weights.

ACKNOWLEDGMENTS

This research was supported in Natural Science Foundation of China 71571026 and 51578112, Liaoning Excellent Talents in University LR2015008 and the Fundamental Research Funds for the Central Universities (YWF-16-BJ-J-40 and DUT16YQ104).

REFERENCES

Adeli, H. (2001), Neural networks in civil engineering: 1989–2000, *Computer-Aided Civil and Infrastructure Engineering*, **16**(2), 126–42.

Adeli, H. & Hung, S.-L. (1994), *Machine Learning: Neural Networks, Genetic Algorithms, and Fuzzy Systems*, John Wiley & Sons, Inc., New York, NY.

Adeli, H. & Yeh, C. (1990), Neural network learning in engineering design, in *Paper presented at the Proceedings of the International Neural Network Conference*, Paris, France. Vol. 1, 412–15.

Al-Deek, H., D'Angelo, M. P. & Wang, M. (1998), Travel time prediction with non-linear time series, in *Paper presented at the Fifth International Conference on Applications of Advanced Technologies in Transportation Engineering*, Newport Beach, CA, 317–42.

Ben-Akiva, M. E. & Lerman, S. R. (1985), *Discrete Choice Analysis: Theory and Application to Travel Demand*, vol. 9, MIT Press, Cambridge, MA.

Bo, Y., Jing, L., Bin, Y. & Zhongzhen, Y. (2010), An adaptive bus arrival time prediction model, *Journal of the Eastern Asia Society for Transportation Studies*, **8**, 1126–36.

Breiman, L. (2001), Random forests, *Machine Learning*, **45**(1), 5–32.

Breiman, L. I., Friedman, J. H., Olshen, R. A. & Stone, C. J. (1984), Classification and regression trees (CART), *Lecture Notes in Computer Science*, **40**(3), 17–23.

Chan, K., Lam, W. & Tam, M. (2009), Real-time estimation of arterial travel times with spatial travel time covariance relationships, *Transportation Research Record: Journal of the Transportation Research Board*, **2121**, 102–109.

Chang, H., Park, D., Lee, S., Lee, H. & Baek, S. (2010), Dynamic multi-interval bus travel time prediction using bus transit data, *Transportmetrica*, **6**(1), 19–38.

Chen, M., Liu, X., Xia, J. & Chien, S. I. (2004), A dynamic bus-arrival time prediction model based on APC data, *Computer-Aided Civil and Infrastructure Engineering*, **19**(5), 364–76.

Chien, S. I.-J., Ding, Y. & Wei, C. (2002), Dynamic bus arrival time prediction with artificial neural networks, *Journal of Transportation Engineering*, **128**(5), 429–38.

Chien, S. I.-J. & Kuchipudi, C. M. (2003), Dynamic travel time prediction with real-time and historic data, *Journal of Transportation Engineering*, **129**(6), 608–16.

Cristianini, N. & Shawe-Taylor, J. (2000), *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, Cambridge, United Kingdom.

Cutler, D. R., Edwards Jr., T. C., Beard, K. H., Cutler, A., Hess, K. T., Gibson, J. & Lawler, J. J. (2007), Random forests for classification in ecology, *Ecology*, **88**(11), 2783–92.

Dharia, A. & Adeli, H. (2003), Neural network model for rapid forecasting of freeway link travel time, *Engineering Applications of Artificial Intelligence*, **16**(7), 607–13.

Dietterich, T. G. (2002), Ensemble learning, *The Handbook of Brain Theory and Neural Networks*, **2**, 110–25.

Friedman, J. H. & Meulman, J. J. (2003), Multiple additive regression trees with application in epidemiology, *Statistics in Medicine*, **22**(9), 1365–81.

Gal, A., Mandelbaum, A., Schnitzler, F., Senderovich, A. & Weidlich, M. (2015), Traveling time prediction in scheduled transportation with journey segments, *Information Systems*, **64**(C), 266–80.

Genuer, R., Poggi, J.-M. & Tuleau-Malot, C. (2010), Variable selection using random forests, *Pattern Recognition Letters*, **31**(14), 2225–36.

Ghimire, B., Rogan, J. & Miller, J. (2010), Contextual land-cover classification: incorporating spatial dependence in land-cover classification models using random forests and the Getis statistic, *Remote Sensing Letters*, **1**(1), 45–54.

Gislason, P. O., Benediktsson, J. A. & Sveinsson, J. R. (2006), Random forests for land cover classification, *Pattern Recognition Letters*, **27**(4), 294–300.

Grimm, R., Behrens, T., Märker, M. & Elsenbeer, H. (2008), Soil organic carbon concentrations and stocks on Barro Colorado Island—digital soil mapping using Random Forests analysis, *Geoderma*, **146**(1), 102–13.

Hamner, B. (2010), Predicting travel times with context-dependent random forests by modeling local and aggregate traffic flow, in *Paper presented at the 2010 IEEE International Conference on Data Mining Workshops (ICDMW)*, Sydney, NSW, Australia.

Ho, T. K. (2002), A data complexity analysis of comparative advantages of decision forest constructors, *Pattern Analysis & Applications*, **5**(2), 102–12.

Iverson, L. R., Prasad, A. M., Matthews, S. N. & Peters, M. (2008), Estimating potential habitat for 134 eastern US tree species under six climate scenarios, *Forest Ecology and Management*, **254**(3), 390–406.

Jeong, R. & Rilett, L. R. (2004), Bus arrival time prediction using artificial neural network model, in *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, Washington DC, 988–93.

