Transportation Research Part C 72 (2016) 45–59

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Headway-based bus bunching prediction using transit smart card data ☆

Haiyang Yu, Dongwei Chen, Zhihai Wu, Xiaolei Ma *, Yunpeng Wang

School of Transportation Science and Engineering, Beijing Key Laboratory for Cooperative Vehicle Infrastructure, Systems, and Safety Control, Beihang University, Beijing 100191, China

ARTICLE INFO

Article history:
Received 19 August 2015
Received in revised form 10 September 2016
Accepted 17 September 2016
Available online 26 September 2016

Keywords:
Bus bunching prediction
Headway irregularity
Least squares support vector machine
Transit smart card

ABSTRACT

Bus bunching severely deteriorates the quality of transit service with poor on-time performance and excessive waiting time. To mitigate bus bunching, this paper presents a predictive framework to capture the stop-level headway irregularity based on transit smart card data. Historical headway, passenger demands, and travel time are utilized to model the headway fluctuation at the following stops. A Least Squares Support Vector Machine regression is established to detect bus bunching with the predicted headway pattern. An empirical experiment with two bus routes in Beijing is conducted to demonstrate the effectiveness of the proposed approach. The predictive method can successfully identify more than 95% of bus bunching occurrences in comparison with other well-established prediction algorithms. Moreover, the detection accuracy does not significantly deteriorate as the prediction lead time increases. Instead of regularizing the headways at all costs by adopting certain correction actions, the proposed framework can provide timely and accurate information for potential bus bunching prevention and inform passengers when the next bus will arrive. This feature will greatly increase transit ridership and reduce operating costs for transit authorities.

1. Introduction

Transit authorities have been striving to improve transit service quality to attract more transit riders, satisfy passenger demand, and reduce operating cost (Ding et al., 2015, 2016). The level of service enhancement largely relies on whether the transit system can be operated as reliable as it was designed (Bellei and Gkoumas, 2010). However, buses may not always adhere to the planned schedule at each stop, a situation that may cause great frustration for transit riders (Watkins et al., 2011). One of the consequences of an irregular transit service is bus bunching, in which two or more buses that should be evenly running along the same route simultaneously arrive at the same stop. The potential reasons behind this phenomenon are unstable traffic conditions and excessive boarding/alighting passenger demands. If a bus is slightly delayed due to traffic congestion, the bus has to pick up more passengers than expected at the next bus stop. The extra passenger demand further aggravates the bus delay. In contrast, the next bus will experience low passenger load and save more dwell time (Fonzone et al., 2015). This outcome can exponentially expand to the following stops and eventually yield little or zero headway for the two buses. Bus bunching increases the waiting time for passengers and deteriorates in-vehicle comfort level due to overcrowding (Bie et al.,

☆ This article belongs to the Virtual Special Issue on: Data-driven smart-city-enabled traffic system modeling, analysis, and optimization.

* Corresponding author.
E-mail addresses: hyyu@buaa.edu.cn (H. Yu), chendongwei@buaa.edu.cn (D. Chen), zy1513318@buaa.edu.cn (Z. Wu), xiaolei@buaa.edu.cn (X. Ma), ypwang@buaa.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.trc.2016.09.007
0968-090X/

2015). Transit authorities may also lose loyal customers with such unreliable transit service and cause revenue reduction. When bus bunching occurs, an overcrowded bus is usually followed by a near-empty bus. This situation will waste the already limited resources and increase the operating costs for both transportation agencies and transit companies.

Transit authorities have developed corrective measures to overcome the negative effect of bus bunching. The typical countermeasures include bus holding, speed adjustment, stop-skipping, short-turning, and passenger demand reduction. However, these corrective methods are reactive and may dissatisfy the passengers waiting at stops or sitting in the bus. For instance, stop-skipping allows drivers to intentionally bypass several stops for service recovery. However, this may change the destinations of those passengers who are already onboard and frustrate or anger them. Adopting proactive strategies to monitor irregular headways and alert possible bus bunching in advance to prevent transit service breakdown is preferable. Therefore, the objective of this study is to predict the occurrence of bus bunching through irregular headway fluctuation. From the perspective of passengers, knowing the time interval between two consecutive bus runs (i.e., stop-level headway) helps them anticipate the arrival of the next bus. With this information, passengers can more efficiently plan their trips to save time. From the perspective of transit operators, the stop-level headway is a good indicator of bus bunching. If the predicted headway is too small, bus bunching will likely happen, and corresponding countermeasures can be made before two buses simultaneously arrive at the next stop.

The recent advent of Automatic Fare Collection (AFC) (i.e., Smart Card) can reduce the high cost and manpower resources for manual data collection method, as well as accelerate the passenger boarding and alighting process through contact-less cards and card readers. The data gathered by the AFC system contains abundant temporal and spatial information of individual passengers and can be utilized to quantify transit performance, improve transit operation, and understand travel behavior. This study aims to leverage transit smart card data into bus bunching detection by forecasting stop-level headway in varying time horizons. Although smart card data suffer from several data quality issues (Robinson et al., 2014), they can store each passenger's boarding time and location. This information can be used to calculate the stop-level bus headway and passenger demands along with inter-stop travel time. Both passenger demand and travel time uncertainties are highly correlated to the bus bunching phenomenon and can be fed back into intelligent algorithms to prevent the upcoming bus bunching in the following stops. Passengers can likewise reduce their anxiety by knowing when their next buses will arrive.

The major contributions of this study are threefold: (1) A novel data-driven approach is proposed to predict the fluctuation of bus headway at each stop based on smart card data. (2) The emergence of bus bunching can be found by mining the headway irregularity at consecutive bus stops. (3) The lead time (i.e., the number of stops ahead of the stop where bus bunching map occurs) to detect bus bunching occurrences is investigated. The sooner bus bunching can be found, the more efficiently transit operators can adopt preventive countermeasures.

The remainder of this paper is organized as follows. Section 2 summarizes the existing literature on bus bunching and headway prediction, followed by a brief introduction of data collection and preprocessing efforts. Section 3 outlines the methodological framework, where Least Squares Support Vector Machine (LS-SVM) regression is introduced to predict the stop-level bus headway, and the predicted headway fluctuation can be further mined to identify the occurrence of bus bunching. Section 4 presents the algorithm results based on the smart card data from two routes in Beijing and tests the prediction accuracies of both headways and bus bunching events by comparing them with other algorithms. In addition, a sensitivity study is undertaken to analyze how early a bus bunching notice can be given with multi-stop intervals. Section 5 concludes the work and elaborates on future research directions.

2. Literature review

In an ideal situation, the headway between two consecutive bus runs at each stop should be constant along the same route. However, in reality, the headways for different stops become irregular due to severe traffic congestion, unexpected passenger demands, heterogeneous bus driver behavior, and unreasonable bus bay layouts. An irregular headway at a specific stop will have a snowball effect that further deteriorates the reliability of the bus schedule. Consequently, bus platoons may end up arriving at the same time. In this case, bus bunching is caused by headway fluctuation, which can be defined as the time difference between the arrival times for two consecutive buses at the same stop. Thus, various complex algorithms are developed to predict bus arrival time. These algorithms can be generally categorized as follows: forecasting model based on historical data (Chen et al., 2003), time series model (D'Angelo et al., 1999), artificial neural network (ANN) model (Chien et al., 2002; Jeong and Rilett, 2004; Padmanaban et al., 2010; Lin et al., 2013), support vector machine model (Yu et al., 2007, 2011), Kalman Filter algorithm (Shalaby et al.,2003; Chien and Kuchipudi, 2003; Chen et al., 2004), nearest-neighbor trajectory method (Tiesyte and Jensen, 2008; Chang et al., 2010), regression prediction model (Patnaik et al., 2004; Jeong, 2004; Sinn et al., 2012), and K-nearest neighbor (KNN) (Coffey et al., 2011). Most of these studies rely on GPS data to predict bus arrival time at each stop for a single bus along the same route (as summarized in Table 1). To successfully extract the stop-level bus arrival time, GPS data have to be integrated with external data sources, such as bus stop spatial information, for additional processing (e.g., map matching). Moreover, it is difficult to take into account the stop-level bus delay (i.e., dwell time) because low GPS data updating frequency may impact the estimation accuracy, whereas the dwell time is highly related to bus arrival time prediction (Lin et al., 1999).

Predicting bus bunching is a more challenging task than predicting single-bus arrival times because more than one bus is involved in bus bunching. Both the dwell times and arrival times of different buses at different stops fluctuate and lead to a

Table 1
Summary of bus arrival time prediction methods.

