

Article

Prediction Intervals for Bus Travel Time Based on Road Segment Sharing, Multiple Routes' Driving Style Similarity, and Bootstrap Method

Zhenzhong Yin ^{1,*} , Bin Wang ¹ , Bin Zhang ² and Xinpu Shen ^{3,*} 

¹ School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China; bin.wang.brilliant@gmail.com

² Software College, Northeastern University, Shenyang 110819, China; zhangbin@mail.neu.edu.cn

³ Institute of Reservoir Engineering, College of Petroleum Engineering, China University of Petroleum (Huadong), Qingdao 266580, China

* Correspondence: yinzhenzhong@stumail.neu.edu.cn (Z.Y.); 20180011@upc.edu.cn (X.S.)

Abstract: Providing accurate information about bus travel times can help passengers plan their itinerary and reduce waiting time. However, due to various uncertainty factors and the sparsity of single-route data, traditional travel time predictions cannot accurately describe the credibility of the prediction results, which is not conducive to passengers waiting based on the predicted results. To address the above issues, this paper proposes a bus travel time prediction intervals model based on shared road segments, multiple routes' driving style similarity, and the bootstrap method. The model first divides the predicted route into segments, dividing adjacent stations shared by multiple routes into one section. Then, the hierarchical clustering algorithm is used to group all drivers in multiple bus routes in this section according to their driving styles. Finally, the bootstrap method is used to construct a bus travel time prediction interval for different categories of drivers. The travel time data sets of Shenyang 239, 134, and New Area Line 1 were selected for experimental verification. The experimental results indicate that the quality of the prediction interval constructed using a data set fused with multiple routes is better than that constructed using a single-route data set. In the two cases studied, the MPIW of the three time periods decreased by 101.04 s, 151.72 s, 33.87 s, and 126.58 s, 127.47 s, 17.06 s, respectively.

Keywords: bus travel time prediction; prediction intervals; multiple bus routes; hierarchical clustering; driving style similarity



Citation: Yin, Z.; Wang, B.; Zhang, B.; Shen, X. Prediction Intervals for Bus Travel Time Based on Road Segment Sharing, Multiple Routes' Driving Style Similarity, and Bootstrap Method. *Appl. Sci.* **2024**, *14*, 2935. <https://doi.org/10.3390/app14072935>

Academic Editors: Ryan Cheuk Pong Wong, Fangni Zhang and Jintao Ke

Received: 19 February 2024

Revised: 25 March 2024

Accepted: 28 March 2024

Published: 30 March 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the continuous acceleration of urbanization and the expansion of the urban scale, the number of motor vehicles and the road traffic flow have sharply increased. Especially in some large and medium-sized cities, traffic congestion is very serious, which brings great inconvenience to citizens' travel. Improving the share rate of public transport is an effective way to solve urban traffic problems, establish the dominant position of urban public transportation, and achieve efficient use of limited road traffic resources [1]. In the process of giving priority to the development of public transport, how to ensure the normal and efficient operation of the public transport system depends not only on the conditions of roads, vehicles, and other facilities but also on the progressiveness of operation management means and technical means. To organize the operation of public transport vehicles more scientifically and reasonably, and to help travelers better plan their itinerary and reduce their waiting time, accurate prediction of bus travel time has become an essential key link [2,3].

The goal of predicting bus travel time is to predict the time it takes for the bus to pass through two points, such as between two stations [3]. The travel time between two

adjacent bus stops also includes the time for bus vehicles to queue up at the intersection and wait to pass through the intersection. Therefore, the factors that affect the accuracy of predicting bus travel time are multifaceted, for example, traffic conditions [4,5], weather factors [4,6], traffic incidents [7], differences in driver's driving behavior [8], changes in passenger flow [9], etc. These uncertain factors increase the uncertainty of bus travel and arrival time. This will cause bus bunching [10] and negative impacts such as increased bus operating costs and longer waiting times for passengers [11,12].

In previous studies, various machine learning-based methods have been proposed for predicting bus travel time. However, most previous studies mainly focused on improving the prediction accuracy of the bus travel time point prediction model. The main drawback of the bus travel time point prediction model is that it cannot provide the confidence interval information of the prediction results [13]. For example, when we travel by bus according to the prediction results of the point prediction model, we will encounter the following two situations: (1) if the predicted results are larger than the actual travel time, we will miss the bus. (2) If the predicted result is smaller than the actual driving time, we may need to wait for a long time [8].

Some scholars have proposed that the prediction intervals (PIs) can be used to quantify the uncertainty of bus travel time point prediction. From the perspective of travelers, predicting the possible range of bus travel time is more meaningful than providing only predicted values for a single arrival time point [14]. Usually, the PIs consist of a lower limit and an upper limit, where the true value falls within these limits. In previous studies, PI techniques mainly include Bayesian techniques [15], Delta techniques [14,15], the bootstrap method [16,17], and the upper and lower bound estimation method (LUBE) [18]. According to previous research results, when the data set is small, the PIs constructed based on the bootstrap method are usually more accurate [19,20].

In addition, the travel time of buses varies greatly at a specific time of the day, and there is also a significant difference in traffic volume between different segments of the road, which significantly affects the accuracy of the travel time prediction. At present, most previous studies use the same prediction model to predict the travel time of the whole line. This method can not accurately predict the travel time of different sections in different periods [21]. At the same time, due to the long interval between bus departures on a single route in the real world, the sparsity of the bus travel time data set results in suboptimal quality of the PIs generated solely from a single route data set [8]. Previous studies have recognized that if several bus routes share a portion of the route, they can benefit from each other's predictions, as in [3,22,23]. Based on inspiration from the above research, this paper integrates driving data from multiple routes and proposes a bus travel time PIs model based on the similarity of multiple routes' driving styles and the bootstrap method. The effectiveness of this method was verified through real-world data.

The main contributions of this paper can be summarized as follows.

- We investigated the impact of multiple routes' driving style similarity on travel time PIs and proposed a new method to predict bus travel time PIs;
- We found that segmenting the predicted route is beneficial for improving the quality of bus travel time PIs;
- Compared to previous studies, the method proposed in this article has been validated for its effectiveness through real-world data, which can further improve the quality of PIs.

The rest of this paper is organized as follows. Section 2 summarizes the related work. The process of the bootstrap method used to construct PIs is introduced in Section 3. Section 4 introduces the model framework and the construction steps of the bus travel time PIs model. Section 5 presents a discussion of the results of the experiment. The conclusion of this paper is summarized in Section 6.

2. Related Work

Machine learning has been widely applied to predict the travel time of buses. Multiple machine learning-based prediction methods have been proposed in the previous literature, including linear regression (LR) [24,25], support vector regression (SVR) [3,7,26,27], artificial neural network (ANN) [3,24,28], random forest (RF) [6,29], the Kalman filter (KF) [30,31], and deep learning prediction models [4,21,32–34]. Among them, the LR models are only used for comparison with other models in most studies [33]. SVR and ANN have become the most widely used models because of their strong nonlinear fitting ability and ability to map complex nonlinear relationships. For example, in [7], the authors demonstrated for the first time the feasibility of applying SVR to predict travel time and demonstrated that SVR is suitable for traffic data analysis. Reddy et al. [11] used support vector machines and V-vector regression with linear kernel functions for prediction. In [35], the authors proposed a bus travel time prediction model with a forgetting factor based on SVM. In [29], the authors used three different algorithms, namely random forest, projection pursuit regression, and support vector machine, to carry out extensive experiments on each route in the study. In [27], the authors proposed a new road segment-based method for predicting bus travel time, selecting the basic prediction model by comparing the performance of SVM, ANN, and k-NN. In [6], the authors proposed the neighbor-based random forest (RFNN) method to predict bus travel time, which has been calibrated and validated with real-world data. Although the results of RFNN show high accuracy, its long computation time is currently not suitable for real-time prediction. To improve the prediction accuracy of travel time and avoid the disadvantage of using a single model, many studies have tried to use hybrid models to predict the travel time of buses. For example, reference [36–39] added the KF algorithm to machine learning-based methods. In [23], the authors proposed a new model combining queuing theory and machine learning to predict bus travel time. Compared to a single model, the results of mixed models indicate that they can provide better performance.

The above machine learning-based methods are all based on shallow learning architectures, but they struggle to handle the correlation between spatiotemporal data factors, and the relationships between these factors are more complex and nonlinear [21]. In recent years, researchers have proposed some research based on deep neural networks. In [32], the authors proposed using recurrent neural networks (RNNs) to predict the arrival time of buses by utilizing long-range correlations between multiple time steps. Petersen et al. [33] utilized the nonstationary spatiotemporal correlations present in urban public transportation networks to propose a multi-output, multi-time-step, deep neural network that utilizes a combination of convolutional and long short-term memory (LSTM) layers to discover complex patterns that traditional methods cannot capture. In [4], the authors proposed a method (TP-SCF) to automatically learn the different traffic conditions of different bus routes and train a separate prediction model (LSTM) for each different traffic mode to improve prediction accuracy. In [34], the authors developed a Geo-conv LSTM model that can extract the subsequent spatial features of the entire bus travel sequence through a 1D convolutional neural network (CNN) while also capturing the temporal dependencies between sub-sequences through the LSTM network. Khaled et al. [21] proposed using the NMF algorithm to group route links with similar traffic patterns into different groups and training separate CNN models for each group to improve the accuracy of model predictions.

