Journal of Electronic Research and Application, 2021, Volume 5, Issue 6
http://ojs.bbwpublisher.com/index.php/JERA
ISSN Online: 2208-3510
ISSN Print: 2208-3502

RHMX: Bus Arrival Time Prediction via Mixed Model

Fei Jia[1], Huibing Zhang[1], Xiaoli Hu[2]*

[1]Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, Guangxi Zhuang Autonomous Region, China

[2]Teaching Practice Department, Guilin University of Electronic Technology, Guilin 541004, Guangxi Zhuang Autonomous Region, China

****Corresponding author:*** Xiaoli Hu, huxiaoli@guet.edu.cn

Abstract: With the widespread use of information technologies such as IoT and big data in the transportation business, traditional passenger transportation has begun to transition and upgrade into intelligent transportation, providing passengers with a better riding experience. Giving precise bus arrival times is a critical link in achieving urban intelligent transportation. As a result, a mixed model-based bus arrival time prediction model (RHMX) was suggested in this work, which could dynamically forecast bus arrival time based on the input data. First, two sub-models were created: bus station stopping time prediction and interstation running time prediction. The former predicted the stopping time of a running bus at each downstream station in an iterative manner, while the latter projected its running time on each downstream road segment (stations as the break points). Using the two models, a group of time series data on interstation running time and bus station stopping time may be predicted. Following that, the time series data from the two sub-models was fused using long short-term memory (LSTM) to generate an approximate bus arrival time. Finally, using Kalman filtering, the LSTM prediction results were dynamically updated in order to eliminate the influence of aberrant data on the anticipated value and obtain a more precise bus arrival time. The experimental findings showed that the suggested model's accuracy and stability were both improved by 35% and 17%, respectively, over AutoNavi and Baidu.

Keywords: IoT; Big data; Bus arrival time prediction; Long short-term memory; Kalman filtering

Publication date: November 2021; ***Online publication:*** November 30, 2021

1. Introduction

With the evolution of urban public transportation systems and changes in user travel preferences, providing consumers with exact bus arrival time prediction service has become a vital component in the development of intelligent transportation. In order to construct a city-level real-time dynamic bus transfer system and execute the intelligent transportation concept of "Mobility as a Service," bus arrival time must be properly forecasted. As a result, bus arrival time prediction technology has become a hotspot for study, which may be examined in the following ways:

(1) Bus arrival time prediction model based on historical data

The historical data model was first used to estimate bus arrival times. Based on historical data, such a model uses a big data platform to assess the bus running laws in each time frame, and then collects bus running time in each time frame [1-5]. The historical data model is based on the original data, which means that the data from the earlier moment must not alter significantly from the current data. Any emergency circumstance may cause data to be modified rapidly, resulting in a significant loss of accuracy, therefore the stability is weak.

(2) Bus arrival time prediction model based on Kalman filtering

According to the optimal estimated value at the previous moment, the Kalman filtering algorithm can forecast the present optimal estimated value. Because of its powerful antijamming capacity, this model has a high level of stability. When utilized to address the bus arrival time prediction problem, it is cable to reduce the disturbance from the external environment, with a relatively high forecast accuracy [6-7]. Furthermore, it may dynamically alter the projected bus arrival time, resulting in good timeliness; nevertheless, the prediction effect decreases as the number of iterations increases.

(3) Bus arrival time prediction model based on particle filtering

Particle filtering, which can solve the problems existing in the Kalman filtering algorithm, approximately expresses a probability density function by seeking for a group of random samples propagated in a state space, and replaces integral operation with sample mean, so as to acquire the minimum variance estimate of the system state [8]. The bus arrival time prediction problem has been investigated by many scholars through particle filtering [9-12], with good nonlinear processing capacity, but the sample diversity will be degraded due to the resampling process.

(4) Bus arrival time prediction model based on regression model

A regression model is a mathematical equation that is used to objectively assess the relationship between independent variables and dependent variables in order to establish the causal relationship. The dependent variable, bus arrival time, is often predicted based on influencing factors such as weather and distance between road segments, which are independent variables [13-14]. However, because it is difficult to determine, this mathematical model is rarely employed.

(5) Bus arrival time prediction model based on neural network

Even if no formal model is constructed, neural networks can be used to tackle some practical problems due to its strong adaptative ability, generalization ability, and nonlinear processing capacity. When it comes to bus arrival time prediction, neural network models may learn not only the relationships between different vehicles on the same route at different times, but also the spatial relationships between distinct road segments. Many researchers have looked into this [15-18], and high prediction accuracy has been established.

2. Model Introduction

2.1. Feature engineering

The prediction of bus stopping time is influenced by both station dwell time and bus running time, which are affected mainly by the following factors:

The related data were collected specific to the above influencing factors, and a total of 708,270 records were included. In order to reduce the prediction error, the data of all holidays and festivals were eliminated, while only the data of working days and rest days were reserved. The GPS data reissued after a delay were placed into the GPS series in chronological order. The data points (totally 1,387) with GPS abnormality were eliminate, and the overall data size collected in this study could be neglected. The GPS data without weather factors were supplemented by historical weather data.

The bus arrival time was influenced by a variety of parameters, as shown in **Table 1.**, although only a few of them were effective. As a result, elements from the bus arrival time prediction model have to be chosen carefully in order to extract the most useful features from the original data.

(1) Feature selection of bus stopping time prediction model

In this study, multiple hypothesis testing was adopted as the feature selection model for this model. In order to judge the correlations between features and variables, F value should be calculated through formulas (1) and (2):

$$ri = \frac{(X[:,i] - mean(X[:,i]))(y - mean(y))}{std(X[:,i])std(y)} \tag{1}$$

$$F = \frac{r_i^2}{1 - r_i^2} * (n-2) \tag{2}$$

Where $X[:,i]$ represents the feature i, $mean(X[:,i])$ is the mean value of feature i, y is the target value, $mean(y)$ stands for the mean value of target value, $std(X[:,i])$ is the standard deviation of feature i, and $std(y)$ is the standard deviation of target value. P value could be calculated according to the F value and degree of freedom, which served as the basis for feature selection. Through the parameter adjustment, it was suitable to take the significance level as 0.042. In other words, the original hypothesis was rejected if the P value was smaller than 0.042, and the feature was correlated with the dependent variable. If the P value was greater than 0.042, the feature was uncorrelated with the dependent variable. The P values of all features are listed in **Table 2**.