Author(s)	Data	Algorithm
Chen et al. (2003)	AVL and APC data	Forecasting model based on historical data
D'Angelo et al., 1999	Speed, traffic flow, occupancy	Time series
Chien et al. (2002)	Simulation data in CORSIM (link travel times and stop-to-stop travel times)	ANNs
Jeong and Rilett (2004)		
Padmanaban et al. (2010)	GPS data	
Lin et al. (2013)	GPS data and AFC data	
Yu et al. (2007, 2011)	Segment, weather, travel time, headway based on GPS or survey data	Support vector machines
Shalaby et al. (2003)	AVL and APC data	Kalman filters
Chien and Kuchipudi (2003)	Travel time	
Chen et al. (2004)	Travel time based on APC data	
Tiesyte and Jensen (2008)	Real-time vehicle trajectories	Nearest-neighbor trajectory method
Patnaik et al. (2004)	APC data	Regression-based prediction model
Jeong (2004)	AVL data	
Sinn et al. (2012)	Real-time GPS data	
Coffey et al. (2011)	AVL data	KNNs

high stochastic process. In recent years, a series of studies began emerging to disentangle the complicated interaction and mitigate bus bunching. Feng and Figliozzi (2011) provided a method to identify the temporal and spatial location of bus bunching events and analyze the causes of those events. Nair et al. (2015) developed a real-time system based on General Transit Feed Specification (GTFS) data feed to predict bus bunching. Albright and Figliozzi (2012) presented a regression model to analyze the impact of Transit Signal Priority (TSP) strategies on bus bunching. Chen et al. (2013) proposed two regression models to verify and estimate the influence of bus bunching on dwell time. They found that dwell time is closely correlated with the passenger boarding demand when bus bunching occurs. Bartholdi and Eisenstein (2012) proposed a self-correcting method to dynamically equalize unstable headways. Hammerle et al. (2005) utilized Automatic Vehicle Location (AVL) and Automatic Passenger Counting (APC) data from the Chicago Transit Authority to calculate transit service reliability measures, such as headway regularity, to monitor bus bunching occurrence. Pilachowski (2009) presented a continuum approximation framework to model the behavior of bus bunching. Wang (2014) designed a FlexiFare bus system to reduce the passenger demand via dynamic transit fare for bus bunching prevention. Daganzo (2009) proposed an adaptive control mechanism based on real-time headway information to dynamically determine bus holding times and eliminate bus bunching. Later, Daganzo and Pilachowski (2011) extended the headway-based control method by introducing an adaptive control scheme to adjust bus cruising speeds through bus-to-bus communication. Strathman et al. (2002) and Kimpel et al. (2008) used the AVL and APC data from Portland to improve transit schedule and operation. They found that bus operators are a dominant factor in causing running time variation. Moreira-Matias et al. (2012a,b, 2014) proposed a sequence mining algorithm to capture the irregular bus headways that may induce bus bunching.

The majority of previous research studied the bus bunching phenomenon through either analyzing the statistical relationship between bus bunching and other contributing factors (e.g., dwell time, headway, driver behavior, etc.) or by developing corrective control strategies for mitigation. Very few studies developed proactive countermeasures to detect the emergence of bus bunching before bus service failure. By learning the irregularity of headways from the historical smart card data, we aim to alert when and where the possible bus bunching will occur and provide a real-time headway forecasting system for passengers to know where the next bus will arrive. These strategies will greatly benefit transit authorities in improving transit service quality and increasing ridership for revenue gains.

3. Data collection and processing

3.1. Data collection

The data used in this study are obtained from Beijing AFC system, which covers more than 1000 bus routes and 17 subway lines. More than 13 million daily smart card transactions are generated for all bus routes due to the special fare discount for smart card holders (75% reduction for students, and 50% reduction for regular passengers). These discounts greatly stimulate the wide usage of transit smart cards, with more than 90% penetration rate in Beijing (Ma et al., 2012). As of the end of 2014, two types of AFC systems existed: flat fare and distance-based fare. The flat-fare based system does not require passengers to tap their cards when alighting, whereas the distance-based system records both boarding and alighting locations for passengers (Ma et al., 2013; Ma and Wang, 2014). Therefore, the key information for the Beijing AFC system includes smart card identification, transaction timestamp, route number, boarding stop, and alighting stop. Notably, the passenger boarding stops on the flat-fare based buses are not stored by smart card readers because all passengers pay the same amount of fares regardless of where they get on and off the buses. Therefore, this study mainly focuses on predicting bus bunching on distance-based fare buses, where both boarding and alighting demands for each stops can be estimated. The smart card data from two routes from July 1, 2012 to November 1, 2012 were collected based on the above criteria. Fig. 1 demonstrates the bus stop layouts for both routes (routes A and B) in a Geographic Information System (GIS) map, where the dashed circles indicate the studied areas. The bus routes and stops are respectively highlighted in black and red. The rationale for choosing these sites is the high occurrence rate of bus bunching at the two selected segments. For Route A, the bus bunching occurrence at stop 49686 will be detected based on the previous stops with IDs from 49863 to 49687. For Route B, the bus bunching occurrence at stop 42915 will be detected based on the previous stops with IDs from 42910 to 42914.

3.2. Data processing

Predicting the emergence of bus bunching is necessary to capture the irregularity of stop-level headways. Therefore, the headways should be calculated based on raw smart card data. Headway is defined as the time interval between two consecutive bus runs at a certain stop. The Beijing transit system (except the subway) allows passengers to board and alight simultaneously. When a bus arrives, one door (front door in regular buses or middle door in articulated buses) is used for boarding

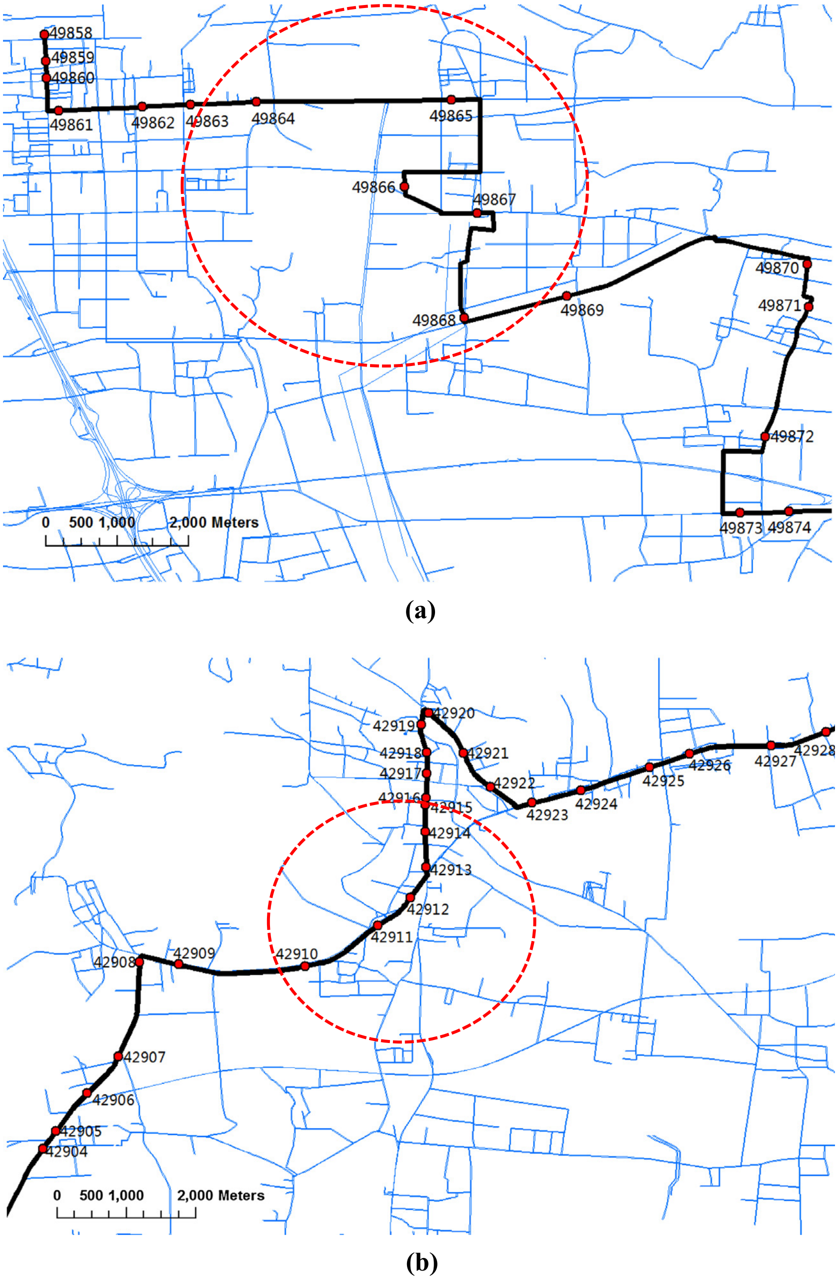


Fig. 1. Spatial distribution of bus stops: (a) Route A and (b) Route B.

and the remaining doors are used for alighting. Thus, the arrival time at each stop can be approximated as the smart card tapping time of the first boarding passenger. Let us define the arrival time of the bus run i at stop j for a specific route as $T_{i,j}$, and the arrival times for bus run i can be expressed as

$$T_i = \{T_{i,1}, T_{i,2}, \ldots, T_{i,s}\}, \tag{1}$$

where s is the total number of bus stops along this route.