Unfortunately, although all of the above studies strive to improve the accuracy of bus travel time prediction, these studies cannot provide information about travel time such as confidence level. Even the most advanced models currently cannot accurately predict the travel time of buses and can only minimize errors as much as possible. Currently, only a few articles have studied the problem of bus travel time PI prediction. Reference [14] studied the application of delta technology in constructing travel time prediction intervals for buses and highways and proposed a neural network model based on a genetic algorithm, which has the method of automatic selection and adjustment of hyperparameters. Khosravi et al. [15] constructed travel time prediction intervals for buses and highways using Bayesian and

Delta methods. The experimental results show that the PIs constructed using Bayesian techniques have stronger robustness to neural network structures and good coverage probability, while the delta method is superior to the Bayesian method in terms of PI narrowness. In [13], the authors demonstrated a bootstrap method based on the maximum likelihood technique to construct PIs. The authors quantified the contribution of each source of uncertainty to the overall prediction uncertainty. Finally, the effectiveness of the proposed method was verified by predicting the travel time of Melbourne's bus routes in Australia. In [8], the authors constructed a bus travel time PIs prediction model based on driving style similarity using hierarchical clustering technology and the bootstrap method and applied it to the travel time PIs prediction of one bus route in Shenyang. Table 1 provides a comprehensive review of research on bus travel time point prediction and PI prediction.

Table 1. Overview of bus travel time point prediction and PI prediction.

Paper	Data Type	Method	Spatial	Temporal	Cluster	Target
[24]	AVL	ANN, LR, HA	2 routes	6 months	No	Point Prediction
[3]	Surveys data	SVM, ANN, K-NN, LR	8 routes	3 days	No	Point Prediction
[26]	Unknown	SVM, ANN	1 route	1 month	No	Point Prediction
[27]	GPS	SVM, ANN, K-NN	3 routes	1 month	Yes	Point Prediction
[28]	GPS	HA, ANN, KF	1 route	6 months	Yes	Point Prediction
[6]	GPS	RFNN	2 routes	3 days	No	Point Prediction
[29]	Unknown	RF, PPR, SVM	6 routes	3 months	No	Point Prediction
[30]	AVL, APC	KF	1 route	5 days	No	Point Prediction
[31]	GPS	KF	1 route	30 days	No	Point Prediction
[38]	APC	ANN + KF	1 route	12 months	No	Point Prediction
[39]	On-board survey	SVM + KF	1 route	1 month	No	Point Prediction
[4]	GPS	LSTM	30 routes	2 months	Yes	Point Prediction
[21]	GPS	CNN	5 routes	6 months	Yes	Point Prediction
[32]	GPS	RNN	47 routes	1 month	No	Point Prediction
[33]	GPS	ConvLSTM	1 route	5 months	No	Point Prediction
[34]	GPS	Geo-convLSTM	2 routes	3 months	No	Point Prediction
[14]	GPS	Delta	1 route	6 months	No	PI Prediction
[15]	GPS	Delta, Bayesian	1 route	6 months	No	PI Prediction
[13]	GPS	Bootstrap + ANN	1 route	6 months	No	PI Prediction
[8]	GPS	Bootstrap + ANN	1 route	1 month	Yes	PI Prediction
Ours	GPS	Bootstrap + ANN	3 routes	1 month	Yes	PI Prediction

Note: AVL—automatic vehicle location, APC—automatic passenger counter, HA—Historical Average.

3. Construction of PIs Using the Bootstrap Method

3.1. Mathematical Description of the Problem

Due to the limitation of the bus departure time interval (10–30 min), only one data record of the travel time can be extracted during each time cycle from departure to arrival. Therefore, the travel time data of buses have typical characteristics of small samples. It is reasonable to use the bootstrap method to predict bus travel time PIs. Referring to reference [16,40], assuming the existence of a set of data $D = \{(x_i, y_i), i = 1, 2, \dots, N\}$, there is a non-linear mapping relationship $y(x_i)$ between the target value y and the input variable x . Taking into account the measurement error, the travel time can be expressed as:

$$y_i^* = y(x_i) + \epsilon(x_i) \quad (1)$$

where y_i^* indicates the measured value, $y(x_i)$ represents the true value of the i -th sampling, and $\epsilon(x_i)$ represents noise. Assuming that the output of the prediction model is $\hat{y}_i(x_i)$, the error of the model can be expressed as:

$$y_i^* - \hat{y}_i = [y(x_i) - \hat{y}_i(x_i)] + \epsilon(x_i) \quad (2)$$

where $y(x_i) - \hat{y}_i(x_i)$ represents the error between the predicted value of the model and the real value, $\epsilon(x_i)$ represents the noise of the data, the predicted variance can be expressed as:

$$\sigma_y^2(x_i) = \sigma_{\hat{y}}^2(x_i) + \sigma_{\epsilon}^2(x_i) \quad (3)$$

If $\sigma_{\hat{y}}^2(x_i)$ and $\sigma_{\epsilon}^2(x_i)$ are estimated, then $\sigma_y^2(x_i)$ can be calculated, and then the prediction intervals can be estimated based on the variance value. Next, in Section 3.2, we will introduce the process of using the bootstrap method to estimate $\sigma_{\hat{y}}^2(x_i)$ and $\sigma_{\epsilon}^2(x_i)$.

3.2. Bootstrap Methodology

The bootstrap method is a computer technology-based resampling method proposed by Professor Efron [41]. The method of constructing confidence intervals and prediction intervals is commonly used to simulate the population distribution by resampling observed data with replacement and generating regenerated samples. Compared to other methods, this method can generate more reliable PI without the need to calculate complex matrices [16].

Suppose that there is a data sample $X = \{x_1, x_2, \dots, x_n\}$ with a sample capacity of N , and B samples with a capacity of M are extracted from the original sample X . $X^* = \{x_1^*, x_2^*, \dots, x_m^*\}$, usually $M = N$, finally producing B bootstrap samples $X_1^*, X_2^*, \dots, X_B^*$. The value of B is generally within the range of 20~200 and can meet the requirements of most applications [19]. In this paper, $B = 30$ is selected.

The mean of point prediction of travel time for B samples using artificial neural network models is [40]:

$$\hat{y}(x_i) = \frac{1}{B} \sum_{b=1}^B \hat{y}_b(x_i) \quad (4)$$

Among them, $\hat{y}_b(x_i)$ is the prediction of the i -th sample generated by the b -th bootstrap model. Then, use the variance of the predicted results from the B models to estimate the model variance [16,40].

$$\sigma_{\hat{y}}^2(x_i) = \frac{1}{B-1} \sum_{b=1}^B (\hat{y}_b(x_i) - \hat{y}(x_i))^2 \quad (5)$$

In order to construct PI, it is also necessary to estimate the variance of the error $\sigma_{\epsilon}^2(x_i)$ [16]:

$$\sigma_{\epsilon}^2(x_i) = E\{(y_i^* - \hat{y}(x_i))^2\} - \sigma_{\hat{y}}^2(x_i) \quad (6)$$

The sum of squared residuals is [40]:

$$r^2(x_i) = \max\left((y_i^* - \hat{y}(x_i))^2 - \sigma_{\hat{y}}^2(x_i), 0\right) \quad (7)$$

where $\hat{y}(x_i)$ and $\sigma_{\hat{y}}^2(x_i)$ can be calculated from Equations (4) and (5). Combine residuals with input variable set samples to build a new data set $D_{r2} = \{(x_i, r^2(x_i))\}_{i=1}^n$. Through the data set D_{r2} , train a new neural network, that is, the $L + 1$ learning model to estimate $\sigma_{\epsilon}^2(x_i)$. In order to maximize the probability of the observed sample in D_{r2} , therefore, the new model introduces the maximum likelihood estimation method and establishes a new objective function to train the model. Defined as [40]

$$C_{BS} = \frac{1}{2} \sum_{i=1}^n \left(\ln(\sigma_{\epsilon}^2(x_i)) + \frac{r^2(x_i)}{\sigma_{\epsilon}^2(x_i)} \right) \quad (8)$$

After estimating $\sigma_{\hat{y}}^2(x_i)$ and $\sigma_{\epsilon}^2(x_i)$, the PIs with a confidence level of $(1 - \alpha)\%$ can be constructed [16]:

$$U^{(\alpha)}(x_i) = \hat{y}(x_i) + t_{df}^{1-\frac{\alpha}{2}} \sqrt{\sigma_{\hat{y}}^2(x_i) + \sigma_{\epsilon}^2(x_i)} \quad (9)$$

$$L^{(\alpha)}(x_i) = \hat{y}(x_i) - t_{df}^{1-\frac{\alpha}{2}} \sqrt{\sigma_y^2(x_i) + \sigma_e^2(x_i)} \quad (10)$$

where $U^{(\alpha)}(x_i)$ and $L^{(\alpha)}(x_i)$ are the upper and lower boundaries that construct PI, respectively. $t_{df}^{1-\frac{\alpha}{2}}$ is the $1 - \alpha/2$ quantile of the t distribution function with the degree of freedom df . Usually, the value of df is set to B .

