

Received November 22, 2018, accepted December 4, 2018, date of publication December 14, 2018, date of current version January 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2886549

Research on Travel Time Prediction Model of Freeway Based on Gradient Boosting Decision Tree

JUAN CHENG, GEN LI, AND XIANHUA CHEN^{ID}

School of Transportation, Southeast University, Nanjing 211189, China

Corresponding author: Xianhua Chen (chenxh@seu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 51478114 and Grant 51778136.

ABSTRACT To improve the prediction accuracy of traffic flow, a travel time prediction model based on gradient boosting decision tree (GBDT) is proposed. In order to test the applicability of GBDT, models with different prediction horizons (5 min ahead, 10 min ahead, and 15 min ahead) are established. The 11 variables are viewed as candidates in this paper. Different from other machine learning algorithms as black boxes, GBDT can provide interpretable results through variable importance. In the proposed model, the variable importance shows that for different prediction horizons, the most important influence variable is uniform, which is travel time in the current period. Traffic conditions in the current period have the greatest influence on the predicted travel time. Compared with the back propagation neural network model and the support vector machine model, the proposed GBDT model can produce more accurate prediction results, especially in multi-step prediction, indicating that GBDT is a promising method in travel time prediction.

INDEX TERMS Different prediction horizons, freeway, gradient boosting decision tree (GBDT), machine learning, traffic flow, travel time prediction.

I. INTRODUCTION

With the rapid growth of car ownership, traffic congestion has become one of the most serious social problems. To reduce traffic jam, many transportation infrastructures have been built. Recently, Advanced Traffic Information Systems (ATIS) has drawn great attention as it can help the traffic managers make full use of the transportation infrastructures. ATIS is a subsystem of Intelligent Transportation Systems (ITS), which aims at providing timely and reliable traffic information to the public [1]. Based on accurate traffic information, on the one hand, traffic managers can improve traffic management decisions; on the other hand, travelers can make better travel choices [2], [3]. Travel time has been one of the most important and intuitionistic performance measures of a transportation system [4]. With accurate travel time, more informative decision in terms of trip generation and routing can be made by travelers individually. Traffic condition can become more stable and traffic congestion can be mitigated collectively [5]. Therefore, accurate travel time prediction algorithm is imperative.

In recent years, travel time prediction has become a hot research field of ITS. Accurate real-time travel time prediction is the premise of realizing traffic control and

management and traffic guidance, which is the key factor for transforming ITS from passive response to active action. Travel time prediction is also one of the most important issues to be solved in areas such as traffic control systems and traffic flow guidance systems. Accurate prediction of travel time has important practical significance and application value for alleviating urban traffic congestion and avoiding resource waste.

Travel time can be achieved directly or indirectly. Direct methods measure travel time using probe vehicle, records at toll stations, tracking of cell phones, and many other technologies [6]. Indirect methods infer travel time using measured traffic volume, speed, and occupancy from point sensors (e.g., loop detector, video camera) along the vehicle trajectory [7], [8]. In previous literature, the inference of travel time has gone through various approaches, including traffic flow theory, time series analysis, statistical regression and artificial neural network [5], [8]–[10]. The development of traffic information acquisition technology has provided us with a large amount of traffic data, which offer us an opportunity to develop a more accurate travel time prediction based on data mining.

The paper introduces a new data mining technique called Gradient Boosting Decision Tree (GBDT) for travel time prediction. Compared with traditional parametric models, data mining algorithms can be deeply explored implicit relationships between variables.

II. LITERATURE REVIEW

Travel time prediction plays a crucial role in dynamic traffic management decision and travel decision. Therefore, a wide range of algorithms of travel time prediction had been presented in the literature.

Early researches were based on mathematical statistics [11]–[13], for example, Zhang and Rice [14] proposed a linear model with the varying coefficient. The coefficient varied with the smoothing function of the departure time. The method was simple to implement and widely applicable to sensor data. A simple moving average method with constant weight and two adaptive moving average methods were developed by Farokhi Sadabadi *et al.* [15]. The results showed that the three moving average methods had similar accuracy in short-term prediction. However, the prediction accuracy declined with the increase of prediction time. Fei *et al.* [16] applied a dynamic model to predict the freeway travel time. In this model, the prediction process was considered as a random process and the prediction travel time distribution of its associated confidence intervals could be produced as results. This model combined historical traffic information with existing traffic measures. The results of the experiment showed that the model could provide accurate and reliable travel time prediction under different traffic states. Using data of the floating car, a traffic flow prediction model based on the Kalman filter model was proposed by Nanthawichit *et al.* [17]. The state equation was represented by a macroscopic traffic flow model. The model was tested under various traffic states, which were applied to the short-term prediction of travel time. Taking into account noise of the model, Chu *et al.* [18] developed a travel time prediction method based on adaptive Kalman filtering, which combined the data of loop detector with floating car. Because of easy to implementation and low computational effort, models based on mathematical statistics method are widely used. However, the accuracy is generally low.

Another type of travel time prediction method is machine learning algorithm [19], [20], including K-Nearest Neighbor (KNN), Support Vector Machines (SVM) [21], and neural networks [22]. A KNN model was presented by Qiao *et al.* [23], which could produce accurate prediction results under different traffic states. A bus travel time prediction model based on the SVM was proposed by Reddy *et al.* [24]. The model used V-Support vector regression as a linear kernel function and used the data collected by public bus equipped with a GPS system to validate. The result showed that the accuracy of the model was significantly improved under the condition of high variance. Zeng and Zhang [25] raised a state-space neural network model with time delays. The input of the model included the parameters

of current road segment (speed, occupancy, traffic flow), and the parameters of downstream and upstream sections (speed, occupancy, traffic flow), which was verified by the data collected in the Houston freeway. The results indicated that the model was more accurate than ordinary neural network. To predict the travel time of trucks, three kinds of machine learning methods were presented by Sun *et al.* [26]. The three models were based on KNN, SVM, and Random Forests algorithms respectively. Experimental results showed that the accuracy of models based on the SVM and Random Forests respectively were higher than that of KNN. Models based on machine learning methods need large amounts of computation, but the high accuracy drives scholars to shift their research focus to these methods.

In summary, the analysis of the literature mentioned above shows that there are many methods for predicting travel time. Mathematical statistics models provide interpretable parameters and a simple model structure, while machine learning models do not assume any particular model structure of the data, but treat it as unknown. Due to the nonlinear characteristics of traffic flow, machine learning methods for travel time prediction are overall more accurate than the methods based on mathematical statistics. Therefore, travel time prediction model has gradually transferred to machine learning methods. However, most machine learning model is lack of result interpretability, which limits the application in travel time prediction.

In recent years, data mining and machine learning have gradually come into sight. The latest research has applied the Random Forests model to the traffic prediction [27], [28]. Compared with traditional parametric models, machine learning algorithms can be deeply excavated implicit relationships between variables. However, there are fewer studies on the application of GBDT in the transportation field. Based on GBDT, Ma *et al.* [29] presented a model to predict the incident clearance time using different types of explanatory variables. The comparison result showed that the proposed model outperforms BP neural network, support vector machine, and Random Forests. However, the nonlinear relationship between influence variable and response variable was not investigated. Yang *et al.* [30] raised a short-term traffic volume prediction model based on GBDT. Whereas in the proposed model, variables related to traffic states such as density were not considered. A taxi travel time prediction model using Random Forests and Gradient Boosting was proposed by Gupta *et al.* [31]. In this study, the prediction accuracy of Gradient Boosting Regression model was greater than Random Forests Regression model. While variable importance was not analyzed. Zhang *et al.* [32] developed an urban travel time prediction model based on Spatiotemporal Gradient Boosted Regression Tree (STGBRT). The proposed model accounted for spatiotemporal correlations extracted from historical and real-time traffic data for adjacent and target links. The prediction accuracy was elevated in both half-an-hour predictions and one-hour predictions. Nevertheless, the parameters of the proposed model only considered

TABLE 1. The commonly used loss function and gradient.

Name	Loss function	Negative gradient	Application
Squared loss function	$\frac{1}{2}[y_i - f(x_i)]^2$	$y_i - f(x_i)$	
Absolute loss function	$ y_i - f(x_i) $	$sign[y_i - f(x_i)]$	
Huber-M loss function	$\begin{cases} \frac{1}{2}[y_i - f(x_i)]^2, & y_i - f(x_i) \leq \delta_m \\ \delta_m(y_i - f(x_i) - \frac{1}{2}\delta_m), & y_i - f(x_i) > \delta_m \end{cases}$	$\begin{cases} y_i - f(x_i), & y_i - f(x_i) \leq \delta_m \\ \delta_m sign[y_i - f(x_i)], & y_i - f(x_i) > \delta_m \end{cases}$ where $\delta_m = \alpha th - quantile\{ y_i - f(x_i) \}$	Regression
Exponential loss function	$e^{-y_i f(x_i)}, y_i \in \{-1, 1\}$	κ th component: $I(y_i = \zeta_k) - p_k(x_i)$	Classification

the number of trees and the learning rate. A travel time prediction model based on gradient boosting method was presented by Zhang and Haghani [33], but the input variables were only time-related variables. Variables related to traffic states such as density and flow were not considered. Li and Bai [34] developed a freight vehicle travel time prediction model based on Gradient Boosting Regression Tree. The pre-start prediction model used the data-time feature as influential variables and made use of trajectory data as the real-time information to perform post-start prediction. The prediction accuracy of the two models was significantly improved. But there was no comparison with other similar approaches. Variable takes into account the mean speed, Yet variables related to traffic states were also not considered.