Table 1. Influencing Factors of Bus Arrival Time Prediction

Influencing factor	Concrete type
Weather factors	Temperature, wind direction, wind power, humidity, reporting time, concrete weather type
Vehicle factors	Vehicle No., vehicle direction, longitude and latitude of vehicle position, next station No., bus distance to the next station, instantaneous speed of bus, signal intensity at vehicle terminal, running distance on the same day, GPS uploading time
Station factors	Station longitude and latitude, quantity of institutions of higher learning around station, quantity of middle schools, quantity of primary schools, quantity of scenic spots, quantity of bus routes, quantity of dining places, quantity of shopping places
Time factors	Date type (holidays and festivals, working days, rest days) and time frames
Road factors	Road segment No., quantity of traffic lights, quantity of confluences, length of road segment, quantity of institutions of higher learning, quantity of middle schools, quantity of primary schools, quantity of kindergartens, quantity of scenic spots around road segment, quantity of bus routes on the road segment, quantity of dining places around road segment, quantity of shopping places around road segment

Table 2. P Values of Features of Bus Stopping Time Prediction Model

Feature	P value	Whether to choose the feature	Feature	P value	Whether to choose the feature	Feature	P value	Whether to choose the feature
Bus No.	0.021	Yes	Wind power	0.033	Yes	Quantity of kindergartens	0.036	Yes
Next station No.	0.025	Yes	Humidity	0.032	Yes	Quantity of dining places	0.022	Yes

Signal intensity	0.065	No	Date type	0.029	Yes	Quantity of shopping places	0.029	Yes
Running distance	0.038	Yes	Quantity of station routes	0.031	Yes	Station longitude	0.028	Yes
Time frame	0.025	Yes	Quantity of scenic spots	0.036	Yes	Station latitude	0.025	Yes
Weather	0.045	No	Quantity of institutions of higher learning	0.026	Yes			
Tempera ture	0.036	Yes	Quantity of middle schools	0.031	Yes			
Wind direction	0.032	Yes	Quantity of primary schools	0.032	Yes			

As seen in **Table 2**, there were totally 19 features with P values greater than 0.042. It is common sense that signal intensity is only associated with data transmission quality but unrelated to bus stopping time prediction. Only six weather conditions were considered in this study, namely, light rain/overcast/cloudy/sunny/rainy/foggy. The weather features could already characterize the temperature, wind direction, wind power and air humidity, so they were excluded.

(2) Feature selection of bus interstation running time prediction model

Multiple hypothesis testing was still selected for the feature selection of bus interstation running time. Through the parameter adjustment, a suitable significance level was 0.05, and the P values of all features are presented in **Table 3**:

Table 3. P Values of Features of Running Time Prediction Model

Feature	**P value**	**Whether to choose the feature**	**Feature**	**P value**	**Whether to choose the feature**	**Feature**	**P value**	**Whether to choose the feature**
Bus No.	0.021	Yes	Data uploading time	0.041	Yes	Quantity of confluen ces	0.036	Yes
Direction of bus head	0.025	Yes	Weather	0.039	Yes	Length of road segment	0.032	Yes

Current bus longitude	0.04	Yes	Temperature	0.053	No	Quantity of bus routes	0.039	Yes
Current bus latitude	0.038	Yes	Wind direction	0.054	No	Quantity of scenic spots	0.033	Yes
Next station No.	0.025	Yes	Wind power	0.054	No	Quantity of institutions of higher learning	0.035	Yes
Current speed	0.045	Yes	Air humidity	0.056	No	Quantity of middle schools	0.044	Yes
Signal intensity	0.066	No	Date type	0.041	Yes	Quantity of primary schools	0.045	Yes
Bus running distance on the same day	0.042	Yes	Quantity of traffic lights	0.04	Yes	Quantity of kindergartens	0.042	Yes
Quantity of dinning places around road segment	0.032	Yes	Quantity of shopping places around road segment	0.036	Yes			

(3) Acquisition of bus interstation running time

In order to select a proper bus interstation running time prediction model, the approximate value of bus interstation running time should be acquired from historical data through the following method:

Algorithm 1: Estimation of running time on road segment

```
Input: Original GPS dataset RawGpsData$_i$={gpsdata$_1$, gpsdata$_2$,…, gpsdata$_n$}
Output: Bus running time set R$_j$={r$_1$, r$_2$,…,r$_n$}on road segment
Initialize the dataset m = 1, $D_i^m, PD_i^m, GD_j^k$// of bus i
while m <= length (RawGpsData) do
    if RawGpsData$_m$∈$d^n$then
        $D_i^n$.append(RawGpsData$_m$);
    m = m + 1;
end while
$PD_i^m$= sortByTime($D_i^n$) //GPS dataset arranged in chronological order
$GD_{mj}^k$ = getSetOftrip($PD_i^m$); //GPS dataset of bus m in each trip on each road segment
while j <= length($GD_{mj}^k$) do
    Pet = $gd_{mj}^k$[time]
```

```
        Pst = $gd_{m1}^{k}$[time];
        Rt = Pet – Pst;
        R$_{j.}$append(Rt);
        j=j+1;
end while
return R$_j$;
```

First, the original GPS data were classified according to the bus number. The GPS dataset under each bus number was D_i^m= $\{\{d_1^1,\dots,d_z^1\}$,…,$\{d_1^n,\dots,d_z^n\}\}$. Next, the data points under each bus number were ranked in a chronological order, thus obtaining a ranked set PD_i^m= $\{\{pd_1^1,\dots,pd_z^1\}$,…,$\{pd_1^n,\dots,pd_z^n\}\}$. The GPS dataset PD_i^m of bus m was acquired, and then the GPS dataset GD_j^k = $\{\{gd_1^1,\dots,gd_j^1\}$,…,$\{gd_1^k,\dots,gd_j^k\}\}$ of bus m in each strip on each road segment was acquired according to the serial number field of next station in the original GPS dataset. Subsequently, GD_j^k was traversed to acquire the first GPS point $gd_1^k$and the last GPS data point gd_j^k of bus m in the trip k on each road segment, and the bus running time Rt on this road segment was obtained by deducting the uploading time Pst of the first GPS data point gd_1^k from the uploading time Rt of the last GPS data point gd_j^k, as shown in **Figure 1**.