Consequently, the headways for each stop for two consecutive bus runs $i, i+1$ can be computed as

$$H = \{h_1, h_2, \ldots, h_s\}, \quad \text{where } h_i = T_{i+1,j} - T_{i,j}. \tag{2}$$

Iteratively executing the above procedure can generate the sequence of headways for both route A and route B. These calculated headways from smart card data will be regarded as ground truth and used for the following bus bunching prediction.

4. Methodology

Predicting the occurrence of bus bunching involves three steps. First, the influential factors that are relevant to headway variability are identified and calculated from smart card data. Next, the LS-SVM algorithm is utilized to forecast the bus headways for the next stop on the basis of historical data generated from the first step. Finally, a relationship between predicted headways and bus bunching occurrence is established to detect the irregular fluctuation of headways.

4.1. Variable selection

A great quantity of work has acknowledged that abnormal passenger loads, unreliable traffic condition, and heterogeneous bus driver behavior contribute to the emergence of bus bunching (Feng and Figliozzi, 2011; Daganzo, 2009). High passenger demands increase the dwell time for the preceding bus but leave no pickup for the following bus, thus shortening the headway. Accordingly, the number of boarding passengers and number of alighting passengers at the previous stop are incorporated as candidate variables. Let the boarding and alighting demands at stop $j-1$ for bus run i be respectively defined as $b_{i,j-1}$ and $a_{i,j-1}$. Similarly, the boarding and alighting demands at stop $j-1$ for bus run $i+1$ are denoted as $b_{i+1,j-1}$ and $a_{i+1,j-1}$. These four-dimensional data can be readily computed by counting the total number of smart card transactions occurring at each stop. Given the impact caused by the unexpected travel congestion, the link travel time $TT_{i,j-1}$ for bus run i between stop $j-1$ and stop j is calculated in Eq. (3), which equals the difference between the arrival times between stops (Chen et al., 2012) and can be calculated as

$$TT_{i,j-1} = T_{i,j} - T_{i,j-1}, \tag{3}$$

where $T_{i,j}$ and $T_{i,j-1}$ respectively represent the arrival times for bus run i at stop j and stop $j-1$.

4.2. LS-SVM regression for bus headway prediction

With the candidate variables identified, the following task is to predict the headway at stop $j(h_j)$ given the six input dimensions of the headway at stop $j-1$ (h_{j-1}), passenger loads at stop $j-1$ for bus run $i+1(b_{i+1,j-1}$ and $a_{i+1,j-1})$, passenger loads at stop $j-1$ for bus run i $(b_{i,j-1}$ and $a_{i,j-1})$, and the link travel time for bus run $i(TT_{i,j-1})$. This task is tantamount to establishing an input–output relationship using Eq. (4):

$$\hat{h}_j = f(h_{j-1}, b_{i+1,j-1}, a_{i+1,j-1}, b_{i,j-1}, a_{i,j-1}, TT_{i,j-1}), \tag{4}$$

where $\hat{h}_j$ denotes the predicted headway at stop j.

The LS-SVM algorithm is employed to capture the highly nonlinear and stochastic nature of bus headway. Different from the traditional SVM algorithm that solves a convex quadratic programming problem, the LS-SVM algorithm solves a set of linear equations to find the global optima (Suykens et al., 2002a,b), thus demonstrating high efficiency and low computational efforts for large-scale datasets (Wang and Hu, 2005). These features motivate us to adopt the LS-SVM algorithm to predict the emergence of bus bunching for hundreds of bus routes in real-time fashion. The objective of the LS-SVM algorithm is to optimize a cost function with a penalized regression error presented as follows:

$$\begin{aligned} &\min_{\boldsymbol{\omega},b,\mathbf{e}} J(\boldsymbol{\omega}, \mathbf{e}) = \tfrac{1}{2}\|\boldsymbol{\omega}\|^2 + \tfrac{1}{2}\gamma\sum_{k=1}^{N} e_k^2 \\ &s.t.\ y_k = \langle \boldsymbol{\omega}, \varphi(\mathbf{x_k})\rangle + b + e_k, k = 1, \ldots, N \end{aligned} \tag{5}$$

Given a set of datasets as

$$D = \{(\mathbf{x_1}, y_1), \ldots, (\mathbf{x_k}, y_k), \ldots, (\mathbf{x_N}, y_N)\}, \quad \mathbf{x_k} \in R^n, y_k \in R, \tag{6}$$

where $\langle \cdot, \cdot \rangle$ represents the dot product for two vectors, $\boldsymbol{\omega}$ denotes the weight vector, $\varphi(\cdot): R^n \to R^{n_h}$ is a nonlinear function that maps the input space to a high-dimensional feature space for linear transformation, b and e_k are the bias and error terms respectively, and γ is a positive constant for regularization to avoid overfitting. In the context of bus headway prediction, y_k is the predicted headway as $\hat{h}_j$ in Eq. (4), while $\mathbf{x}_k$ is the input matrix similar to those input variables in the function of Eq. (4). LS-SVM aims to minimize the structural risk to avoid the issue of overfitting by applying the Lagrangian function in Eq. (7):

$$L(\mathbf{w}, b, \mathbf{e}, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{\omega}\|^2 + \frac{1}{2}\gamma\sum_{k=1}^{N} e_k^2 - \sum_{k=1}^{N}\alpha_k\{\langle \boldsymbol{\omega}, \varphi(\mathbf{x_k})\rangle + b + e_k - y_k\}, \tag{7}$$

where $\alpha_k \in R$ is the Lagrange multiplier. Optimizing the Lagrangian function is equivalent to calculating the following equations. The optimal model parameters can then be estimated using Eq. (8).

$$\begin{cases} \dfrac{\partial L}{\partial \boldsymbol{\omega}} = 0 \to \boldsymbol{\omega} = \displaystyle\sum_{k=1}^{N}\alpha_k\varphi(\mathbf{x_k}) \\ \dfrac{\partial L}{\partial b} = 0 \to \displaystyle\sum_{k=1}^{N}\alpha_k = 0 \\ \dfrac{\partial L}{\partial e_k} = 0 \to \alpha_k = \gamma e_k \\ \dfrac{\partial L}{\partial \alpha_k} = 0 \to \langle \boldsymbol{\omega}, \varphi(\mathbf{x_k})\rangle + b + e_k - y_k = 0 \end{cases}. \tag{8}$$

Solving the above equations can generate the following linear system:

$$\begin{bmatrix} 0 & \mathbf{1}^T \\ \mathbf{1} & \boldsymbol{\Omega} + \frac{\mathbf{I}}{\gamma} \end{bmatrix}\begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}, \tag{9}$$

where $y = [y_1; \ldots; y_N]$, $\mathbf{1} = [1; \ldots; 1]$, $\alpha = [\alpha_1; \ldots; \alpha_N]$, and $\mathbf{I}$ denotes the identity matrix with $N \times N$ dimensions.

$$\boldsymbol{\Omega}_{ij} = \langle \varphi(\mathbf{x_i}), \varphi(\mathbf{x_j})\rangle = K(\mathbf{x_i}, \mathbf{x_j}) \quad i, j = 1, \ldots, N. \tag{10}$$

$K(\cdot, \cdot)$ is defined as the kernel function that can meet Mercer's theorem (Vapnik, 2010). Then, the estimated values for the LS-SVM model can be rewritten as

$$f(\mathbf{x}) = \sum_{k=1}^{N}\alpha_k K(\mathbf{x}, \mathbf{x}_k) + b, \tag{11}$$

where α_k and b can be calculated using Eq. (9), and the kernel function can be generally chosen in the form of a linear function, polynomial function, Gaussian function, and radial basis function. In this study, the radial basis function is adopted for its good performance in short-term traffic state prediction (Zhang and Liu, 2009). The form of the radial basis function can be written as in Eq. (12).

$$K(\mathbf{x_i}, \mathbf{x_j}) = \exp\left(-\frac{\|\mathbf{x_i} - \mathbf{x_j}\|^2}{\sigma^2}\right), \tag{12}$$

where $\|\cdot, \cdot\|$ indicates the distance between two input vectors and is usually defined as the Euclidean distance, and σ is a constant that should be determined through a cross-validation process in the following section.

Compared with the original SVM algorithm, the LS-SVM optimizes the least squares loss function rather than the ε-insensitive loss function, thus avoiding solving the quadratic programming problem for less computational time.

4.3. Bus bunching detection

The occurrence of bus bunching can be measured by the headway regularity and should be defined based on a threshold value. The Transit Capacity and Quality of Service Manual (TCQSM) recommends using the probability that a transit vehicle's actual headway is larger than half of the scheduled headway (TCRP, 2003) to identify headway irregularity. The Beijing public transit system employs the headway-based strategy for bus dispatching. It cannot provide a fixed timetable for passengers because most route schedules change over time depending on varying passenger demands, time of day, and traffic conditions. Therefore, this study uses the headway at the first stop as the "scheduled" headway. A more rigorous criterion for bus bunching detection is taken in this study as shown in Eq. (13) (Moreita-Matias et al., 2014).

$$\textit{Bus Bunching} = \begin{cases} 1 \ \textit{if} \ h_i < \frac{h_1}{4} \\ 0 \ \textit{otherwise} \end{cases}. \tag{13}$$

h_i can be generated by the predicted headway at any given stop, and h_1 is the headway for the first stop on the same route.