In view of this, the hot-spot algorithm in machine learning—Gradient Boosting Decision Tree is used to establish a travel time prediction model. The influence of variables on travel time can be deeply excavated through GBDT.

The main contribution of this paper is (1) Accurately predicting travel time with different prediction horizons. (2) Providing a flexible framework to adopt different types of predictors as the input variables (such as traffic flow, speed, density, occupancy, number of vehicles and data-time variables). (3) Understanding the diverse influences of different variables on the predicted travel time, exploring the nonlinear relationship between variables and the predicted travel time, and having good interpretability.

The rest of the paper is structured as followed. Section III will give the methodology of GBDT to build a travel time prediction model followed by Section IV, which describes the data used in this paper. Results and discussions are presented in Section V. Finally, the conclusions and future work are outlined in Section VI.

III. METHODOLOGY

This section presents the methodology and learning algorithms of GBDT. GBDT is an iterative decision tree algorithm as originally derived by Friedman [35].

A. THE THEORY OF GRADIENT BOOSTING DECISION TREE

1) LOSS FUNCTION

The loss function is a function to measure loss and error, which reflects the credibility of the model. The smaller the

loss function is, the higher accuracy of the model is. Currently, there are various loss functions in GBDT [36]–[38]. The most commonly used loss functions and corresponding gradients are shown in TABLE 1.

2) GRADIENT DESCENT ALGORITHM

Supervised learning is an important kind of machine learning algorithms. For supervised learning, we suppose there are N training samples, $\{(X^1, y^1), (X^2, y^2), \dots, (X^N, y^N)\}$, where

$X^i = \{x_i^1, x_i^2, \dots, x_i^K\}$ is the feature vector of the sample i , and y^i is the response variable of the sample, namely the target variable. The response variable of a sample can be either a continuous value (in regression problems) or a discrete value (in classification problems). The objective of machine learning is to find a mapping function $F(X)$ between the independent variable X^i and target variable y^i by using the training data. In order to find the optimal function, a loss function $L(y, F(x))$ is usually set for the model. Optimal mapping function $F^*(X)$ can be obtained through minimizing the loss function, as

$$F^*(X) = \underset{F(X)}{\operatorname{argmin}} L(y, F(x)) \tag{1}$$

When $F(X)$ is a linear regression problem, the mapping function $F(X)$ is,

$$F(X, \theta) = \theta X = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n \tag{2}$$

Assuming $L(\theta)$ is the loss function of the linear regression problem, and then the optimal mapping function can be obtained by,

$$\theta^* = \operatorname{argmin}_{\theta} L(\theta) = \operatorname{argmin}_{\theta} L(y, F(X, \theta)) \tag{3}$$

$$F^*(X) = F^*(X, \theta) \tag{4}$$

The optimization problem can be effectively solved by Gradient Descent (GD) algorithm, which is also called the steepest descent algorithm. The theoretical basis is that the loss function always decreases the fastest in the direction of the negative gradient (the gradient is the maximum direction derivative of a function at a certain point). The GD algorithm reduces function value along the negative gradient direction when optimizing the objective function.

When solving the minimum value of loss function, the minimum loss function and the parameter values of the model can

Algorithm 1 The Gradient Descent Algorithm

Step 1 initialize the parameter θ_0
 Step 2 for $m = 1, 2 \dots M$, repeat
 (1) Calculate the negative gradient $d_m = -\frac{\partial}{\partial \theta} f(\theta)|_{\theta_m}$
 (2) Calculate the step-size $\rho_m = \underset{\rho}{\operatorname{argmin}} f(\theta_{m-1} + \rho \cdot d_m)$
 (3) Update $\theta_m = \theta_{m-1} + \rho_m \cdot d_m$
 Step 3 Until the termination condition, then $\theta^* = \theta_0 + \sum_{m=1}^M \rho_m \cdot d_m$

be obtained by the iterative method in the negative gradient direction. Algorithm 1 illustrates the basic steps of the GD algorithm.

3) GRADIENT BOOSTING ALGORITHM
 a: OPTIMIZATION IN FUNCTION SPACE

The difference between boosting method and conventional machine learning is that the optimization is held out in the function space. $F_0(X) = f_0(X)$ is set as the initial value according to the GD algorithm described above. Taking the function $F(X)$ as a whole, the optimal function is obtained after $M - th$ iterations.

$$F^*(X) = \sum_{i=0}^M f_i(x) \tag{5}$$

where $f_i(x)(i = 1, 2 \dots M)$ is the function increments, that $f_i(X) = -\rho_i \cdot g_m(X)$. Among them, the new base-learner is the largest loss function associated with negative gradients [39]. Therefore, for $M - th$ iterations, the negative gradient is shown in (6).

$$g_m = -\left[\frac{\partial L(y, F(X))}{\partial F(X)} \right]_{F(X)=F_{m-1}(X)} \tag{6}$$

where g_m is the direction in which the loss function drops the fastest when $F(X) = F_{m-1}(X)$. For each step, a new decision tree is aimed to correct the error made by its previous base-learner, so the current model is updated to (7).

$$F_m(X) = F_{m-1}(X) + \rho_m \cdot h_m(X, \alpha_m) \tag{7}$$

Assuming $h(X, \alpha_m)$ is the base-learner. The step-size of GD is calculated as (8).

$$\rho_m = \underset{\rho}{\operatorname{argmin}} \sum_{n=1}^N [L(y_i F_{m-1}(x_i)) + \rho \cdot h(X_i, \alpha)] \tag{8}$$

The function $f(X)$ is determined by the parameter α , which is shown in (9).

$$f(X) = -\rho \cdot h(X; \alpha) \tag{9}$$

b: GRADIENT BOOSTING ALGORITHM

Boosting method is one of the most important methods in ensemble learning, which generates base models sequentially. The step of boosting method is shown as follows.

- Step 1: Each sample has weights, initial of which is identical;
- Step 2: Base-learner 1 is used to learn the training samples;
- Step 3: When the learning is completed, the weight of the wrong samples is increased, and the weight of the correct samples is reduced;
- Step 4: Base-learner 2 is used to learn;
- Step 5: Repeat Steps 2-4 obtain M base-learners;
- Step 6: Combine the results of the M base-learners as the ultimate learning result.

The weight of each base-learner in the boosting method is different. The specific process of boosting is shown in Fig 1.

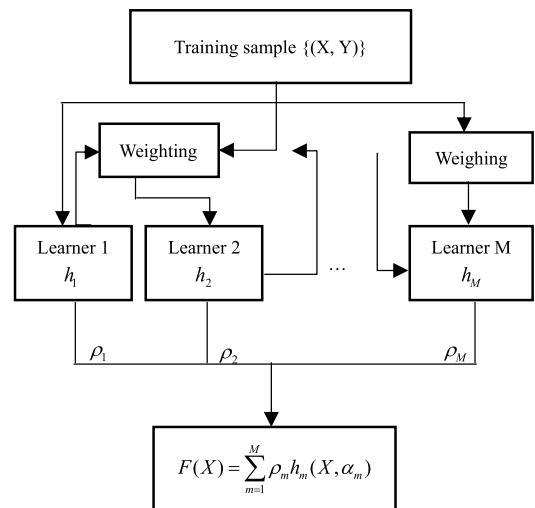


FIGURE 1. Boosting method.

It can be seen from Fig. 1 that the ultimate prediction result of boosting method is the combination of M base-learner results, showed in (10).

$$F(X) = \sum_{m=1}^M \rho_m \cdot h_m(X, \alpha_m) \tag{10}$$

Similar to the optimization in function space described earlier, then

$$F_m(X) = \sum_{i=0}^m -\rho_i \cdot h(X; \alpha_i) \tag{11}$$

The training value for each sample is shown in (12).

$$\bar{y}_i = -\left[\frac{\partial L(y_i, F(X_i))}{\partial F(X_i)} \right]_{F(X)=F_{m-1}(X)} \tag{12}$$

The establishment of the aforementioned model is the process of solving the gradient, so it is also called the boosting method based on gradient. Algorithm 2 illustrates the basic steps of the Gradient Boosting algorithm.

Algorithm 2 The Gradient Boosting algorithm

Step 1 initialize $F_0(X)$ with a constant, $F_0(X) =$

$$\operatorname{argmin}_{\rho} \sum_{i=1}^N L(y_i, \rho)$$

 Step 2 for $m = 1, 2, \dots, M$, repeat
 (1) Calculate the negative gradient

$$\bar{y}_i = -\left[\frac{\partial L(y_i, F(X_i))}{\partial F(X_i)}\right]_{F(X)=F_{m-1}(X)}, i = 1, 2, \dots, N$$

 (2) Calculate $\alpha_m = \operatorname{argmin}_{\alpha, \beta} \sum [\bar{y}_i - \beta \cdot h(X_i, \alpha)]^2$
 (3) Calculate the step-size

$$\rho_m = \operatorname{argmin}_{\rho} \sum_{n=1}^N [L(y_i F_{m-1}(x_i)) + \rho \cdot h(X_i, \alpha)]$$

 (4) Update $F_m(X) = F_{m-1}(X) + \rho_m \cdot h_m(X, \alpha_m)$
 Step 3 Until the termination condition, then output the final model $F_m(X)$

In the above algorithm, the meaning of $\alpha_m = \operatorname{argmin}_{\alpha, \beta} \sum [\bar{y}_i - \beta \cdot h(X_i, \alpha)]^2$ is to measure the proximity between the base-learner and the negative gradient by using the square error at each step of boosting, and choose the base-learner with the smallest fitting error.