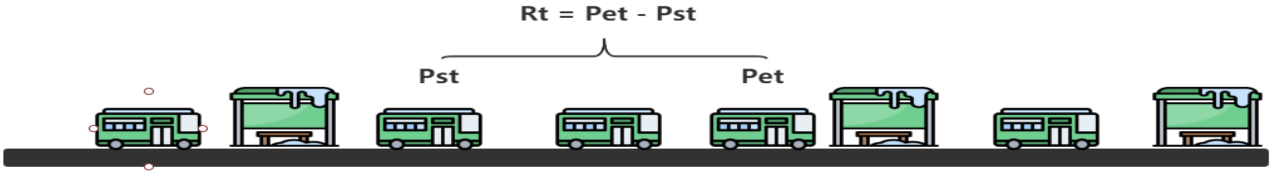

Figure 1. Calculation of Running Time on Road Segment

(4) Acquisition of bus stopping time

In order to choose a suitable bus stopping time prediction model, the approximate value of bus stopping time should be sought from the historical data through the following method:

The calculation steps of bus stopping time were basically identical with those of bus running time. The difference lied in that the last GPS point Pe of bus m in the trip k on the road segment n and its first GPS point Ps on the road segment n+1 were acquired in the traversal process of GD_j^k. The stopping time Ts of bus m at station n+1 in the trip k was calculated by deducting the uploading time Pet of Pe from the uploading time Pst of Ps, as shown in **Figure 2**.

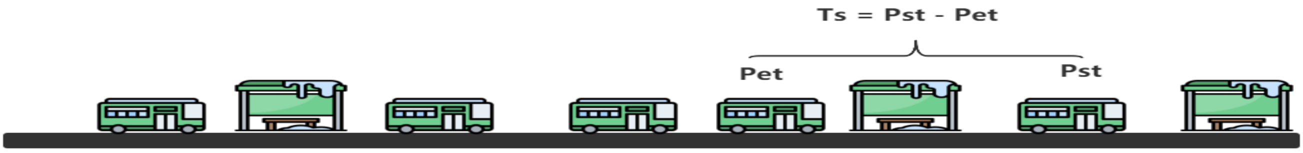

Figure 2. Calculation of Bus Stopping Time

2.2. Framework of arrival time prediction model

In the running process, a bus had two running states: interstation running state and stopping state at station. Without consideration of emergencies, the estimation of bus arrival time was a linear superposition of running time on each road segment and stopping time at each station, but the actual bus running laws could not be accurately described through such simple linear superposition. Since many factors should be considered in bus running process, the prediction result obtained through linear superposition deviated a lot from the reality in case of any emergency. Therefore, a new bus arrival time prediction model was proposed in this chapter to eliminate the error of such linear superposition, with the model framework as shown in **Figure 3**:

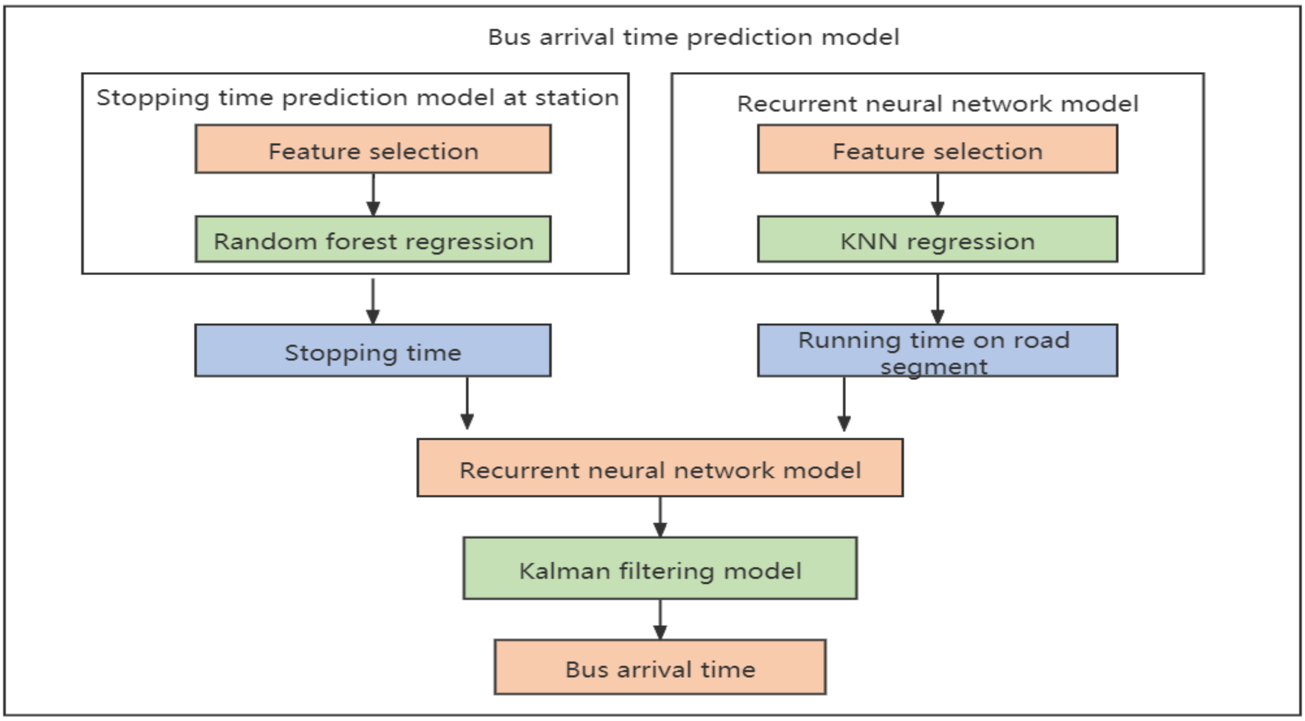
Bus arrival time prediction model
Stopping time prediction model at station
Feature selection
Random forest regression
Recurrent neural network model
Feature selection
KNN regression
Stopping time
Running time on road segment
Recurrent neural network model
Kalman filtering model
Bus arrival time

Figure 3. Bus Arrival Time Prediction Based on Fixed Model

The blended model was principally partitioned into three sections: two sub-models and one fixed model, separately being halting time expectation model at station, interstation running time forecast model and LSTM blended model, among which the first was utilized to anticipate the transport preventing time at each station from the spot of takeoff to the objective, the subsequent model was utilized to foresee the transport running time on every street portion from the spot of flight to the objective, and the contribution of LSTM blended model was the time series information consolidating the running time on every street fragment and halting time at each station yield by the two sub-models. Kalman separating model was utilized to progressively address the results, lastly acquire the anticipated appearance season of transport at the objective.