5. Result analysis and comparison

5.1. Algorithm evaluation and comparison

As discussed in the data collection section, two bus routes with four-month smart card data are utilized to validate the accuracy and robustness of the proposed algorithm. For each bus route, six consecutive bus stops are selected. The validation process is divided into two stages: the first stage evaluates the performance of the LS-SVM algorithm for headway prediction, and the second stage performs a classification test on bus bunching by comparing the predicted binary events (based on the predicted headways) with ground truth. If bus bunching occurs, the event is denoted as 1; otherwise, it is denoted as 0. Data from the first three months are used for model parameter calibrations, and the remaining one-month data are used for testing. To measure the effectiveness of the LS-SVM used on bus headway prediction, the Mean Absolute Percentage Errors (MAPE) and Root Mean Square Errors (RMSE) are computed as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{h(i)-\hat{h}(i)}{\bar{h}}\right| \times 100\% \tag{14}$$

and

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}[h(i)-\hat{h}(i)]^2}, \tag{15}$$

where $h(i)$ and $\hat{h}(i)$ respectively represent the observed bus headway and predicted bus headway at bus run i. $\bar{h}$ is the mean of the observed bus headways; it aims to prevent zero denominators caused by zero headway values (Yamin and Shahidehpour, 2004). Bus bunching detection is essentially a binary classification test based on the prediction headway, and the positive value is defined as the occurrence of bus bunching. In the context of bus bunching detection, the accuracy, sensitivity, and specificity are used for performance evaluation indices. Sensitivity is also known as the true positive rate, and it measures the percentage of events with bus bunching that are correctly identified as such. Specificity is the true negative rate that measures the percentage of events without bus bunching that are correctly identified as such (Ma et al., 2015).

$$Accuracy(Acc) = \frac{number\ of\ correctly\ predicted\ events}{total\ number\ of\ events}. \tag{16}$$

Sensitivity and specificity can be respectively defined as

$$Sensitivity(SES) = \frac{number\ of\ events\ correctly\ predicted\ as\ bus\ bunching}{total\ number\ of\ bus\ bunching\ events} \tag{17}$$

$$Specificity(SPC) = \frac{number\ of\ events\ correctly\ predicted\ as\ no\ bus\ bunching}{total\ number\ of\ no\ bus\ bunching\ events} \tag{18}$$

Four well-established algorithms, ANN, KNN, Random Forest (RF), and Gaussian Process Regression (GPR), are compared with the LS-SVM algorithm in terms of prediction performance on both headway estimation and bus bunching classification. These algorithms are implemented in Matlab with normalized raw data. The fivefold cross-validation technique is used to determine the optimal parameter setting for LS-SVM. To maintain fairness in comparison, each algorithm is run multiple times with varying parameter settings, and the algorithm with the optimal performance is chosen for comparison with the LS-SVM algorithm. The ANN algorithms are executed 10 times with a maximum 1000 iterations to eliminate result fluctuation, whose learning rate is set as 0.001, and number of units in the hidden layer is set as 10. The K value of KNN is set as 10, and the output is weighted by the reciprocals of Euclidean distances between test points and its neighbors. The performance of GPR relies on the selection of covariance function (Rasmussen and Williams, 2006). In this study, the squared exponential function as presented in Eq. (19) is chosen to calculate the covariance:

$$K_{SE}(x_i, x_j) = \sigma_s^2 \exp\left(-\frac{1}{2l^2}(x_i - x_j)^2\right) + \sigma_n^2\delta_{ij} \tag{19}$$

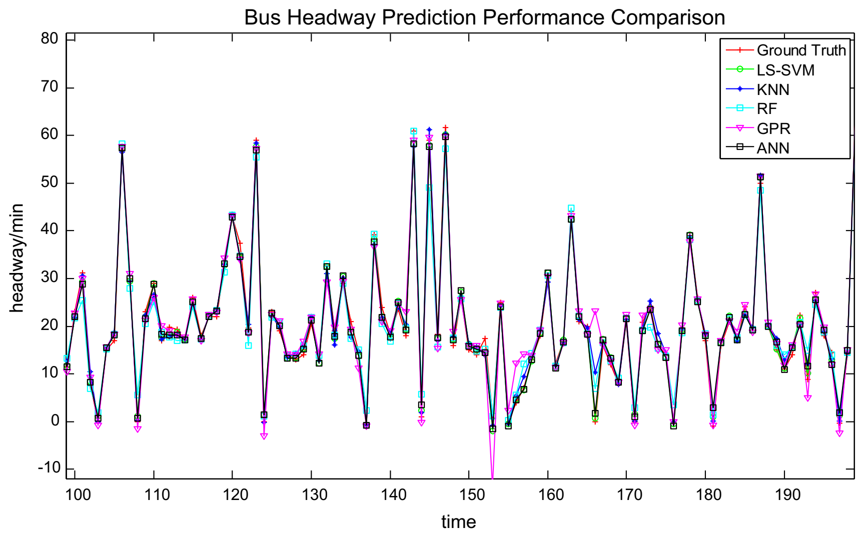
where σ_s^2 is the signal variance, l is the length scale, and σ_n^2 is the noise variance. The optimal setting of the above three parameters can be achieved by maximizing the marginal log likelihood function with the hyperparameters. GPML toolbox is utilized to implement GPR (Rasmussen and Nickisch, 2015). The prediction performance metrics for Routes A and B are presented in Table 2, where the optimal results are in bold font. The results are visualized in Fig. 2.

For Route A, the average headway at stop 49868 is 19.8 min with a standard deviation of 14.6 min. For Route B, the average headway at stop 42915 is 17.0 min with a standard deviation of 13.7 min. Route A has a higher bus bunching rate (17.5%) than Route B (14.4%), and this result coincides with the headway irregularities for both routes presented in Fig. 2. This result implies that bus bunching is likely to occur on the routes with high headway deviation.

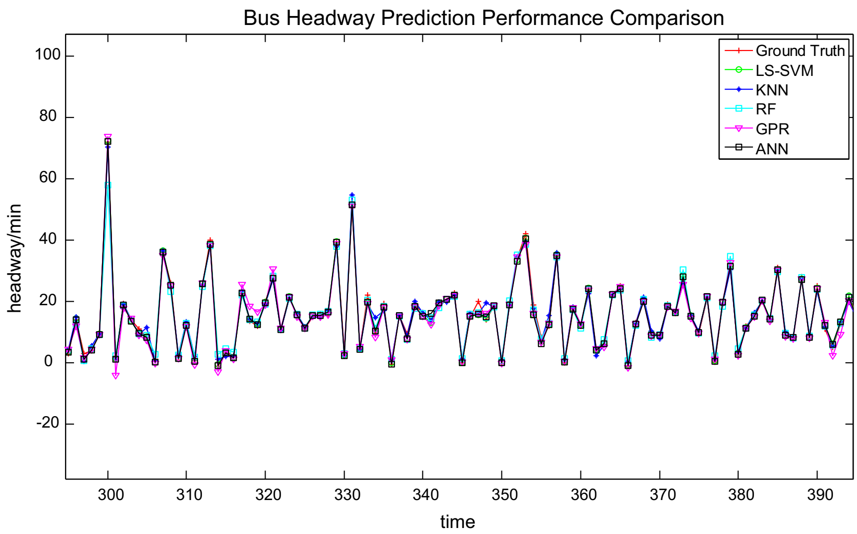
The LS-SVM algorithm receives the best prediction performance for Routes A and B with corresponding MAPE values of 5.41% and 4.97%. Predicting the headway for Route A is more challenging than predicting the headway for Route B because the headway for Route A exhibits a more irregular pattern, as demonstrated in Fig. 2. This fact explains why the MAPE value for Route A is higher than that for Route B. As for the other comparative algorithms, GPR performs the worst, followed by either the NN or KNN algorithms. GPR is a kernel-based regression technique to model nonlinear multivariate relationships, and it assumes that the random variables follow Gaussian distributions. This rule may not be successfully applied to predict

Table 2
Performance comparison of bus headway one-step-ahead prediction.