4) GRADIENT BOOSTING DECISION TREE ALGORITHM

GBDT algorithm uses a regression tree model of Classification and Regression Trees (CART), which is shown in (13).

$$f(x) = \sum_{j=1}^J c_j \cdot I(x \in R_j) \quad (13)$$

Equation (13) indicates that the data space is divided into J units (J leaf nodes), each of which has fixed output value c_j . Combining Algorithm 2 and Equation (13), GBDT can solve most of the regression models.

However, GBDT might suffer the overfitting problem, which could be effectively solved by introducing the learning rate η at each iteration. Learning rate is a regularization method in GBDT. To further improve the generalization of GBDT, random subsample was introduced by Freidman.

The subsample fraction is proposed to obtain new training data sets by randomly sampled without returning in each step of boosting. Meanwhile, a new feature vector set is obtained randomly sampled without returning in each step. The regression tree is fitted using the new training data sets and feature vector sets. Utilizing the subsample fraction is the second way to regularize GBDT.

5) VARIABLE IMPORTANCE AND PARTIAL DEPENDENCE FUNCTION

α : VARIABLE IMPORTANCE

Variable importance is calculated in the model by the number of time variables appear in the decision tree and performance improvement of the model after each split [35], [36].

$$I_k^2(F) = \frac{1}{M} \sum_{m=1}^M I_k^2(T_m) \quad (14)$$

$$I_k^2(T_m) = \sum_{j=1}^J E_j^2 1_j(X^k) \quad (15)$$

where in (14)-(15), T_m is the m -th decision tree in GBDT F with J leaf nodes. $1_j(X^k)$ is the indicator function that whether variable X^k was chosen as split variable at node j in the decision tree T_m . E_j^2 is the squared error improvement of the corresponding node after selecting the variable X^k to split. $I_k^2(F)$ is the importance value of the variable X^k in the Gradient Boosting Decision Tree F . $I_k^2(T_m)$ is the importance value of the variable X^k in the decision tree T_m .

b : PARTIAL DEPENDENCE FUNCTION

Through GBDT, the partial dependence function of the trained model $F(X)$ for variables can be calculated and the corresponding function graph can be drawn. Equation (16) is the calculation formula for partial dependence function.

$$F(X_S) = E_{X_C} [F(X_S)] = \frac{1}{N} \sum_{i=1}^N F(X_S, X_{iC}) \quad (16)$$

where X_S is a subset of variables that need to calculate partial dependent functions. X_C is the complementary set of X_S . $X_S \cup X_C = X_K$ is a subset of variables in the model. X_{iC} is the value of X_C in training sample, where $i = 1, \dots, N$.

B. PARAMETERS OF MODEL

In GBDT, there are five parameters need to be determined, which is presented in Table 2.

TABLE 2. Parameters in GBDT.

Parameters	Description
J	number of leaf nodes in a single regression tree
η	learning rate
S_a	number of attribute sampling
frd	subsample fraction
M	number of regression trees

1) NUMBER OF LEAF NODES IN A SINGLE REGRESSION TREE J

In the process of constructing GBDT, a weaker regression tree is used to improve the model at each step. Therefore, for each regression tree, it will not be allowed to grow and pruned fully. All regression trees are limited to the same size by J . The recommended value of J is 5-8 [40]. For specific problems, J can be determined through experiments.

2) LEARNING RATE η

Learning rate η ($0 < \eta \leq 1$) is a regularization method used to control the contribution of a single decision tree in the model, which can prevent over-fitting of GBDT. There is a tradeoff between the number of regression trees M and learning rate η . With the same number of regression trees, the smaller η is, the higher the training risk is. Smaller learning rate requires a large number of regression trees to obtain

the same training risk. In general, a small η ($\eta < 0.1$) with a large M is advisable, while smaller η will increase the training time.

3) SUBSAMPLE FRACTION frd

The Subsample fraction frd ($0 < frd \leq 1$) is another regularization method to prevent over-fitting. Unlike Random Forests, sampling without replacement is used in GBDT. If frd is 1, the whole sample is used as a subsampling. If frd is less than 1, part of sample participates are used, which avoids over-fitting but increases the fitting deviation of the sample. Therefore, the value of frd cannot be too low, which is recommended as [0.5, 0.8].

4) NUMBER OF ATTRIBUTE SAMPLING S_a

In developing Random Forests, Breiman [41] demonstrated that additional benefits might be acquired if a subset of predictors is randomly selected as potential splitters during the tree construction process. In order to improve the precision of the model, GBDT combines the idea of Random Forests. In the process of constructing the decision tree, each node uses part of the variable as the candidate variable.

5) NUMBER OF REGRESSION TREES M

The number of regression trees M is also called iterations. In the growth process of GBDT, each iteration will generate a regression tree. Increasing the number of regression trees will improve the prediction accuracy of the training data sets, but correspondingly increase the training time. In order to choose the appropriate M to prevent over-fitting, the data need to be divided into two parts, one for training and one for testing. With the increment of M , the error of test data sets decreases first and then increases. Therefore, M needs to be selected by the error of test data sets.

IV. DATA PREPARATION

A. SOURCE OF DATA

To validate the proposed model, field data collected in Nanjing were used. Unfortunately, the amount of data was still insufficient. Thus, we critically reviewed existing researches and found that using microsimulation software was a good solution and VISSIM has been used for validation by several researchers [42]–[46]. Thus, we used field data to calibrate VISSIM software and got sufficient data to verify the proposed model. VISSIM is used for modeling and analyzing the operation of urban traffic and public traffic under various traffic conditions (lane set, traffic structure, traffic signals, bus station and so on), which is an effective tool for evaluating the design of traffic engineering and urban planning [47].

1) SELECTION OF A SIMULATION SECTION

Nanjing Airport freeway between the Airport Interchange and Lukou Interchange, with the length of 1048.28m and 4 lanes in one direction, is selected as the research area. Time

detectors are set at both ends of the selected freeway section. The route diagram is presented in Fig. 2.

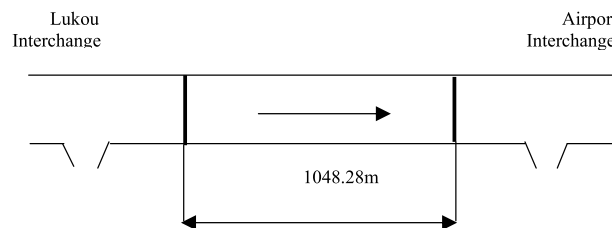


FIGURE 2. The study area.

2) DETERMINATION OF SIMULATION PARAMETERS

a: TRAFFIC FLOW

The VISSIM simulation software is calibrated according to real hourly traffic flow in Nanjing Airport freeway from Nanjing to Airport investigated by airport toll station at 9:00-15:00 on August 22, 2017. Since the real traffic flow does not include congestion, in order to cover the state of free-flow, transition, and congestion in the freeway, the traffic flow increases 600Veh/h from the real measured value of the previous period during 15: 00-17: 00, which reflect the state of congestion. Only increase the number of vehicles does not necessarily result in congestion. However, based on the transition state, we guarantee that all variables are constant, and continue to increase the traffic flow to characterize the congestion state. The input of traffic flow is shown in TABLE 3.

b: VEHICLE TYPE

The user-defined taxi type is 1, and the vehicle color is blue; the truck type is 2, and the vehicle color is yellow; the bus type is 3, and the vehicle color is blue; the car type is 4, and the vehicle color is red.

c: SPEED DISTRIBUTION

In the freeway, the expected speed of car, truck, and bus is 120,100 and 100 km/h. The speed distribution of cars, trucks, buses, and taxis is shown in Fig. 3.

d: VEHICLE PROPORTION

Through investigation, the vehicle proportion on the freeway is car: truck: bus: taxi = 0.42:0.12:0.26:0.2.

e: TIME DETECTOR

In the freeway section, time detectors are set up to collect travel time data of the traffic flow.

B. DESIGN OF EXPERIMENTAL SCHEME

In the process of experiment, the dynamic changing process of the freeway traffic flow was simulated by changing the input traffic flow, including the state of free-flow, transition, and congestion.

TABLE 3. The input of traffic flow.

Time Segments (s)	9:00-9:30	9:30-10:00	10:00-10:30	10:30-11:00
Simulation Time Segments (s)	0-1800	1800-3600	3600-5400	5400-7200
Traffic Flow (veh/h)	800	1200	1600	2000
Time Segments (s)	11:00-11:30	11:30-12:00	12:00-12:30	12:30-13:00
Simulation Time Segments (s)	7200-9000	9000-10800	10800-12600	12600-14400
Traffic Flow (veh/h)	2400	2600	2800	3000
Time Segments (s)	13:00-13:30	13:30-14:00	14:00-14:30	14:30-15:00
Simulation Time Segments (s)	1400-16200	16200-18000	18000-19800	19800-21600
Traffic Flow (veh/h)	3600	4200	4800	5400
Time Segments (s)	15:00-15:30	15:30-16:00	16:00-16:30	16:30-17:00
Simulation Time Segments (s)	21600-23400	23400-25200	25200-27000	27000-28800
Traffic Flow (veh/h)	6000	6600	7200	7800

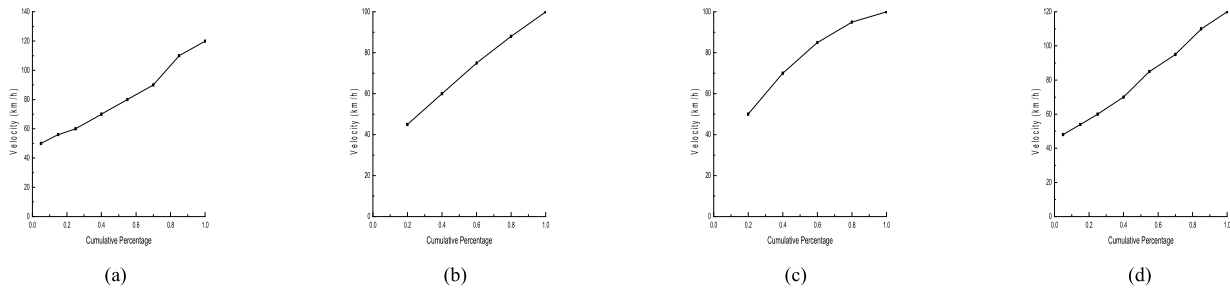


FIGURE 3. The speed distributions. (a) Speed distributions of cars. (b) Speed distributions of trucks. (c) Speed distributions of buses. (d) Speed distributions of taxis.