4. Proposed Method

4.1. Model Architecture

The architecture of the bus travel time PI model based on the similarity of driving styles across multiple routes and the bootstrap method proposed in this article is shown in Figure 1. This mainly includes dividing the predicted route into segments, using hierarchical clustering technology to construct the original sample data set D_{HC} , and then using the bootstrap method to resample the original sample data set and generate prediction intervals. Refer to Section 4.2 for detailed steps.

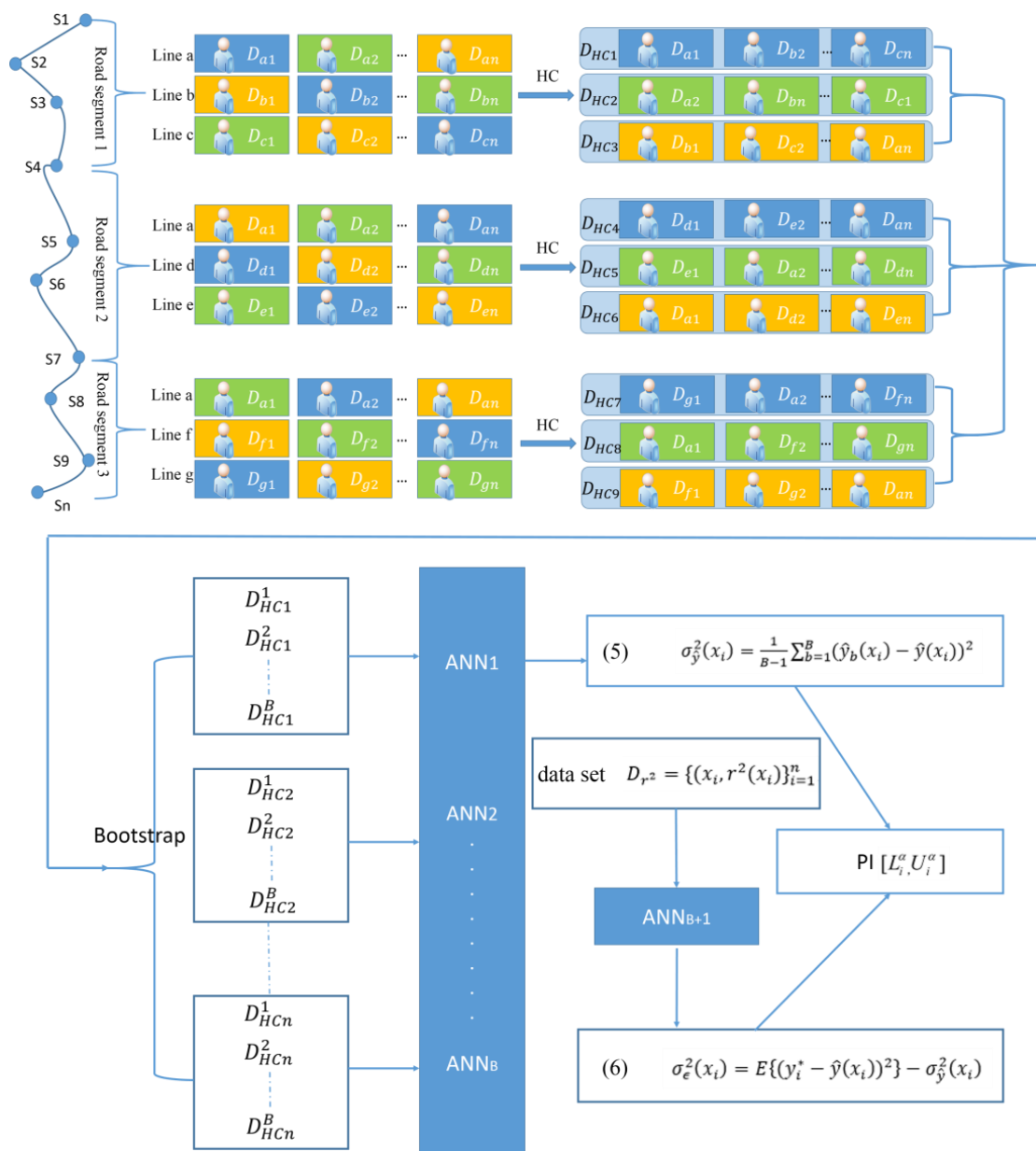


Figure 1. Model architecture.

4.2. Specific Procedures

The specific steps to construct a prediction interval for bus travel time are as follows:

(1) Division of time period and road segment. In order to more accurately classify bus drivers on multiple routes into different types according to their driving styles, this study considers both spatial and temporal characteristics. First, divide the time periods. Divide a day into three time periods: morning peak hour (7:00–9:00), off-peak hour (9:00–16:00), and evening peak hour (16:00–19:00). The purpose of time division is to ensure that bus drivers have similar patterns in their travel time within the same time period. Second, divide the predicted bus route into sub-segments. The segment between two bus stops with multiple routes having duplicate segments is used as the basic prediction unit.

(2) Construct the original sample data set D_{HC} of bus travel time using the hierarchical clustering method. Perform hierarchical clustering on all drivers on sub-road segments according to their driving styles, and use the travel time data sets of different categories of drivers as the original sample data set D_{HC} , which includes n sub-sample data sets $D_{HC1} \dots D_{HCn}$, each representing the historical travel time data set of a category of drivers over a period of time.

(3) Resampling of the original sample data set. The bootstrap method is used to resample each subsample set in data set D_{HC} B times (where $B = 30$) with replacement, resulting in B bootstrap sample sets.

(4) Prediction of travel time points based on ANN. In this step, select a three-layer neural network, including the input layer, the hidden layer, and the output layer. The factors (inputs) considered in this study are the following: X_1 day of the week, X_2 road segment number, and X_3 departure time. The resampled sample data set is trained to obtain B trained ANN models, and then the test data set is used to predict the B point prediction results and their squared residuals $r^2(x_i)$.

(5) Training the $b + 1$ neural network model. According to Equation (7), construct a squared residual data set $D_{r^2} = \{(x_i, r^2(x_i))\}_{i=1}^n$ to train the $b+1$ neural network model. Use the test sample data set and the above-trained models to obtain the prediction value, prediction model variance, and variance of the errors.

(6) Constructing travel time prediction intervals for sub-segments. Construct a travel time prediction interval for the corresponding road segment based on Equations (9) and (10).

5. Experiments

5.1. Data Collection

The data in this study were provided by Shenyang Municipal Bureau of Big Data. The software used in this study mainly includes IBM SPSS Statistics 26 and MATLAB R2016a. Experiments were conducted using three bus routes in Shenyang, namely 239, 134, and New Area Line 1. Each bus in the above three routes is equipped with a GPS positioning device, which collects bus position data every 5 s. The onboard serial number of each bus is unique and corresponds to a fixed bus driver. This study takes the segment of the 239 bus route from the 4th station (Jianshe Road Weigong Street) to the 13th station (Peace Square) as the main predicted segment. The traffic flow and the number of passengers in different directions on the same road section are not the same, which will lead to a difference in travel time. Therefore, this study uses the driving data in the same direction (from west to east). Some parts of the driving segments of the three bus routes are shared. According to the shared parts, the predicted segment is divided into three sub-segments, as shown in Figure 2. Among them, sub-segment 1 spans four bus stations from Jianshe Road Weigong Street Station to Tiexi Square Station. There are 239 bus routes and the new district line 1 overlapping with sub-segment 1. Sub-segment 2 spans four bus stations from Tiexi Square Station to Shengli South Street Station on Nanwu Road. In sub-segment 2, there are three bus routes with overlapping driving intervals: 239, 134, and New Area Line 1. Sub-segment 3 runs from Shengli South Street Station on Nanwu Road to Peace Square Station and also spans four bus stops. There are overlapping driving segments of 239 and 134 in sub-segment 3.