Algorithm	Route A		Route B	
	MAPE	RMSE	MAPE	RMSE
LS-SVM	**5.41%**	**1.3439**	**4.97%**	**3.3495**
KNN	6.71%	2.0933	8.43%	3.9851
ANN	9.99%	7.2992	7.46%	7.3128
RF	8.70%	2.7804	8.08%	3.7639
GPR	10.24%	4.1122	9.63%	5.7565



(a)



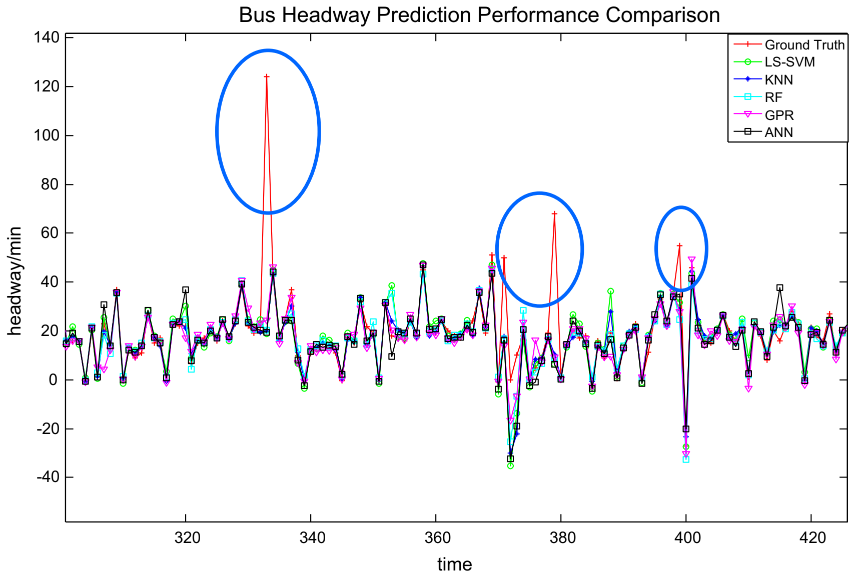
(b)

Fig. 2. Results of bus headway one-step-ahead prediction: (a) Stop 49868 for Route A and (b) Stop 42915 for Route B.

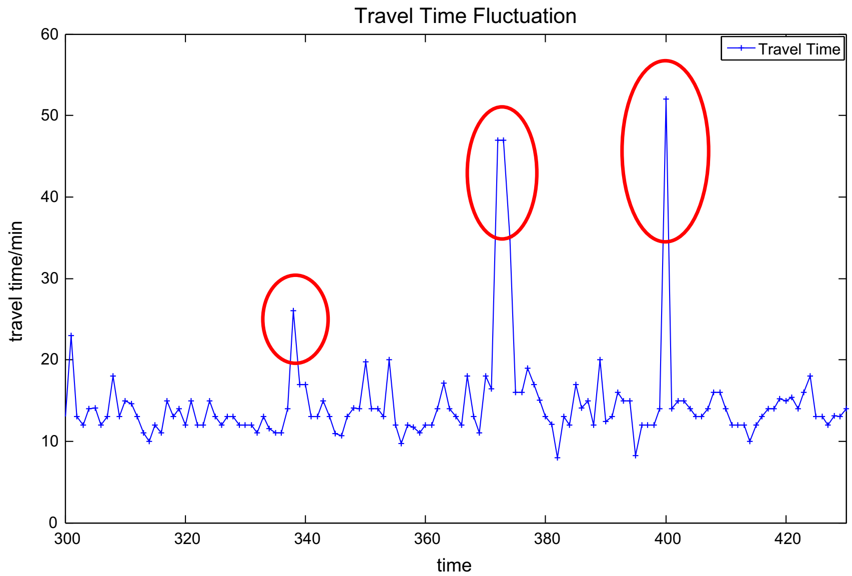
Table 3
Performance comparison of bus bunching one-step-ahead prediction.

Algorithm	Route A					Route B				
	Size	# of BB	SES (%)	SPC (%)	ACC (%)	Size	# of BB	SES (%)	SPC (%)	ACC (%)
LS-SVM	652	94	**100**	**99.82**	**99.85**	756	132	**100**	98.72	**98.94**
KNN	652	94	93.62	99.46	98.62	756	132	87.12	98.24	96.30
ANN	652	94	97.87	99.64	99.39	756	132	84.85	**99.04**	96.56
RF	652	94	85.11	99.64	97.55	756	132	90.91	98.08	96.83
GPR	652	94	82.98	98.92	96.63	756	132	90.91	98.40	97.09

Note: Size = test data sample size; # of BB = number of bus bunching; SES = Sensitivity; SPC = Specificity; ACC = Accuracy.

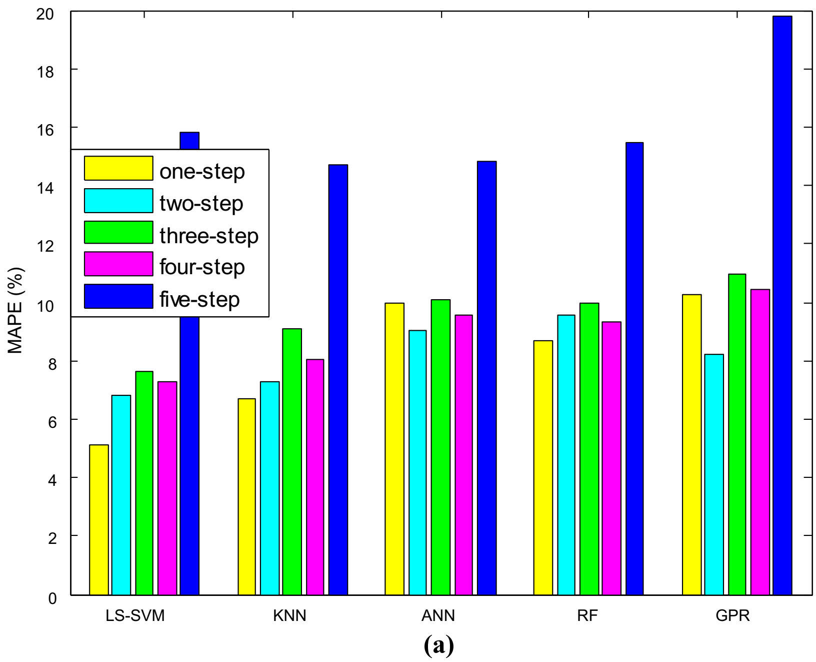


(a)



(b)

Fig. 3. Analysis of extreme bus headways on Stop 49868 for Route A: (a) results of five-step-ahead prediction and (b) travel time fluctuation.



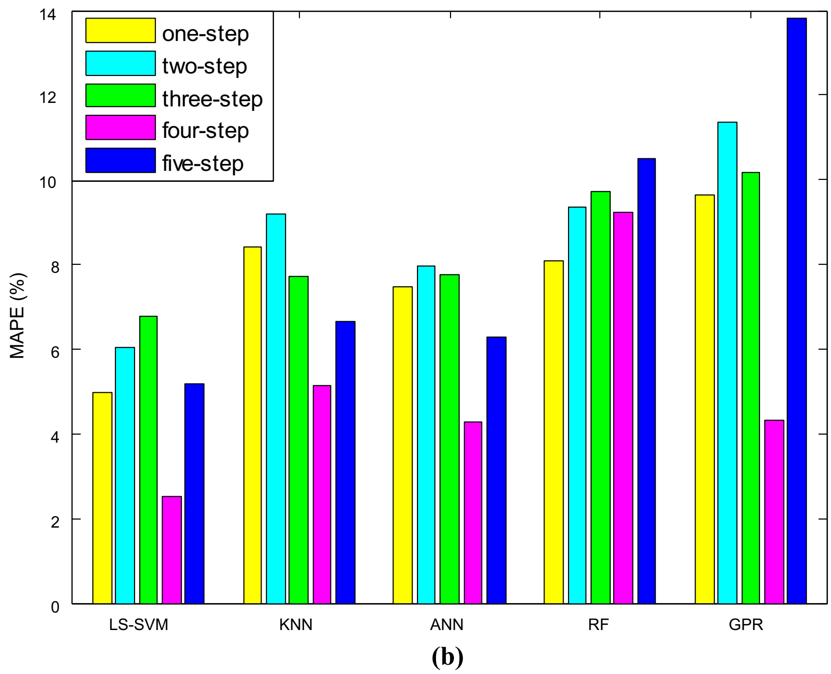


Fig. 4. MAPE comparison for multi-step-ahead headway predictions: (a) Route A and (b) Route B.

bus headways. The neural network can cope with stochastic and nonlinear data but is easily trapped in the local optima during its initialization, thus causing a high MSE value because of learning failures (see Table 2). Therefore, from the perspective of effectiveness and robustness, the LS-SVM outperforms all other algorithms.

On the basis of the predicted headways, the next step is to estimate the occurrence of bus bunching using Eq. (13). The results are presented in Table 3, where the best performance is highlighted in bold.

As demonstrated in Table 3, the LS-SVM outperforms other algorithms in terms of sensitivity and accuracy for two routes. The LS-SVM successfully identifies all bus bunching events for Routes A and B and respectively generates false alarm probabilities as low as 0.17% and 1.28%.