Using different random seed number, the experiment simulated 133 times and the simulation time was 28800s. At last 133 sets of data were obtained, which represented 133 days' data of 9:00-17:00. Data of 133 days were divided into two data sets, in which 27-133 days of data were used as training data sets and 1-26 days of data were used as test data sets. The 5th day of data in the test data sets was the real travel time obtained by field experiment, which was acquired on August 22, 2017.

Travel time was obtained at the sampling interval of 300 seconds. T_i was used to represent the travel time at time step i (i is the current period), where $i = 1, 2, \dots, 93$, represents 93 time steps from 9:00 to 17:00.

C. VARIABLES OF THE MODEL

1) TRAFFIC STATE PARAMETER

In the Highway Capacity Manual [48], the freeway traffic state was divided into six grades (namely A to F) according to the average speed and density, shown in TABLE 4.

TABLE 4. The standard of traffic state classification [48].

Traffic state	Density Range (pc/mi/ln)	Design Speed (mi/h)		
		75 Speed (mi/h)	65 Speed (mi/h)	55 Speed (mi/h)
A	[0,11)	≥ 75	≥ 65	≥ 55
B	[11,18)	[74.8,75)	≥ 65	≥ 55
C	[18,26)	[70.6,74.8)	[64.6,65)	≥ 55
D	[26,35)	[62.2,70.6)	[59.7,64.6)	[54.7,55)
E	[35,45)	[53.3,62.2)	[52.2,59.7)	[50,54.7)
F	>45	[0,62.2)	[0,52.2)	[0,50)

Note: In order to keep the data neat, the unit used pc/mi/ln and mi/h in

In this study, traffic state parameters refer to the standard of traffic state classification of the freeway, let $x = 1$ to 6 for representing the traffic state A to F of freeway respectively. The paper combined existing traffic state levels and described the freeway at a lower level. Therefore, the traffic state of the freeway was divided into three categories. The free-flow state included the traffic state A and B, namely $x_f = 1$ and 2; the transition state included the traffic state C and D, namely

$x_t = 3$ and 4; the congestion state included the traffic state E and F, namely $x_c = 5$ and 6, which is shown in TABLE 5. The traffic state parameter is $X = \{x_f, x_t, x_c\}$.

TABLE 5. Traffic state parameter.

Traffic State	Traffic State Parameter	Traffic State of The Paper
A	1	free-flow
B	2	
C	3	
D	4	transition
E	5	congestion
F	6	

2) VARIABLES OF THE MODEL

Traffic flow, speed, and density are three basic parameters that characterize traffic flow characteristics and influence travel time of the vehicle. In addition, occupancy and number of vehicles will also have a certain impact on travel time. Therefore, traffic flow, speed, density, occupancy, and number of vehicles are introduced as input variables in the model.

Other factors that have been discussed in previous studies are also considered. Thus, input variable of the model is the relevant parameters at time step i (that is, the current period), where $i = 1, 2 \dots 93$, represents 93 periods from 9:00 to 17:00. The target variable of the model, namely the predicted travel time, is travel time at time step $i+1$, which is denoted by T_{i+1} . The explanation of each variable is shown in TABLE 6.

TABLE 6. Selected explanatory variables.

Variables	Definition	Type of Variables	
T_i	travel time of the vehicle at time step i (travel time in the current period)	input variable	
T_{i-1}	travel time of the vehicle at time step $i-1$		
T_{i-2}	travel time of the vehicle at time step $i-2$		
ΔT_i	changes of travel time over two adjacent time steps i and $i-1$, $\Delta T_i = T_i - T_{i-1}$		
ΔT_{i-1}	changes of travel time over two adjacent time steps $i-1$ and $i-2$, $\Delta T_{i-1} = T_{i-1} - T_{i-2}$		
Q_i	traffic flow at time step i		
K_i	density at time step i		
V_i	speed at time step i		
R_i	occupancy at time step i		
N_i	number of vehicles at time step i		
X_i	traffic state parameter at time step i		
T_{i+1}	travel time of the vehicle at time step $i+1$		target variable (the predicted variable)

V. RESULTS AND DISCUSSION

GBDT models with different prediction horizons (5 min ahead, 10 min ahead and 15 min ahead) were established respectively. Data of 133 days were divided into two data sets, of which 27-133 days of data used as training data sets and 1-26 days of data used as test data sets.

SPM 8.2, a data mining software developed by Salford Systems, was used to establish the GBDT model [49].

Mean Absolute Deviation (MAD), Mean Absolute Percent Error (MAPE) and Relative Error (RE) were selected as evaluation criteria.

$$MAD = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \tag{17}$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \tag{18}$$

$$RE = \frac{y_j - \hat{y}_j}{y_j} * 100\% \tag{19}$$

where in (17)-(19), n is the total number of samples. y_i is the real value of travel time. \hat{y}_i is the predicted value of travel time.

A. PARAMETER DETERMINATION

1) NUMBER OF LEAF NODES IN A SINGLE REGRESSION TREE J

In order to determine the number of leaf nodes in a single decision tree J , for different prediction horizons, the models of the training data sets are established according to $J = 2, 3 \dots 11$, respectively by keeping the other parameters constant. The number of decision trees is automatically determined by MAD of the test data sets. For different prediction horizons, the prediction accuracy of various J is shown in Fig. 4. J is determined according to MAD of the model. It can be observed from Fig. 4 that J are 8, 9, and 10 when the prediction horizon is 5 min ahead (1-step ahead), 10 min ahead (2-step ahead), and 15 min ahead (3-step ahead).

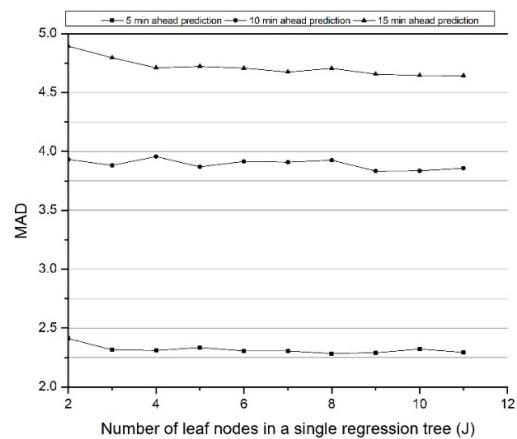


FIGURE 4. MAD of various numbers of leaf nodes in single regression trees for different prediction horizons.

2) LEARNING RATE η

The determination of the learning rate η is similar to J . For different prediction horizons, the models of the training data sets were established with various η respectively by keeping J constant. The number of decision trees is automatically determined by MAD of the test data sets. For different prediction

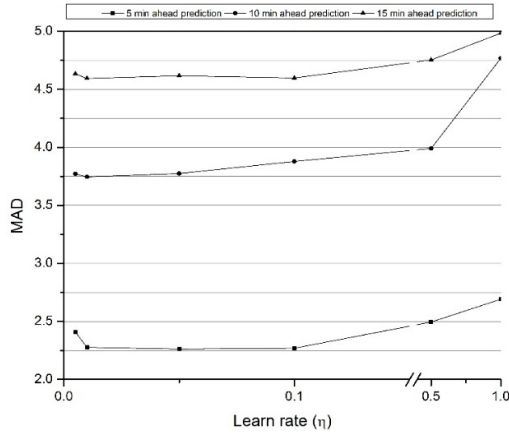


FIGURE 5. MAD of various learning rate for different prediction horizons.

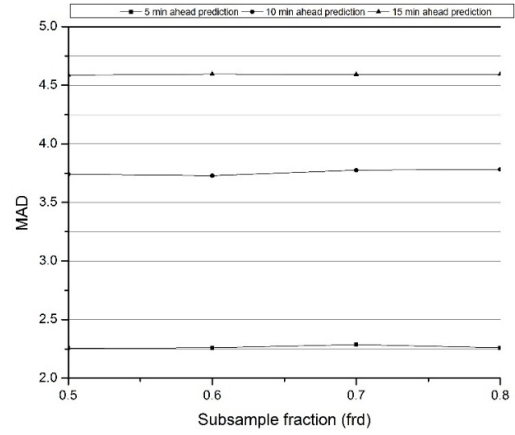


FIGURE 7. MAD of various subsample fraction for different prediction horizons.