2.2.1. Stopping time prediction model at station

In this study, random forest regression model was selected to predict the bus stopping time after a comparison with multiple models. In the constructed random forest model, boostrap method was selected to divide each tree model training set. To construct the model, the maximum quantity of features (stn), minimum sample size of leaf nodes (tln) and quantity of subtrees (ctn) used by a single decision tree should be determined. In the end, the mean value of all subtree prediction results was taken as the predicted value of bus stopping time at station, as seen in formula (3). The established random forest model is as shown in **Figure 4**.

$$S_t = \frac{\sum_1^{ct_n} PT_i}{ct_n} (1 \leq i \leq ct_n) \qquad (3)$$

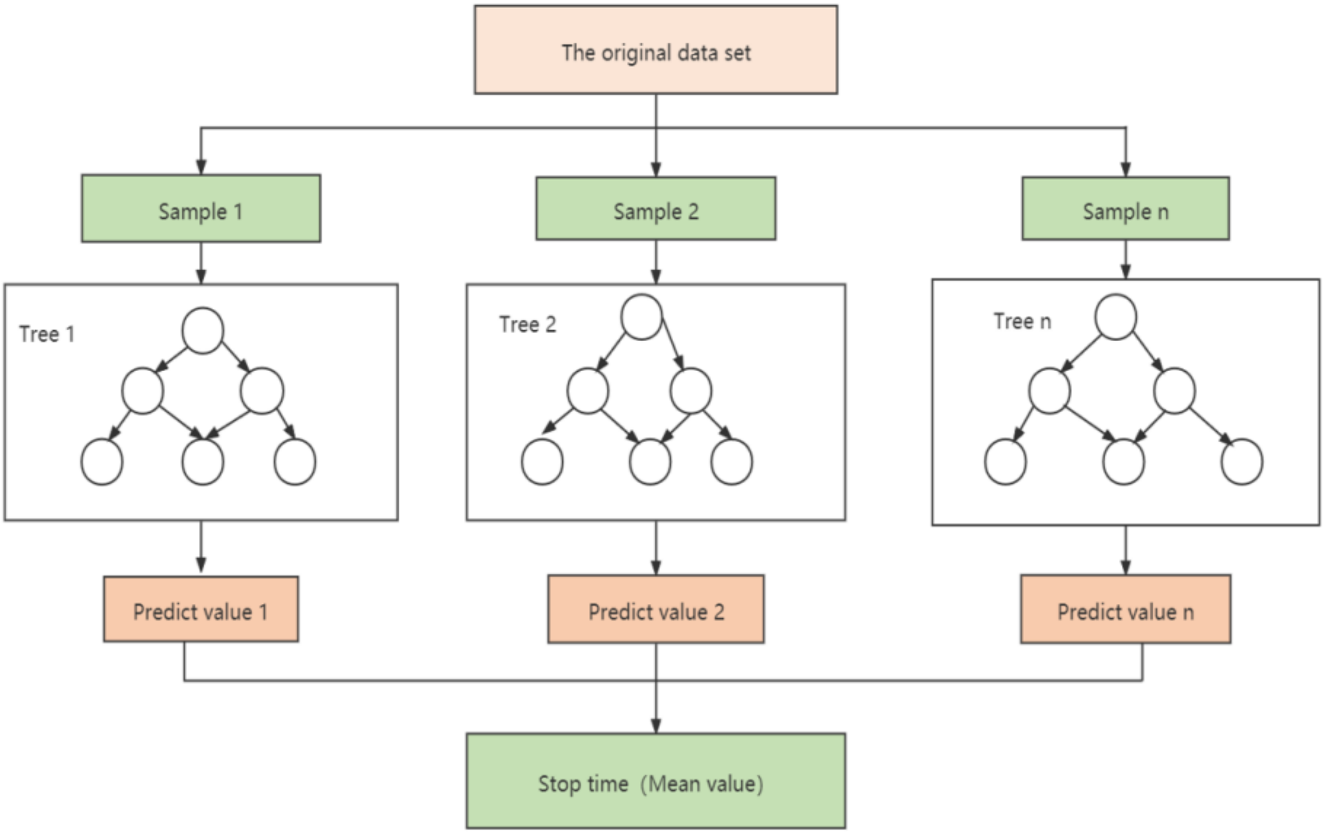

Figure 4. Random Forest Regression

2.2.2. Interstation running time prediction model

In this study, the KNN regression model (**Figure 5**) was used to predict the bus interstation running time.

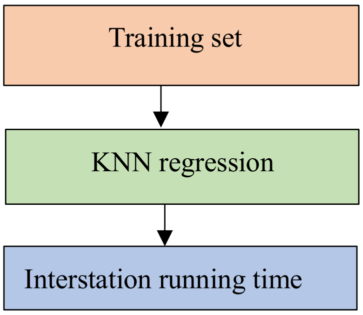

Figure 5. Interstation Running Time Prediction Model

The training set X= {x1, x2,…,x3} was input into KNN. The similarity D (formula 4) of samples was calculated using Manhattan distance, and the output value was namely the interstation running time.

$$D = \left|x_i - x_j\right| + \left|y_i - y_j\right| \tag{4}$$

2.2.3. Fusion of related sub-models

The bus stopping time prediction model and running time prediction model were introduced in 2.2.1 and 2.2.2. The prediction of bus arrival time at station was linear accumulation or nonlinear fusion of stopping time at station and interstation running time. As the bus arrival time prediction was a nonlinear problem, while LSTM showed favorable advantages in processing nonlinear time series data, LSTM was then used to fuse the outputs of the two models. The abstract graph of the mixed model is as shown in **Figure 6**:

The inputs of the whole model were the influence characteristics of interstation running time and stopping time at station. Through the two sub-models, the predicted value series R={r1,r2,…,rn} of interstation running time and predicted value series S={s1,s2,…,sn} of stopping time at station could be obtained, which were generated by iterating the two sub-models. By fusing the two models in a chronological order, the time series Xt={s1,r1,s2,r2,s3,r3,…, rn-1,sn} of bus at each downstream station on each road segment could be obtained, which was then taken as the input of LSTM to obtain an approximate output. Without consideration of emergencies, the bus arrival time predicted by the model was shortened with the bus running, so this process basically presented linear changes. Kalman achieved a significant effect in linear filtering, so a filtering layer was added to the output of LSTM, and the final model output was the bus arrival time. In this way, when the arrival time of a bus in a trip was predicted, the corresponding eigenvalue was generated through the bus position updating each time, and then substituted into the mixed model to generate an output bus arrival time, so this was a dynamic prediction process.