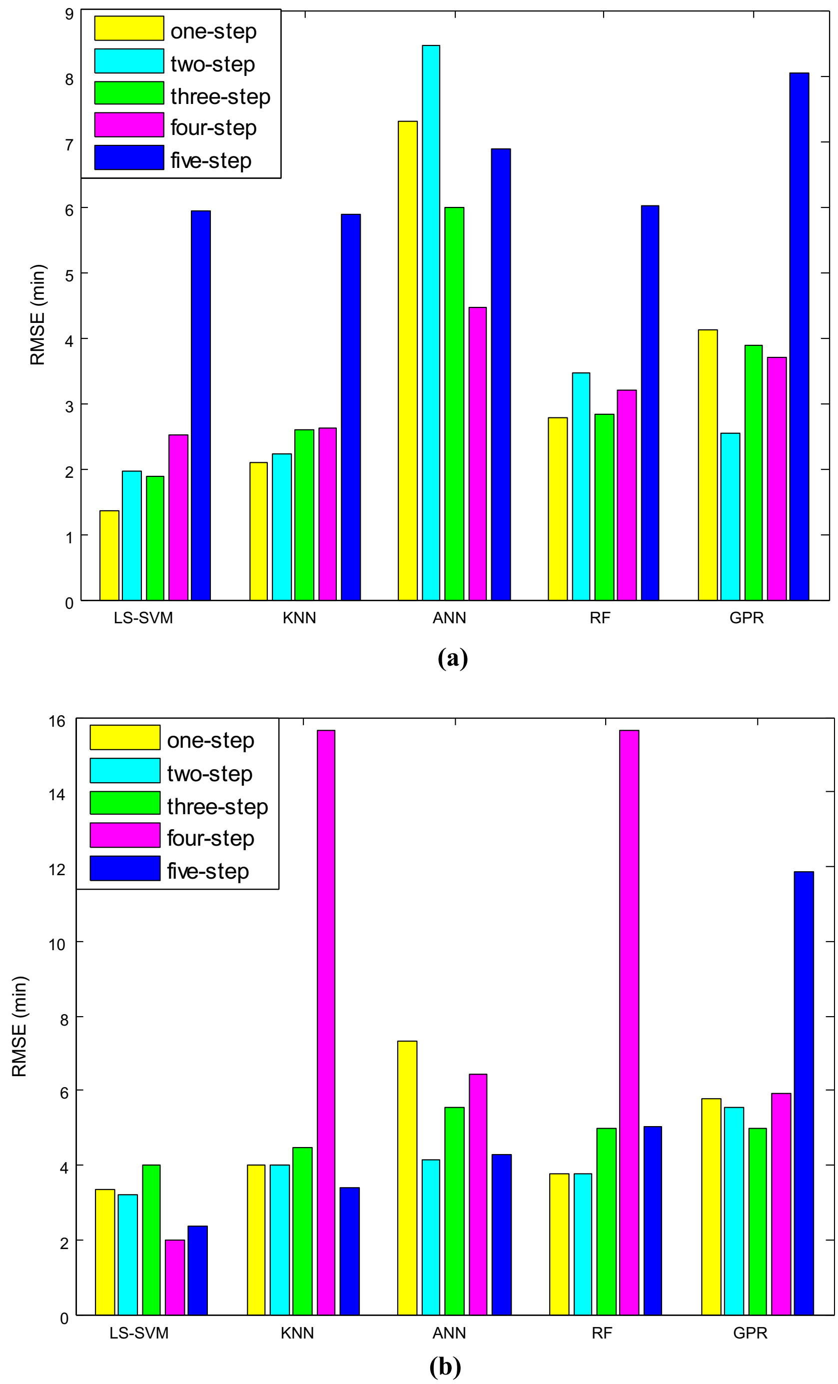


Fig. 5. RMSE comparison for multi-step-ahead headway predictions: (a) Route A and (b) Route B.

5.2. Analysis of multi-step-ahead prediction results

A multi-step prediction analysis is utilized for two underlying reasons: (1) No boarding and alighting demand may occur at a particular bus stop, and so the number of boarding and alighting passengers for bus bunching prediction at the two consecutive stops will be unavailable. (2) Transit operators are more interested in how early a bus bunching notice can be released. The sooner a bus bunching is detected, the more effective is the corresponding reactive control strategies. There-

Table 4
Performance comparison of bus headway two-step-ahead prediction.

Algorithm	Route A		Route B	
	MAPE	RMSE	MAPE	RMSE
LS-SVM	**6.82%**	**1.9707**	**6.05%**	**3.2269**
KNN	7.27%	2.2184	9.17%	3.9845
ANN	9.06%	8.4504	7.97%	4.1344
RF	9.58%	3.4527	9.37%	3.7836
GPR	8.23%	2.5438	11.34%	5.5539

Table 5
Performance comparison of bus headway three-step-ahead prediction.

Algorithm	Route A		Route B	
	MAPE	RMSE	MAPE	RMSE
LS-SVM	**7.63%**	**1.8817**	**6.78%**	**4.0160**
KNN	9.11%	2.5792	7.72%	4.4668
ANN	10.07%	5.9997	7.75%	5.5259
RF	9.96%	2.8372	9.73%	4.9867
GPR	10.94%	3.8851	10.19%	4.9997

Table 6
Performance comparison of bus headway four-step-ahead prediction.

Algorithm	Route A		Route B	
	MAPE	RMSE	MAPE	RMSE
LS-SVM	**7.26%**	**2.5174**	**2.50%**	**1.9941**
KNN	8.02%	2.6210	5.14%	15.6407
ANN	9.55%	4.4681	4.27%	6.4230
RF	9.32%	3.1961	9.24%	15.6704
GPR	10.46%	3.6914	4.33%	5.9344

Table 7
Performance comparison of bus headway five-step-ahead prediction.

Algorithm	Route A		Route B	
	MAPE	RMSE	MAPE	RMSE
LS-SVM	15.80%	5.9399	**5.19%**	**2.3652**
KNN	**14.68%**	**5.8779**	6.65%	3.4094
ANN	14.84%	6.8900	6.27%	4.2713
RF	15.44%	6.0225	10.51%	5.0473
GPR	19.77%	8.0360	13.82%	11.8483

Table 8
Performance comparison of bus bunching two-step-ahead prediction.

Algorithm	Route A					Route B				
	Size	# of BB	SES (%)	SPC (%)	ACC (%)	Size	# of BB	SES (%)	SPC (%)	ACC (%)
LS-SVM	743	87	**100**	99.85	**99.87**	824	138	**95.65**	98.83	**98.30**
KNN	743	87	**100**	99.54	99.60	824	138	88.41	97.81	96.24
ANN	743	87	98.85	**100**	**99.87**	824	138	78.26	**99.27**	95.75
RF	743	87	94.25	99.85	99.19	824	138	86.96	98.54	96.60
GPR	743	87	97.70	99.70	99.46	824	138	88.41	97.23	95.75

fore, we use the input information from the upstream stops that are multiple stops (from 2 stops to 5 stops) away from the predicted bus stop and evaluate their prediction performance as follows.

As the prediction step increases, the prediction metrics for all algorithms do not significantly deteriorate. LS-SVM possesses the best performance for Routes A and B, except for one scenario with a five-step-ahead prediction for Route A. A close examination of this exception reveals that high data sparseness can be observed in the test data, where more extreme headways (caused by unusually long traffic delays) occur away from the general shape of bus headway distribution than in other

Table 9
Performance comparison of bus bunching three-step-ahead prediction.

Algorithm	Route A					Route B				
	Size	# of BB	SES (%)	SPC (%)	ACC (%)	Size	# of BB	SES (%)	SPC (%)	ACC (%)
LS-SVM	1078	187	**98.40**	**99.55**	**99.35**	590	59	**88.14**	98.12	**97.12**
KNN	1078	187	**98.40**	99.33	99.17	590	59	**88.14**	97.93	96.95
ANN	1078	187	91.44	99.44	98.05	590	59	32.20	**99.44**	92.71
RF	1078	187	97.33	99.44	99.07	590	59	61.02	98.87	95.08
GPR	1078	187	97.86	**99.55**	99.26	590	59	74.58	97.74	95.42

Table 10
Performance comparison of bus bunching four-step-ahead prediction.

Algorithm	Route A					Route B				
	Size	# of BB	SES (%)	SPC (%)	ACC (%)	Size	# of BB	SES (%)	SPC (%)	ACC (%)
LS-SVM	826	72	97.22	**99.87**	99.64	203	6	**66.67**	**100**	**99.01**
KNN	826	72	**100**	99.73	**99.76**	203	6	**66.67**	**100**	**99.01**
ANN	826	72	94.44	**99.87**	99.39	203	6	**66.67**	99.49	98.52
RF	826	72	88.89	**99.87**	98.91	203	6	0	**100**	97.04
GPR	826	72	95.83	99.47	99.15	203	6	**66.67**	**100**	**99.01**

Table 11
Performance comparison of bus bunching five-step-ahead prediction.