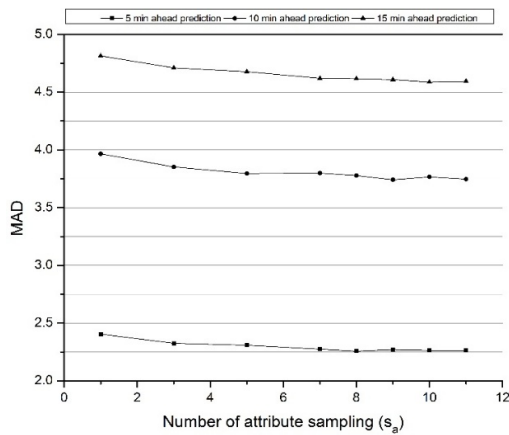


FIGURE 6. MAD of various number of attribute sampling for different prediction horizons.

horizons, the prediction accuracy of η is shown in Fig. 5. As indicated in Fig. 5 that the model error decreases as the learning rate increases. However, when the learning rate of 5 min ahead prediction (1-step ahead) is decreased from 0.05 to 0.01, the learning rate of 10 min ahead prediction (2-step ahead) and learning rate of 15 min ahead prediction (3-step ahead) is decreased from 0.01 to 0.005, the error is increased. The reason may be that the low learning rate has caused overfitting. η is determined according to MAD of the model. From Fig. 5 that η are 0.05, 0.01, and 0.01 when the prediction horizon is 5 min ahead (1-step ahead), 10 min ahead (2-step ahead), and 15 min ahead (3-step ahead).

3) NUMBER OF ATTRIBUTE SAMPLING S_a

In order to improve the accuracy of the model, sampled variables are used as the candidate variables in the process of constructing the decision tree. The candidate variables are called attribute sampling numbers. In reference (25), Freidman takes $\text{int}\sqrt{k}$ (k is the number of variables) as the number of attribute samples. The paper has 11 variables, using the method of Freidman that $\text{int}\sqrt{k} = 3$. In order to find the best number of attribute samples, for different prediction horizons,

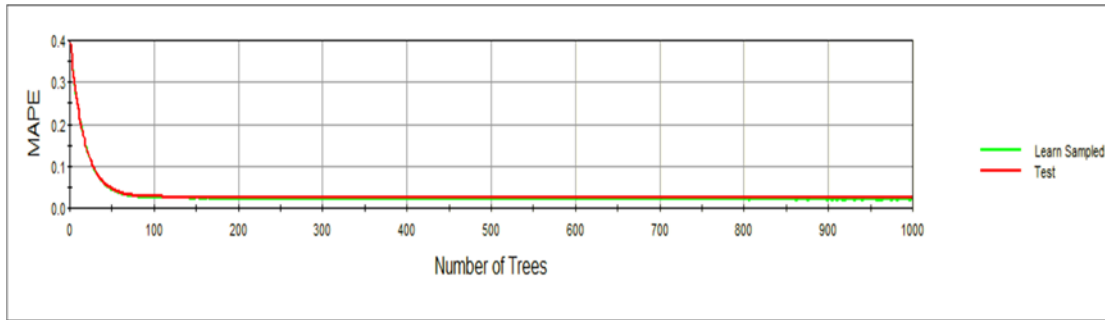
TABLE 7. Variable importance for different prediction horizons.

Variable	5 min Ahead Prediction (1-step ahead)		10 min Ahead Prediction (2-step ahead)		15 min Ahead Prediction (3-step ahead)	
	Order	Relative Importance (%)	Order	Relative Importance (%)	Order	Relative Importance (%)
T_i	1	100	1	100	1	100
V_i	2	54.42	4	40.89	3	55.93
X_i	3	47.87	2	51.70	6	30.70
K_i	4	37.00	6	23.93	2	63.75
T_{i-1}	5	31.79	5	24.00	7	26.63
T_{i-2}	6	30.93	3	49.83	5	52.63
Q_i	7	27.73	7	17.95	8	22.11
N_i	8	22.15	10	9.25	9	13.71
ΔT_i	9	21.32	9	9.85	11	10.44
ΔT_{i-1}	10	19.81	11	8.95	10	11.98
R_i	11	19.56	8	10.54	4	54.08

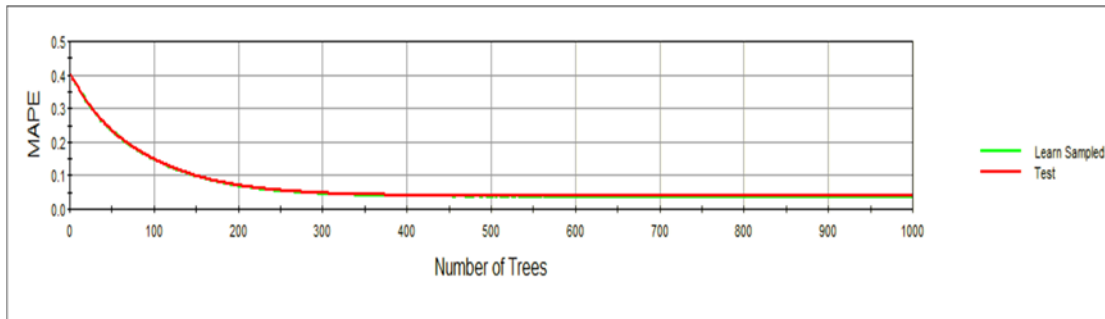
TABLE 8. Maape of different models.

Prediction Horizon	Data Set	GBDT	BP Neural Network	SVM
5 min ahead prediction	Training data	0.0199	0.0270	0.0374
	Test data	0.0245	0.0299	0.0422
10 min ahead prediction	Training data	0.0359	0.0449	0.0647
	Test data	0.0394	0.0470	0.0654
15 min ahead prediction	Training data	0.0426	0.0578	0.0682
	Test data	0.0466	0.0599	0.0695

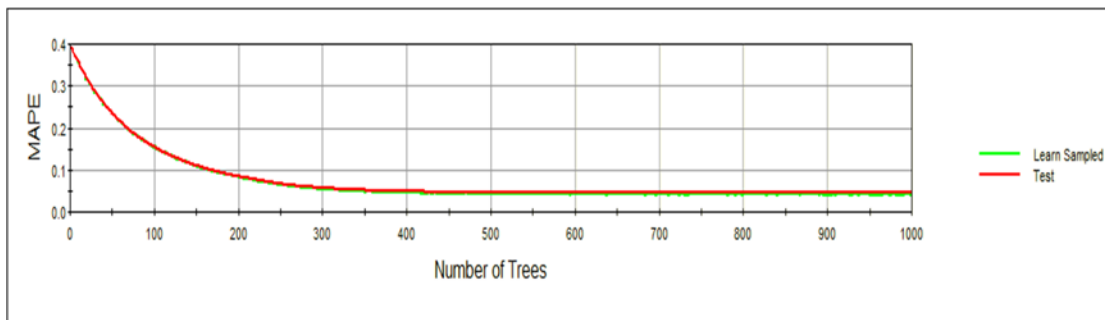
the models of the training data sets are established according to $S_a = 1, 3, 5, 7, 8, 9, 10, 11$, respectively by keeping J and η constant. The number of decision trees is automatically determined by MAD of the test data sets. For different prediction horizons, the prediction accuracy of various S_a is shown in Fig. 6. S_a is determined according to MAD of the model. As can be seen in Fig. 6 that S_a are 8, 9, and 10 when



(a)



(b)



(c)

FIGURE 8. Learning result of GBDT for different prediction horizons. (a) Learning result of GBDT for 5 min ahead (1-step ahead) prediction. (b) Learning result of GBDT for 10 min ahead (2-step ahead) prediction. (c) Learning result of GBDT for 15 min ahead (3-step ahead) prediction.

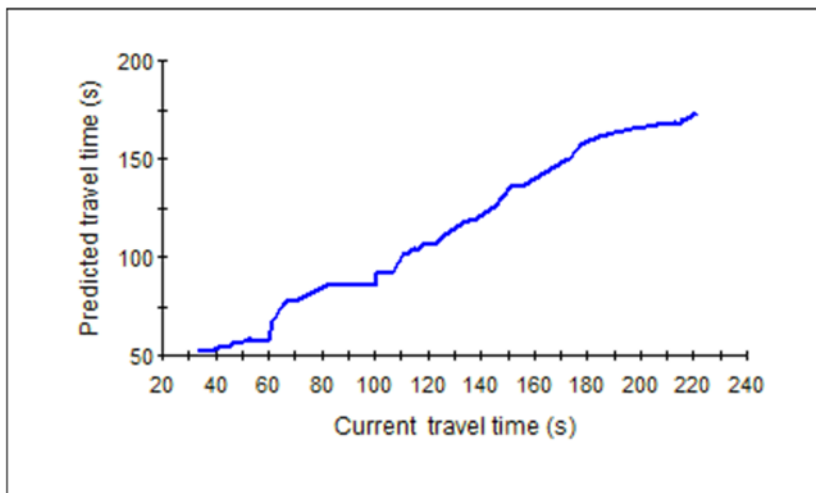
the prediction horizon is 5 min ahead (1-step ahead), 10 min ahead (2-step ahead), and 15 min ahead (3-step ahead).

4) SUBSAMPLE FRACTION frd

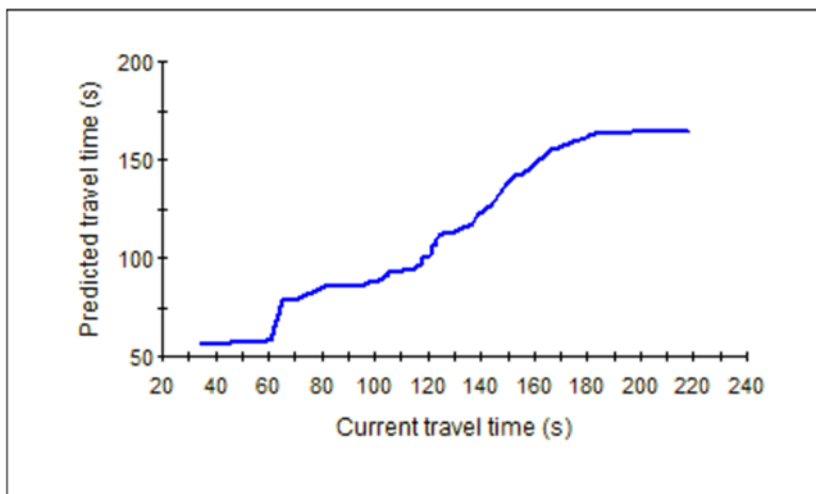
The sampling fraction of the training data sets is generally 0.5-0.8. In order to find the best value of frd , for different prediction horizons, the models of the training data sets are established with $frd = 0.5, 0.6, 0.7, 0.8$, respectively by keeping J, η and S_a constant. The number of decision trees is automatically determined by MAD of the test data sets. For different prediction horizons, the prediction accuracy of various frd is shown in Fig. 7. frd is determined according to MAD of the model. As showed in Fig. 7 that frd are 0.5, 0.6, and 0.5 when the prediction horizon is 5 min ahead

(1-step ahead), 10 min ahead (2-step ahead), and 15 min ahead (3-step ahead).