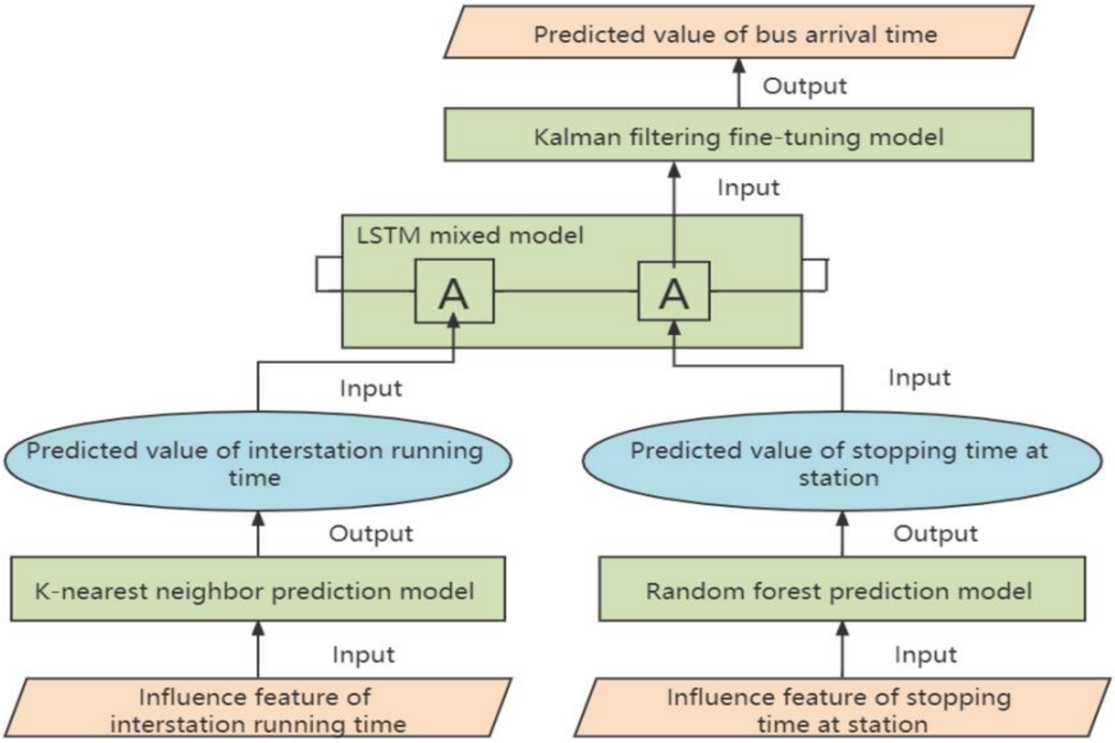
Predicted value of bus arrival time
Output
Kalman filtering fine-tuning model
Input
LSTM mixed model
A
A
Input
Input
Predicted value of interstation running time
Predicted value of stopping time at station
Output
Output
K-nearest neighbor prediction model
Random forest prediction model
Input
Input
Influence feature of interstation running time
Influence feature of stopping time at station

Figure 6. Abstract Graph of Mixed Model

3. Experimental analysis

The performance indexes of the proposed algorithm were analyzed using the root mean square error (RMSE) and mean absolute error (MAE) of formulas (5) and (6), where RMSE could be used to judge the model stability and MAE to discriminate its accuracy.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m}(y_{true} - y_{pre})^2}{m}} \tag{5}$$

$$MAE = \frac{\sum_{i=1}^{m}|y_{true} - y_{pre}|}{m} \tag{6}$$

Where m denotes the sample size, y_{true} stands for the true bus arrival time, and y_{pre} is the predicted arrival time.

3.1. Comparison of stopping time prediction models at station

Five regression models were chosen as the bus stopping time models at station and then compared in the aspects of RMSE and MAE, as shown in **Figure 7**:

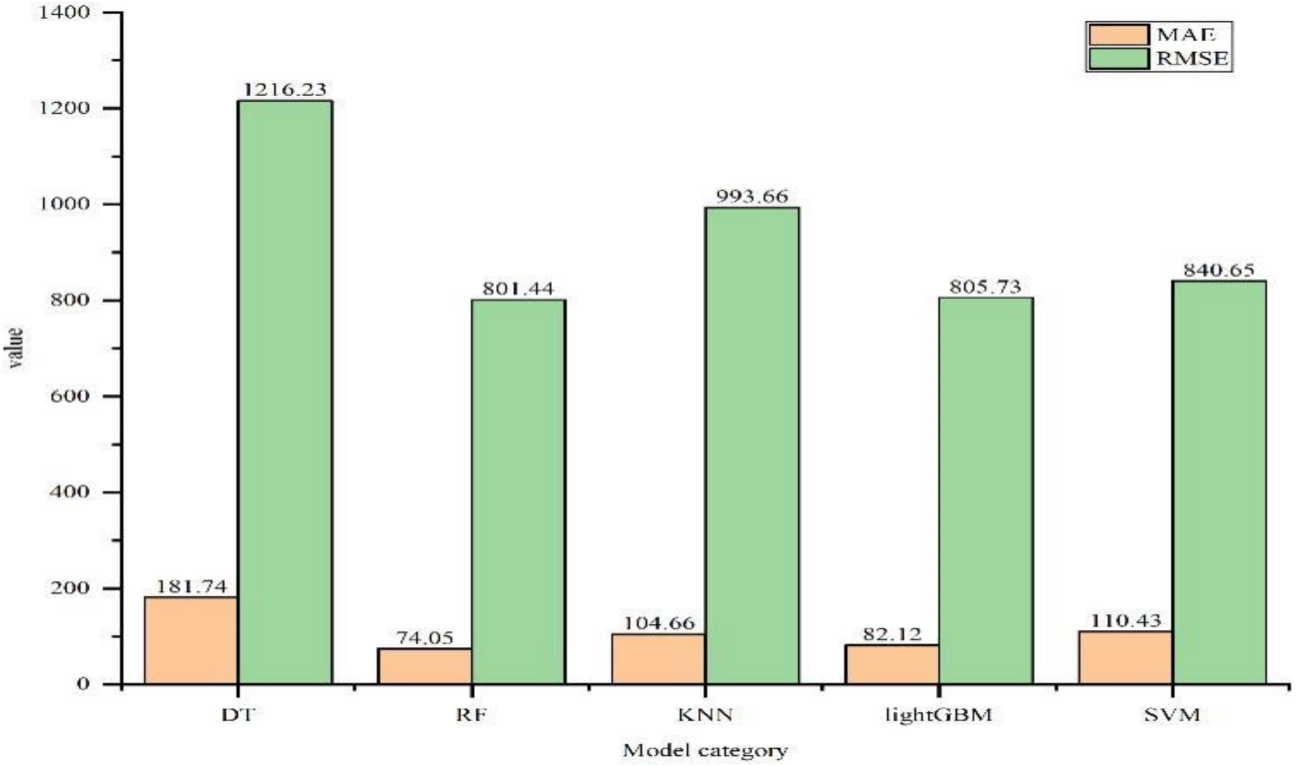

Figure 7. Model Comparison in MAE and RMSE

Figure 7 shows that, when compared to the other models, random forest and light GBM performed better in terms of accuracy and stability, implying that ensemble learning could take full advantage of each base learner's advantages to compensate for its own disadvantages and achieve a better prediction effect. Because the study object was a regression problem, the decision tree model, which was advantageous in the classification problem, performed poorly in this study. The prediction accuracy of KNN and SVM differed little, although SVM was more stable. As a result, the station stopping time was predicted using a random forest model with improved accuracy and stability.