Jeong, R. H. (2005), The prediction of bus arrival time using automatic vehicle location systems data, Texas A&M University, College Station, TX.

Jiang, X. & Adeli, H. (2004), Clustering-neural network models for freeway work zone capacity estimation, *International Journal of Neural Systems*, **14**(03), 147–63.

Jiang, X. & Adeli, H. (2005), Dynamic wavelet neural network model for traffic flow forecasting, *Journal of Transportation Engineering*, **131**(10), 771–79.

Kalman, R. E. (1960), A new approach to linear filtering and prediction problems, *Journal of Basic Engineering*, **82**(1), 35–45.

Leshem, G. & Ritov, Y. (2007), Traffic flow prediction using Adaboost algorithm with random forests as a weak learner, *International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering*, **1**(1), 1–6.

Liaw, A. & Wiener, M. (2002), Classification and regression by Random Forest, *R News*, **2**(3), 18–22.

Ließ, M., Glaser, B. & Huwe, B. (2012), Uncertainty in the spatial prediction of soil texture: comparison of regression tree and random forest models, *Geoderma*, **170**, 70–79.

McLachlan, G., Do, K.-A. & Ambroise, C. (2005), *Analyzing Microarray Gene Expression Data*, vol. 422, John Wiley & Sons, Hoboken, NJ.

Moreira, J. P. C. L. M. (2008), Travel time prediction for the planning of mass transit companies: a machine learning approach, Universidade do Porto, Porto, Portuguese Republic.

Park, D. C., El-Sharkawi, M., Marks, R., Atlas, L. & Damborg, M. (1991), Electric load forecasting using an artificial neural network, *IEEE Transactions on Power Systems*, **6**(2), 442–49.

Park, S. H., Jeong, Y. J. & Kim, T. J. (2007), Transit travel time forecasts for location-based queries, *Journal of the Eastern Asia Society for Transportation Studies*, **7**, 1859–69.

Patnaik, J., Chien, S. & Bladikas, A. (2004), Estimation of bus arrival times using APC data, *Journal of Public Transportation*, **7**(1), 1–20.

Peters, J., Verhoest, N., Samson, R., Boeckx, P. & De Baets, B. (2008), Wetland vegetation distribution modelling for the identification of constraining environmental variables, *Landscape Ecology*, **23**(9), 1049–65.

Sesnie, S. E., Gessler, P. E., Finegan, B. & Thessler, S. (2008), Integrating Landsat TM and SRTM-DEM derived variables with decision trees for habitat classification and change detection in complex neotropical environments, *Remote Sensing of Environment*, **112**(5), 2145–59.

Shalaby, A. & Farhan, A. (2003), Bus travel time prediction model for dynamic operations control and passenger information systems, in *Paper prepared for presentation at the 82nd Annual Meeting of the Transportation Research Board*, Washington DC, January 2003.

Smith, B. L., Williams, B. M. & Oswald, R. K. (2002), Comparison of parametric and nonparametric models for traffic flow forecasting, *Transportation Research Part C: Emerging Technologies*, **10**(4), 303–21.

Steele, B. M. (2000), Combining multiple classifiers: an application using spatial and remotely sensed information for land cover type mapping, *Remote Sensing of Environment*, **74**(3), 545–56.

Tam, M. L. & Lam, W. H. (2009), Short-term travel time prediction for congested urban road networks, in *Paper presented at the Transportation Research Board 88th Annual Meeting*, Washington DC.

Thomas, T., Weijermars, W. & Van Berkum, E. (2010), Predictions of urban volumes in single time series, *IEEE Transactions on Intelligent Transportation Systems*, **11**(1), 71–80.

van Hinsbergen, C. I., Van Lint, J. & Van Zuylen, H. (2009), Bayesian committee of neural networks to predict travel times with confidence intervals, *Transportation Research Part C: Emerging Technologies*, **17**(5), 498–509.

Vapnik, V. (2013), *The Nature of Statistical Learning Theory Neural Networks*. Springer Science & Business Media, New York, NY.

Vapnik, V. N. (1999), An overview of statistical learning theory, *IEEE Transactions on Neural Networks*, **10**(5), 988–99.

Wall, Z. & Dailey, D. (1999), An algorithm for predicting the arrival time of mass transit vehicles using automatic vehicle location data, in *Paper presented at the 78th Annual Meeting of the Transportation Research Board, National Research Council*, Washington DC.

Williams, B. M. & Hoel, L. A. (2003), Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results, *Journal of Transportation Engineering*, **129**(6), 664–72.

Wu, M. & Adeli, H. (2001), Wavelet-neural network model for automatic traffic incident detection, *Mathematical and Computational Applications*, **6**(2), 85–96.

Yang, R.-M., Zhang, G.-L., Liu, F., Lu, Y.-Y., Yang, F., Yang, F., Yang, M., Zhao, Y. G. & Li, D.-C. (2016), Comparison of boosted regression tree and random forest models for mapping topsoil organic carbon concentration in an alpine ecosystem, *Ecological Indicators*, **60**, 870–78.

Yu, B., Yang, Z.-Z., Chen, K. & Yu, B. (2010), Hybrid model for prediction of bus arrival times at next station, *Journal of Advanced Transportation*, **44**(3), 193–204.

Yu, B., Yang, Z. & Yao, B. (2006), Bus arrival time prediction using support vector machines, *Journal of Intelligent Transportation Systems*, **10**(4), 151–58.