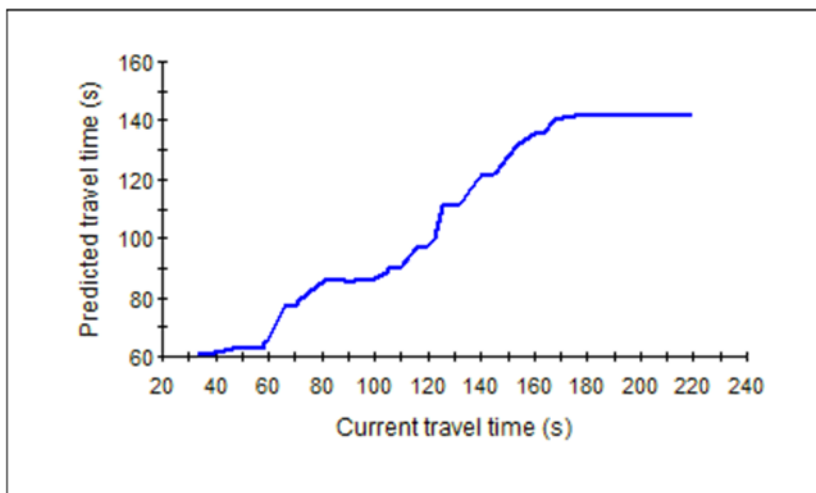
In summary, all parameters of GBDT have been determined. When the prediction horizon is 5 min ahead (1-step ahead), all parameters are $\{J, \eta, S_a, frd\} = \{8, 0.05, 8, 0.5\}$. When the prediction horizon is 10 min ahead (2-step ahead), all parameters are $\{J, \eta, S_a, frd\} = \{9, 0.01, 9, 0.6\}$. When the prediction horizon is 15 min ahead (3-step ahead), all parameters are $\{J, \eta, S_a, frd\} = \{11, 0.01, 10, 0.5\}$. The learning results of the models are different with different prediction horizons, as indicated in Fig. 8. When the prediction horizon is 5 min ahead (1-step ahead), the MAPE is 0.0198, and the cumulative number of regression trees is 950. When the prediction horizon is 10 min ahead (2-step ahead), the MAPE is 0.0365, and the cumulative number



(a)

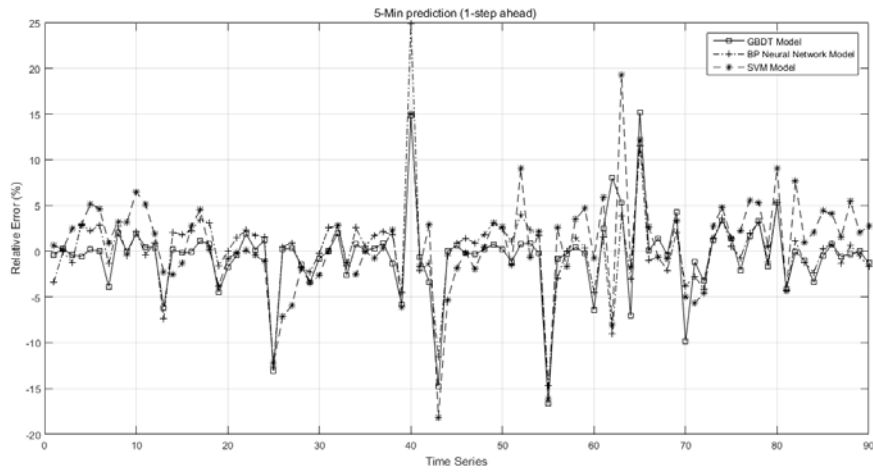


(b)

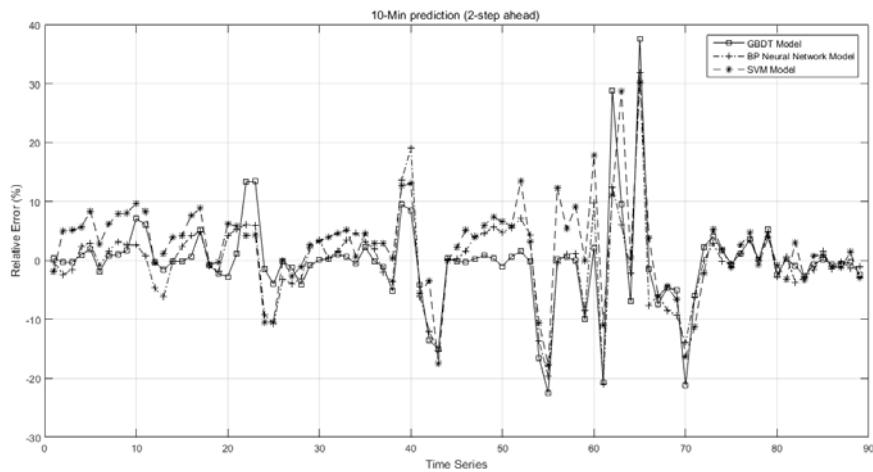


(c) 1

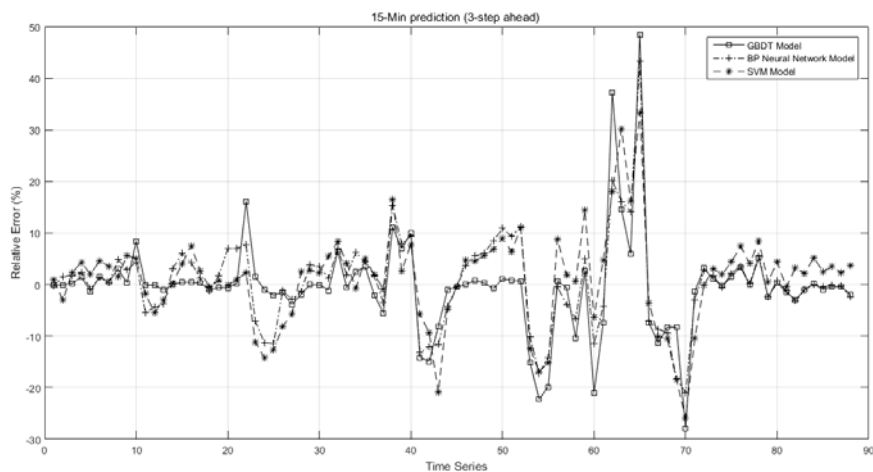
FIGURE 9. Partial dependence function graph for different prediction horizon. (a) 5 min ahead prediction. (b) 10 min ahead prediction. (c) 15 min ahead prediction.



(a)



(b)



(c)

FIGURE 10. Error comparison of the models for different prediction horizons. (a) Error comparison of three model in 5 min ahead prediction. (b) Error comparison of three model in 10 min ahead prediction. (c) Error comparison of three model in 15. min ahead prediction.

of regression trees is 923. When the prediction horizon is 15 min ahead (3-step ahead), the MAPE is 0.0418, and the cumulative number of regression trees is 511.

B. EXPLANATION AND DISCUSSION OF THE MODEL

Input variables of the model usually have different effects on the dependent variable. Exploring the impact of a single input variable on the dependent variable can dig the hidden information of data. The greater importance value of the variable is, the stronger influence on the model is. Variable importance for different prediction horizons is shown in TABLE 7.

From TABLE 7, the contribution values of input variables in the model are different. When the prediction horizon is 5 min ahead (1-step ahead), the most important variables are T_i , V_i , X_i , K_i , and T_{i-1} . When the prediction horizon is 10 min ahead (2-step ahead), the most important variables are T_i , X_i , T_{i-2} , V_i , and T_{i-1} . When the prediction horizon is 15 min ahead (3-step ahead), the most important variables are T_i , K_i , V_i , R_i , and T_{i-2} . For different prediction horizons, the most important variable is the same, that is, travel time in the current period T_i . Travel time of the current period has the greatest influence on travel time of the next period. As expected, the immediate previous traffic state will influence the traffic in the near future. For different prediction horizons, speed of the vehicle V_i in the current period has a greater influence on the predicted travel time, because speed is the most intuitive reflection of travel time. For different prediction horizons, the influences of Q_i , N_i , ΔT_i , and ΔT_{i-1} on the model are relatively small, indicating that flow rate and number of vehicles cannot directly affect travel time. The influence of time difference on the model is less than travel time of the two periods closest to the predicted travel time.

Since the most important influence variables are the same for different prediction horizons, the partial dependence function graphs between predicted travel time and travel time in the current period are shown in Fig. 9.