3.2. Comparison of interstation running time prediction models

The following 5 models were also selected as the interstation running time prediction models and comparatively analyzed in RMSE and MAE, as shown in **Figure 8**.

Figure 8 shows that while the KNN regression model had a higher prediction accuracy, it did not have a significant advantage over the stability. The Light GBM and SVM models performed well in terms of stability, but not as well as the KNN model in terms of accuracy. The random forest model was sensitive to noise, and buses would unavoidably encounter traffic jams during their journey, causing the trained model to become stuck in an overfitting problem. As a result, the random forest model did not perform well in prediction. The decision tree model had the worst results. As a result, the KNN regression model was used to accurately forecast the bus interstation running duration.

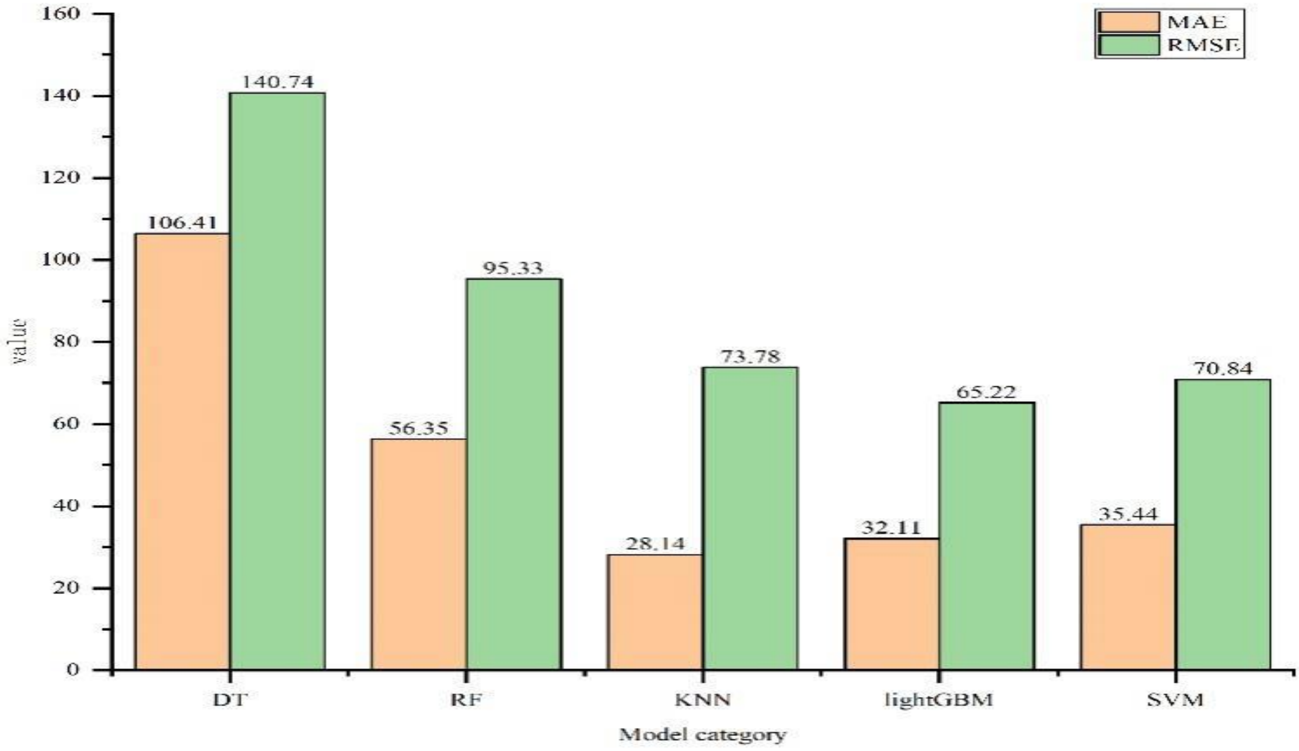

Figure 8. Comparison of RMSE and MAE Among Interstation Running Time Prediction Models

3.3. Performance comparison of mixed models

The bus stopping time prediction model at station and bus running time prediction model were already depicted in 2.2.1 and 2.2.2. The bus stopping time at each station and running time on each road segment could be acquired in an iterative way, and then linearly accumulated to obtain its arrival time. Or, the stopping time at each station and running time on each road segment could be substituted into the LSTM model to acquire the bus arrival time through nonlinear fusion. The accuracy and stability of the two methods were compared as shown in **Figure 9**:

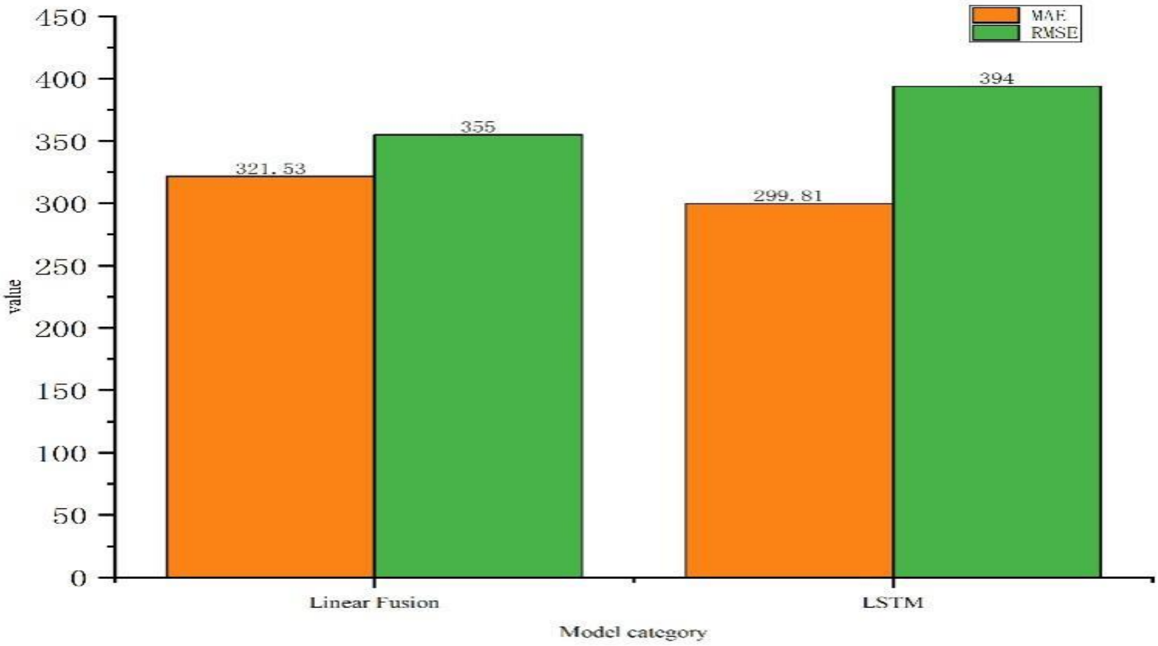

Figure 9. Comparison of RMSE and MAE Between Linear Fusion and Nonlinear Fusion

As shown in **Figure 9**, the model accuracy was improved through the nonlinear fusion of LSTM, because the bus arrival time prediction was actually a nonlinear process. Some sudden factors could not be learned simply through the linear superposition, while LSTM was a neural network model in essence, with relatively strong adaptive ability, so the prediction accuracy could be higher. Out of full consideration, LSTM nonlinear fusion was chosen in this study.

3.4. Comparison of filtering performance

In order to further eliminate the influence of abnormal data on the model, a filtering layer was added to the output of LSTM. The output of LSTM was input into the filtering layer as observed value to further optimize the model. The comparison results between the two filters are as shown in **Figure 10**:

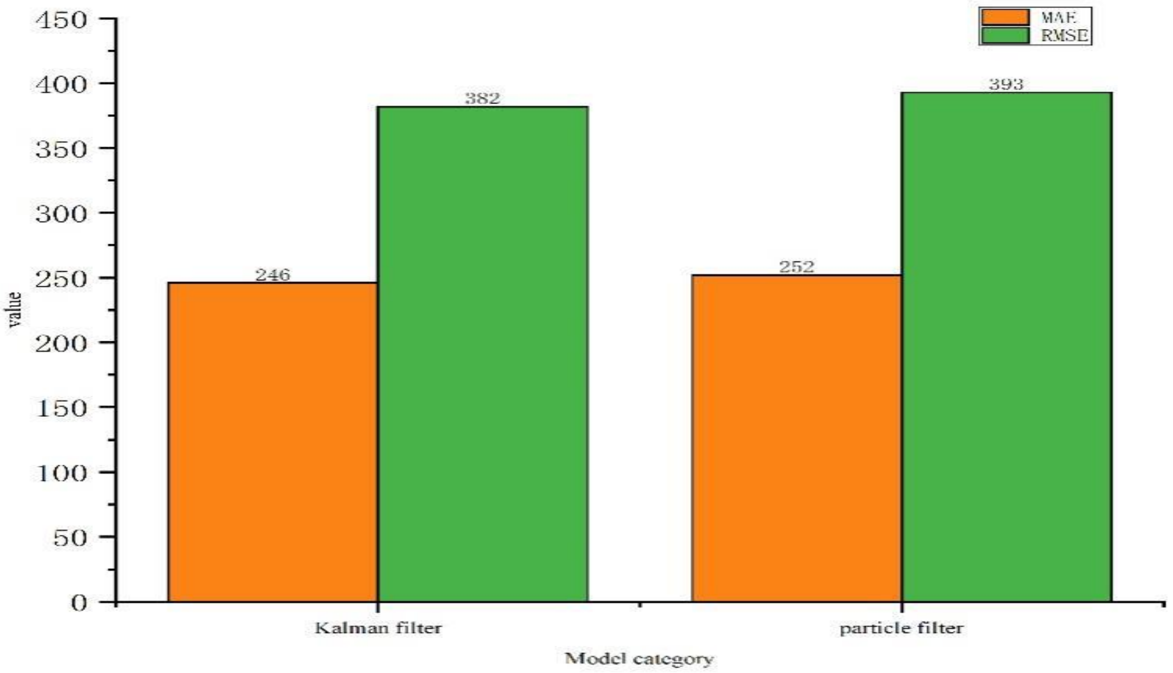

Figure 10. Comparison of RMSE and MAE Two Filtering Algorithms

It could be seen from **Figure 10** that no matter whether Kalman filtering or particle filtering was added, the model output was improved in both stability and accuracy, and the two differed little in the accuracy and stability. Given that Kalman filtering was fast, while the model required good timeliness and the prediction of arrival time was a linear process under most circumstances, Kalman filtering algorithm was chosen as the filtering layer in this study.

3.5. Model comparison

In order to embody the advantages of the proposed model, the proposed model was compared with commonly used AutoNavi and Baidu bus arrival time prediction models in the aspects of accuracy and stability as below.

As shown in **Figure 11**, the proposed mixed model had good performance in both stability and accuracy, because AutoNavi and Baidu only relied upon static data when predicting the bus arrival time and could hardly make an accurate prediction for the actual bus running status. However, the mixed model fused real-time bus data based on static data, so it accorded with the reality more and its prediction accuracy was higher.