Algorithm	Route A					Route B				
	Size	# of BB	SES (%)	SPC (%)	ACC (%)	Size	# of BB	SES (%)	SPC (%)	ACC (%)
LS-SVM	961	122	73.77	**99.52**	96.25	482	52	76.92	**99.30**	**96.89**
KNN	961	122	**91.80**	**99.52**	**98.54**	482	52	**78.85**	98.60	96.47
ANN	961	122	87.70	99.40	97.92	482	52	63.46	**99.30**	95.44
RF	961	122	86.07	99.40	97.71	482	52	71.15	98.84	95.85
GPR	961	122	86.89	98.93	97.40	482	52	23.08	99.53	91.29

scenarios (see Fig. 3). The LS-SVM fails to predict the extreme high bus headways at those circled data points as presented Fig. 3(a), and the possible reason causing such failures may lie in the highly-congested traffic condition, which can be measured by the sudden rises of travel times circled in Fig. 3(b). The times when the extreme travel times occurred match with the emergence of irregular bus headways. Unfortunately, these extreme values are not sufficient to be learned by LS-SVM. This situation is one of the primary drawbacks of LS-SVM (Suykens et al., 2002a; Mall and Suykens, 2015). LS-SVM cannot effectively handle outliers or noises due to the lack of sparsity (Yang et al., 2014). A possible remedy is to introduce a weighted LS-SVM model for robust regression (Suykens et al., 2002b). For the remaining cases, LS-SVM outperforms the other algorithms with the smallest MAPE and RMSE values. Another interesting finding is that the prediction lead time generates very little impact on the accuracy of bus headway prediction. This result implies that transit management centers can alert the bus drivers to the following buses with irregular headways in a very early fashion.

Figs. 4 and 5 intuitively demonstrate the changes of MAPE and RMSE for Routes A and B under the circumstances of multi-step-ahead (i.e., from one-step-ahead prediction to five-step-prediction) headway predictions.

The bus bunching events can also be detected based on the predicted headways. The results are presented in the following tables (see Tables 4–7), where the best performance is highlighted in bold.

In terms of total accuracy, LS-SVM wins six of eight times for both Route A and Route B. It ranks second when the lead time is four stops away from the target stop for Route A. However, LS-SVM fails in the case of five-step-ahead prediction, where its lowest accuracy and sensitivity can be witnessed. This result corresponds to the poor performance of LS-SVM when performing five-step-ahead bus headway prediction in Table 7. In the context of bus bunching detection, sensitivity should weigh more than specificity because bus bunching is a relatively rare event compared with normal bus operations. Low sensitivity indicates that most bus bunching occurrences are identified as normal operations with no additional preventative actions, while low specificity means that several normal bus operations are mistakenly detected as bus bunching. Therefore, one should pay more attention to the former than the latter. The above tables show that the sensitivity for Route B is generally higher than that for Route A. In most cases of Route B, LS-SVM receives the optimal result but lags behind KNN in the five-step-ahead prediction by 1.93%. All algorithms perform poorly in the four-step-ahead prediction, with a sensitivity as low as 66.67%. This result is due to a small sample size of only six bus bunching occurrences. Very few passengers swipe their smart cards at stop 42911, and thus a low matching rate of total training and testing records is obtained. For Route A, KNN achieves slightly better performance than LS-SVM, especially in the four-step-ahead prediction and five-step-ahead prediction. The difference, however, is marginal except for the five-step-ahead prediction. The success of KNN primarily results from its robustness to noise data with a large K value (Tang et al., 2016), but LS-SVM may be less effective in these scenarios. However, when performing regression-based prediction on bus headways, LS-SVM yields more accurate results than KNN. Considering the high computation cost of KNN, LS-SVM is preferable to KNN in real-time bus bunching detection (see Tables 8–11, where the best performance is highlighted in bold).

6. Conclusion

Transit authorities strive to improve transit service reliability by enhancing on-time performance and schedule adherence. However, the emergence of bus bunching severely deteriorates the quality of transit service and increases passenger waiting time with uneven passenger loadings. To mitigate the adverse effect of bus bunching, this paper proposes a bus bunching prediction framework that leverages transit smart card data for stop-level headway irregularity mining. Historical headways, passenger demands, and link travel time are extracted as influential factors for bus bunching and are inputted to the LS-SVM regression to forecast the headway deviation at the next stop, where the occurrence of bus bunching can be detected by thresholding the predicted headway with the planned bus schedule. Four-month smart card data from two routes in Beijing are utilized to evaluate the algorithm performances of the headway and bus bunching predictions. In most scenarios, LS-SVM outperforms well-established artificial intelligent algorithms in terms of accuracy and robustness. A multi-step prediction experiment is also conducted to account for the uncertainty caused by passenger boarding and alighting demands, as well as to investigate how early a bus bunch notice can be given to transit operators. We find that the bus bunching detection performance does not significantly decay as the lead time increases.

This research contributes to existing literature on the following aspects: (1) A data-driven approach is presented to predict the occurrence of bus bunching by using transit smart card data, whereas most of the previous studies focused on developing reactive actions to maintain the stability of the headway at the cost of passengers' dissatisfaction and schedule unreliability. By providing early warning information on when and where the bus bunching may occur, transit authorities can employ proactive countermeasures to mitigate bus bunching, and therefore improve the quality of transit service for operating cost savings and increasing ridership (Ding et al., 2014). (2) From the perspective of passengers, knowing when the next bus will arrive will alleviate their anxiety when waiting at a bus stop and help them choose a less crowded vehicle. These goals can be realized by predicting stop-level headway in an accurate manner. (3) This study investigates the impact of different stop horizons on the accuracy of bus bunching detection, which will provide sufficient time for transit operators to adopt preventive measures.

The present work can be further strengthened in the following aspects. For instance, bus driver behavior and overcrowding caused by multiple buses for different routes arriving at the same stop should be considered in the prediction model to improve accuracy. From the perspective of algorithm improvements, LS-SVM can be enhanced to incorporate sparseness into the model (Jiang et al., 2016). In addition, external data sources such as GPS data can be integrated with smart card data to generate more reliable dwell time estimation. Both data types can then be fed back jointly into the transit management center via a real-time data transmission system. The system design scheme lays a foundation for future work to apply theoretical research into practice.

Acknowledgment

The authors would like to appreciate the funding support from the National Natural Science Foundation of China (51408019, 51308021, U1564212, 71402011), Science and Technology Research Foundation for Transportation [2015318221020], National Key Technologies R & D Program of China (2014BAG01B03), Beijing Nova Program (z151100000315048), and the Fundamental Research Funds for the Central Universities.

References

Albright, E., Figliozzi, M.A., 2012. Analysis of the impacts of transit signal priority on bus bunching and performance. In: Proceedings of the Conference on Advanced Systems for Public Transport (CASPT), June 2012, Santiago, Chile.
Bartholdi III, J.J., Eisenstein, D., 2012. A self-coordinating bus route to resist bus bunching. Transport. Res. Part B Methodol. 46 (4), 481–491.
Bellei, G., Gkoumas, K., 2010. Transit vehicles' headway distribution and service irregularity. Publ. Transport 2 (4), 269–289.
Bie, Y., Gong, X., Liu, Z., 2015. Time of day intervals partition for bus schedule using GPS data. Transport. Res. Part C Emerg. Technol. 60, 443–456.
Chang, H., Park, D., Lee, S., Lee, H., Baek, S., 2010. Dynamic multi-interval bus travel time prediction using bus transit data. Transportmetrica 6 (1), 19–38.
Chen, M., Chien, S., Liu, X., Brickey, J., 2003. Application of APC/AVL archived data support system. In: TRB 82nd Annual Meeting (CD-ROM), Washington, D.C.
Chen, M., Liu, X., Xia, J., Chien, S.I., 2004. A dynamic bus-arrival time prediction model based on APC data. Comput.-Aided Civil Infrastruct. Eng. 19 (5), 364–376.
Chen, G., Yang, X., An, J., Zhang, D., 2012. Bus-arrival-time prediction models: link-based and section-based. J. Transport. Eng. 138 (1), 60–66.
Chen, G., Huang, L., Liu, H., Yang, X., 2013. Effects of bunching at stops on bus dwell time. Int. IEEE Conf. Intell. Transport. Syst., 1499–1502
Chien, S.I., Kuchipudi, C.M., 2003. Dynamic travel time prediction with real-time and historic Data. J. Transport. Eng. 129 (6), 608–616.
Chien, S.I., Ding, Y., Wei, C., 2002. Dynamic bus arrival time prediction with artificial neural networks. J. Transport. Eng. 128 (5), 429–438.
Coffey, C., Pozdnoukhov, A., Calabrese, F., 2011. Time of arrival predictability horizons for public bus routes. ACM Sigspatial International Workshop on Computational Transportation Science, pp. 1–5.
Daganzo, C.F., 2009. A headway-based approach to eliminate bus bunching: systematic analysis and comparisons. Transport. Res. Part B Methodol. 43 (10), 913–921.
Daganzo, C.F., Pilachowski, J., 2011. Reducing bunching with bus-to-bus cooperation. Transport. Res. Part B Methodol. 45 (1), 267–277.