From Fig. 9, one can see that travel time in the current period have a highly nonlinear relationship with the predicted travel time for different prediction horizons. Meanwhile, the impact trend of the partial dependent function between travel time in the current period and predicted travel time is the same for different prediction horizons. However, as the prediction horizon increases, the change rate of the curve gradually decreases, which demonstrate that travel time in the current period has less influence on the predicted travel time. In particular, when $T_i > 170$, the curve changes most obviously for different prediction horizons. As the prediction horizons increase, the curve gradually becomes a straight line. It indicates that along with the increase of prediction horizons, even the most important influence variable T_i has a weaker effect on the predicted travel time. It also reveals that the model's predicted performance drops with the increase of prediction horizons.

C. ACCURACY OF THE MODEL

To test the effectiveness of the GBDT model, the BP (Back Propagation) neural network model and the SVM model were also established by using the same data. BP neural network [50] is a multi-layer feedforward network trained by error inverse propagation algorithm, which was proposed by a team of scientists led by Rumelhart and McClelland in 1986. SVM [51] was first proposed by Corinna Cortes and Vapnik in 1995. Both methods have been widely used for travel time prediction and other problems. For different prediction horizons, the BP neural network model with a three-layer feedforward perceptron algorithm and the SVM model with radial basis function (RBF) as the kernel function was used to predict travel time. Then the models were tested using 1-26 days of data sets. The training and test errors of different models are shown in TABLE 8.

As indicated in TABLE 8, for different prediction horizons, the prediction accuracy of the GBDT model is better than the BP neural network model and the SVM model. As the prediction horizons increase, the performances of the three models all drop. In comparison, the GBDT model is the least sensitive to the prediction horizons and can maintain good prediction performance. It reveals that the GBDT mode is a promising method for travel time prediction. Meanwhile, the advantages of GBDT are more noticeable in multi-step prediction.

Fig. 10 shows the comparison between travel time of the 5th day in the test data sets (real travel time) and travel time obtained with various models for different prediction horizons. It can be observed in Fig. 10 that the GBDT model can accurately predict the change of travel time.

VI. CONCLUSION

In this paper, GBDT is proposed for travel time prediction, which is a hotspot algorithm in machine learning and can deeply explore the complex relationships between variables. For different prediction horizons, GBDT model is established with 11 variables, namely travel time in current period T_i , traffic flow in current period Q_i , speed in current period V_i , density in current period K_i , number of vehicle in current period N_i , occupancy in current period R_i , traffic state parameter in current period X_i , travel time in previous period T_{i-1} , etc. Using different random seed number, the experiment simulated 133 times with VISSIM simulation software, representing 133 days of traffic conditions. Taking 27-133 days data as training data sets and 1-26 days data as test data sets, results show that among all prediction horizons, the most important influence variable is the same, that is, travel time in the current period T_i , which reflects that travel time of the current period has the greatest influence on travel time of the next period. Comparison results show that GBDT outperforms BP neural network model and SVM model in travel time prediction, especially in multi-step prediction. In general, GBDT is a promising method for travel time prediction.

Comparing with other machine learning algorithms, the GBDT model cannot only produce more accurate prediction results but also provide us an opportunity to understand the diverse influences of different variables and nonlinear relationships between variables and prediction results.

The development of traffic information acquisition technology (such as GPS trajectories, smartphone and road sensors) has provided us with a large amount of traffic data, which offer us an opportunity to develop a more accurate travel time prediction model based on data mining. Further research will use the data to verify the model. Moreover, the data obtained by VISSIM in this paper, which limited the diversity of data. In the future, the variables of weather, characters of drivers, and other variables affecting travel time will be incorporated into the model.

REFERENCES

- [1] M. A. Abdel-Aty, R. Kitamura, and P. P. Jovanis, "Using stated preference data for studying the effect of advanced traffic information on drivers' route choice," *Transp. Res. C, Emerg. Technol.*, vol. 5, no. 1, pp. 39–50, 1997, doi: [10.1016/S0968-090X\(96\)00023-X](https://doi.org/10.1016/S0968-090X(96)00023-X).
- [2] D. M. Miranda and S. V. Conceição, "The vehicle routing problem with hard time windows and stochastic travel and service time," *Expert Syst. Appl.*, vol. 64, pp. 104–116, Dec. 2016, doi: [10.1016/j.eswa.2016.07.022](https://doi.org/10.1016/j.eswa.2016.07.022).
- [3] Y. Wang et al., "A two-stage algorithm for origin-destination matrices estimation considering dynamic dispersion parameter for route choice," *PLoS ONE*, vol. 11, no. 2, p. e0149827, 2016, doi: [10.1371/journal.pone.0146850](https://doi.org/10.1371/journal.pone.0146850).
- [4] W.-H. Lee, S.-S. Tseng, and S.-H. Tsai, "A knowledge based real-time travel time prediction system for urban network," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 4239–4247, 2009, doi: [10.1016/j.eswa.2008.03.018](https://doi.org/10.1016/j.eswa.2008.03.018).
- [5] H. van Lint, "Reliable travel time prediction for freeways," Ph.D. dissertation, Dept. Transport Planning, Delft Univ. Technol., Delft, The Netherlands, 2004.
- [6] L. Sun, J. Yang, and H. Mahmassani, "Travel time estimation based on piecewise truncated quadratic speed trajectory," *Transp. Res. A, Policy Pract.*, vol. 42, no. 1, pp. 173–186, 2008, doi: [10.1016/j.tra.2007.08.004](https://doi.org/10.1016/j.tra.2007.08.004).
- [7] B. Coifman, "New methodology for smoothing freeway loop detector data: Introduction to digital filtering," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1554, no. 1, pp. 142–152, 1996, doi: [10.3141/1554-18](https://doi.org/10.3141/1554-18).
- [8] B. Coifman, "Estimating travel times and vehicle trajectories on freeways using dual loop detectors," *Transp. Res. A, Policy Pract.*, vol. 36, no. 4, pp. 351–364, May 2002, doi: [10.1016/S0965-8564\(01\)00007-6](https://doi.org/10.1016/S0965-8564(01)00007-6).
- [9] Y. Zhang and Y. Xie, "Forecasting of short-term freeway volume with v support vector machines," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2024, pp. 92–99, Jan. 2008.
- [10] S. Shekhar and B. M. Williams, "Adaptive seasonal time series models for forecasting short-term traffic flow," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2024, pp. 116–125, Jan. 2008, doi: [10.3141/2024-14](https://doi.org/10.3141/2024-14).
- [11] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transp. Res. C Emerg. Technol.*, vol. 19, no. 4, pp. 606–616, 2011, doi: [10.1016/j.trc.2010.10.002](https://doi.org/10.1016/j.trc.2010.10.002).
- [12] L. Li, X. Chen, Z. Li, and L. Zhang, "Freeway travel-time estimation based on temporal-spatial queueing model," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1536–1541, Sep. 2013, doi: [10.1109/TITS.2013.2256132](https://doi.org/10.1109/TITS.2013.2256132).
- [13] S. I.-J. Chien and C. M. Kuchipudi, "Dynamic travel time prediction with real-time and historic data," *J. Transp. Eng.*, vol. 129, no. 6, pp. 608–616, 2003, doi: [10.1061/\(ASCE\)0733-947X\(2003\)129:6\(608\)](https://doi.org/10.1061/(ASCE)0733-947X(2003)129:6(608)).
- [14] X. Zhang and J. A. Rice, "Short-term travel time prediction," *Transp. Res. C, Emerg. Technol.*, vol. 11, nos. 3–4, pp. 187–210, 2003, doi: [10.1016/S0968-090X\(03\)00026-3](https://doi.org/10.1016/S0968-090X(03)00026-3).
- [15] S. K. Farokhi, M. Hamed, and A. Haghani, "Evaluating moving average techniques in short-term travel time prediction using an AVI data set," in *Proc. 89th Annu. Meeting, Transp. Res. Board*, Washington, DC, USA, 2010.
- [16] X. Fei, C.-C. Lu, and K. Liu, "A Bayesian dynamic linear model approach for real-time short-term freeway travel time prediction," *Transp. Res. C, Emerg. Technol.*, vol. 19, no. 6, pp. 1306–1318, 2011, doi: [10.1016/j.trc.2010.10.005](https://doi.org/10.1016/j.trc.2010.10.005).
- [17] C. Nanthawichit, T. Nakatsuji, and H. Suzuki, "Application of probe-vehicle data for real-time traffic-state estimation and short-term travel-time prediction on a freeway," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1855, pp. 49–59, Jan. 2003, doi: [10.3141/1855-06](https://doi.org/10.3141/1855-06).
- [18] L. Chu, S. Oh, and W. Recker, "Adaptive Kalman filter based freeway travel time estimation," in *Proc. 84th TRB Annu. Meeting*, Washington, DC, USA, 2005, pp. 1–21.
- [19] N. Julio, R. Giesen, and P. Lizana, "Real-time prediction of bus travel speeds using traffic shockwaves and machine learning algorithms," *Res. Transp. Econ.*, vol. 59, pp. 250–257, Nov. 2016, doi: [10.1016/j.retrec.2016.07.019](https://doi.org/10.1016/j.retrec.2016.07.019).
- [20] A. Gal, A. Mandelbaum, F. Schnitzler, A. Senderovich, and M. Weidlich, "Traveling time prediction in scheduled transportation with journey segments," *Inf. Syst.*, vol. 64, pp. 266–280, Mar. 2017, doi: [10.1016/j.is.2015.12.001](https://doi.org/10.1016/j.is.2015.12.001).
- [21] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus arrival time prediction using support vector machines," *J. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 151–158, 2006, doi: [10.1080/15472450600981009](https://doi.org/10.1080/15472450600981009).
- [22] S. I.-J. Chien, Y. Ding, and C. Wei, "Dynamic bus arrival time prediction with artificial neural networks," *J. Transp. Eng.*, vol. 128, no. 5, pp. 429–438, 2002, doi: [10.1061/\(ASCE\)0733-947X\(2002\)128:5\(429\)](https://doi.org/10.1061/(ASCE)0733-947X(2002)128:5(429)).
- [23] W. Qiao, A. Haghani, C.-F. Shao, and J. Liu, "Freeway path travel time prediction based on heterogeneous traffic data through nonparametric model," *J. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 438–448, 2016, doi: [10.1080/15472450.2016.1149700](https://doi.org/10.1080/15472450.2016.1149700).
- [24] K. K. Reddy, B. A. Kumar, and L. Vanajakshi, "Bus travel time prediction under high variability conditions," *Current Sci.*, vol. 111, no. 4, pp. 700–711, 2016, doi: [10.18520/cs/v111/i4/700-711](https://doi.org/10.18520/cs/v111/i4/700-711).
- [25] X. Zeng and Y. Zhang, "Development of recurrent neural network considering temporal-spatial input dynamics for freeway travel time modeling," *Comput.-Aided Civil Infrastructure Eng.*, vol. 28, no. 5, pp. 359–371, 2013, doi: [10.1111/mice.12000](https://doi.org/10.1111/mice.12000).
- [26] X. Sun, H. Zhang, F. Tian, and L. Yang, "The use of a machine learning method to predict the real-time link travel time of open-pit trucks," *Math. Problems Eng.*, vol. 2018, no. 7, 2018, Art. no. 4368045, doi: [10.1155/2018/4368045](https://doi.org/10.1155/2018/4368045).
- [27] J. Mendes-Moreira, A. M. Jorge, J. F. de Sousa, and C. Soares, "Comparing state-of-the-art regression methods for long term travel time prediction," *Intell. Data Anal.*, vol. 16, no. 3, pp. 427–449, 2012, doi: [10.3233/IDA-2012-0532](https://doi.org/10.3233/IDA-2012-0532).
- [28] B. Hamner, "Predicting travel times with context-dependent random forests by modeling local and aggregate traffic flow," in *Proc. IEEE Int. Conf. Data Mining Workshops*, Dec. 2010, pp. 1357–1359, doi: [10.1109/ICDMW.2010.128](https://doi.org/10.1109/ICDMW.2010.128).
- [29] X. Ma, C. Ding, S. Luan, Y. Wang, and Y. Wang, "Prioritizing influential factors for freeway incident clearance time prediction using the gradient boosting decision trees method," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2303–2310, Sep. 2017, doi: [10.1109/TITS.2016.2635719](https://doi.org/10.1109/TITS.2016.2635719).
- [30] S. Yang, J. Wu, Y. Du, Y. He, and X. Chen, "Ensemble learning for short-term traffic prediction based on gradient boosting machine," *J. Sensors*, vol. 2017, Mar. 2017, Art. no. 7074143, doi: [10.1155/2017/7074143](https://doi.org/10.1155/2017/7074143).
- [31] B. Gupta et al., "Taxi travel time prediction using ensemble-based random forest and gradient boosting model," *Adv. Big Data Cloud Comput.*, pp. 63–78, Apr. 2018, doi: [10.1007/978-981-10-7200-0_6](https://doi.org/10.1007/978-981-10-7200-0_6).
- [32] F. Zhang, X. Zhu, T. Hu, W. Guo, C. Chen, and L. Liu, "Urban link travel time prediction based on a gradient boosting method considering spatiotemporal correlations," *ISPRS Int. J. Geo-Inf.*, vol. 5, no. 11, p. 201, 2016, doi: [10.3390/ijgi5110201](https://doi.org/10.3390/ijgi5110201).
- [33] Y. Zhang and A. Haghani, "A gradient boosting method to improve travel time prediction," *Transp. Res. C, Emerg. Technol.*, vol. 58, pp. 308–324, Sep. 2015, doi: [10.1016/j.trc.2015.02.019](https://doi.org/10.1016/j.trc.2015.02.019).
- [34] X. Li and R. Bai, "Freight vehicle travel time prediction using gradient boosting regression tree," in *Proc. 15th IEEE Int. Conf. Mach. Learn. Appl.*, Dec. 2016, pp. 1010–1015, doi: [10.1109/ICMLA.2016.0182](https://doi.org/10.1109/ICMLA.2016.0182).
- [35] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001, doi: [10.1214/aos/1013203451](https://doi.org/10.1214/aos/1013203451).