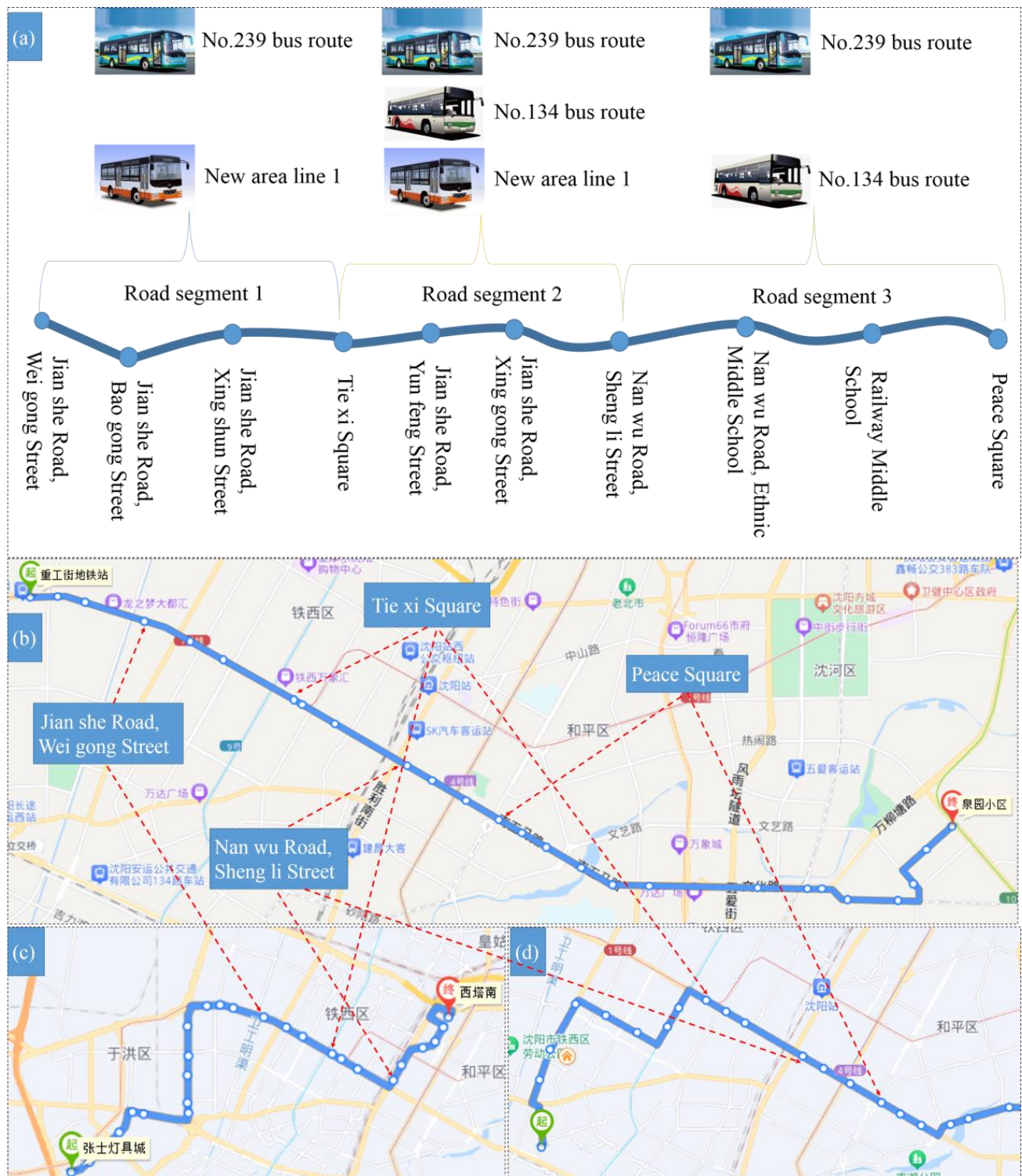


Figure 2. (a) Schematic diagram of multiple routes' shared road segment division, (b) route map of bus No. 239, (c) route map of new area line 1, (d) route map of bus No. 134.

Among them, the travel time data set of 9 drivers was selected for each route as experimental data, and 27 drivers for a total of 3 routes is shown in Table 2. The experimental data are from 4 to 22 January 2016 (Monday to Friday). The data set is divided into three sub-data sets: D1 (4–15 January 2016), D2 (18–20 January 2016) and D3 (21–22 January 2016). Data sets D1 and D2 are used for the training data set, and data set D3 is the test sample.

Table 2. List of driver codes for 3 routes.

No. 239	No. 134	New Area Line 1
902334	903687	902708
902335	903688	902709
902340	903689	902710
902347	903692	902711
902349	903696	902713
902351	903698	902718
902353	903708	902722
902355	903712	902724
902359	903690	902730

5.2. PI Assessment Indexes

In related research [14,16], the most important feature of PIs is their coverage probability. PI coverage probability (*PICP*) is measured by calculating the number of covered target values [16]:

$$PICP = \frac{1}{N} \sum_{i=1}^N C_i \quad (11)$$

where:

$$C_i = \begin{cases} 1, & \text{if } t_i \in [L_i, U_i] \\ 0, & \text{if } t_i \notin [L_i, U_i] \end{cases} \quad (12)$$

where N is the number of samples. L_i and U_i represent the upper and lower bounds of the PI corresponding to the i -th sample.

Another important indicator is the mean prediction interval width (*MPIW*), which is used to quantify the width of PI [14]. The definition is as follows:

$$MPIW = \frac{1}{n} \sum_{i=1}^n (U(X_i) - L(X_i)) \quad (13)$$

Meanwhile, in some literature [14,42], two additional evaluation indicators are also used to supplement the evaluation of PI, namely *NMPIW* and *CWC*. Assuming that the target range R is known, normalized *MPIW* (*NMPIW*) can be calculated as follows [14]:

$$NMPIW = \frac{MPIL}{R} \quad (14)$$

NMPIW represents the percentage of the average width of PIs in the underlying target.

Reference [42] developed a coverage width criterion (*CWC*) consisting of *PICP* and *NMPIW*, which simultaneously evaluates PIs from both coverage probability and width perspectives:

$$CWC = NMPIW * \left(1 + \gamma(PICP)e^{(-\eta(PICP-\mu))} \right) \quad (15)$$

where $\gamma(PICP)$ is given by:

$$\gamma = \begin{cases} 0, & PICP \geq \mu \\ 1, & PICP < \mu \end{cases} \quad (16)$$

In Equation (15), η and μ are the two hyperparameters that control *CWC*, as suggested in reference [18]. The η And μ values are set to 50 and 0.9, respectively. *CWC* provides an effective compromise between the information and the precision of the PI. It is worth noting that the smaller the *CWC*, the better [16].

5.3. Results and Analysis

5.3.1. Clustering Results of Driving Styles for Multiple Routes Drivers

We use a hierarchical clustering algorithm to classify the driving style of drivers, which can help us discover the internal relationships and hierarchical structure between data

objects without the need to pre-set the number of clustering categories. The result is shown in Figures 3–5. Referring to previous research [8,10], through an experimental comparison of clustering results at different levels, the hierarchical clustering results divided by blue lines in the figure are selected as an example. By comparing the subgraphs in Figures 3–5, it can be found that the driving style of the same driver on the same road segment is not the same at different periods. For example, the clustering results of driver 902335 on road segment 1 are as follows: during the morning peak period (7:00–9:00), the driving style of drivers 902713, 902730, 902349, 902718, 902724, and 902347 is similar. During off-peak hours (9:00–16:00), the driving style of drivers numbered 902334, 902349, 902340, 902347, and 902353 is similar. During the evening peak hours (16:00–19:00), the driving style of the drivers numbered 902349, 902708, 902722, 902351, 902334, and 902347 is similar. This result indicates that the division of periods is very necessary. At the same time, the results of the clustering of driving styles on different segments of the road during the same time period are also different. Taking driver number 902335 as an example in Figure 3, during the morning peak period (7:00–9:00), the clustering results of 902335 on road segment 1 show that the driving styles of drivers with numbers 902713, 902730, 902349, 902718, 902724, and 902347 are similar. On road segment 2, the driving style of driver 902335 is similar to that of drivers numbered 902351, 902359, 902353, 902722, 902724, and 902713. On road segment 3, the driving style of driver 902335 is only similar to that of driver 902334. This result indicates that the division of the predicted route into segments is also very important.