D'Angelo, M.P., Al-Deek, H.M., Wang, M.C., 1999. Travel-time prediction for freeway corridors. Transport. Res. Rec. J. Transport. Res. Board 1676, 184–191.
Ding, C., Lin, Y., Liu, C., 2014. Exploring the influence of built environment on tour-based commuter mode choice: a cross-classified multilevel modeling approach. Transport. Res. Part D: Transport Environ. 32, 230–238.
Ding, C., Mishra, S., Lin, Y., Xie, B., 2015. Cross-nested logit choice model of joint travel mode and departure time choice for urban commuting trips: case study in Maryland-Washington, DC Region. J. Urban Plan. Develop. 141 (4), 04014036.
Ding, C., Wang, Y., Yang, J., Liu, C., Lin, Y., 2016. Spatial heterogeneous impact of built environment on household auto ownership levels: evidence from analysis at traffic analysis zone scales. Transport. Lett. 8 (1), 26–34.
Feng, W., Figliozzi, M., 2011. Empirical findings of bus bunching distributions and attributes using archived AVL/APC bus data. Int. Conf. Chin. Transport. Prof., 4330–4341
Fonzone, A., Schmöcker, J.D., Liu, R., 2015. A model of bus bunching under reliability-based passenger arrival patterns. Transport. Res. Part C: Emerg. Technol. 7, 164–182.
Hammerle, M., Haynes, M., Mcneil, S., 2005. Use of Automatic Vehicle Location and Passenger Count Data to Evaluate Bus Operations. Experience of the Chicago Transit Authority, Illinois.
Jeong, R., 2004. The prediction of bus arrival time using automatic vehicle location systems data. Dissertation Abstracts International, vol. 65–12, Section: B, page: 6530. Co-Chairs: Laurence R.
Jeong, R., Rilett, L.R., 2004. Bus arrival time prediction using artificial neural network model. In: The International IEEE Conference on Intelligent Transportation Systems. Proceedings, pp. 988–993.
Jiang, H., Zou, Y., Zhang, S., Tang, J., Wang, Y., 2016. Short-term speed prediction using remote microwave sensor data: machine learning versus statistical model. Math. Probl. Eng. (1965), 1–13
Kimpel, T.J., Strathman, J.G., Callas, S., 2008. Improving scheduling through performance monitoring. Lect. Notes Econ. Math. Syst. 600, 253–280.
Lin, W.H., Zeng, J., Zeng, L., 1999. Experimental study of real-time bus arrival time prediction with GPS data. Transp. Res. Rec. (1)
Lin, Y., Yang, X., Zou, N., Jia, L., 2013. Real-time bus arrival time prediction: case study for Jinan, China. J. Transport. Eng. 139 (11), 1133–1140.
Ma, X., Wang, Y., 2014. Development of a data-driven platform for transit performance measures using smart card data and GPS data. J. Transport. Eng. 140 (12), 04014063.
Ma, X., Wang, Y., Chen, F., Liu, J., 2012. Transit smart card data mining for passenger origin information extraction. J. Zhejiang Univ. Sci. C 13 (10), 750–760.
Ma, X., Wu, Y., Wang, Y., Chen, F., Liu, J., 2013. Mining smart card data for transit riders' travel patterns. Transport. Res. Part C – Emerg. Technol. 36, 1–12.
Ma, X., Yu, H., Wang, Y., Wang, Y., 2015. Large-scale transportation network congestion evolution prediction using deep learning theory. PLoS ONE 10 (3).
Mall, R., Suykens, J.A., 2015. Very sparse LSSVM reductions for large-scale data. IEEE Transact. Neur. Netw. Learn. Syst. 26 (5), 1086–1097.
Moreira-Matias, L., Ferreira, C., Gama, J., Mendes-Moreira, J., Sousa, J.F.D., 2012a. Bus Bunching Detection by Mining Sequences of Headway Deviations. Springer, Berlin, Heidelberg.
Moreira-Matias, L., Ferreira, C., Gama, J., Mendes-Moreira, J., Sousa, J.F.D., 2012b. Bus Bunching Detection: A Sequence Mining Approach. Ubiquitous Data Mining.
Moreira-Matias, L., Gama, J., Mendes-Moreira, J., Sousa, J.F.D., 2014. An Incremental Probabilistic Model to Predict Bus Bunching in Real-time. Advances in Intelligent Data Analysis XIII Lecture Notes in Computer Science, vol. 8819, pp. 227–238.
Nair, R., Bouillet, E., Gkoufas, Y., Versceure, O., Mourad, M., Yashar, F., Perez, R., Perez, J., Brtant, G., 2015. Data as a resource: real-time predictive analytics for bus bunching. In: 94th Annual Meeting of the Transportation Research Board, Washington, D.C.
Padmanaban, R.P.S., Divakar, K., Vanajakshi, L., Subramanian, S.C., 2010. Development of a real-time bus arrival prediction system for Indian traffic conditions. IET Intel. Transport Syst. 4 (3), 189–200.
Patnaik, J., Chien, S., Bladikas, A., 2004. Estimation of bus arrival times using APC data. J. Publ. Transport. 7 (1), 128001–128100.
Pilachowski, J.M., 2009. An Approach to Reducing Bus Bunching. Dissertations & Theses - Gradworks.
Rasmussen, C.E., Nickisch, H., 2015. The GPML Toolbox version 3.5 Available online <http://www.gaussianprocess.org/gpml/code/matlab/doc/manual.pdf> (accessed on 10 September, 2016).
Rasmussen, C.E., Williams, C.K.I., 2006. Gaussian Processes for Machine Learning. The MIT Press.
Robinson, S., Narayanan, B., Toh, N., Pereira, F., 2014. Methods for pre-processing smartcard data to improve data quality. Transport. Res. Part C Emerg. Technol. 49, 43–58.
Shalaby, A., Eng, P., Farhan, A., Sc, M.A., 2003. Bus travel time prediction model for dynamic operations control and passenger information systems. Steroids 53 (1–2), 131–147.
Sinn, M., Yoon, J.W., Calabrese, F. and Bouillet, E., 2012. Predicting Arrival Times of Buses Using Real-time GPS Measurements, pp. 1227–1232.
Strathman, J.G., Kimpel, T.J., Dueker, K.J., Gerhart, R.L., Callas, S., 2002. Evaluation of transit operations: data applications of Tri-Met's automated Bus Dispatching System. Transportation 29 (3), 321–345.
Suykens, J.A.K., Brabanter, J.D., Lukas, L., Vandewalle, J., 2002a. Weighted least squares support vector machines: robustness and sparse approximation. Neurocomputing 48 (1–4), 85–105.
Suykens, J.A.K., Gestel, T.V., Brabanter, J.D., Moor, B.D., Vandewalle, J., 2002b. Least square support vector machine. Euphytica 2 (2), 1599–1604.
Tang, J., Zou, Y., Ash, J., Zhang, S., Liu, F., Wang, Y., 2016. Travel time estimation using freeway point detector data based on evolving fuzzy neural inference system. PLoS ONE 11 (2).
TCRP Report 100: Transit Capacity and Quality of Service Manual, 2nd ed., 2003. Transportation Research Board of the National Academies, Washington, D.C.
Tiesyte, D., Jensen, C.S., 2008. Similarity-based prediction of travel times for vehicles traveling on known routes. In: ACM Sigspatial International Symposium on Advances in Geographic Information Systems, Acm-Gis 2008, November 5–7, 2008, Irvine, California, USA, Proceedings, pp. 1–10.
Vapnik, B.V., 2010. Statistical Learning Theory. John Wiley, New York.
Wang, T., 2014. A FlexiFare Bus (F2B) system to reduce bus bunching problem. In: International Conference on Progress in Informatics and Computing.
Wang, H., Hu, D., 2005. Comparison of SVM and LS-SVM for regression. In: International Conference on Neural Networks and Brain, 2005. Icnn&B, pp. 279–283.
Watkins, K.E., Ferris, B., Borning, A., Rutherford, G.S., Layton, D., 2011. Where Is My Bus? Impact of mobile real-time information on the perceived and actual wait time of transit riders. Transport. Res. Part A Policy Practice 45 (8), 839–848.
Yamin, H.Y., Shahidehpour, S.M., 2004. Bidding strategies using price based unit commitment in a deregulated power market. Electr. Power Compo. Syst. 32 (32), 229–245.
Yang, X., Tan, L., He, L., 2014. A robust least squares support vector machine for regression and classification with noise. Neurocomputing 140, 41–52.
Yu, B., Yang, Z.Z., Lin, J.Y., 2007. Bus arrival time prediction using support vector machines. J. Intell. Transport. Syst. 10 (4), 151–158.
Yu, B., Lam, W.H.K., Mei, L.T., 2011. Bus arrival time prediction at bus stop with multiple routes. Transport. Res. Part C Emerg. Technol. 19 (6), 1157–1170.
Zhang, Y., Liu, Y., 2009. Traffic forecasting using least squares support vector machines. Transportmetrica 5 (3), 193–213.