- [36] L. Breiman, J. H. Friedman, and R. Olshen, "Classification and regression trees," *Encyclopedia Ecol.*, vol. 57, no. 1, pp. 582–588, 1984, doi: [10.2307/2530946](https://doi.org/10.2307/2530946).
- [37] M. Schmid and T. Hothorn, "Flexible boosting of accelerated failure time models," *Bioinformatics*, vol. 9, no. 1, p. 269, 2008, doi: [10.1186/1471-2105-9-269](https://doi.org/10.1186/1471-2105-9-269).
- [38] M. Schmid, T. Hothorn, K. O. Maloney, D. E. Weller, and S. Potapov, "Geoadditive regression modeling of stream biological condition," *Environ. Ecological Statist.*, vol. 18, no. 4, pp. 709–733, 2011, doi: [10.1007/s10651-010-0158-4](https://doi.org/10.1007/s10651-010-0158-4).
- [39] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers Neuroinformatics*, vol. 7, no. 7, pp. 1–21, 2013, doi: [10.3389/fnbot.2013.00021](https://doi.org/10.3389/fnbot.2013.00021).
- [40] J. Elith, J. R. Leathwick, and T. Hastie, "A working guide to boosted regression trees," *J. Animal Ecol.*, vol. 77, no. 4, pp. 802–813, 2010, doi: [10.1111/j.1365-2656.2008.01390.x](https://doi.org/10.1111/j.1365-2656.2008.01390.x).
- [41] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324).
- [42] M. S. Ghanim and G. Abu-Lebdeh, "Real-time dynamic transit signal priority optimization for coordinated traffic networks using genetic algorithms and artificial neural networks," *J. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 327–338, 2015, doi: [10.1080/15472450.2014.936292](https://doi.org/10.1080/15472450.2014.936292).
- [43] G. Zhang and Y. Wang, "Optimizing minimum and maximum green time settings for traffic actuated control at isolated intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 1, pp. 164–173, Mar. 2011, doi: [10.1109/TITS.2010.2070795](https://doi.org/10.1109/TITS.2010.2070795).
- [44] F. Zheng and H. V. Zuylen, "The development and calibration of a model for urban travel time distributions," *J. Intell. Transp. Syst.*, vol. 18, no. 1, pp. 81–94, 2014, doi: [10.1080/15472450.2013.802155](https://doi.org/10.1080/15472450.2013.802155).
- [45] Y. Xie, H. Zhang, N. H. Gartner, and T. Arsava, "Collaborative merging strategy for freeway ramp operations in a connected and autonomous vehicles environment," *J. Intell. Transp. Syst.*, vol. 21, no. 2, pp. 136–147, 2017, doi: [10.1080/15472450.2016.1248288](https://doi.org/10.1080/15472450.2016.1248288).
- [46] X. Hua, W. Wang, Y. Wang, and Z. Pu, "Optimizing phase compression for transit signal priority at isolated intersections," *Transport*, vol. 32, no. 4, pp. 386–397, 2017, doi: [10.3846/16484142.2017.1345787](https://doi.org/10.3846/16484142.2017.1345787).
- [47] A. Louati, S. Darmoul, S. Elkosantini, and L. B. Said, "An artificial immune network to control interrupted flow at a signalized intersection," *Inf. Sci.*, vols. 433–434, pp. 70–95, Apr. 2018, doi: [10.1016/j.ins.2017.12.033](https://doi.org/10.1016/j.ins.2017.12.033).
- [48] *HCM2010: Highway Capacity Manual*, 5th ed., National Research Council, Washington, DC, USA, 2010.
- [49] M. Gualtieri, C. A. Rowan, and K. TaKeaways, "The forrester wave: Big data predictive analytics solutions, Q1 2013," Forrester Res., Cambridge, MA, USA, Tech. Rep., 2013.
- [50] F. Zheng and H. V. Zuylen, "Urban link travel time estimation based on sparse probe vehicle data," *Transp. Res. C, Emerg. Technol.*, vol. 31, pp. 145–157, Jun. 2013, doi: [10.1016/j.trc.2012.04.007](https://doi.org/10.1016/j.trc.2012.04.007).
- [51] C.-H. Wu, J.-M. Ho, and D. T. Lee, "Travel-time prediction with support vector regression," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 4, pp. 276–281, Dec. 2004, doi: [10.1109/TITS.2004.837813](https://doi.org/10.1109/TITS.2004.837813).



JUAN CHENG received the bachelor's degree from Shandong Jiaotong University, in 2006, and the master's degree from East China Jiaotong University, in 2009. She is currently pursuing the Ph.D. degree with Southeast University, Nanjing, China. She has presented her work at the Conference of ICTTE. Her main research areas include the intelligent transportation systems, traffic flow theory, and traffic simulation.



GEN LI received the bachelor's degree from Southeast University, in 2010, where he is currently pursuing the Ph.D. degree in transportation engineering. He has published his works in several journals and conferences. His research interests include microscopic traffic flow theory, traffic simulation, and machine learning.



XIANHUA CHEN received the bachelor's degree from Chang'an University and the master's and Ph.D. degrees from Southeast University, Nanjing, China. He is currently a Professor with Southeast University. He has published his research works in several international journals and conferences. His research interests include functional pavement, high-performance road construction materials, vehicle interaction, and asphalt roadbed and high railway foundation protection technology.

He has chaired and attended several international conferences.

• • •