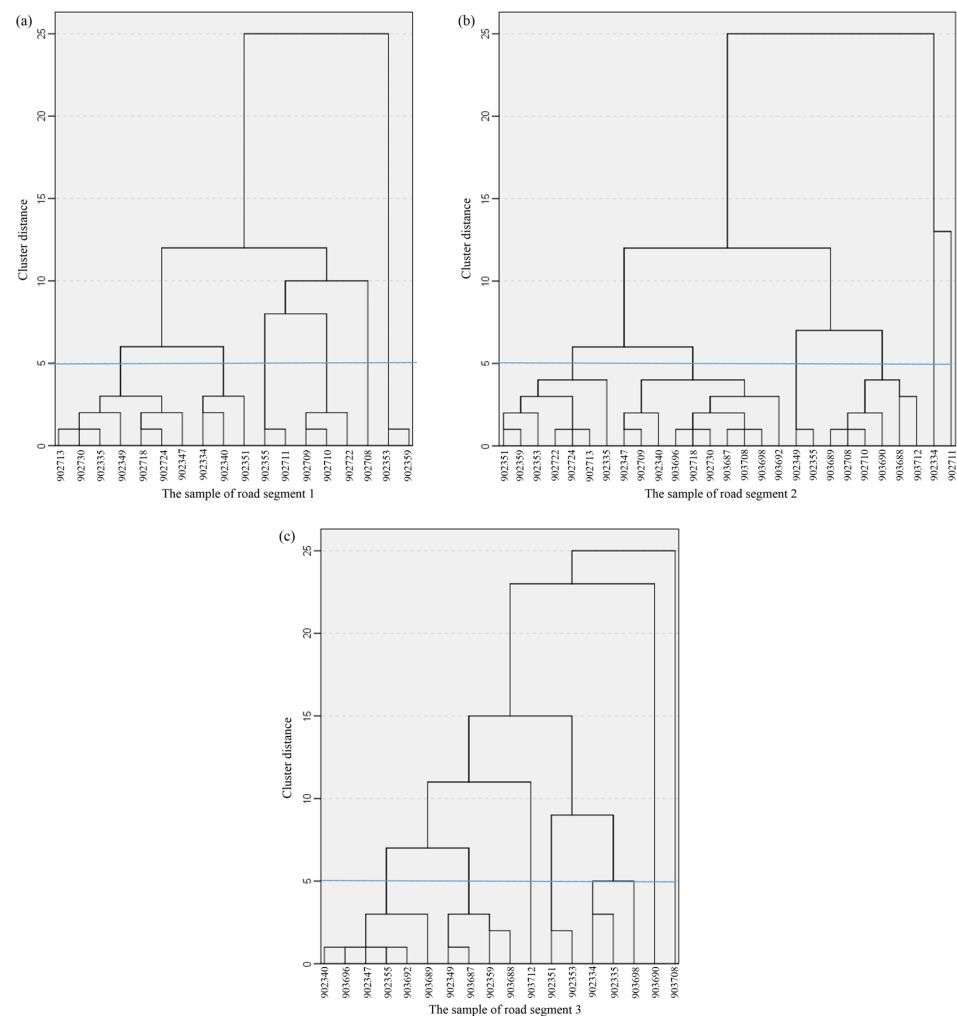


Figure 3. The clustering results during the morning peak time (7:00–9:00) of the three road segments. (a) Result of road segment 1, (b) result of road segment 2, (c) result of road segment 3.

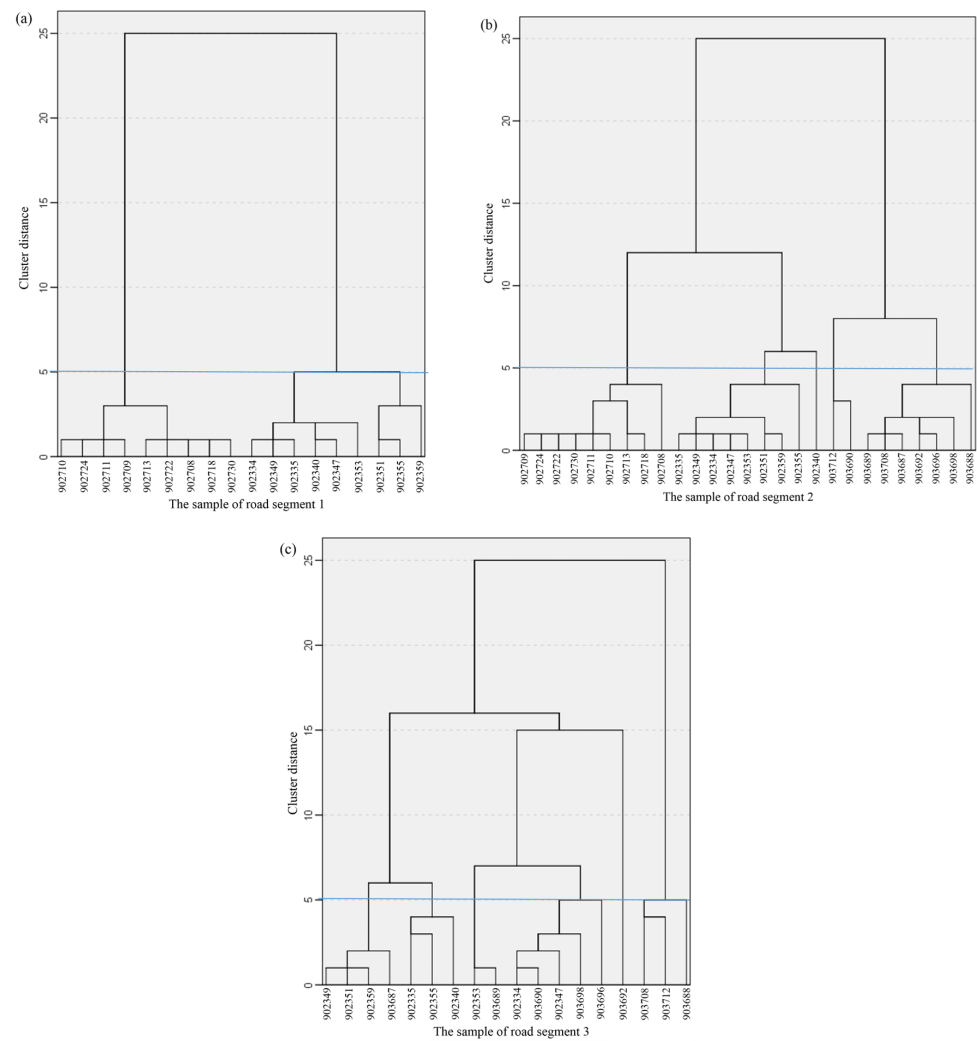


Figure 4. The clustering results during off-peak time (9:00–16:00) of the three road segments. (a) Result of road segment 1, (b) result of road segment 2, (c) result of road segment 3.

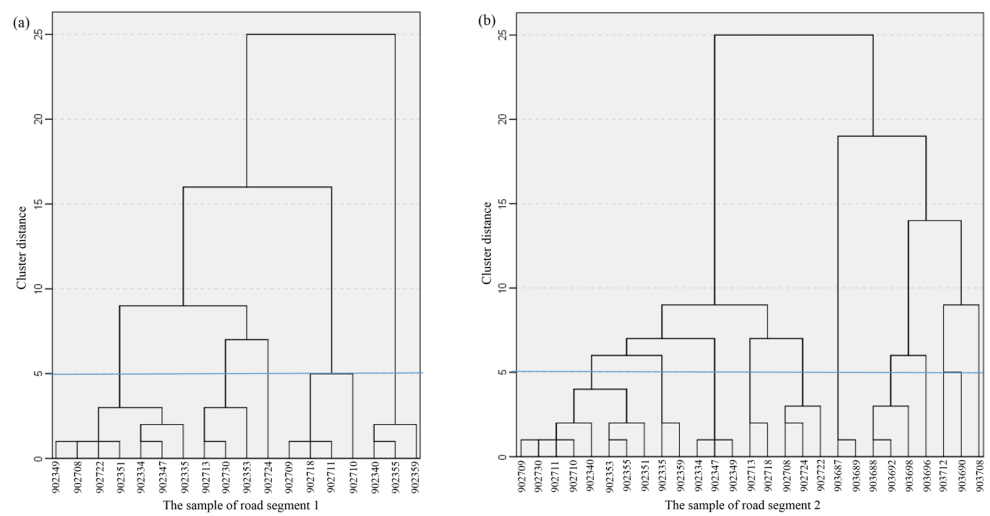


Figure 5. Cont.

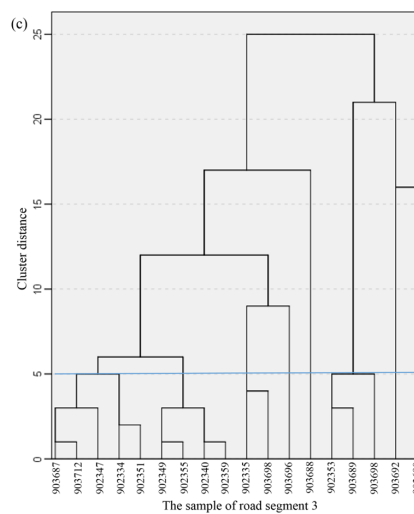


Figure 5. The clustering results during the evening peak time (16:00–19:00) of the three road segments. (a) Result of road segment 1, (b) result of road segment 2, (c) result of road segment 3.

5.3.2. Travel Time Interval Prediction Results

According to the driving style clustering results in Section 5.3.1, two drivers numbered 902335 and 902359 are selected as the travel time PI prediction objects in this section. The departure schedule of the two drivers to be predicted from 21 January 2016 to 22 January 2016 is shown in Table 3.

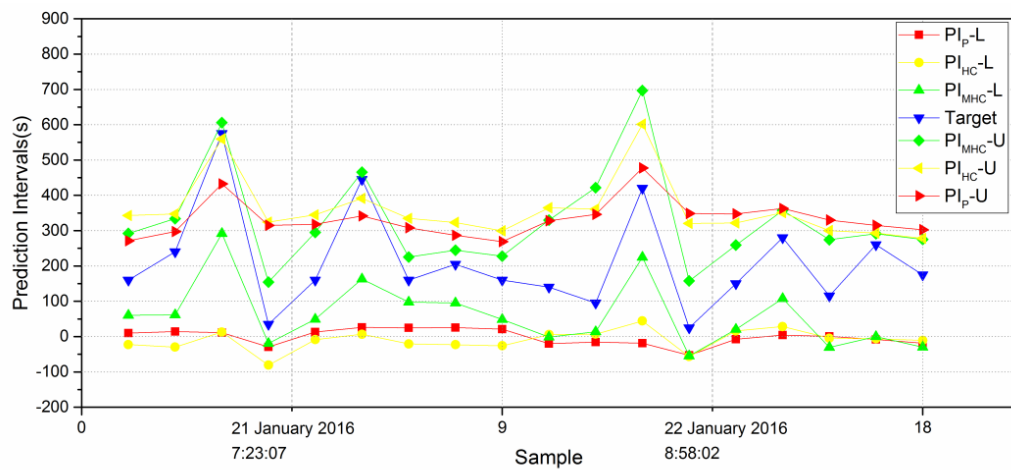
Table 3. The predicted departure schedule of the drivers.