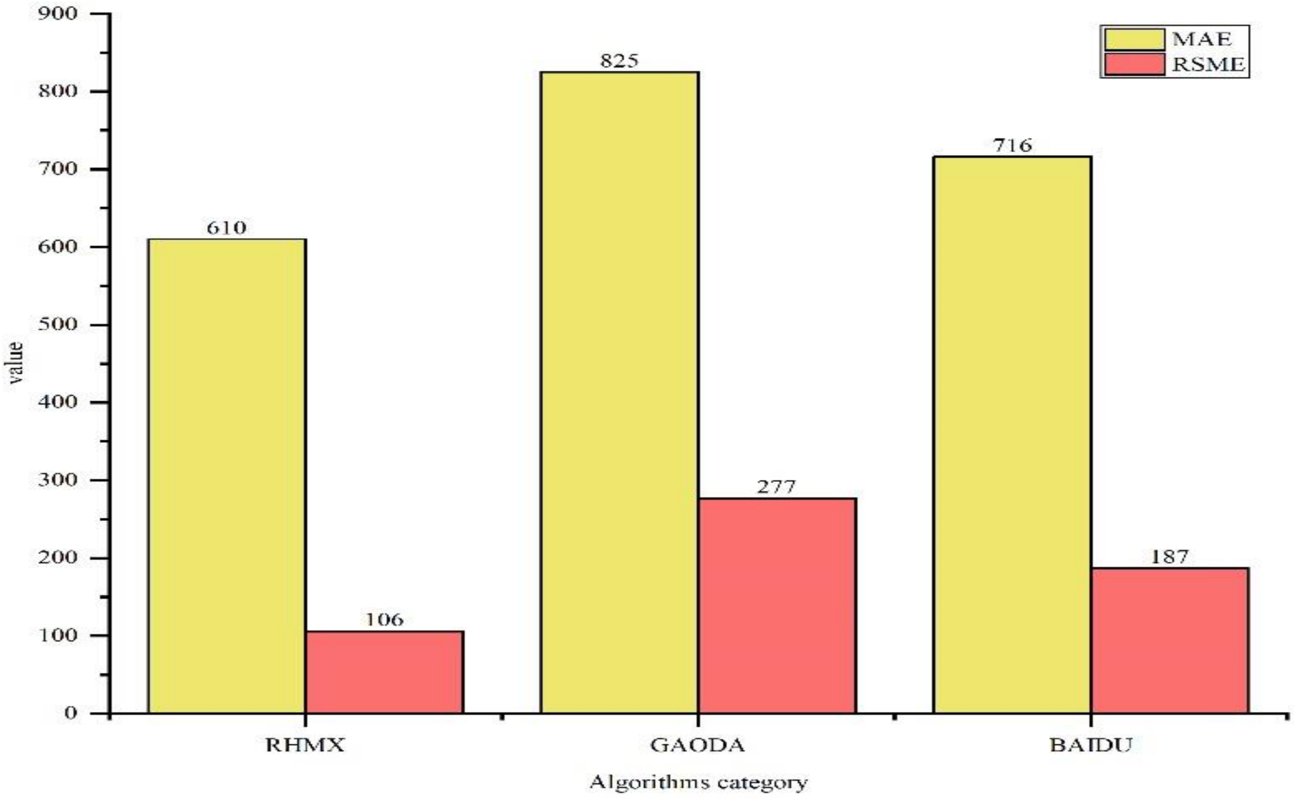

Figure 11. Comparison of RMSE and MAE Between Two Filtering Algorithms

4. Conclusion

A mixed model for forecasting bus arrival time was developed in this study. First, two sub-models were built in this model: stopping time prediction at station and interstation running time, where the former predicted the stopping time of a running bus at each downstream station in an iterative manner and the latter predicted its running time on each downstream road segment in an iterative manner. The two models might be used to predict one group of interstation running time and station halting time series data. Following that, the time series data from the two sub-models was fused using LSTM to produce an approximate bus arrival time. Finally, Kalman filtering was utilized to dynamically change the LSTM prediction results in order to avoid aberrant data influencing the anticipated value and obtain more accurate bus arrival time. This model could dynamically predict the bus arrival time based on the input data, but because the bus running time on the downstream road segment and stopping time at each station were predicted one by one iteratively, cumulative error was generated to some extent, with the cumulative error of the bus closer to the downstream station being greater. As a result, by including a residual error module in the iteration process, the error can be further decreased

Funding

This work was supported by Guilin Scientific Research and Technology Development Plan (2020010304).

Disclosure statement

The author declares no conflict of interest.

References

[1] Gong J, Liu M, Zhang S, 2018, Hybrid Dynamic Prediction Model of Bus Arrival Time Based on Weighted of Historical and Real-Time GPS Data, Control & Decision Conference. IEEE.

[2] Marko C, Marjan L, 2018, Bus-Arrival Time Prediction Using Bus Network Data Model and Time Periods. Future Generation Computer Systems, 3-5.

[3] Chen G, Yang X, Zhang D, et al., 2017, Historical Travel Time Based Bus-Arrival-Time Prediction Model. ICCTP 2011: Towards Sustainable Transportation Systems.

[4] Yu B, Lam WHK, Tam ML, 2011, Bus Arrival Time Prediction at Bus Stop with Multiple Routes. Transportation Research Part C Emerging Technologies, 19(6): 1157-1170.

[5] Van Eekelen R, McLernon DJ, Van Wely M, et al., 2018, External Validation of A Dynamic Prediction Model for Repeated Predictions of Natural Conception over Time. Pubmed, 2268-2275.

[6] Bachu A, Subramanian SC, 2016, Pattern-Based Bus Travel Time Prediction under Heterogeneous Traffic Conditions. 93rd Annual Meeting-Transportation Research Boards.

[7] Kong He, et al., 2021, Kalman Filtering Under Unknown Inputs and Norm Constraints. Automatica, 133.

[8] Li X, Gao W, Zhang J, 2020, A Novel Hybrid Unscented Particle Filter based on Firefly Algorithm for Tightly-Coupled Stereo Visual-Inertial Vehicle Positioning. Cambridge University Press, 73(3): 121-55.

[9] Dhivyabharathi B, Kumar BA, Vanajakshi L, 2016, Real Time Bus Arrival Time Prediction System under Indian Traffic Condition. IEEE International Conference on Intelligent Transportation Engineering. IEEE, 18-22.

[10] Dhivyabharathi B, Kumar BA, Vanajakshi L, et al., 2017, Particle Filter for Reliable Bus Travel Time Prediction Under Indian Traffic Conditions. Transportation in Developing Economies, 3(2): 1-11.

[11] Hans E, Chiabaut N, Leclercq L, et al., 2018, Real-time Bus Route State Forecasting Using Particle Filter: An Empirical Data Application. Transportation Research Procedia. 2018.

[12] Dhivyabharathi B, Kumar BA, Vanajakshi L, et al., 2017, Particle Filter for Reliable Bus Travel Time Prediction Under Indian Traffic Conditions. Transportation in Developing Economies, (2).

[13] Huang YP, et al., 2021, Bus Arrival Time Prediction and Reliability Analysis: An Experimental Comparison of Functional Data Analysis and Bayesian Support Vector Regression. Applied Soft Computing Journal, 111.

[14] Zhou Y, Yao L, et al., 2017, Bus Arrival Time Calculation Model Based on Smart Card Data. Transportation Research Part C, 74: 81-96.

[15] Deng F, Deng F, Su P, et al., 2020, Travel Time Prediction of Urban Road based on BP Neural Network, Journal of Physics: Conference Series, 1651(1).

[16] Xu W, Rong W, 2021, Neural Network Model based on Travel Planning for Travel Time Prediction, Journal of Physics: Conference Series, 1883(1).

[17] Petersen NC, Rodrigues F, Pereira FC, 2019, Multi-output Bus Travel Time Prediction with Convolutional LSTM neural Network. Expert Systems with Applications, 120: 426-435.

[18] Xie ZY, He YR, Chen CC, et al., 2021, Multistep Prediction of Bus Arrival Time with the Recurrent Neural Network, Mathematical Problems in Engineering, 76:72-97.