Time Period	21 January 2016		22 January 2016	
	902335	902359	902335	902359
7–9	7:23:07	7:13:38	8:58:02	7:05:45
9–16	10:06:04	9:57:11	11:55:28	9:37:12
	14:19:48	12:57:05 15:14:05	14:38:04	12:45:11 15:58:15
16–19	16:42:22	17:36:00	17:00:06	

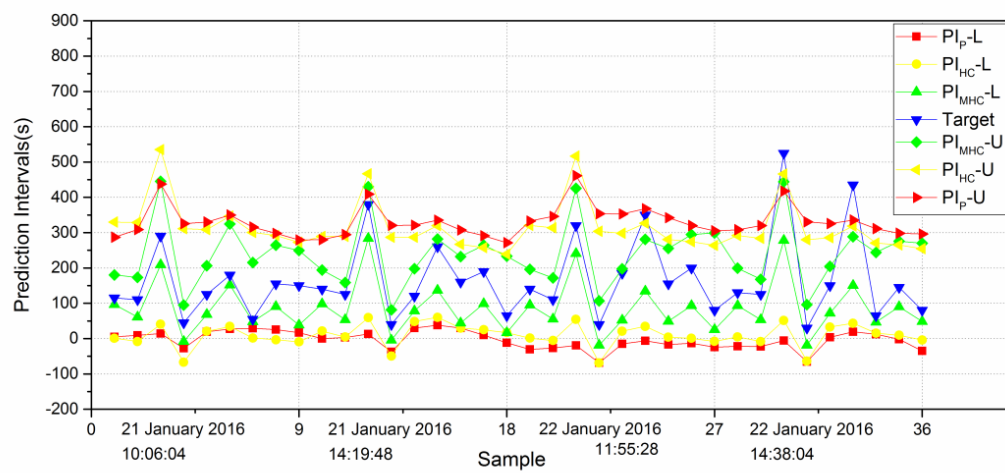
The experiment used the MATLAB R2016a software package to build PIs running on a Core I7–4790, 3.6–GHZ CPU with 8–GB RAM. Each experiment was repeated 20 times, and the average value was calculated as the final result. Meanwhile, to obtain high-confidence prediction information, we set the confidence level to 90%. The predicated PI results are shown in Figures 6 and 7, respectively.

To compare the effectiveness of the proposed method, we compared it with the reference results [8], which used a data set of Route 239 to construct personalized travel time prediction intervals and prediction intervals after driving style clustering, respectively. For simplicity, we use P, HC, and MHC subscripts to represent the PI results generated from a single driver data set, a single bus route hierarchical clustering data set, and a multiple bus routes hierarchical clustering data set, respectively. For example, taking 902335 as an example, PI_P represents the PI result generated using only the data set of drivers with ID 902335. PI_{HC} represents the PI result generated by the hierarchical clustering of driving styles using all drivers on a single route (239). PI_{MHC} represents the PI result generated by hierarchical clustering of all drivers on multiple routes according to their driving styles.

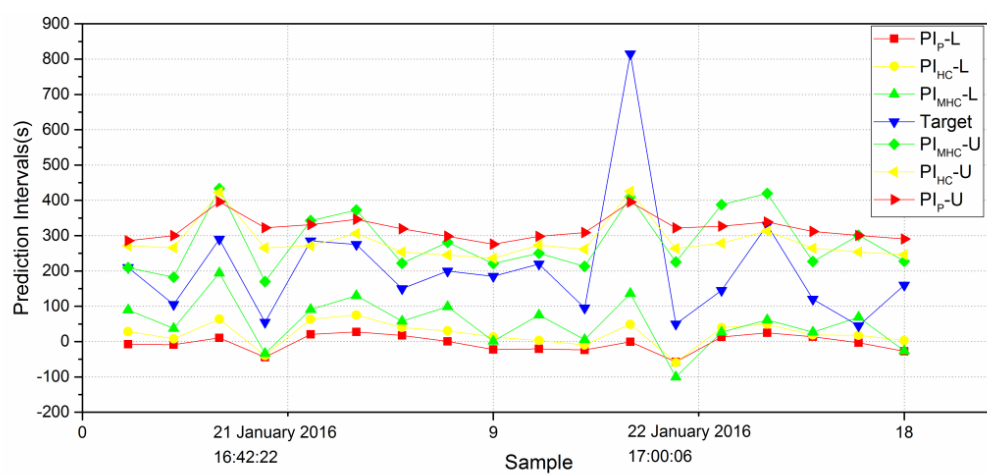
The results of the travel time PIs of driver 902335 are shown in Figure 6. The prediction results are divided into three subgraphs according to time periods, where subgraphs a, b, and c, respectively, show the PIs results of driver 902335 during morning peak hours, off-peak hours, and evening peak hours.



(a)

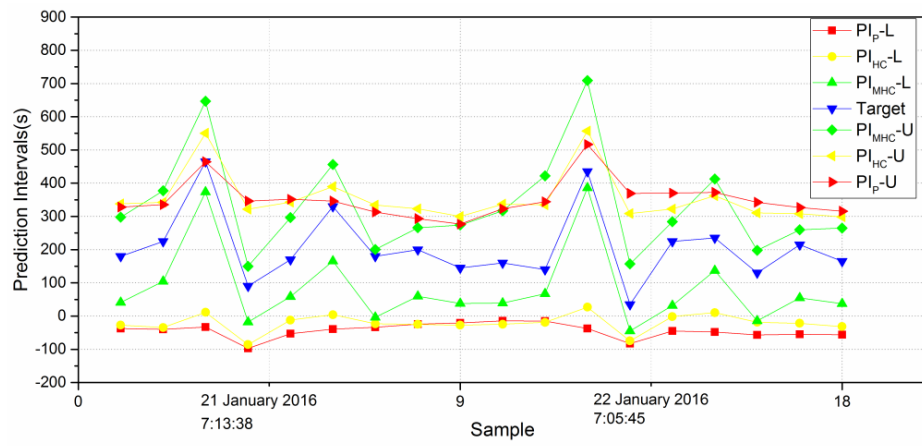


(b)

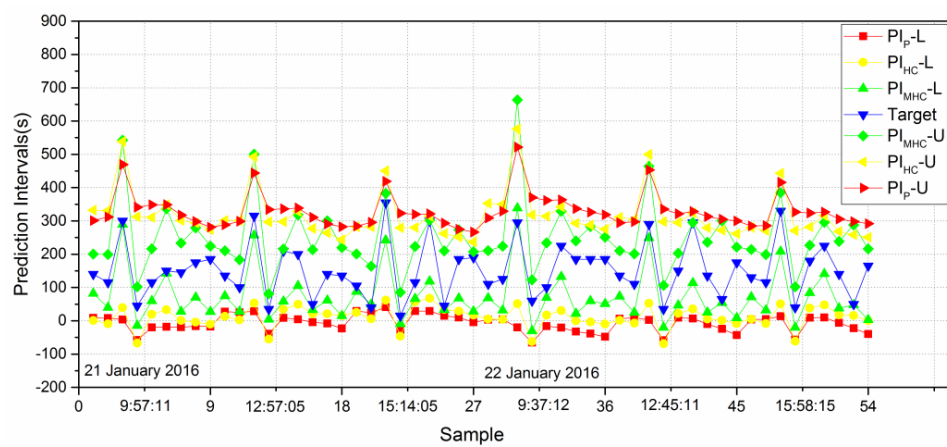


(c)

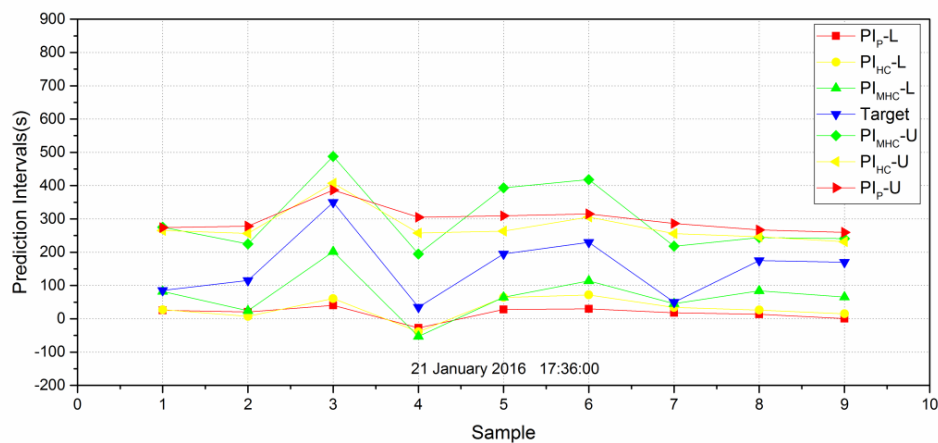
Figure 6. PI results of 902335. (a) The morning peak time (7:00–9:00), (b) the off-peak time (9:00–16:00), (c) the evening peak time (16:00–19:00).



(a)



(b)



(c)

Figure 7. PI results of 902359. (a) The morning peak time (7:00–9:00), (b) the off-peak time (9:00–16:00), (c) the evening peak time (16:00–19:00).

It is not difficult to see from the results that the prediction results of the three models PI_{MHC} , PI_{HC} , and PI_P can all include most of the true values within the prediction interval. Meanwhile, the width of PI_{MHC} based on multiple routes sharing prediction results is significantly narrower than that of PI_{HC} and PI_P in the three time periods. And the

predicted values can track the changes in the measured values well. During the morning peak time, as shown in Figure 6a, only the predicted interval of PI_{MHC} fully includes the true values, and the predicted interval width is the narrowest. This indicates that the prediction interval of PI_{MHC} is significantly better than the other two models. During the off-peak time, as shown in Figure 6b, as the fluctuation range of the true value decreases, the predicted interval width of PI_{MHC} is significantly narrower compared to the other two periods. This indicates that PI_{MHC} prediction performs better during periods of stable traffic flow. However, when there is a drastic fluctuation in the actual value, as shown in Figure 6c during the evening peak period, the prediction error is relatively large and the true value falls outside the range. We believe that the traffic situation at this time has been affected by an unexpected event.

Figure 7a–c show the interval prediction results of driver 902359 during morning peak hours, off-peak hours, and evening peak hours, respectively. It can also be seen from the results that the prediction results of the three models PI_{MHC} , PI_{HC} , and PI_P can all include most of the true values within the prediction interval. At the same time, the width of PI_{MHC} is significantly narrower than PI_{HC} and PI_P in the prediction results based on the sharing of multiple route segments in three time periods. And the predicted values can track the changes in the measured values well. However, there was no significant fluctuation in the actual value and almost all true values were within the range.

Next, we will quantitatively evaluate the generated PI and further analyze the effectiveness of interval prediction. By calculating the PICP, MPIW, NMPIW, and CWC, the accuracy of prediction results for each interval can be accurately compared.

Table 4 shows the values of PICP, MPIW, NMPIW and CWC for the prediction interval when the confidence level is 90%. It can be seen that almost all predicted PICP values for each PI of PI_{MHC} exceed the set confidence level (90%). In all cases, the MPIW value of PI_{MHC} is smaller than the corresponding MPIW values of PI_{HC} and PI_P . For example, in case 902335, the MPIW values in three time periods decreased by 101.04 s, 151.72 s, and 33.87 s in PI_{MHC} compared to PI_{HC} , respectively. Compared to PI_P , it decreased by 67.25 s, 180.29 s, and 96.79 s, respectively, indicating that the predicted interval width significantly narrowed without reducing the PICP. The values of the corresponding NMPIW also decreased, with PI_{MHC} decreasing by 18.44, 17.43, and 11.08 compared to PI_{HC} , respectively. Compared to PI_P , it decreased by 27.5, 38.87, and 28.69, respectively. Meanwhile, the corresponding CWC values were also smaller (decreased by 102.09, 17.43, and 1210.65, respectively). Similarly, in Table 4, it can be observed that the predicted results of case 902359 show a similar trend over three time periods. The value of MPIW was decreased by 126.58 s, 127.47 s, and 17.06 s in PI_{MHC} compared to PI_{HC} , respectively. Compared to PI_P , it decreased by 147.16 s, 158.4 s, and 51.27 s, respectively. The corresponding NMPIW values also decreased, with PI_{MHC} decreasing by 25.42, 21.66, and 12.82 compared to PI_{HC} , respectively. Compared to PI_P , it decreased by 31.53, 31.65, and 25.41, respectively. Meanwhile, the corresponding CWC value is also smaller. Furthermore, it indicates that the interval prediction quality of the method proposed in this article is better, which can shorten the waiting time of passengers and provide more reliable waiting time recommendations.

Table 4. PIs characteristics for test samples when the confidence level is 90%.

Case Study	Model	PICP (%)	MPIW	NMPIW (%)	CWC
902335	7:00–9:00	PI_P	88.89	63.03	160.73
		PI_{HC}	88.89	53.97	137.62
		PI_{MHC}	100.00	35.53	35.53
	9:00–16:00	PI_P	94.44	72.05	72.05
		PI_{HC}	91.67	50.61	50.61
		PI_{MHC}	91.67	33.18	33.18
	16:00–19:00	PI_P	94.44	71.73	71.73
		PI_{HC}	83.33	54.12	1320.4
		PI_{MHC}	88.89	43.04	109.75

Table 4. *Cont.*

Case Study	Model	PICP (%)	MPIW	NMPIW (%)	CWC
902359	7:00–9:00	PI _P	94.44	395.85	64.51
		PI _{IHC}	100.00	375.27	58.4
		PI _{MHC}	100.00	248.69	32.98
	9:00–16:00	PI _P	100.00	336.43	57.31
		PI _{IHC}	100.00	305.5	47.32
		PI _{MHC}	96.3	178.03	25.66
	16:00–19:00	PI _P	100.00	281.1	67.91
		PI _{IHC}	100.00	246.89	55.32
		PI _{MHC}	100.00	229.83	42.5

6. Conclusions

Travel time is an important indicator in intelligent transportation systems, and currently, most research on travel time focuses on deterministic prediction, requiring decision-makers to evaluate the reliability of results through the error value of the model. This article conducts in-depth research on the uncertainty of travel time prediction. To further improve the quality of bus travel time PIs, this paper outlines a multi-faceted approach comprising segmentation of routes, clustering of drivers based on driving styles, and bootstrap method application. We propose a bus travel time PIs model based on road sharing and driving style similarity for multiple routes. The superiority of the proposed method was demonstrated through experiments on real data from multiple bus routes in Shenyang city. We found that the width of PIs based on multiple routes' segment sharing significantly narrows. In the two cases studied, the MPIW for the three time periods decreased by 101.04 s, 151.72 s, 33.87 s, and 126.58 s, 127.47 s, 17.06 s, respectively. And the predicted values can better track the changes in the measured values. The findings suggest potential implications for improving public transportation systems, emphasizing the importance of incorporating shared data and driving behavior analysis for more accurate predictions. The results of this study can serve as a reference for public transportation management agencies to arrange the operation of public transportation vehicles more scientifically and reasonably and also help travelers better plan their itinerary and reduce their waiting time.

Future research can be conducted from the following two aspects: (1) combining deep learning models to explore the impact of deep-level spatiotemporal factors on travel time prediction intervals; (2) developing a component for the prediction of traffic accidents to enhance the predictive ability of existing models in dealing with unexpected situations.

Author Contributions: Conceptualization, B.Z. and Z.Y.; methodology, Z.Y.; software, B.W.; validation, Z.Y. and B.W.; writing—original draft preparation, Z.Y.; writing—review and editing, Z.Y.; supervision, X.S.; project administration, B.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Key Project of the National Natural Science Foundation of China (U1908212), the Central Government-Guided Local Science and Technology Development Fund Project (1653137155953), and Liaoning Province “Takes the Lead” Science and Technology Research Project (2021jh1/10400006).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The authors wish to express their gratitude to Shenyang Municipal Bureau of Big Data for providing the experimental data set.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bhat, C.R.; Sardesai, R. The impact of stop-making and travel time reliability on commute mode choice. *Transp. Res. Part B Meth.* **2006**, *40*, 709–730. [\[CrossRef\]](#)
2. Song, X.; Tian, J.; Tao, P.; Li, H.; Wu, C. Traffic State Estimation of Bus Line with Sparse Sampled Data. *IEEE Access* **2020**, *8*, 216127–216140. [\[CrossRef\]](#)
3. Yu, B.; Lam, W.H.K.; Tam, M.L. Bus arrival time prediction at bus stop with multiple routes. *Transp. Res. Part C Emerg. Technol.* **2011**, *19*, 1157–1170. [\[CrossRef\]](#)
4. He, P.; Jiang, G.; Lam, S.K.; Sun, Y. Learning heterogeneous traffic patterns for travel time prediction of bus journeys. *Inf. Sci.* **2020**, *512*, 1394–1406. [\[CrossRef\]](#)
5. Huang, Y.P.; Chen, C.; Su, Z.C.; Chen, T.S.; Sumalee, A.; Pan, T.L.; Zhong, R.X. Bus arrival time prediction and reliability analysis: An experimental comparison of functional data analysis and Bayesian support vector regression. *Appl. Soft Comput.* **2021**, *111*, 107663. [\[CrossRef\]](#)
6. Yu, B.; Wang, H.; Shan, W.; Yao, B. Prediction of bus travel time using random forests based on near neighbors. *Comput. Aided Civ. Infrastruct. Eng.* **2017**, *33*, 333–350. [\[CrossRef\]](#)
7. Wu, C.-H.; Ho, J.-M.; Lee, D.T. Travel-time prediction with support vector regression. *IEEE Trans. Intell. Transp. Syst.* **2004**, *5*, 276–281. [\[CrossRef\]](#)
8. Yin, Z.; Zhang, B. Construction of Personalized Bus Travel Time Prediction Intervals Based on Hierarchical Clustering and the Bootstrap Method. *Electronics* **2023**, *12*, 1917. [\[CrossRef\]](#)
9. Shalaby, A.; Farhan, A. Bus travel time prediction model for dynamic operations control and passenger information systems. In Proceedings of the 82nd Annual Meeting of the Transportation Research Board, Washington, DC, USA, 12–16 January 2003.
10. Yin, Z.; Zhang, B. Bus Travel Time Prediction Based on the Similarity in Drivers' Driving Styles. *Future Internet* **2023**, *15*, 222. [\[CrossRef\]](#)
11. Reddy, K.K.; Kumar, B.A.; Vanajakshi, L. Bus travel time prediction under high variability conditions. *Curr. Sci.* **2016**, *111*, 700–711. [\[CrossRef\]](#)
12. O'Sullivan, A.; Pereira, F.C.; Zhao, J.; Koutsopoulos, H.N. Uncertainty in Bus Arrival Time Predictions: Treating Heteroscedasticity With a Metamodel Approach. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3286–3296. [\[CrossRef\]](#)
13. Mazloumi, E.; Rose, G.; Currie, G.; Moridpour, S. Prediction intervals to account for uncertainties in neural network predictions: Methodology and application in bus travel time prediction. *Eng. Appl. Artif. Intell.* **2011**, *24*, 534–542. [\[CrossRef\]](#)
14. Khosravi, A.; Mazloumi, E.; Nahavandi, S.; Creighton, D.; Van Lint, J.W.C. A genetic algorithm-based method for improving quality of travel time prediction intervals. *Transp. Res. Part C Emerg. Technol.* **2011**, *19*, 1364–1376. [\[CrossRef\]](#)
15. Khosravi, A.; Mazloumi, E.; Nahavandi, S.; Creighton, D.; Van Lint, J.W.C. Prediction intervals to account for uncertainties in travel time prediction. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 537–547. [\[CrossRef\]](#)
16. Khosravi, A.; Nahavandi, S.; Srinivasan, D.; Khosravi, R. Constructing Optimal Prediction Intervals by Using Neural Networks and Bootstrap Method. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *26*, 1810–1815. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Khosravi, A.; Nahavandi, S.; Creighton, D. Quantifying uncertainties of neural network-based electricity price forecasts. *Appl. Energy* **2013**, *112*, 120–129. [\[CrossRef\]](#)
18. Khosravi, A.; Nahavandi, S.; Creighton, D.; Atiya, A.F. Lower upper bound estimation method for construction of neural network-based prediction intervals. *IEEE Trans. Neural Netw.* **2011**, *22*, 337–346. [\[CrossRef\]](#)
19. Efron, B.; Tibshirani, R.J. *An Introduction to the Bootstrap*; Chapman & Hall: New York, NY, USA, 1993.
20. Lins, I.D.; Droguett, E.L.; Moura, M.D.C.; Zio, E.; Jacinto, C.M. Computing confidence and prediction intervals of industrial equipment degradation by bootstrapped support vector regression. *Reliab. Eng. Syst. Saf.* **2015**, *137*, 120–128. [\[CrossRef\]](#)
21. Alkilane, K.; Alfateh, M.T.E.; Shen, Y. Travel time prediction based on route links' similarity. *Neural. Comput. Appl.* **2023**, *35*, 3991–4007. [\[CrossRef\]](#)
22. Bai, C.; Peng, Z.-R.; Lu, Q.-C.; Sun, J. Dynamic Bus Travel Time Prediction Models on Road with Multiple Bus Routes. *Comput. Intell. Neurosci.* **2015**, *2015*, 432389. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Gal, A.; Mandelbaum, A.; Schnitzler, F.; Senderovich, A.; Weidlich, M. Traveling time prediction in scheduled transportation with journey segments. *Inf. Syst.* **2017**, *64*, 266–280. [\[CrossRef\]](#)
24. Jeong, R.; Rilett, L.R. Bus arrival time prediction using artificial neural network model. In Proceedings of the 7th IEEE Intelligent Transportation System Conference, Washington, DC, USA, 3–6 September 2004. [\[CrossRef\]](#)
25. Patnaik, J.; Chien, S.; Bladikas, A. Estimation of bus arrival times using APC data. *J. Public Transp.* **2004**, *7*, 1–20. [\[CrossRef\]](#)
26. Yu, B.; Yang, Z.Z.; Yao, B.Z. Bus arrival time prediction using support vector machines. *J. Intell. Trans. Syst.* **2006**, *10*, 151–158. [\[CrossRef\]](#)
27. Ma, J.; Chan, J.; Ristanoski, G.; Rajasegarar, S.; Leckie, C. Bus travel time prediction with real-time traffic information. *Transp. Res. Part C Emerg. Technol.* **2019**, *105*, 536–549. [\[CrossRef\]](#)
28. Fan, W.; Gurmu, Z. Dynamic Travel Time Prediction Models for Buses Using Only GPS Data. *Int. J. Transp. Sci. Technol.* **2015**, *4*, 353–366. [\[CrossRef\]](#)
29. Mendes-Moreira, J.; Jorge, A.M.; Freire de Sousa, J.; Soares, C. Improving the accuracy of long-term travel time prediction using heterogeneous ensembles. *Neurocomputing* **2015**, *150*, 428–439. [\[CrossRef\]](#)

30. Shalaby, A.; Farhan, A. Prediction models of bus arrival and departure times using AVL and APC data. *J. Public Transp.* **2004**, *7*, 41–61. [\[CrossRef\]](#)
31. Vanajakshi, L.; Subramanian, S.C.; Sivanandan, R. Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses. *IET Intell. Transp. Syst.* **2009**, *3*, 1–9. [\[CrossRef\]](#)
32. Pang, J.; Huang, J.; Du, Y.; Yu, H.; Huang, Q.; Yin, B. Learning to Predict Bus Arrival Time from Heterogeneous Measurements via Recurrent Neural Network. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 3283–3293. [\[CrossRef\]](#)
33. Petersen, N.C.; Rodrigues, F.; Pereira, F.C. Multi-output bus travel time prediction with convolutional LSTM neural network. *Expert Syst. Appl.* **2019**, *120*, 426–435. [\[CrossRef\]](#)
34. Lee, G.; Choo, S.; Choi, S.; Lee, H. Does the Inclusion of Spatio-Temporal Features Improve Bus Travel Time Predictions? A Deep Learning-Based Modelling Approach. *Sustainability* **2022**, *14*, 7431. [\[CrossRef\]](#)
35. Yu, B.; Ye, T.; Tian, X.-M.; Ning, G.-B.; Zhong, S.-Q. Bus travel- time prediction with a forgetting factor. *J. Comput. Civ. Eng.* **2014**, *28*, 06014002. [\[CrossRef\]](#)
36. Liu, H.; Van Lint, H.; Van Zuylen, H.; Zhang, K. Two distinct ways of using kalman filters to predict urban arterial travel time. In Proceedings of the 2006 IEEE Intelligent Transportation Systems Conference (ITSC), Toronto, ON, Canada, 17–20 September 2006; pp. 845–850.
37. Van Lint, J.W.C. Incremental and online learning through extended kalman filtering with constraint weights for freeway travel time prediction. In Proceedings of the 2006 IEEE Intelligent Transportation Systems Conference (ITSC), Toronto, ON, Canada, 17–20 September 2006; pp. 1041–1046.
38. Chen, M.; Liu, X.; Xia, J.; Chien, S.I. A dynamic bus-arrival time prediction model based on APC data. *Comput.-Aided Civ. Infrastruct. Eng.* **2004**, *19*, 364–376. [\[CrossRef\]](#)
39. Yu, B.; Yang, Z.-Z.; Chen, K.; Yu, B. Hybrid model for prediction of bus arrival times at next station. *J. Adv. Transp.* **2010**, *44*, 193–204. [\[CrossRef\]](#)
40. Heskes, T. Practical confidence and prediction intervals. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 2–5 December 1996; MIT Press: Cambridge, MA, USA, 1997; pp. 176–182.
41. Efron, B. Bootstrap methods: Another look at the jackknife. *Ann. Stat.* **1979**, *7*, 1–26. [\[CrossRef\]](#)
42. Khosravi, A.; Nahavandi, S.; Creighton, D. Construction of optimal prediction intervals for load forecasting problem. *IEEE Trans. Power Syst.* **2010**, *25*, 1496–1503. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.